

**AAP-33**

**NATO GLOSSARY  
OF  
AVIONICS TERMS AND DEFINITIONS**

**GLOSSAIRE  
DES TERMES ET DEFINITIONS  
D'AVIONIQUE  
POUR  
L'OTAN**



NATO UNCLASSIFIED

AAP-33

NORTH ATLANTIC TREATY ORGANIZATION  
MILITARY AGENCY FOR STANDARDIZATION (MAS)  
NATO LETTER OF PROMULGATION

March 1998

1. AAP-33 - STANDARDIZED AVIONICS TERMINOLOGY AND ABBREVIATIONS - is a NATO UNCLASSIFIED publication. The agreement of nations to use this publication is recorded in STANAG 3908.
2. AAP-33 is effective upon receipt.



A. GRØNHEIM  
Major General, NOAF  
Chairman MAS

- III -

NATO UNCLASSIFIED

ORIGINAL

RECORD OF CHANGE

CHANGE DATE	DATE ENTERED	EFFECTIVE DATE	BY WHOM ENTERED

TABLE OF CONTENTS

<b>CHAPTER/CHAPITRE 1 ENGLISH TERMS AND DEFINITIONS TERMES ET DEFINITIONS ANGLAIS</b>	<b>1</b>
<b>CHAPTER/CHAPITRE 2 TERMES ET DEFINITIONS FRANCAIS FRENCH TERMS AND DEFINITIONS</b>	<b>1</b>
<b>CHAPTER/CHAPITRE 3 ENGLISH TO FRENCH CROSS-REFERENCE RENOI ANGLAIS A FRANCAIS</b>	<b>1</b>
<b>CHAPTER/CHAPITRE 4 RENOI FRANCAIS A ANGLAIS FRENCH TO ENGLISH CROSS-REFERENCE</b>	<b>1</b>
<b>CHAPTER/CHAPITRE 5 LIST OF ABBREVIATIONS LISTE DES ABBREVIATIONS</b>	<b>1</b>

CHAPTER/CHAPITRE 1ENGLISH TERMS ANDDEFINITIONS TERMES ETDEFINITIONS ANGLAISabort

To terminate a process prior to completion.  
(ANSI/IEEE)

absolute machine code

Machine language code that must be loaded into fixed storage locations at each use and may not be relocated.  
(ANSI/IEEE) Contrast with relocatable machine code.

abstract machine

- a. A representation of the characteristics of a process or machine. (ANSI/IEEE)
- b. A module that processes inputs as though it were a machine. (ANSI/IEEE)

abstraction

- a. A view of a problem that extracts the essential information relevant to a particular purpose and ignores the remainder of the information.  
(ANSI/IEEE)
- b. The process of forming an abstraction.  
(ANSI/IEEE)

acceptance testing

Formal testing conducted to determine whether or not a system satisfies its acceptance criteria and to enable the customer to determine whether or not to accept the

system. (ANSI/IEEE) See also qualification testing, system testing.

### access

To obtain data from a storage medium, or former core store.

### access control mechanism

Hardware or software features, operating procedures, or management procedures designed to permit authorized access and prevent unauthorized access to a computer system. (ANSI/IEEE)

### access time

- a. The time interval between the instant at which data is called for from a storage device and the instant delivery begins.
- b. The time interval between the instant at which data is requested to be stored and the instant at which storage is started.

### accessibility

The extent to which software facilitates selective use or maintenance of its components. (ANSI/IEEE)

### accumulator

A register in which the result of an arithmetic or logic operation is formed.

### accuracy

The degree of freedom from error, that is, the degree of conformity to truth or to a rule. Accuracy is contrasted with precision. For example, four-place numerals are less precise than six-place numerals, nevertheless a properly computed four-place numeral might be more accurate than an improperly computed six-place numeral. (ANSI)

**acronym**

An abbreviation, such as UNO, NATO, ALGOL, COBOL, formed from the initial letters of a name consisting of several words.

**actual parameter**

An argument or expression used within a call to a subprogram to specify data or program elements to be transmitted to the subprogram. (ANSI/IEEE) Contrast with formal parameter.

**Ada**

A computer programming language used for especially military applications.

**Ada programming support environment (APSE)**

A collection of software tools to support Ada applications.

**adaptability**

The ease with which software allows differing system constraints and user needs to be satisfied. (ANSI/IEEE)

**adaptive maintenance**

Maintenance performed to make a software product usable in a changed environment. (ANSI/IEEE)

**address**

- a. A character or group of characters that identifies a register, a particular part of storage, or some other data source or destination. (ISO)
- b. To refer to a device or an item of data. (ISO)



---

**address generation**

The derivation, by means of a formula or algorithm applied to a record key, of the physical address of the record on a mass-storage unit such as a disk.

---

**address space**

The range of addresses available to a computer program. (ANSI/IEEE)

---

**ALGOL**

Acronym for Algorithmic Language, a programming language for defining scientific and mathematical problems.

---

**algorithm**

- a. A finite set of well-defined rules for the solution of a problem in a finite number of steps. Example: a full statement of an arithmetic procedure for evaluating  $\sin x$  to a stated precision. (ADatP-2)
- b. A finite set of well-defined rules for solution of a problem in a finite number of steps; for example, a complex specification of a sequence of arithmetic operations for evaluating  $\sin x$  to give precision. (ISO)
- c. A finite set of well-defined rules that gives a sequence of operations for performing a specific task. (ANSI/IEEE)
- d. A program or subroutine may be an algorithm, or may consist of a number of algorithms.

---

**algorithm analysis**

The examination of an algorithm to determine its correctness with respect to its intended use, to determine its operational characteristics, or to understand it more fully in order to modify, simplify, or improve it. (ANSI/IEEE)

**alias**

- a. An additional name for an item. (ANSI/IEEE)
- b. An alternate label. For example, a label and one or more aliases may be used to refer to the same data element or point in a computer program. (ANSI)

**alignment**

The shifting of operands involved in an arithmetic operation so that their positions are in the same relative positions within computer words.

**allocate**

An operating system service used by a task to request memory for message or I/O buffers.

**allocated baseline**

A milestone activity that marks the completion of the software analysis phase. The allocated baseline captures the mapping between the architecture and software requirements.

**alpha test**

The test stage conducted at the developer's site by a customer. These tests are conducted in a controlled environment.

**analog**

Pertaining to representation by means of continuously variable physical quantities. (ADatP-2)

**analogue-to-digital (A/D)**

The conversion of analogue signals to digital data.

**analytical model**

A representative of a process or phenomenon by a set of solvable equations. (ANSI/IEEE) Contrast with simulation.

**application(s) software**

That software which is written to accomplish a desired functional task and is to be distinguished from operating systems, executives, and supervisory programmes generally.

**application-oriented language**

- a. A computer-oriented language with facilities or notations applicable primarily to a single application area; for example, a language for statistical analysis or machine design. (ANSI/IEEE)
- b. A problem-oriented language whose statements contain or resemble the terminology of the occupation or profession of the user. (ANSI)

**apt**

Acronym for Automatically Programmed Tools, a programming language for the numerical control of machine tools.

**aptitude test**

A test designed to discover whether an applicant has the basic aptitude for a particular job. In the case of programming, the test will usually consist of logical and semantic problems.

**architectural design**

- a. The process of defining a collection of hardware and software components and their interfaces to establish a framework for the development of a computer system. (ANSI/IEEE)

- b. The result of the architectural design process.  
(ANSI/IEEE)

### architecture

In avionics, a representation of the hardware and software components of a system and their interrelationships, considered from the viewpoint of the whole system.

### archival memory

Functionally similar to secondary memory except that off-line storage media are used.

### arithmetic unit

The unit of a computing system that contains the circuits that perform arithmetic operations.

### assemble

- a. To prepare a machine language programme from a symbolic language programme by substituting absolute codes for symbolic operation codes and absolute or relocatable addresses for symbolic absolute or relocatable addresses for symbolic addresses, including, in the machine language programme, routines which may be called from a programmed library. (ADatP-2)
- b. To translate a program expressed in an assembly language into a machine language and perhaps to link subroutines. Assembling is usually accomplished by substituting machine language operation codes for assembly language operation codes and by substituting absolute addresses, immediate addresses, relocatable addresses, or virtual addresses for symbolic addresses.  
(ANSI/IEEE) Contrast with compile, interpret.

### assembler

- a. A computer programme that assembles. (ADatP-2)

- b. A computer program used to assemble. (ISO)  
Contrast with compiler, interpreter.  
(ANSI/IEEE)

### assembly language

A symbolic programming language from which a corresponding machine programme is assembled. The instructions in a machine language program are usually generated in a one-to-one correspondence with those in the assembly language programme. (ADatP-2)

### assembly language or assembly code

A programming language in which the numerical form of a machine code instruction is represented by a mnemonic, e.g. "ADX" for a numerical code meaning "add contents of store location into accumulator".

### assertion

A logical expression specifying a program state that must exist or a set of conditions that program variables must satisfy at a particular point during program execution; for example, A is positive and A is greater than B. (ANSI/IEEE)

### assignment statement

An instruction used to express a sequence of operations, or used to assign operands to specified variables, or symbols, or both. (ANSI)

### associative memory

Another term for Content-Addressable Memory.

### asynchronous

Relating to an event or device which is not coordinated with or controlled by a common clock.

**audit**

An independent review for the purpose of assessing compliance with software requirements, specifications, baselines, standards, procedures, instructions, codes, and contractual and licensing requirements. (ANSI/IEEE)

**autocode**

A computer language which may be similar to a natural language such as English, or to algebra, and is "problem-oriented" rather than "machine-oriented".

**automated design tool**

A software tool that aids in the synthesis, analysis, modelling, or documentation of a software design. Examples include simulators, analytic aids, design representation processors, and documentation generators. (ANSI/IEEE)

**automated test generator**

A software tool that accepts as input a computer program and test criteria, generates test input data that meet these criteria, and sometimes, determines the expected results. (ANSI/IEEE)

**automated verification system**

A software tool that accepts as input a computer program and a representation of its specification, and produces, possibly with human help, a correctness proof or disproof of the program. (ANSI/IEEE)

**automated verification tools**

A class of software tools used to evaluate products of the software development process. These tools aid in the verification of such characteristics as correctness, completeness, consistency, traceability, testability, and adherence to standards. Examples include design analyzers, automated verification systems, static analyzers, dynamic analyzers, and standards enforcers. (ANSI/IEEE)

**automatic test equipment (ATE)**

General and special purpose test equipment used to apply a programmed sequence of tests to verify proper operation of, or identify/isolate faults in, systems or sub-systems.

**availability**

The ratio of effective hours of operation to the total number of hours of scheduled operation expressed as a percentage.

**availability model**

A model used for predicting, estimating, or assessing availability. (ANSI/IEEE)

**avionics**

The application of electronics in aviation.

**back porch**

The portion of a composite video signal that follows the horizontal synchronizing pulse and extends to the trailing edge of the corresponding blanking pulse.

**back-up**

Provisions made for the recovery of data files or software, for restart of processing, or for use of alternative computer equipment after a system failure or a disaster. (ANSI/IEEE)

**back-up programmer**

The assistant leader of a chief programmer team; a senior-level programmer whose responsibilities include contributing significant portions of the software being developed by the team, aiding the chief programmer in reviewing the work of the other team members, substituting for the chief programmer when necessary, and having an overall technical understanding of the software being developed. (ANSI/IEEE)

**backplane**

The structure that provides a physical interface for the modules/cards that comprise a system. The interface consists of module/card connectors into which the modules/cards are inserted to maintain interconnection continuity through the backplane structure.

**base address**

A given address from which an absolute address is derived by combination with a relative address. (ADatP-2)

**base registers**

A high speed storage location that contains the address of the first data word or instruction of a segment. All addresses within the segment are referenced to the start of the segment, and during execution, the physical address of each memory reference is calculated by simply adding the contents of the base register to the relative addresses contained within the segment. Thus, relocation is easily accomplished by simply moving a segment in memory and placing the new address in the base register.

**baseline**

- a. A specification or product that has been formally reviewed and agreed upon, that thereafter serves as the basis for further development, and that can be changed only through formal change control procedures. (ANSI/IEEE)
- b. A configuration identification document or a set of such documents formally designed and fixed at a specific time during a configuration item's life cycle. Baselines, plus approved changes from those baselines, constitute the current configuration identification. For configuration management there are three baselines, as follows:
  - (1) Functional baseline. The initial approved functional configuration.



- (2) Allocated baseline. The initial approved allocated configuration.
- (3) Product baseline. The initial approved or conditionally approved product configuration identification. (MIL-STD-973)

### baseline management

The application of technical and administrative direction to designate the documents and changes thereto which formally identify, designate and control the configuration item at specific times during its life cycle.

### batch

A type of operating system where jobs are processed one at a time and a job must complete its execution before the next is begun.

### begin-end block

A sequence of design or programming statements bracketed by begin and end delimiters and characterized by a single entrance and single exit. (ANSI/IEEE)

### benchmark

A test designed to evaluate the performance of a computer program, according to specified criteria.

### best fit

An algorithm for memory allocation that searches the memory free list for the unused memory block that is closest in size to that needed by the requesting task.

### beta testing

The test stage conducted at one or more customer by the end-user of the software. Unlike alpha testing, the

developer is generally not present. Therefore, the beta testing is a "live" application of the software in an environment that cannot be controlled by the developer. The customer records all problems (real or imagined) that are encountered during this phase and reports these to the developer at regular intervals. As a result of problems reported during beta test, the software developer makes modifications and then prepares to release the software product to the entire customer base.

### binary

- a. Characterized by a selection, choice or condition that has two possible different values or states. (ADatP-2)
- b. Of a fixed radix numeration system, having a radix of two. (ADatP-2)

### binary digit (bit)

A digit in the binary scale of notation, i.e., 0 or 1. It is commonly used to represent two distinct characteristics, e.g., on-off, yes-no. Usually abbreviated bit. (ADatP-2)

### binary element

A constituent element of data that may take either of two values or states. (ADatP-2)

### binding

The assigning of a value or referent to an identifier; for example, the assigning of a value to a parameter or the assigning of an absolute address, virtual address, or device identifier to a symbolic address or label in a computer program. (ANSI/IEEE)

### bit

Contraction of binary digit. (ADatP-2)

**bit rate**

The number of bits per unit of time which pass a given point in a given channel or which a device can accept (input) or deliver (output). (ADatP-2)

**black box**

- a. A generic term used to describe an unspecified device which performs a special function or in which known inputs produce known outputs in a fixed relationship.
- b. A generic term for any integral unit or device, but most often used to refer to electronic devices. (Computer Dictionary)

**block (n)**

A physical unit of data on a magnetic storage medium such as a tape or disk.

**block (v)**

To assemble data into blocks for processing.  
(ADatP-2)

**block diagram**

A diagram of a system, instrument or computer in which the principal parts are represented by suitably associated geometrical figures to show both the basic functions and the functional relationship among the parts. Contrast with flowchart. (ANSI)

**block-structured language**

A design or programming language in which sequences of statements are demarcated, usually with begin and end delimiters. (ANSI/IEEE)

**blocking factor**

The maximum number of records held in a block on a backing-store medium, for a particular file.

**boolean**

Originally meant "pertaining to processes in the algebraic system invented by the English mathematician George Boole (1815-1964)". When used as a noun in programming it refers to a switch, held as a single bit, which can be in one of two states.

**boolean logic**

A deductive system concerned with classes, propositions and two state circuit elements, represented by symbols and terms such as AND, OR, NAND.

**boot**

To use a bootstrap. (ISO)

**bootstrap**

- a. A short computer program that is permanently resident or easily loaded into a computer, whose execution brings another, larger program, such as an operating system or its loader, into memory. (ANSI/IEEE)
- b. A set of instructions that causes additional instructions to be loaded until the complete computer program is in storage. (ISO)
- c. A technique or device designed to bring itself into a desired state by means of its own action; for example, a machine routine whose first few instructions are sufficient to bring the rest of itself into the computer from an input device. (ANSI)
- d. That part of a computer program used to establish another version of the computer program. (ANSI)

### bootstrap loader

An input routine in which preset computer operations are used to load a bootstrap. (ISO)

### bottom-up

Pertaining to an approach that starts with the lowest level software components of a hierarchy and proceeds through progressively higher levels to the top level component; for example, bottom-up programming, bottom-up testing. (ANSI/IEEE) Contrast with top-down.

### bottom-up design

The design of a system starting with the most basic or primitive components and proceeding to higher level components that use the lower level ones. (ANSI/IEEE) Contrast with top-down design.

### bottom-up testing

Opposite to top-down testing. Firstly, the functions making up a module are tested individually. Then they are integrated to form a module and this is tested. After each module has been tested, the modules are integrated and, finally, the subsystem is tested. This sort of testing involves first testing the modules at lower levels in the hierarchy, and working up in the hierarchy of modules until the final module is tested. Drivers must be constructed for the lower level modules which present these modules with appropriate inputs. Bottom-up testing has the disadvantage that no demonstrably working program is available until the very last module has been tested.

### branch

In computer programming, a jump to another instruction not the instruction immediately following the branch instruction in the program as written. Branches may be conditional, depending on the result of a test, or unconditional.

**branch testing**

The most widely used measure of test completeness is branch coverage. A set of tests is considered complete if every program branch is traversed at least once during some tests. A branch is a transfer of control from one program statement to the next, either through sequential execution of statements or through the execution of a go-to. Complete branch testing is a necessary criteria for correctness in the sense that it is essential to test all branches in order to find all errors in the program. The use of branch testing requires the availability of a program testing tool that can be used to keep track of which branches are traversed during a sequence of tests.

**broadcast**

In a data bus system, the operation of the system in such a manner that information transmitted by the bus controller or a remote terminal is addressed to more than one of the remote terminals connected to the data bus.

**buffer**

- a. A routine or storage used to compensate for a difference in rate of flow of data, or time of occurrence of events, when transferring data from one device to another. (ADatP-2)
- b. In avionics, an isolating circuit used to prevent a driven circuit from influencing the driving circuit.
- c. Both a hardware and a software term. an intermediate location where data, e.g. in the form of records, is stacked up until it can be accepted by another hardware unit (such as the central processor), or by a procedure in a program.

**bug**

- a. A mistake or malfunction. (ADatP-2)
- b. A reference mark in a display.
- c. See fault.

**build**

An operational version of a software product incorporating a specified subset of the capabilities that the final product will include. (ANSI/IEEE)

**building block**

An individual unit or module that is utilized by higher-level programs or modules. (ANSI/IEEE)

**built-in test equipment (BITE)**

Any device which is a part of an equipment or system and is used for the express purpose of testing the equipment or system.

**built-in-test (BIT)**

A test approach using BITE (Built In Test Equipment) or self-test hardware or software to test all or part of a unit.

**bulk storage**

See secondary memory.

**bus**

See highway. (ADatP-2)

**bus controller**

In data bus systems, the terminal assigned the task of initiating information transfers on the data bus.

**bus monitor**

In data bus systems, the terminal assigned the task of receiving bus traffic and extracting selected information to be used at a later time.

byte

- a. A binary element string operated upon as a unit and usually shorter than a computer word.  
(ADatP-2)
- b. A group of binary digits (usually eight) addressable as a single unit.

C

A computer programming language which can be described as a middle-lever, having characteristics and capabilities of both high- and low-level languages. Those are capabilities available in assembly language, but rarely in high-level languages. The developers, Dennis Ritchie and Brian Kernighan, gave C the speed and flexibility of assembly language, and the data structures and control constructs, (such as if-then, for, which) of high-level languages.

cache

A small but very high speed memory buffer situated between the processor and the main memory in a computer system. It operates on the principle that certain memory locations tend to be accessed very often (normally for reads). Thus, when a main memory location is read, it is stored in the cache at the same time. Any further read references to this location are automatically routed to the cache. A write access usually writes to both main memory and the cache. Since a cache represents many non-continuous main memory locations, content-addressable registers are usually used to determine when a main memory location is currently stored in the cache.

case

A multi-branch conditional statement that allows for selective execution of bounded groups of program statements depending upon the value of a control expression. (ANSI/IEEE) See also control structure.



**central processing unit (CPU)**

- a. The part of a computer that controls the interpretation and execution of instructions. (RTCA)
- b. The heart or brain of a computer installation consisting of logic, arithmetic and control units, embodied in electronic circuitry.

**certification**

The process of confirming that a system, software subsystem, or computer program is capable of satisfying its specified requirements in an operational environment. Certification usually takes place in the field under actual conditions, and is utilized to evaluate not only the software itself, but also the specifications to which the software was constructed. Certification extends the process of verification and validation to an actual or simulated operational environment. (ANSI/IEEE)

**chain**

Items of data, e.g. in a table, each of which is connected to its successor, and perhaps to another members of the chain, by a pointer.

**chained list**

A list in which the items may be dispersed but in which each item contains an identifier for locating the next item to be considered. (ANSI)

**change control**

The process by which a change is proposed, evaluated, approved or rejected, scheduled, and tracked. (ANSI/IEEE)

**channel**

A special-purpose computer system devoted to handling input/output independently of the main processor of the computer system. A channel can access primary

storage directly to store or retrieve information. Three types of channels are selector, multiplexer, and block-multiplexer. Communication between a processor and a channel is ordinarily handled by polling or by interrupts.

#### chief programmer

The leader of a chief programmer team; a senior-level programmer whose responsibilities include producing key portions of the software assigned to the team, coordinating the activities of the team, reviewing the work of the other team members, and having an overall technical understanding of the software being developed. (ANSI/IEEE)

#### chief programmer team

A software development group that consists of a chief programmer, a backup programmer, a secretary/librarian, and additional programmers and specialists as needed, and that employs support procedures designed to enhance group communication and to make optimum use of each member's skills. (ANSI/IEEE)

#### chip

A piece of semiconductor material which performs a specific function(s).

#### chip set

A group of mutually related chips which perform integrated function(s).

#### cobol

Acronym for Common Business Oriented Language, a high-level programming language for commercial applications.

#### code (n)

- a. A set of unambiguous rules specifying the manner in which data may be represented in a

discrete form. Synonymous with coding scheme.  
(ADatP-2)

- b. In telecommunications, a set of rules and conventions according to which the signals representing data can be formed, transmitted, received, and processed. (ADatP-2)
- c. A set of items, such as abbreviations, representing the members of another set.  
(ADatP-2)

#### code (v)

- a. To represent data or a computer programme in a symbolic form that can be accepted by a data processor. (ADatP-2)
- b. To write a routine. (ADatP-2)
- c. To write or transcribe data in a format suitable for key-punching. (ADatP-2)

#### code audit

An independent review of source code by a person, team, or tool to verify compliance with software design documentation and programming standards. Correctness and efficiency may also be evaluated. (ANSI/IEEE)

#### code generation

After the syntax of a program has been analyzed, the last task of compilation is the generation of object code. When the parser recognizes a portion of a source program according to some rule of the grammar, the corresponding routine is executed. Such routines are often called semantic routines and they generate object code directly, so it is common to refer to them as code-generation routines. In more complex compilers, the semantic routines might generate an intermediate form of the program that would be analyzed further in an attempt to generate more efficient code.

**code generator**

A program or program function, often part of a compiler, that transforms a computer program from some intermediate level of representation (often the output of a parser) into a lower level representation such as assembly code or machine code. (ANSI/IEEE)

**coder**

A person who writes but does not usually design computer programmes. (ADatP-2)

**cohesion**

The degree to which the tasks performed by a single program module are functionally related. (ANSI/IEEE)  
Contrast with coupling.

**command and control systems**

Software application used to either generate vehicle commands or transmit these commands from the control center.

**command language**

A set of procedural operators with a related syntax, used to indicate the functions to be performed by an operating system. (ISO) Synonymous with control language.

**command response**

The mode of operation of a data bus system such that remote terminals receive and transmit data only when commanded to do so by the bus controller.

**comment**

Information embedded within a computer program, command language, or set of data that is intended to provide clarification to human readers and that does not affect machine interpretation. (ANSI/IEEE)

---

**common area**

An area used by more than one segment of a program, or more than one program.

---

**comparator**

A software tool used to compare two computer programs, files, or sets of data to identify commonalties or differences. Typical objects of comparison are similar versions of source code, object code, data base files, or test results. (ANSI/IEEE)

---

**compatibility**

The state which exists between two or more computers which can execute the same machine code instruction repertoire and drive the same peripherals. (ADH WP)

---

**compile**

- a. To prepare a machine language programme from a computer programme written in another programming language by making use of the overall logic structure of the programme, or generating more than one computer instruction for each symbolic statement, or both, as well as performing the function of an assembler. (ADatP-2)
- b. To translate a higher order language program into its relocatable or absolute machine language code equivalent. (ANSI/IEEE) Contrast with assemble, interpret.

---

**compiler**

- a. A programme that compiles. (ADatP-2)
- b. A computer program used to compile. (ISO) Contrast with assembler, interpreter. (ANSI/IEEE)

- c. A computer software that translates other programs written in high-level languages such as Fortran, Pascal, C etc., into machine code.

#### compiler generator

A translator or an interpreter that is used to construct compilers. (ISO) Synonymous with metacompiler.

#### complexity

The degree of complication of a system or system component, determined by such factors as the number and intricacy of interfaces, the number and intricacy of conditional branches, the degree of nesting, the types of data structures, and other system characteristics.  
(ANSI/IEEE)

#### complexity measure

A quantitative attribute of a computer program which measures its logical weight. Mathematically expressed via formula  $v(g)=e-n+2$  wherein (e) is the number of arcs and (n) is the number of nodes.

#### component

A basic part of a system or program. (ANSI/IEEE)

#### computer

A data processor that can perform substantial computation, including numerous arithmetic operations or logic operations, without intervention by a human operator during a run. (ADatP-2)

#### computer architecture

The structural and functional definition of a computer as viewed in terms of its machine instruction set and input/output capabilities. (GLOS-1)

**computer code**

A machine code for a specific computer. (ANSI)

**computer data**

Data available for communication between or within computer equipment. Such data can be external (in computer-readable form) or resident within the computer equipment and can be in the form of analog or digital signals. (ANSI/IEEE)

**computer instruction**

A machine instruction for a specific computer.  
(ADatP-2)

**computer network**

A complex consisting of two or more interconnected computers. (ADatP-2)

**computer program abstract**

A brief description of a computer program, providing sufficient information for potential users to determine the appropriateness of the computer program to their needs and resources. (ANSI/IEEE)

**computer programme**

A series of instructions or statements in a form acceptable to a computer, prepared in order to achieve a certain result. (ADatP-2)

**computer resources**

The totality of available and useful computer equipment, programs, documentation, services, supplies, and personnel. In an allocation sense "resources" is more specifically oriented to available memory and allowable running time to accomplish a specific job or project. (GLOS-1)

**computer system**

A functional unit, consisting of one or more computers and associated software, that uses common storage for all or part of a program and also for all or part of the data necessary for the execution of the program; executes user-written or user-designated programs; performs user-designated data manipulation, including arithmetic operations and logic operations; and that can execute programs that modify themselves during their execution. A computer system may be a standalone unit or may consist of several interconnected units. (ISO) Synonymous with ADP system, computing system.

**computer word**

A word stored in one computer location and capable of being treated as a unit. Synonymous with machine word. (ADatP-2)

**conciseness**

The compactness of the program in terms of lines of code.

**concurrent processes**

Processes that may execute in parallel on multiple processors or asynchronously on a single processor. Concurrent processes may interact with each other, and one process may suspend execution pending receipt of information from another process or the occurrence of an external event. (ANSI/IEEE) Contrast with sequential processes.

**conditional control structure**

A programming control structure that allows alternative flow of control in a program depending upon the fulfilment of specified conditions; for example, case, if... then... else... (ANSI/IEEE)

**configuration**

In avionics the arrangement of parts of a system; it may be in either a physical or logical sense.



**configuration audit**

The process of verifying that all required configuration items have been produced, that the current version agrees with specified requirements, that the technical documentation completely and accurately describes the configuration items, and that all change requests have been resolved. (ANSI/IEEE)

**configuration control**

The process of evaluating, approving or disapproving, and coordinating changes to configuration items after formal establishment of their configuration identification. (ANSI/IEEE)

**configuration control board**

The authority responsible for evaluating and approving or disapproving proposed engineering changes, and ensuring implementation of the approved changes. (ANSI/IEEE)

**configuration identification**

The process of designating the configuration items in a system and recording their characteristics. (ANSI/IEEE)

**configuration item**

A collection of hardware or software elements treated as a unit for the purpose of configuration management. (ANSI/IEEE)

**configuration management**

The process of identifying and defining the configuration items in a system, controlling the release and change of these items throughout the system life cycle, recording and reporting the status of configuration items and change requests, and verifying the completeness and correctness of configuration items. (ANSI/IEEE) See also change control, configuration

identification, configuration control, configuration status accounting, configuration audit.

### configuration status accounting

The recording and reporting of the information that is needed to manage a configuration effectively, including a listing of the approved configuration identification, the status of proposed changes to the configuration, and the implementation status of approved changes. (MIL-STD-973)

### confinement

- a. Prevention of unauthorized alteration, use, destruction, or release of data during authorized access. (ANSI/IEEE) See also integrity.
- b. Restriction on programs and processes so that they do not access or have influence on data, programs, or processes other than that allowed by specific authorization. (ANSI/IEEE)

### connection

A reference in one part of a program to the identifier of another part (that is, something found elsewhere). (ANSI/IEEE)

### consistency

The use of uniform design and documentation techniques throughout the software development project.

### control data

Data used to influence the operation of a routine either by selecting it or by modifying it. (ADatP-2)

### control display unit

In avionics, a unit which is used to interface a given sub-system or system to an aircrew member. The

unit displays information and is used to enter commands and data into the sub-system/system.

#### control statement

A programming language statement that affects the order in which operations are performed. (ANSI/IEEE)

#### control structure

A construct that determines the flow of control through a computer program. (ANSI/IEEE)

#### conversational

Pertaining to an interactive system that provides for interaction between a user and a system similar to a human dialog. (ANSI/IEEE)

#### conversion

Modification of existing software to enable it to operate with similar functional capability in a different environment; for example, converting a program from FORTRAN to Ada, converting a program that runs on one computer to run on another computer. (ANSI/IEEE)

#### core avionics

That portion of an avionics system that is applicable to a wide range of operational missions.

#### core store

The memory store of a central processing unit, consisting in most present-day machines of ferrite magnetic cores.

#### coroutines

Two or more modules that can call each other, but that are not in a superior to subordinate relationship. (ANSI/IEEE)

**corrective maintenance**

Maintenance performed specifically to overcome existing faults. (ISO)

**correctness**

- a. The extent to which software is free from design defects and from coding defects; that is, fault free. (ANSI/IEEE)
- b. The extent to which software meets its specified requirements. (ANSI/IEEE)
- c. The extent to which software meets user expectations. (ANSI/IEEE)

**coupling**

- a. A measure of the interdependence among modules in a computer program. (ANSI/IEEE) Contrast with cohesion.
- b. Coupling is a measure of program or system complexity. It describes how an innocent change in one module causes a totally unwarranted problem in another module. Module high interdependent are said to be tightly coupled, which is a undesirable characteristics in structured design.

**critical design review**

A formal review of the contractor's detailed design prior to determine whether the design adequately addresses all of the performance/interface requirements of the CI (Configuration Item). The CDR should be a formal technical review of the design as depicted by the specification and flow diagrams, sufficiently detailed to enable the programmer to code, compile, and debug a computer program, to assure that design requirements have been met before beginning coding. It is held normally prior to fabrication/production design release. Note: Successful completion of the CDR does not constitute customer agreement that the design will meet the specification requirements for a computer program.

**critical module**

A module that has one or more of the following characteristics.

- a. Address several requirements.
- b. Has a high level of control (resides relatively high in the program structure).
- c. Is complete or error prone.
- d. Has definite performance requirements. Critical modules should be tested as early as possible. In addition regression testing should focus on critical module function.

**critical piece first**

Pertaining to an approach to software development that focuses on implementing the most critical aspects of a software system first. The critical piece may be defined in terms of services provided, degree of risk, difficulty, or some other criterion. (ANSI/IEEE)

**critical section**

A segment of code to be executed mutually exclusively with some other segment of code that is also called a critical section. Segments of code are required to be executed mutually exclusively if they make competing uses of a computer resource or data item. (ANSI/IEEE)

**criticality**

A classification of a software error or fault based upon an evaluation of the degree of impact of that error or fault on the development or operation of a system (often used to determine whether or when a fault will be corrected). (ANSI/IEEE)

---

**cross-assembler**

An assembler that executes on one computer but generates object code for a different computer. (ANSI/IEEE)

---

**cross-compiler**

A compiler that executes on one computer but generates assembly code or object code for a different computer. (ANSI/IEEE)

---

**data**

- a. A representation of facts, concepts, or instructions in a formalized manner suitable for communications, interpretation, or processing by humans or by automatic means. (ADatP-2)
- b. Any representations such as characters or analog quantities to which meaning is or might be assigned. (ADatP-2)

---

**data abstraction**

The result of extracting and retaining only the essential characteristic properties of data by defining specific data types and their associated functional characteristics, thus separating and hiding the representation details. (ANSI/IEEE)

---

**data base**

- a. A set of data, part of the whole of another set of data, and consisting of at least one file that is sufficient for a given purpose or for a given data processing system. (ISO/ADatP-2)
- b. A collection of data fundamental to a system. (ANSI/IEEE)
- c. A collection of data fundamental to an enterprise. (ANSI/IEEE)

- d. A structured organizing of large quantities of data which may be accessed by different programs using different subsets of the data.

#### data bus

- a. In general terms, a bus which is functionally intended to transfer information from one point to another.
- b. In a STANAG 3838 AVS system, all the hardware including twisted shielded pair cables, isolation resistors, transformers, etc., required to provide a data path between the bus controller and all the associated remote terminals.

#### data dictionary

- a. A collection of the names of all data items used in a software system, together with relevant properties of those items; for example, length of data item, representation, etc. (ANSI/IEEE)
- b. A set of definitions of data flows, data elements, files, data bases, and processes referred to in a levelled data flow diagram set. (ANSI/IEEE)

#### data domain

An approach to obtaining an estimate of operational reliability. In principal, if all sets of input data values upon which the computer program must operate are identified, an estimate of the reliability of the program could be obtained by running the program for all such possible sets. In practice a methodology is used to select sample data sets from the total of such sets which are representative of intended operational usage and the program is run for those sets only. The results of the runs on the sample data sets are used to compute reliability estimates for the program in its intended operational environment. (DAN 238)

**data driven design**

A software design derived from an analysis of the input and output system data.

**data flowchart**

A flowchart representing the path of data through a problem solution. It defines the major phases of the processing as well as the various data media used. Synonymous with data flow diagram. (ANSI)

**data structure**

A formalized representation of the ordering and accessibility relationships among data items without regard to their actual storage configuration. (ANSI/IEEE)

**data type**

A class of data characterized by the members of the class and the operations that can be applied to them; for example, integer, real, logical. (ANSI/IEEE)

**deadlock**

A situation that occurs when all tasks within a system are suspended waiting for resources that have already been assigned to other tasks that are also waiting for additional resources. Thus, the system can perform no useful work unless tasks are destroyed and their resources reclaimed.

**debug**

To detect, locate, and remove mistakes from a routine or malfunctions from a computer. (ADatP-2)

**debugging**

- a. Testing of programs under development by locating and removing the "bugs" or error in them.



- b. The process of locating, analyzing, and correcting suspected faults. (IEEE) Compare with testing.

---

**decision table**

- a. The representation in tabular form of a series of tests to be applied to data. Decision tables are not only a convenient substitute for flowcharts in many cases, but can also be converted into programming language statements by special preprocessor programs.
- b. A table of all contingencies that are to be considered in the description of a problem, together with the actions to be taken. Decision tables are sometimes used in place of flowcharts for problem description and documentation. (ANSI)

---

**delivery**

- a. The point in the software development cycle at which a product is released to its intended user for operational use. (ANSI/IEEE)
- b. The point in the software development cycle at which a product is accepted by its intended user. (ANSI/IEEE)

---

**demand paging**

A paging algorithm in a virtual storage system where pages are not loaded into memory until a reference is made to that page.

---

**design**

- a. The process of defining the software architecture, components, modules, interfaces, test approach, and data for a software system to satisfy specified requirements. (ANSI/IEEE)
- b. The result of the design process. (ANSI/IEEE)

---

**design analysis**

- a. The evaluation of a design to determine correctness with respect to stated requirements, conformance to design standards, system efficiency, and other criteria.  
(ANSI/IEEE)
- b. The evaluation of alternative design approaches. (ANSI/IEEE) See also preliminary design.

---

**design analyzer**

An automated design tool that accepts information about a program's design and produces such outputs as module hierarchy diagrams, graphical representations of control and data structure, and lists of accessed data blocks. (ANSI/IEEE)

---

**design language**

A language with special constructs and, sometimes, verification protocols used to develop, analyze, and document a design. (ANSI/IEEE)

---

**design methodology**

A systematic approach to creating a design, consisting of the ordered application of a specific collection of tools, techniques, and guidelines.  
(ANSI/IEEE)

---

**design phase**

The period of time in the software life cycle during which the designs for architecture, software components, interfaces, and data are created, documented, and verified to satisfy requirements. (ANSI/IEEE)

---

**design requirement**

Any requirement that impacts or constrains the design of a software system or software system component; for example, functional requirements, physical requirements, performance requirements, software

development standards, software quality assurance and standards. (ANSI/IEEE)

### design review

A formal meeting at which the preliminary or detailed design of a system is presented to the user, customer, or other interested parties for comment and approval. (ANSI/IEEE)

### design specification

A specification that documents the design of a system or system component; for example, a software configuration item. Typical contents include system or component algorithms, control logic, data structures, data set-use information, input/output formats, and interface descriptions. (ANSI/IEEE) See also requirements specification.

### desk checking

The manual simulation of program execution to detect faults through step-by-step examination of the source code for errors in logic or syntax. (ANSI/IEEE)

### desk-checking

"Dry-running" or "Playing computer": the testing of a program without using the computer, by applying test data and checking the result of each instruction when performed on it.

### detailed design

- a. The process of refining and expanding the preliminary design to contain more detailed descriptions of the processing logic, data structures, and data definitions, to the extent that the design is sufficiently complete to be implemented. (ANSI/IEEE)
- b. The result of the detailed design process. (ANSI/IEEE)

development methodology

A systematic approach to the creation of software that defines development phases and specifies the activities, products, verification procedures, and completion criteria for each phase. (ANSI/IEEE)

device

A unit of peripheral hardware, or a piece of system software usually for Input and/or Output.

device driver

Another term for a driver.

diagnostic

Pertaining to the detection and isolation of a malfunction or mistake. (ADatP-2)

diagnostic aid

A routine which assists the debugging of a program under test by, for example, printing out the contents of specified store locations during the running of the program, or when it fails.

digit

A graphic character that represents an integer. Example: one of the characters 0 to 9. (ADatP-2)

digital

Pertaining to digits or to the representation of data or physical quantities by digits. (ADatP-2)

digital computer

- a. A computer in which discrete representation of data are mainly used. (ADatP-2)

- b. A computer that operates on discrete data by performing arithmetic and logic processes on these data. (ADatP-2)

### digital-to-analogue (D/A)

The conversion of digital data to an analogue signal.

### direct access

- a. Pertaining to the process of obtaining data from, or placing data into, storage where the time required for such access is independent of the location of the data most recently obtained or placed in storage. (ADatP-2)
- b. Pertaining to a storage device in which the access time is effectively independent of the location of the data. (ADatP-2)

### direct memory access (DMA)

A method used to supply memory locations with data, or extract data from memory locations, without CPU intervention.

### directed graph

A graph whose edges are unidirectional. (ANSI/IEEE)

### discrete signal

A signal that takes only two states to represent any single function or piece of information.

### disk

A magnetic storage device on which data is recorded on a number of concentric, circular tracks. A disk unit contains one or more disks that may be either interchangeable, when they can be removed and replaced, or fixed.

---

**dispatcher**

Another term for a schedule.

---

**distributed architecture**

A system architecture in which the functions are partitioned into physically separate units.

---

**distributed processing**

- a. A method of operation in which routine processing is carried out locally by satellite computers and only those data which are required to be processed centrally need to be transmitted to the central computer facility. (ADatP-2)
- b. In avionics, processing accomplished by a system in which the computing functions are partitioned among several elements of the system.

---

**documentation**

- a. The creating, collecting, organizing, storing, citing, and disseminating of documents or the information recorded in documents. (ANSI)
- b. A collection of documents or information on a given subject. (ANSI)
- c. Typical documentation might consists of: the analysis of the problem by a system analyst; flowcharts of the program written to solve it; written descriptions of the program; and printed computer listings.

---

**domain testing**

A method of reliably determining a set of test data which will bring 'domain' and boundary errors to light. It involves analysis of the program by 'mathematical methods' involving hypergeometry.

---

**driver**

- a. A system software module that directly controls the data transfer to and from I/O peripherals.
- b. A software module used in bottom-up testing to make modules ready for testing.
- c. A program than exercises a system or system component by simulating the activity of a higher level component. (ANSI/IEEE)

---

**dual coding**

A development technique in which two functionally identical versions of a program are developed from the same specification by different programmers or different programming teams. The resulting source code may be in the same or different languages. The purpose of dual coding is to provide for error detection, increase reliability, provide additional documentation, or reduce the probability of systematic programming errors or compiler errors influencing the end result. (ANSI/IEEE)

---

**dump**

- a. Data that have been dumped. (ISO)
- b. To write the contents of a storage, or of part of a storage, usually from an internal storage to an external medium, for specific purpose such as to allow other use of the storage, as a safeguard against faults or errors, or in connection with debugging. (ISO)
- c. A dump is the periodical transferring of information from the computer memory to a peripheral medium, or from one medium to another. It preserves the current state of a program and its data.

---

**duplex operation**

Duplex (or full duplex) operation refers to communication between two points in both directions simultaneously. (ADatP-2)

---

**dynamic allocation**

The allocation of addressable storage and other resources to a program while the program is executing. (ANSI/IEEE)

---

**dynamic analysis**

The process of evaluating a program based on execution of the program. (ANSI/IEEE) Contrast with static analysis.

---

**dynamic analyzer**

A software tool that aids in the evaluation of a computer program by monitoring execution of the program. Examples include instrumentation tools, software monitors, and tracers. (ANSI/IEEE) Contrast with static analyzer.

---

**dynamic binding**

Binding performed during execution of a program. (ANSI/IEEE) Contrast with static binding.

---

**dynamic code analysis**

See dynamic analysis

---

**dynamic priority**

A form of scheduling in that task's execution priority varies depending on the system environment. For example, it is common for time-sharing systems to initially give large tasks low priority, but raise the priority periodically as the task waits to execute. This assures that low priority tasks will not be locked out of execution time by small, high priority tasks.



**dynamic restructuring**

- a. The process of changing software components or structure while a system is running.  
(ANSI/IEEE)
- b. The process of restructuring a data base or data structure during program execution.  
(ANSI/IEEE)

**dynamic storage allocation**

The allocation of storage, such as working areas, to a program only when it is needed, and not permanently throughout the running of the program.

**editor**

A computer program that permits selective revision of computer stored data. (ANSI/IEEE)

**efficiency**

The extent to which software performs its intended functions with a minimum consumption of computing resources. (ANSI/IEEE)

**egoless programming**

An approach to software development based upon the concept of team responsibility for program development. Its purpose is to prevent the programmer from identifying so closely with his or her output that objective evaluation is impaired. (ANSI/IEEE)

**electro-optical**

Dealing with the translation of electromagnetic radiation in the visible and infrared portion of the spectrum into a format suitable for electronic application and vice versa.

**electronic attitude director indicator**

A display that provides attitude and director information using a cathode ray tube (CRT) or light-emitting matrix. Other information, such as altitude, angle of attack, airspeed, etc., also may be displayed. Various modes, such as take-off, cruise, weapon delivery, landing, etc., can usually be selected.

**embedded computer system**

A computer system that is integral to a larger system whose primary purpose is not computational; for example, a computer system in a weapon, aircraft, command and control, or rapid transit system. (ANSI/IEEE)

**embedded software**

Software for an embedded computer system.  
(ANSI/IEEE)

**emulation**

The imitation of all or part of one computer system by another, primarily by hardware, so that the imitating computer system accepts the same data, executes the same programs, and achieves the same results as the imitated system. (ISO)

**emulator**

Software run on a host computer that accepts the same input data, executes the same programs and yields the same outputs as the target computer. (RTCA)

**encoding**

A design and programming technique through which information is stored or conveyed by means of a reversible mapping from the domain in which the information exists originally into another domain.  
(ABBOTT)

**entry point**

The first programmer defined instruction obeyed in a program when it is activated. There may be more than one entry point to cover different modes of running the program.

**environment**

The universe within which the system must operate. All the elements over which the designer has no control and that affect the system or its inputs and outputs.

**error**

Any discrepancy between a computed, observed, or measured quantity and the true, specified, or theoretically correct value or condition. (ANSI)

**error analysis**

The process of investigating an observed software fault to identify such information as the cause of the fault, the phase of the development process during which the fault was introduced, methods by which the fault could have been prevented or detected earlier, and the method by which the fault was detected. (ANSI/IEEE)

**error category**

One of a set of classes into which an error, fault, or failure might fall. Categories may be defined for the cause, criticality, effect, life cycle phase when introduced or detected, or other characteristics of the error, fault, or failure. (ANSI/IEEE)

**error data**

A term commonly (but not precisely) used to denote information describing software problems, faults, failures, and changes, their characteristics, and the conditions under which they are encountered or corrected. (ANSI/IEEE)

**error model**

A mathematical model used to predict or estimate the number of remaining faults, reliability, required test time, or similar characteristics of a software system. (ANSI/IEEE)

**error prediction**

A quantitative statement about the expected number or nature of software problems, faults, or failures in a software system. (ANSI/IEEE)

**error procedures**

Procedures in a program that are obeyed when an error, such as invalid data, is detected by the program.

**exception**

An event that causes suspension of normal program execution. (ANSI/IEEE)

**executable object code**

The binary form of code instructions which is directly usable by the CPU. (RTCA)

**execution**

The process of carrying out an instruction or the instructions of a computer program by a computer. (ISO)

**execution time**

The time taken to complete the cycle of events required to perform an instruction, routine or program. (ADatP-2)

**execution time theory**

A theory that uses cumulative execution time as the basis for estimating software reliability. (ANSI/IEEE)

**executive routine**

A routine that controls the execution of other routines. Synonymous with supervisory routine. (ADatP-2)

**exit**

Any instruction in a computer program, in a routine, or in a subroutine, after the execution of which control is no longer exercised by that computer program, that routine, or that subroutine. (ISO)

**expendability**

The degree to which architectural, data or procedural design can be extended.

**expert system**

A program that has a wide base of knowledge in a restricted domain, and uses complex inferential reasoning to perform tasks which a human expert could do.

**external function testing**

The verification of the external system as stated in the external specifications.

**failure**

- a. The termination of the ability of a functional unit to perform its required function. (ISO)
- b. The inability of a system or system component to perform required function within specified limits. A failure may be produced when a fault is encountered. (ANSI/IEEE)
- c. A departure of program operation from program requirements. (ANSI/IEEE)

**failure rate**

The ratio of the number of failures to a given unit of measure; for example, failures per unit of time, failures per number of transactions, failures per number of computer runs. (ANSI/IEEE)

**failure recovery**

The return of a system to a reliable operating state after failure. (ANSI/IEEE)

**fault**

- a. An accidental condition that causes a functional unit to fail to perform its required function. (ISO)
- b. A manifestation of an error in software. A fault, if encountered, may cause a failure. (ANSI/IEEE) Synonymous with bug.

**fault seeding**

The process of intentionally adding a known number of faults to those already in a computer program for the purpose of estimating the number of indigenous faults in the program. (ANSI/IEEE) Synonymous with bug seeding.

**fault tolerance**

The built-in capability of a system to provide continued correct execution in the presence of a limited number of hardware or software faults. (ANSI/IEEE)

**federated architecture**

A computer system architecture in which two or more computers execute tasks under the loose control of a centralized executive programme.

**fibre optics**

The technology by which glass (or other medium) fibres are used to transmit light energy.

**field**

An element in, or part of a record.

**field testing**

The initial operation of the actual hardware-software system in the field in a test mode (limited or full capabilities) to ferret out as many remaining errors as is feasible.

**file**

- a. A collection of records. A personnel file, for example, would contain information about all the individuals in an establishment. The information about each individual would constitute his/her record; such information might include name, age, marital status, salary, grade and so on, and such of these items would constitute a field in his record.
- b. A set of related records treated as a unit.  
(ISO)

**finite state machine**

A computational model consisting of a finite number of states, and transitions between these states.  
(ANSI/IEEE)

**firmware**

Information stored in a read only memory which can be changed by physically reprogramming or replacing the device, but which cannot be changed in the normal operation of the computer.

**first fit**

An algorithm for memory allocation that searches the free list only long enough to find an unused memory block that is large enough to satisfy the requesting task.

**first in-first out (FIFO)**

A data structure in which only the least recent entry is available for access.

**flag**

- a. Any of various types of indicators used for identification, e.g., a wordmark. (ANSI)
- b. A character that signals the occurrence of some condition, such as the end of a word. (ANSI)

**floating-point**

A form of arithmetic in which a number is expressed as an exponent and a mantissa, and the point position of the number is determined by raising the mantissa to the power represented by the exponent.

**flow of control**

The sequence of operations performed in the execution of an algorithm. (ANSI/IEEE)

**flowchart**

A graphical representation for the definition, analysis or solution of a problem, in which symbols are used to represent operations, data, flow, equipment, etc. Contrast with block diagrams. (ANSI)

**foreground/background**

A type of operating system with a combination of real time and either batch or multitasking capabilities. This type of system is used to permit time critical programs to operate in the foreground and execute with



high priority while background assemblies, edits, etc., are also going on at much lower priority. Foreground programs always have priority over background work. This type of operating system usually has the capability to protect the foreground from the background from the background, and is used in applications such as process control.

#### formal language

A language whose rules are explicitly established prior to its use. Synonymous with artificial language. Examples include programming languages, such as FORTRAN and Ada, and mathematical or logical languages, such as predicate calculus. (ANSI/IEEE) Contrast with natural language.

#### formal parameter

A variable used in a subprogram to represent data or program elements to be transmitted to the subprogram by a calling routine. Synonymous with dummy parameter. (ANSI/IEEE) Contrast with actual parameter.

#### formal qualification review

A formal review, normally accomplished incrementally at the procuring facility, of tests reports and test data generated during the formal qualification of a new configuration item (CI) to assure that all test required by the quality assurance provisions of the development specification(s) have been accomplished, and that the CI performs as required by the requirements of the development specification(s).

#### formal specification

- a. A specification written and approved in accordance with established standards.  
(ANSI/IEEE)
- b. In proof of correctness, a description in a formal language of the externally visible behavior of a system or system component.  
(ANSI/IEEE)

**formal testing**

The process of conducting testing activities and reporting results in accordance with an approved test plan. (ANSI/IEEE)

**fortran**

Acronym for Formula Translation, a high-level computer programming language for defining and solving mathematical and scientific problems.

**fragmentation**

The leftover sections of memory which are too small to fill an allocation request. The division of contiguous storage area such as main memory or secondary storage in a way that causes areas to be wasted. For instance, in most paged systems, whole pages are always allocated, even when a program or data segment is smaller than the page size. In segmented systems or in a system with dynamic memory allocation, continually loading and swapping new tasks will always cause small areas of memory to become unusable.

**free**

An operating system service that returns allocated memory blocks to the memory manager for reuse.

**free list**

A list of memory locations that are currently unused and may be allocated by the memory manager to requesting tasks. Free lists are usually organized as linked lists of memory blocks, where each block contains the size of the block and a pointer to the next block in the list.

**front porch**

The portion of a composite video signal that lies between the leading edge of the horizontal blanking pulse and the leading edge of the corresponding synchronizing pulse.

---

**function**

- a. A specific purpose of an entity or its characteristic action. (ANSI)
- b. A subprogram that is invoked during the evaluation of an expression in which its name appears and that returns a value to the point of invocation. (ANSI/IEEE)
- c. flight critical A function for which the occurrence of any failure condition or design error would prevent the continued safe flight and landing of the aircraft.
- d. mission critical A function for which the occurrence of any failure condition or design error would prevent the aircraft crew from commencing, continuing, or completing the assigned mission.
- e. flight essential A function for which the occurrence of any failure condition or design error would reduce the capability of the aircraft or the ability of the crew to cope with adverse operating conditions.
- f. mission essential A function for which the occurrence of any failure condition or design error would degrade the capability of the aircraft or reduce crew options for commencing, continuing, or completing the assigned mission.
- g. other A function for which failures or design errors could not significantly degrade aircraft capability or reduce crew options under any circumstance.

---

**functional baseline**

The Functional Baseline is normally established by the authenticated system/segment specification at the end of the conceptual phase.

---

**functional decomposition**

A method of designing a system by breaking it down into its components in such a way that the components

correspond directly to system functions and subfunctions.  
(ANSI/IEEE)

### functional design

The specifications of the working relations between the parts of a system in terms of their characteristic actions. (ANSI)

### functional requirement

A requirement that specifies a function that a system or system component must be capable of performing.  
(ANSI/IEEE)

### functional specification

A specification that defines the functions that a system or system component must perform. (ANSI/IEEE)

### functional test

In the vast majority of cases, the first type of computer-assisted testing performed is functional test. The goal of this testing is to verify that the program does what it is supposed to do. Normally, the person who has created a program does functional testing. In functional testing, test cases are derived through analysis of the software specification. Functional testing is essentially the traditional approach to testing practiced in a disciplined and systematic manner.

### functional unit

An entity of hardware, software, or both capable of accomplishing a specified purpose. (ISO)

### general purpose computer

A computer that is designed to operate upon a wide variety of problems.

**generator**

A generator produces test data or test cases to exercise the target system. A generator in this case is differentiated from a simulator because it actually creates test data using numerical integrators, random number generators, etc. Once the data are produced by the generator, a simulator might be required to route the data to the system. Generators are useful in a system test environment where "live" data is not available. Useful output of a data generator are tapes of logged data that can be used with a data replay facility for establishing standard test cases. (DAN 134)

**graceful degradation**

A system design objective whereby operating capability is maintained, although with degraded performance, in the face of progressive failure of components.

**graph**

A model consisting of a finite set of nodes having connections called edges or arcs. (ANSI/IEEE)

**ground support equipment (GSE)**

Equipment required to maintain an aircraft and its associated equipment.

**half duplex operation**

Operating method using a circuit designed for duplex operation, but which, on account of the terminal equipment, can be operated alternately only. (ADatP-2)

**handler**

Another term for a driver.

**hardware**

- a. In avionics, physical equipment, as differentiated from the computer programme or

method of use, e.g., mechanical, magnetic, electrical or electronics devices.

- b. The physical machinery in a computer installation.

### head

The part of a magnetic recording device which senses the bit patterns in magnetic fields, on a tape or disk or drum for example, or records such fields. Most heads can both read and write, that is both sense and record.

### head down display

A display of flight, navigation, attack or other information which is located within the cockpit area and requires the aircrew to look inside the cockpit.

### head-up display

A display of flight, navigation, attack, or other information superimposed upon the pilot's forward field of view. (AAP-6)

### helmet mounted display

A display of flight, navigation, attack, or other information on a unit which is attached to the aircrew's helmet.

### helmet mounted sight

A display of simple aiming information on a unit which is attached to the aircrew's helmet.

### heuristic method

Any exploratory method of solving problems in which an evaluation is made of the progress towards an acceptable final solution using a series of approximate results.

---

**hierarchical decomposition**

A method of designing a system by breaking it down into its components through a series of top-down refinements. (ANSI/IEEE)

---

**hierarchical input-process-output**

A design and documentation technique based on development of a top-down hierarchy of system functional elements and Input-Process-Output (IPO) diagrams for each of these elements. The HIPO technique provides an organization chart of the system and a description of the characteristic of the software elements that constitute the system.

---

**hierarchy**

A structure whose components are ranked into levels of subordination according to a specific set of rules. (ANSI/IEEE)

---

**high(er) order language**

- a. A language which allows algorithms and data structures to be represented in a symbolic, machine-independent, form with high information (as opposed to data) content and hence more easily understood by humans than machine or assembly language.
- b. A programming language that usually includes features such as nested expressions, user defined data types, and parameter passing not normally found in lower order languages, that does not reflect the structure of any one given computer or class of computers, and that can be used to write machine independent source programs. A single, higher-order, language statement may represent multiple machine operations. Contrast with machine language, assembly language. (ANSI/IEEE)

---

**highway**

A major channel along which signals travel from one of several sources to one of several destinations. (ADatP-2)

**host computer**

A computer used to prepare programmes for use on another computer.

**identifier**

A symbol whose purpose is to identify, indicate or name a body of data. (ANSI)

**imperfect debugging**

In reliability modelling, the assumption that attempts to correct or remove a detected fault are not always successful. (ANSI/IEEE)

**implementation**

- a. A realization of an abstraction in more concrete terms; in particular, in terms of hardware, software, or both. (ANSI/IEEE)
- b. A machine executable form a program, or a form of a program that can be translated automatically to machine executable form. (ANSI/IEEE)
- c. The process of translating a design into code and debugging the code. (ANSI/IEEE)

**implementation phase**

The period of time in the software life cycle during which a software product is created from design documentation and debugged. (ANSI/IEEE)

**implementation requirement**

Any requirement that impacts or constrains the implementation of a software design; for example, design descriptions, software development standards, programming language requirements, software quality assurance standards. (ANSI/IEEE)



**independent verification and validation**

Verification and validation of a software product by individuals or groups other than those who performed the original design, but who may be from the same organization. The degree of independence must be a function of the importance of the software. (ANSI/IEEE)

**index register**

A register whose content may be added to or subtracted from the operand address prior to or during the execution of a computer instruction.

**indigenous fault**

A fault existing in a computer program that has not been inserted as part of a fault seeding process. (ANSI/IEEE)

**inductive assertion method**

A proof of correctness technique in which assertions are written describing program inputs, outputs, and intermediate conditions, a set of theorems is developed relating satisfaction of the input assertions to satisfaction of the output assertions, and the theorems are proved to be true. (ANSI/IEEE)

**information bits**

In telecommunications, those bits which are generated by the data source and which are not used for error control by the data transmission system.

**information hiding**

The technique of encapsulating software design decisions in modules in such a way that the modules' interfaces reveal as little as possible about the modules' inner workings; thus, each module is a "black box" to the other modules in the system. The discipline of information hiding forbids the use of information

about a module that is not in the module's interface specification. (ANSI/IEEE)

### input assertion

A logical expression specifying one or more conditions that program inputs must satisfy in order to be valid. (ANSI/IEEE)

### inspection

- a. A formal evaluation technique in which software requirements, design, or code are examined in detail by a person or group other than the author to detect faults, violations of development standards, and other problems. (ANSI/IEEE) Contrast with walk-through.
- b. An activity that involves the review of specific documents.

### installation and check-out phase

The period of time in the software life cycle during which a software product is integrated into its operational environment and tested in this environment to ensure that it performs as required. (ANSI/IEEE)

### installation testing

The validation of each particular installation of the system with the intent of pointing out any errors made while installing the system.

### instruction

A statement that specifies an operation and the values or locations of its operands. (ADatP-2) See also computer instructions and machine instruction.

### instruction set

The repertoire of instructions for a given computer.

---

**instruction set architecture (ISA)**

The attributes of a digital computer as seen by a programmer using machine or assembly language. (ISA includes the processor, input/output, instruction sets, their formats, operating codes, addressing modes, memory management and partitioning (if accessible to the programmer), the speed of accessible clocks, interrupt structure, and the manner of use and format of all registers and memory locations that may be directly manipulated or tested. This definition excludes the time or speed of any operation, internal computer partitioning, electrical and physical organization, circuits or components of the computer, manufacturing technology, memory organization, memory cycle time and memory bus widths.)

---

**instrumentation tool**

A software tool that generates and inserts counters or other probes at strategic points in another program to provide statistics about program execution, such as how thoroughly the program's code is exercised. (ANSI/IEEE)

---

**integration**

See system integration.

---

**integration testing**

- a. An orderly progression of testing in which software elements, hardware elements, or both are combined and tested until the entire system has been integrated. (ANSI/IEEE)
- b. The verification of the interfaces among system parts (modules, components, and subsystems). The progress of integration testing is often measured by the number of instructions coded and tested, or the number of functions or modules implemented and tested. The measure of progress should be supplemented by the use of reliability models which estimate and project the mean time between software failures (MTBF). Integration testing is followed by either simulation testing or initial field testing, which can be viewed as the ultimate integration test. Integration testing is a systematic technique for constructing the program

structure while at the same time conducting tests to uncover errors associated with interfacing. The objective is to take unit-tested modules and build a program structure that has been dictated by design. As integration testing is conducted, the tester should identify critical modules.

### integrity

The extent to which unauthorized access to or modification of software or data can be controlled in a computer system. (ANSI/IEEE)

### interactive

Pertaining to a system in which each user entry causes a response from the system. (ANSI/IEEE)

### interblock gap

The spaces between blocks of data recorded, for example on magnetic tape.

### interface

- a. A boundary or common point to two or more entities through which information flow takes place. (ADatP-2)
- b. In avionics, the means by which functional elements are connected. May be physical, electrical, mechanical or functional and may be used to pass information.
- c. A shared boundary. An interface might be a hardware component to link two devices or it might be a portion of storage or registers assessed by two or more computer programs (ANSI)
- d. To interact or communicate with another system component. (ANSI/IEEE)
- e. Originally a hardware term referring to the control circuitry between two hardware units.

Now applies also to software, referring, for instance, to the parameters passed between a program and its subroutines.

### interface requirement

A requirement that specifies a hardware, software, or data base element with which a system or system component must interface, or that sets forth constraints on formats, timing, or other factors caused by such an interface. (ANSI/IEEE)

### interface specification

A specification that sets forth the interface requirements for a system or system component. (ANSI/IEEE)

### interface testing

Testing conducted to ensure that program or system components pass information or control correctly to one another. (ANSI/IEEE)

### interleaving

Inserting segments of one program into another program sequentially so that the two programs can, in effect, be executed simultaneously. (Computer Dictionary)

### interoperability

The ability of two or more systems to exchange information and to mutually use the information that has been exchanged. (ANSI/IEEE)

### interpret

To translate and to execute each source language statement of a computer program before translating and executing the next statement. (ISO) Contrast with assemble, compile.

---

**interpreter**

- a. A computer programme that translates and executes each source language statement before translating and executing the next one. (ADatP-2)
- b. A device that prints on a punched card that data already punched in the card. (ADatP-2)

---

**interrupt (n)**

An event that alters the sequence in which a processor executes instructions. It is generated by the hardware of the computer system. When an interrupt occurs, the operating system

- a. Gains control,
- b. Saves the state of the interrupted process. In many systems this information is stored in the interrupted process's process control block (PCB),
- c. Analyze the interrupts and passes control to the appropriate routine to handle the interrupt.

There are six interrupt types. These are.

- (1) SVC (supervisor call) interrupts.
- (2) I/O interrupt.
- (3) External interrupts.
- (4) Restart interrupts.
- (5) Program check interrupts.
- (6) Machine check & timer interrupts.

---

**interrupt (v)**

To stop a process in such a way that it can be resumed. (ANSI)

**iteration**

- a. The process of repeatedly executing a given sequence of programming language statements until a given condition is met or while a given condition is true. (ANSI/IEEE)
- b. A single execution of a loop. (ANSI/IEEE)

**JCL**

An abbreviation for Job Control Language. A special computer command language designed for use in batch systems to inform the systems software and computer operator of unique requirements for the running of a computer program.

**job**

A collection of tasks, grouped and run together in order to perform a specific function.

**jump**

See BRANCH

**karnaugh map**

A rectangular diagram of a logic function of variables drawn with overlapping sub-rectangles such that each intersection of the over-lapping rectangles represents a unique combination of the logic variables and such that an intersection is shown for all combinations.

**kernel**

- a. A nucleus or core, as in the kernel of an operating system. (ANSI/IEEE)
- b. An encapsulation of an elementary function. Kernels can be combined to form some or all of

an operating system or set of firmware.  
(ANSI/IEEE)

- c. A model used in computer selection studies to evaluate computer performance. (ANSI/IEEE)

### key

One or more characters, within a set of data, that contains information about the set, including its identification. (ISO)

### kilo (K)

Used to express the number 210 (1024) in digital terminology (e.g., 16K words of memory = 16,384 words).

### label

- a. One or more characters, within or attached to a set of data, that contain information about the set, including its identification. (ISO)
- b. An identification record for a tape or disk file. (ANSI/IEEE)
- c. In computer programming, an identifier of an instruction. (ISO)

### language processor

A computer program that performs such functions as translating, interpreting, and other tasks required for processing a specified programming language; for example, a FORTRAN processor, a COBOL processor. (ISO)

### large scale integration

Technology by which thousands of electronic devices are fabricated on a single semiconductor chip. See also VLSI, MSI, VHSIC.



**last in-first out**

A data structure in which data is extracted sequentially with the most recent entry accessed first.

**latency**

- a. The time between the completion of the interpretation of an address and the start of the actual transfer from the addressed location. Latency includes the delay associated with access to storage devices such as magnetic drums and delay lines. (ADatP-2)
- b. In avionics data bus systems, the time delay between the bus controller issued command to transmit data and the receipt of this data at the appropriate terminal or sub-system.

**least significant bit (LSB)**

The bit in a word which represents the lower power of the radix.

**length register**

A high speed memory location used in conjunction with a base register in task security implementations. The base register contains lower physical address of the segment and the contents of the base register plus the contents of the limit register is the highest physical address of the segment.

**level of documentation**

A description of required documentation indicating its scope, content, format, and quality. Selection of the level may be based on project cost, intended usage, extent of effort, or other factors. (ANSI/IEEE)

**lexical**

Pertaining to the words of a language; opposed to syntactical or grammatical, which means "pertaining to the relationship of words in the permitted structures of a language". Lexical analysis is concerned with the

identification of individual words in a statement and is the first function performed by a compiler.

#### light emitting diode (LED)

A semiconductor diode that serves as a light source.

#### line replaceable unit (LRU)

Any unit which can be replaced at line level of maintenance.

#### link

A branch instruction at the end of a subroutine to return control to some point in the main program from which the subroutine is called.

#### link list

A data structure in which each element contains a pointer to its predecessor or successor (singly linked list) or both (doubly linked list).

#### link(age) editor

A programme that produces a load module by transforming object modules into an acceptable format for a computer.

#### link(age) loader

See link(age) editor.

#### linker

See link(age) editor.

**liquid crystal display**

A device which uses liquid crystal technology to display information.

**list**

- a. An ordered set of items of data. (ISO)
- b. To print or otherwise display items of data that meet specified criteria. (ANSI)

**list processing**

A method of processing data in the form of lists. Usually, chained lists are used so that the logical order of items can be changed without altering their physical locations. (ISO)

**listing**

A computer output in the form of a human-readable list. (ANSI/IEEE)

**load**

To read data, commonly computer instructions, into internal storage or registers, in particular, to copy into registers. (ADatP-2)

**load map**

A computer generated list that identifies the location or size of all or selected parts of a memory-resident computer program or of memory-resident data. (ANSI/IEEE)

**load module**

- a. The output of linkage editor. (ADatP-2)
- b. A programme in a format suitable for loading into main storage for execution. (ADatP-2)

**loader**

In avionics, the programme that causes another programme to be loaded into a computer memory.

**log**

The record of a computer run, often in the form of a sequence of messages output by a console typewriter.

**log on**

A mechanism by which a computer system user identifies himself/herself and gains access to system facilities.

**logical address**

A means of addressing data and/or program instructions in such a way as to be independent of the physical memory structure of a given machine. When the program is run, this logical address is translated on the target machine into a physical address within the machine structure.

**logical device**

A reference to an I/O device by name or number without regard to the extra nature of the I/O device. At execution time, the operator, or another task assigns a physical device to the logical device name or number. The I/O subsystem then assures that the proper device drivers are utilized to perform the I/O transfers.

**logical file**

A file independent of its physical environment. Portions of the same logical file may be located in different physical files, or several logical files or parts of logical files may be located in one physical file. (ANSI/IEEE)

**logical instruction**

An instruction that operates on binary digits, using Boolean Logic.

**logical record**

A collection of items independent of their physical environment. Portions of the same logical record may be located in different physical records. (ANSI)

**loop**

A set of instructions that may be executed repeatedly while a certain condition prevails. (ISO)

**machine**

A term often used for a computer.

**machine code**

- a. An operation code that a machine is designed to recognize. (ADatP-2)
- b. The absolute numbers, names, or symbols assigned by the machine designer to any part of the machine.
- c. Same as operation code. (Computer Dictionary)
- d. Consist of bit patterns recorded as magnetic impulses, each pattern causing a different function of computer to be performed.

**machine cycle**

An algorithm which defines a sequence of operations performed by an abstract machine to determine which additional operations are to be performed and to which data objects they are to be applied. If an abstract machine is defined as having a machine cycle, that machine cycle is executed regularly and whenever the abstract machine has completed the performance of the

operation called for by the previous machine cycle.  
(ABBOTT)

#### machine instruction

An instruction that can be recognized and executed by the computer for which it is designed). (ADatP-2)

#### machine language

A language that is used directly by a machine.  
(ADatP-2)

#### machine word

See computer word.

#### machine-independent

Pertaining to procedures or programmes created without regard for the actual devices which will be used to process them. (ADatP-2)

#### macro-instruction

An Instruction in a source language that is to be replaced by a defined sequence of instructions in the same source language. The macro-instruction may also specify values for parameters in the instructions that are to replace it. (ISO)

#### macro-processor

The portion of some assemblers and compilers that allows a programmer to define and use macros.  
(ANSI/IEEE)

#### mailbox

A system data structure that handles task communication. Tasks send messages to and receive messages from Mailboxes.

---

**main storage**

Computer storage which is a part of the central processing unit and which is used to hold the programmes or parts of programmes which are currently being executed. Synonymous with primary storage. (ADatP-2)

---

**maintainability**

- a. The ease with which software can be maintained. (ANSI/IEEE)
- b. The ease with which maintenance of a functional unit can be performed in accordance with prescribed requirements. (ISO)
- c. Ability of an item under stated conditions of use to be retained in, or restored to, within a given period of time, a specified state in which it can perform its required functions when maintenance is performed under stated conditions and while using prescribed procedures and resources. (ANSI/ASQC A3-1978)

---

**map program**

A compiler or assembler feature that generates a load map. (ANSI/IEEE)

---

**mass storage**

Backing-store of large capacity, such as magnetic disk.

---

**mass store**

See secondary memory.

---

**master file**

A file of basic, current data, periodically updated, and used in the interim for, e.g. the extraction of data for reports.

**master library**

A software library containing formally released versions of software and documentation. (ANSI/IEEE)  
Contrast with production library.

**matrix**

In hardware, a frame, e.g. of magnetic cores; in mathematics, an array of numbers; in software, often synonymous with tables.

**media**

Devices or material which act as carriers of the software in object or source form, e.g., program listings, magnetic tape, magnetic discs, cartridge files, cassettes, punched tape, punched cards, etc. (RTCA)

**medium scale integration (MSI)**

Technology by which approximately 10 to 1,000 electronic devices are fabricated on a single semiconductor chip.

**memory device**

An article of hardware capable of storing machine readable computer programs and associated data. May be an integrated circuit (IC) chip, a circuit card containing ICs, a core memory, a disk, a magnetic tape, etc. (RTCA)

**memory management**

A system of adaptive control that allocates memory and schedules the central processor in order to maximize performance. (DAN 595)

**message**

A unit of communication between tasks. In a typical system, a message usually contains a header with information about message size, the sending task, etc.



**metalanguage**

A language used to specify a language or languages.  
(ANSI/IEEE)

**metric**

A measure of the extent or degree to which the software possesses and exhibits a certain characteristic, quality property or attribute. (GLOS-1)

**microcode**

The elementary instructions which are used to control the operation of individual circuit elements used; e.g., to implement the ISA.

**microcomputer**

A microprocessor plus other components such as memories, clocks and various interface devices that collectively operate as a stored programme computer. (Microcomputers may be packaged as integrated circuits, integrated circuit sets, a single circuit card, or multiple circuit cards.)

**microprocessor**

Chips or chip sets performing arithmetic and logic functions.

**microprogramme**

A programme in microcode.

**milestone**

A scheduled event for which some project member or manager is held accountable and that is used to measure progress; for example, a formal review, issuance of a specification, product delivery. (ANSI/IEEE)

**mission avionics**

That portion of an avionics system that is unique to a specified operational mission. Contrast with core avionics.

**mnemonic symbol**

A symbol chosen to assist the human memory, e.g., an abbreviation such as 'mpy' for 'multiply'. (ANSI)

**model**

A representation of a real world process, device, or concept. See also analytical model, availability model, error model, reliability model, statistical test model. (ANSI/IEEE)

**modular decomposition**

A method of designing a system by breaking it down into modules. (ANSI/IEEE) See also hierarchical decomposition.

**modular programming**

A technique for developing a system or program as a collection of modules. (ANSI/IEEE)

**modularity**

The extent to which software is composed of discrete components such that a change to one component has minimal impact on other components. (ANSI/IEEE)

**module**

- a. A programme unit that is discrete and identifiable with respect to compiling, combining with other units, and loading, e.g., the input to, or output from, an assembler, compiler, linkage editor, or executive routine. (ADatP-2)

- b. A packaged functional hardware unit designed for use with other components. (ADatP-2)
- c. A convenient unit, either of hardware or software, which can be combined with other units in different ways to form larger wholes.

#### module testing

After each individual function has been tested, it is necessary to test the cooperation of these functions when they are put together as a module. It should be possible to test a module as stand-alone entity, without the presence of other system modules.

#### monitor

A monitor determines which of two or more processes competing for control in order to execute, has priority. It allows that which has priority to take control and execute, and places the other process(es) in a queue to await their turn to take control and execute. (DAN 922)

#### most significant bit

The bit in a word which represents the high power of the radix.

#### multi-level-feedback queue

A preemptive process scheduling discipline that is particularly useful in systems with diversified job mixes.

#### multi-tasking

The concurrent execution of two or more tasks by a computer. (ANSI)

#### multi-user operating system

An operating system that permits more than one user to operate the computer system in such a way that it appears to each user as if he has exclusive control of

the machine. This is done by letting each user nominally execute his/her program for a fixed period of time. These systems are known as Time Sharing Systems (TSS).

#### multifunction display

An electro-optical display that can provide selectable information, usually in a raster, cursive, or matrix format.

#### multilevel security

A mode of operation permitting data at various security levels to be concurrently stored and processed in a computer system, when at least some users have neither the clearance nor the need-to-know for all data contained in the system. (ANSI/IEEE)

#### multiplex

- a. To interleave or simultaneously transmit two or more signals on a single channel. (ADatP-2)
- b. Sometimes used instead of multiple (in contrast to simplex) in the context of redundancy.

#### multiplex data bus

A data transmission technique which uses multiplex technology to connect and integrate multiple sub-systems.

#### multiprocessor

A computer including two or more central processing units that have access to a common main storage. (ADatP-2)

#### multiprogramming

A mode of operation that provides for the interleaved execution of two or more computer programs by a single processor. (ISO)

**MUX**

Abbreviation of multiplex.

**N-ary**

Characterized by a selection, choice, or condition that has no possible different values or states. (ISO)

**natural language**

A language whose rules are based on current usage without being explicitly prescribed. (ISO) Examples include English, Chinese, French, and Swahili. Contrast with formal language.

**nest**

To embed subroutines or data in other subroutines or data at a different hierarchical level such that the different levels of routines or data can be executed or accessed recursively. (ANSI)

**network**

An interconnected or interrelated group of nodes. (ANSI/IEEE)

**node**

- a. A position in tree structure related by pointers either to the nodes above or below itself in the hierarchy, or both.
- b. The representation of a state or an event by means of a point on a diagram. (ANSI)

**non-preemptive scheduling**

A sort of scheduling that requires a task run completion once it begins executing. The scheduler must predict task requirements and attempt to execute these tasks in a manner that minimizes turn-around. Since the scheduler cannot stop task once they are executing, the

resources (maximum processor time, memory; and peripheral devices) needed by the task must be specified by the user before the task is executed. The most simple type of non-preemptive scheduling algorithm is FOFO scheduling. Non-preemptive scheduling techniques attempt to pick the tasks for execution which keep average turn-around time to a minimum. However, once a long task begins execution, all other tasks must wait until it completes regardless of relative priorities.

#### non-volatile memory

Memory which does not require power to retain the stored data. (RTCA)

#### numerical reliability

The probability that an item will perform a required function under stated conditions for a stated period of time. (ANSI/ASQC A3-1978)

#### object code

- a. Output from a compiler or assembler which is itself executable machine code or is suitable for processing to produce executable machine code. (ADatP-2)
- b. The representation of data or a computer program in a symbolic form which is fully compiled or assembled and ready to be loaded into a computer.

#### object module

A module that is the output of an assembler or compiler and is input to a linkage editor. (ADatP-2)

#### object oriented design

A design technique whereby a system is viewed as a collection of objects rather than as functions with messages passed from object to object, each with its own set of associated operations.

object oriented programming

A programming technique based on the notion of objects which allow Object Oriented Designs to be directly implemented. Requires the use of an Object Oriented Language.

object programme

A fully compiled or assembled programme that is ready to be loaded into the computer. Contrast with source programme. (ADatP-2)

open system

A system that implements sufficient open specifications for interfaces, services, and supporting formats to enable properly engineered components to be utilized across a wide range of systems with minimal changes, to interoperate with other components on local and remote systems, and to interact with users in a style that facilitates portability. An open system is characterized by the following:

- a. well defined, widely used, non-proprietary interfaces/protocols;
- b. use of standards which are developed/adopted by industrially recognized standards bodies;
- c. definition of all aspects of system interfaces to facilitate new or additional systems capabilities for a wide range of applications;
- d. explicit provision for expansion or upgrading through the incorporation of additional or higher performance elements with minimal impact on the system.

(IEEE POSIX 1003.0/D15 as modified by the Tri-Service Open Systems Architecture Working Group)

operand

That which is operated upon. An operand is usually identified by an address part of an instruction. (ANSI)

---

**operating system (OS)**

- a. Software for controlling the execution of computer programmes and that may provide scheduling, debugging, input/output control, accounting, compilation, storage assignment, data management, and related services. (ADatP-2)
- b. Software that controls the execution of programs. An operating system may provide service such as resource allocation scheduling, input/output control, and data management. Although operating systems are predominantly software, partial or complete hardware implementations are possible. (ISO)
- c. An operating system provides support in a single spot rather than forcing each program to be concerned with controlling hardware. See system software. (ANSI/IEEE)

---

**operational flight programme (OFP)**

Programme loaded into avionics computer which performs the computations, functions, and algorithms required by the sub-systems for performing a specified mission.

---

**operational reliability**

The reliability of a system or software subsystem in its actual use environment. Operational reliability may differ considerably from reliability in the specified or test environment. (ANSI/IEEE)

---

**operational software (embedded)**

The portion of a computer program, including real time executive and application software, which is directly involved in flight control and avionics processing functions. (GLOS-1)



**operator**

- a. In the description of a process, that which indicates the action to be performed on operands. (ANSI)
- b. A person who operates a machine. (ANSI)

**optical fibre**

Glass (or other medium) fibres through which light energy is transmitted.

**optimization**

Changes in the source code to improve program performance - e.g., run faster or use less space. Optimization changes are not error correction; however, if a change is made to use less space to conform to the specified space constraint, then the term "error" applies. Note: efficiency is a quality characteristic; optimization can be a process which increases efficiency. (GLOS-1)

**output assertion**

A logical expression specifying one or more conditions that program outputs must satisfy in order for the program to be correct. (ANSI/IEEE)

**overhead**

The amount of processing time required by the operating system to perform housekeeping tasks such as paging, swapping, and scheduling. This time is usually expressed as a percentage of total available time. If system overhead is quoted as 20%, then user programs can utilize 48 seconds (80%) of each minute of processor execution time.

**overlay**

A technique used to execute programs larger than the available memory size in systems without paging or segmentation capabilities. To utilize this method, a program must be manually divided into a number of

mutually exclusive groups of software modules. Any software common to more than one overlay is included in the root overlay. The root must remain in system memory at all times and other overlays are loaded and executed one at a time in a designated memory area. This loading is normally performed by the operating system at the request of the root overlay.

### **paged-store**

A store divided into fixed-length partitions.

### **paging**

- a. A method of organizing programmes and their storage requirements into small units (or pages) to enable them to be executed with a limited area of available main storage.  
(ADatP-2)
- b. The division of core-store into fixed-length physical blocks (page frames). Programs and data are divided into blocks (pages) of the same size. Thus =, one page of information can be loaded into one page frame. The length of the pages is determined by the characteristics of the hardware rather than by program logic.

### **parameter**

A variable that is given a constant value for a specific purpose or process. (ANSI)

### **parity bit**

A check bit appended to an array of binary digits to make the sum of all the binary digits, including the check bit, always odd or always even.

### **parse**

To determine the syntactic structure of an artificial or natural language unit by decomposing the unit into more elementary sub-units and establishing the relationships among the sub-units; for example, blocks, statements, and expressions may be decomposed into

statements, expressions, and operators and operands.  
(ANSI/IEEE)

### partial correctness

In proof of correctness, a designation indicating that a program's output assertions follow logically from its input assertions and processing steps. (ANSI/IEEE)  
Contrast with total correctness.

### partitioning

Function A is partitioned from function B if no action of function B can cause a failure in function A.  
(RTCA)

### pascal

A high-level structured computer programming language whose name is dedicated to the French scientist Blaise PASCAL, and used especially for scientific and engineering applications. Pascal is known with its strict rules. Data structures are based upon pascal-line language such as C.

### patch

- a. A modification to an object program made by replacing a portion of existing machine code with modified machine code. (ANSI/IEEE)
- b. To modify an object program without recompiling the source program. (ANSI/IEEE)

### path analysis

Program analysis performed to identify all possible paths through a program, to detect incomplete paths, or to discover portions of the program that are not on any path. (ANSI/IEEE)

**path condition**

A set of conditions that must be met in order for a particular program path to be executed. (ANSI/IEEE)

**path expression**

A logical expression indicating the input conditions that must be met in order for a particular program path to be executed. (ANSI/IEEE)

**path testing**

The path testing approach can only be successful applied to programs that have been designed in a top-down manner using structured programming. Path testing is then used to test each of the program modules separately. Every "logical path" through a program should be tested at least once.

**performance evaluation**

The technical assessment of a system or system component to determine how effectively operating objectives have been achieved. (ANSI/IEEE)

**performance requirement**

A requirement that specifies a performance characteristic that a system or system component must possess; for example, speed, accuracy, frequency. (ANSI/IEEE)

**performance specification**

A specification that sets forth the performance requirements for a system or system component. (ANSI/IEEE)

**performance testing**

A test phase designed to test run-time performance of software within the context of an integrated system. Performance testing occurs throughout all steps in the testing process. Even at the unit level the performance

of an individual module may be assessed as white box tests are conducted. Performance tests are often coupled with stress testing and often require both hardware and software instrumentation. That is, it is often necessary to measure resource utilization (e.g. processor cycles) in an exacting fashion.

### petri net

An abstract, formal model of information flow, showing static and dynamic properties of a system. A Petri net is usually represented as a graph having two types of nodes (called places and transitions) connected by arcs, and markings (called tokens) indicating dynamic properties. (ANSI/IEEE)

### physical address

The address of an existing system memory location.

### physical device

An actual peripheral hardware device such as a line printer, terminal, etc., which is attached to a computer system.

### physical requirement

A requirement that specifies a physical characteristic that a system or system component must possess; for example, material, shape, size, weight. (ANSI/IEEE)

### PL/1

Acronym for Programming Language 1. A computer programming language designed to combine the facilities of both commercial and scientific programming languages such as COBOL and FORTRAN.

### pointer

- a. A data item whose value is the location of another data item. (ANSI/IEEE)

- b. A store location, used as an operand, whose contents are the address of another store location.

### polling

A method of control by a central computer, used when several remote terminals are sharing one line, to permit querying of terminals in accordance with established priorities. (ADatP-2)

### pop

A term used for retrieving information from a stack.

### portability

The ease with which software can be transferred from one computer system or environment to another. (ANSI/IEEE)

### positional notation

A numeration system in which a number is represented by means of a ordered set of digits, such that the value contributed by each digit depends upon its position as well as its value.

### precision

The degree of discrimination with which a quantity is stated. For example, a three-digit numeral discriminates among 1000 possibilities. (ANSI)

### precompiler

A computer program that preprocesses source code, part of which may be unacceptable to a compiler, to generate equivalent code acceptable to the compiler; for example, a preprocessor which converts structured FORTRAN to ANSI-standard FORTRAN. (ANSI/IEEE)

**predicate**

A logical proposition or assertion concerning the state of a program at a given point, having either a true or false value. Concerning program correctness, all such assertions must be axioms or be proved true. (DAN 1153)

**preemptive scheduling**

A means of scheduling where a high priority task will preempt execution of a lower priority task as soon as the high priority task is ready to run, usually after an interrupt. To implement this scheduling algorithm, each task is given an execution algorithm. Generally, the higher the task priority, the more important it is for the task to be executed. Preemptive scheduling can only be implemented in multi-programming systems where tasks may be suspended and restarted. The simplest type of preemptive scheduling is round-robin scheduling.

**preliminary design**

- a. The process of analyzing design alternatives and defining the software architecture. Preliminary design typically includes definition and structuring of computer program components and data, definition of the interfaces, and preparation of timing and sizing estimates. (ANSI/IEEE)
- b. The result of the preliminary design process. (ANSI/IEEE)

**preprocessor**

A computer program that effects some preliminary computation or organization. (ISO)

**primary memory**

See main storage.

**priority scheduling**

A scheduling algorithm where each task is given a priority, and the next task to execute is the one in the

Ready List with the highest priority. Priorities may be static or dynamic. With static priorities, it is possible that some low priority tasks will never run due to the frequency and execution duration of higher priority tasks. This is very unhealthy in a Time-Sharing System Environment where it is necessary to guarantee that all tasks will receive some processor time on a regular basis. For these systems, the task priority is often changed as a function of time, so that the longer a task waits to execute, the higher its priority becomes.

### privileged instruction

An instruction that may be used only by a supervisory program. (ISO)

### problem-oriented

Those computer languages designed to facilitate the programming of problems in a particular field, rather than to take advantage of the facilities of a particular machine. Opposed to machine-oriented.

### procedure

- a. A portion of a computer program which is named and which performs a specific task. Compare with subroutine, subprogram, function, module. (ANSI/IEEE)
- b. The course of action taken for the solution of a problem. (ISO)
- c. The description of the course of action taken for the solution of a problem. (ANSI)

### process

A systematic sequence of operations to produce a specified result. (ADatP-2)

### processor

- a. In hardware, a data processor. (ADatP-2)



- b. In software, a computer programme that includes the compiling, assembling, translating and related functions for a specific programming language. (ADatP-2)

#### product baseline

A milestone activity that identifies the produced software system and provides a basis for modifying the system by correcting errors and incorporating new requirements.

#### production library

A software library containing software approved for current operational use. (ANSI/IEEE)

#### program architecture

The structure and relationships among the components of a computers program. The program architecture may also include the program's interface with its operational environment. (ANSI/IEEE)

#### program block

In problem-oriented languages, a computer program subdivision that serves to group related statements, delimit routines, specify storage allocation, delineate the applicability of labels, or segment paths of the computer program for other purposes. (ANSI)

#### program design language

The technique of using English language statements to describe the logical processes of system functional element. PDL is a tool for program development and is an essential part of the system documentation.

#### program extension

An enhancement made to existing software to increase the scope of its capabilities. (ANSI/IEEE)

---

**program instrumentation**

- a. Probes, such as instructions or assertions, inserted into a computer program to facilitate execution monitoring, proof of correctness, resource monitoring, or other activities. (ANSI/IEEE)
- b. The process of preparing and inserting probes into a computer program. (ANSI/IEEE)

---

**program library**

An organized collection of computer programs that are sufficiently documented to allow them to be used by persons other than their authors. (ISO)

---

**program mutation**

- a. A program version purposely altered from the intended version to evaluate the ability of program test cases to detect the alteration. (ANSI/IEEE)
- b. The process of creating program mutations in order to evaluate the adequacy of program test data. (ANSI/IEEE)

---

**program protection**

The application of internal or external controls to preclude any unauthorized access or modification to a computer program. (ANSI/IEEE)

---

**program specification**

Any specification for a computer program. See design specification, functional specification, performance specification, requirements specification. (ANSI/IEEE)

---

**program synthesis**

The use of software tools to aid in the transformation of a program specification into a program that realizes that specification. (ANSI/IEEE)

**programmable**

Describes a device, component, sub-system or system which is designed to allow its function and/or physical characteristics to be modified.

**programmable read-only memory (PROM)**

A memory which allows only reading of the data stored within it under operational conditions.

**programme**

- a. To design, write, and test programmes. (ADatP-2)
- b. Loosely, to write a routine. (ADatP-2)

**programmer**

A person who designs, writes, and tests computer programmes. (ADatP-2)

**programming**

The design, writing, and testing of programmes. (ADatP-2)

**programming language**

A language used to prepare computer programmes. (High order language and assembly language.) (ADatP-2)

**programming support environment**

An integrated collection of tools accessed via a single command language to provide programming support capabilities throughout the software life cycle. The environment typically includes tools for designing,

editing, compiling, loading, testing, configuration management, and project management. (ANSI/IEEE)

### project notebook

A central repository of written material such as memos, plans, technical reports, etc., pertaining to a project. Synonymous with project file. (ANSI/IEEE)

### project plan

A management document describing the approach that will be taken for a project. The plan typically describes the work to be done, the resources required, the methods to be used, the configuration management and quality assurance procedures to be followed, the schedules to be met, the project organization, etc. (ANSI/IEEE)

### prom

Programmable ROM. A semiconductor ROM in which the stored information can be altered, but not during operation. It can be erased by prolonged application of ultra-violet light, and then rewritten with relatively high power electronics.

### prompt

- a. A message informing a user that a system is ready for the next command, message, or other user action. (ANSI/IEEE)
- b. To inform a user that a system is ready for the next command, element, or other input. (ANSI/IEEE)

### proof of correctness

- a. A formal technique used to prove mathematically that a program satisfies its specifications. See also partial correctness, total correctness. (ANSI/IEEE)
- b. A program proof that results from applying this technique. (ANSI/IEEE)

protection

An arrangement for restricting access to or use of all, or part, of a computer system. (ISO)

protection violation

An attempt by a task to access a protected resource, or perform an unauthorized operation on an assigned resource. For example, attempting to write a read-only file, or attempting to execute a privileged instruction will generate a protection violation.

protocol

- a. A set of conventions or rules that govern the interactions of processes or applications within a computer system or network.  
(ANSI/IEEE)
- b. A set of rules that govern the operation of functional units to achieve communication.  
(ISO)

pseudo code

- a. A combination of programming language and natural language used for computer program design. (ANSI/IEEE)
- b. An informal documentation method representing structured programming logic.

pulse code modulation (PCM)

A means of signal modulation in which the modulating signal waveform is sampled at regular intervals, quantified into discrete steps, and transmitted as a code pattern of pulses.

push

A term used for placing information on a stack.

**pushdown storage**

A storage device that handles data in such a way that the next item to be retrieved is the most recently stored item still in the storage device; i.e., last-in-first-out (LIFO). (ISO)

**qualification testing**

Formal testing, usually conducted by the developer for the customer to demonstrate that the software meets its specified requirements. (ANSI/IEEE)

**quality**

The totality of features and characteristics of a product or service that bears on its ability to satisfy given needs. (ANSI/ASQC A3-1978)

**quality assurance**

A planned and systematic pattern of all actions necessary to provide adequate confidence that the item or product conforms to established technical requirements. (ANSI/IEEE STD 730-1981)

**quality metric**

A quantitative measure of the degree to which software possesses a given attribute that affects its quality. (ANSI/IEEE)

**queue**

A list that is accessed in a first-in, first-out manner. (ANSI/IEEE) Contrast with stack.

**random access**

An access mode in which specific logical records are obtained from or placed into a mass storage file in a nonsequential manner. (ADatP-2)

**random access memory (RAM)**

A memory whose words may be read or written in any desired order.

**read only memory (ROM)**

A memory that cannot be altered during the operational use of the computer.

**ready**

A task state in which a given task is prepared to execute.

**ready list**

A system list of tasks in the Ready task state, meaning that they are ready to be executed on a processor.

**real time**

- a. Pertaining to the actual time during which a physical process transpires. (ADatP-2)
- b. Pertaining to the performance of a computation during the actual time that the related physical process transpires, in order that results of the computation can be used in guiding the physical process. (ADatP-2)

**reconfiguration**

Changing the components or the organization of the components of a system.

**record**

A collection of related data or words treated as a unit. (ISO)

**recovery testing**

A system test that forces the software to fail in a variety of ways and verifies that recovery is properly performed. If recovery is automatic (performed by the system itself), reinitialization, checkpointing mechanisms, data recovery, and restart are each evaluated for correctness. If recovery requires human intervention, the mean time to repair (MTTR) is evaluated whether it is within acceptable limits.

**recursive routine**

A routine that may be used as a subroutine of itself, calling itself directly or being called by another subroutine, one that it itself has called. The use of a recursive routine usually requires the keeping of records of the status of its unfinished uses in, for example, a pushdown list. (ANSI/IEEE)

**redundancy**

The inclusion of duplicate or alternate system elements to improve operational reliability by ensuring continued operation in the event that a primary element fails. (ANSI/IEEE)

**reentrant code**

Code that may be executed simultaneously by more than one task. Thus, the code cannot be self-modifying, and each task must maintain its own data.

**register**

- a. A device capable of storing a specified amount of data such as one word. (ADatP-2)
- b. A device for the temporary storage of one or more words to facilitate arithmetical, logical, or transferral operations. Examples are the accommodator, address, index, instruction, and M-Q registers.
- c. The hardware for storing one or more computer words. (Computer Dictionary)



**regression testing**

Selective retesting to detect faults introduced during modification of a system or system component, to verify that modifications have not caused unintended adverse effects, or to verify that a modified system or system component still meets its specified requirements. (ANSI/IEEE)

**release**

A system service that informs the operating system that a task no longer requires use of a resource.

**reliability**

The probability that a device will perform without failure for a specified time period or amount of usage. (ANSI)

**reliability assessment**

The process of determining the achieved level of reliability of an existing system or system component. (ANSI/IEEE)

**reliability data**

Information necessary to assess the reliability of software at selected points in the software life cycle. Examples include error data and time data for reliability models, program attributes such as complexity, and programming characteristics such as development techniques employed and programmer experience. (ANSI/IEEE)

**reliability growth**

The improvement in software reliability that results from correcting faults in the software. (ANSI/IEEE)

**reliability model**

A model used for predicting, estimating, or assessing reliability. (ANSI/IEEE)

**relocatable machine code**

Machine language code that requires relative addresses to be translated into absolute addresses prior to computer execution. (ANSI/IEEE) Contrast with absolute machine code.

**relocation**

The process of moving the location of a program in system memory. Relocation may either be static or dynamic. Static relocation means that all program addresses are assigned and fixed before the program is loaded into memory. Dynamic relocation refers to a technique that requires all program address to be relative to some base address and programs may be relocated at any time by moving the program code and changing the base address.

**remote terminal**

In a data bus system all terminals not operating as the bus monitor or as a bus controller.

**rendez-vous**

The interaction that occurs between two parallel tasks when one task has called an entry of the other task, and a corresponding accept statement is being executed by the other task on behalf of the calling task. (ANSI/IEEE)

**requirement**

- a. A condition or capability needed by a user to solve a problem or achieve an objective.  
(ANSI/IEEE)
- b. A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed document. The set of all requirements forms the basis for subsequent development of the system or system component.  
(ANSI/IEEE) See also requirements analysis, requirements phase, requirements specification.

**requirements analysis**

- a. The process of studying user needs to arrive at a definition of system or software requirements. (ANSI/IEEE)
- b. The verification of system or software requirements. (ANSI/IEEE)

**requirements phase**

The period of time in the software life cycle during which the requirements for a software product, such as the functional and performance capabilities, are defined and documented. (ANSI/IEEE)

**requirements specification**

A specification that sets forth the requirements for a system or system component; for example, a software configuration item. Typically included are functional requirements, performance requirements, interface requirements, design requirements, and developing standards. (ANSI/IEEE)

**requirements specification language**

A formal language with special constructs and verification protocols used to specify, verify, and document requirements. (ANSI/IEEE)

**resource**

Assets of a computer system that the operating system can use and/or allocate to tasks for their use. Assets such as memory, disk storage space, printers and terminals) main memory are allocated to tasks based on their availability.

**resource allocation**

The operating system function of assigning resources to tasks that request them. Resources such as

peripherals (printers, terminals) and main memory are allocated to tasks based on their availability.

### resource management

A task requests a resource to execute. If there are no resources available, the task waits until the resource becomes available. The resource is then assigned exclusively to the requesting task that proceeds to use it as required. When the task no longer needs to use the resource, the release service is utilized to return the resource to the operating system which then assigns the resource to the next requesting task.

### response time

The elapsed time from entry of command until its execution is complete.

### restore

A system utility that reloads a file to secondary storage from an archive volume such as magnetic tape.

### retirement phase

The period of time in the software life cycle during which support for a software product is terminated.  
(ANSI/IEEE)

### reusability

The extent to which a module can be used in multiple applications. (ANSI/IEEE)

### robustness

The extent to which software can continue to operate correctly despite the introduction of invalid inputs.  
(ANSI/IEEE)

**root compiler**

A compiler whose output is a machine independent, intermediate-level representation of a program. A root compiler, when combined with a machine-dependent code generator, comprises a full compiler. (ANSI/IEEE)

**round robin scheduling**

A means of scheduling whereby tasks in the Ready list are executed in order and entries into the Ready list are always put at the end of the list. Usually, each task is also given a limited time for execution. If after this time it is not complete, it is reentered at the end of the Ready List and the next task is executed.

**rounding**

The truncation of digits at the least significant end of a number; in addition, if the rounding is "up", the last remaining digit is increased by one if the first truncated digit is 5 or more (in the decimal place). Thus if the number 476.58 is rounded up to one fractional digit, it becomes 476.6.

**routine**

A computer programme, or part thereof, that may have some general or frequent use. (ADatP-2)

**run time**

- a. A measure of the time expended to execute a program. While run time ordinarily reflects the expended central processor time, run time may also include peripheral processing and peripheral accessing time; for example, a run time of 5 hours. (ANSI/IEEE)
- b. The instant at which a program begins to execute. (ANSI/IEEE) See also execution time.

**running**

A task state in which the task is executing on a processor.

**scenario**

An automated test control package consisting of test execution control words, test data, and event stimuli used to activate and test a target program. (DAN 1201)

**scheduling**

- a. An operating system module that selects the next job to be admitted into the system and the next process to run.
- b. In operating systems, scheduling refers to the allocation of jobs from a queue to run in an efficient or prioritized manner. (GLOS-1)

**secondary memory**

Also called "bulk storage" of "mass store" these memory facilities are external to the computer, but are directly connected to and controlled by the computer (e.g., on-line magnetic discs or drums).

**security**

The protection of computer hardware and software from accidental or malicious access, use, modification, destruction, or disclosure. Security also pertains to personnel, data communications, and the physical protection of computer installations. (ANSI/IEEE)

**security kernel**

A small, self-contained collection of key security-related statements that works as a privileged part of an operating system. All criteria specified by the kernel must be met for a program or data to be accessed. (ANSI/IEEE)

**security testing**

A test phase that attempts to verify that protection mechanisms built into a system will, in fact, protect it

from improper penetration. During security testing, the tester plays the role of the individual who desires to penetrate the system. The tester may attempt to acquire passwords through external clerical means; may attack the system with custom software designed to break down any defense that have been constructed; may overwhelm the system thereby denying service to others; may purposely cause system errors, hoping to penetrate during recovery; or may browse through public data, hoping to find the key to system entry.

#### segment (n)

- a. A self-contained portion of a computer program that may be executed without the entire computer program necessarily being maintained in internal storage at any one time. (ISO)  
See also component, module, subprogram.
- b. The sequence of computer program statements between two consecutive branch points.  
(ANSI/IEEE) See also path analysis.

#### segment (v)

To divide a computer program into segments. (ISO)

#### segmentation

The partitioning of a program's address space into variable-size blocks (segments) or modules. Segmentation involves breaking a program into logically separable units. For example, one segment may be a subroutine; another may be a data area. Segments that are too large take too much time when being moved to or from primary storage (main memory). Segments that are too small involve too many moves.

#### semantics

The relationships between symbols and their meanings. (ANSI)

**semaphore**

A shared variable used to synchronize concurrent processors by indicating whether an action has been completed or an event has occurred. (ANSI/IEEE)

**send**

A system service used for task communication and synchronization that places a message in the appropriate mailbox. If a task is waiting at the mailbox, it is placed in the Ready List.

**sequential processes**

Processes that execute in such a manner that one must finish before the next begins. (ANSI/IEEE)  
Contrast with concurrent processes.

**serial access**

- a. Pertaining to the sequential or consecutive transmission of data to or from storage. (ADatP-2)
- b. Pertaining to the process of obtaining data from or placing data into storage, where the access time is dependent upon the location of the data most recently obtained or placed in storage. (ADatP-2)

**sharing**

A means of running many users with a limited amount of memory. Sharing generally requires that either paging or segmentation be used.

**shift**

To move the values of the binary digits in a computer word from one position to another.



**side-effect**

Processing or activities performed, or results obtained, secondary to the primary function of a program, subprogram, or operation. (ANSI/IEEE)

**sign bit**

The bit in a computer word, either the first or last, which by its setting indicates whether the numerical content of the word is positive or negative.

**signal processing**

Manipulation of digital or analogue signals to obtain a desired effect.

**simplex operation**

A method of operation in which communication between two stations takes place in one direction at a time.

**simulation**

The representation of certain features of the behavior of a physical or abstract system by the behavior of another system, e.g., the representation of physical phenomena by means of operations performed by a computer or the representation of operations of a computer by those of another computer. (ANSI)

**simulation testing**

The exercise of the software in conjunction with a simulation program and often some peripheral hardware which replicate the real operating environment as closely as possible. The computer used to run the software and the simulation program may be the computer to be used in the field or a development computer.

**simulator**

A device, system, or computer program that represents certain features of the behavior of a physical or abstract system. (ANSI)

**sizing**

The process of estimating the amount of computer storage or the number of source lines that will be required for a system or system component. (ANSI/IEEE)

**software**

Computer programmes, procedures, rules, and any associated documentation concerned with the operation of a data processing system. (ADatP-2)

**software data base**

A centralized file of data definitions and present values for data common to, and located internal to, an operational software system. (ANSI/IEEE)

**software development cycle**

The period of time that begins with the decision to develop a software product and ends when the product is delivered. This cycle typically includes a requirements phase, design phase, implementation phase, test phase, and sometimes, installation and check-out phase. (ANSI/IEEE) Contrast with software life cycle.

**software development library**

A software library containing computer readable and human readable information relevant to a software development effort. (ANSI/IEEE)

**software development notebook**

A collection of material pertinent to the development of a given software module. Contents typically include the requirements, design, technical reports, code listings, test plans, test results, problem reports, schedules, notes, etc. for the module. (ANSI/IEEE)

**software development plan**

A project plan for the development of a software product. (ANSI/IEEE) Synonymous with computer program development plan.

**software development process**

The process by which user needs are translated into software requirements, software requirements are transformed into design, the design is implemented in code, and the code is tested, documented, and certified for operational use. (ANSI/IEEE)

**software documentation**

Technical data or information, including computer listings and printouts, in human-readable form, that describe or specify the design or details, explain the capabilities, or provide operating instructions for using the software to obtain desired results from a software system. (ANSI/IEEE)

**software engineering**

The systematic approach to the development, operation, maintenance, and retirement of software. (ANSI/IEEE)

**software engineering standards**

A set of enforceable guidelines and/or constraints used to ensure consistency during the development period; they may apply to such areas as nomenclature, naming, labelling, coding, computer usage, documentation, procedure, and testing. (DAN 300)

**software experience data**

Data relating to the development or use of software that could be useful in developing models, reliability predictions, or other quantitative descriptions of software. (ANSI/IEEE)

**software factory**

A collected set of tools, methodologies, and a common data base that provide a procedural approach to the successful completion of software projects. (DAN 207)

**software librarian**

The person responsible for establishing, controlling, and maintaining a software library. (ANSI/IEEE)

**software library**

A controlled collection of software and related documentation designed to aid in software development, use, or maintenance. Types include software development library, master library, production library, program library, and software repository. (ANSI/IEEE)

**software life cycle**

The period of time that starts when a software product is conceived and ends when the product is no longer available for use. The software life cycle typically includes a requirements phase, design phase, implementation phase, test phase, installation and check-out phase, operation and maintenance phase, and sometimes retirement phase. (ANSI/IEEE) Contrast with software development cycle.

**software maintenance**

Modification of a software product after delivery to correct faults, to improve performance or other attributes, or to adapt the product to a changed environment. (ANSI/IEEE) See also adaptive maintenance, corrective maintenance.

**software monitor**

A software tool that executes concurrently with another computer program and that provides detailed information about the execution of the other program. (ANSI/IEEE)

---

**software problem report (SPR)**

A report of a perceived problem that the software does not meet the defined requirements or perform intended functions. (RTCA)

---

**software product**

A software entity designated for delivery to a user. (ANSI/IEEE)

---

**software quality**

The totality of features and characteristics of a software product that bear on its ability to satisfy given needs; for example, conform to specifications. (ANSI/IEEE)

---

**software reliability**

- a. The probability that software will not cause the failure of a system for a specified time under specified conditions. The probability is a function of the inputs to and use of the system as well as a function of the existence of faults in the software. The inputs to the system determine whether existing faults, if any, are encountered. (ANSI/IEEE)
- b. The ability of a program to perform a required function under stated conditions for a stated period of time. (ANSI/IEEE)

---

**software repository**

A software library providing permanent, archival storage for software and related documentation. (ANSI/IEEE)

---

**software sneak analysis**

A technique applied to software to identify latent (sneak) logic control paths or conditions that could

inhibit a desired operation or cause an unwanted operation to occur. (ANSI/IEEE)

### software tool

A computer program used to help develop, test, analyze, or maintain another computer program or its documentation; for example, automated design tool, compiler, test tool, maintenance tool. (ANSI/IEEE)

### source code

Code written in the source language (assembly language or HOL) that is in a machine-readable form for input to the assembler or compiler. (RTCA)

### source language

- a. A computer programme expressed in a source language. (ADatP-2)
- b. A language used to write source programs. (ANSI/IEEE)
- c. A language from which statement are translated. (ISO) Contrast with target language. (ANSI/IEEE)

### source listing

A human-readable format of the source code annotated to describe modules, functions, and program flow. (RTCA)

### source programme

A computer programme expressed in a source language. (ADatP-2)

### specification

A document that prescribes, in a complete, precise, verifiable manner, the requirements, design, behavior, or other characteristics of a system or system component. (ANSI/IEEE) See also design specification, formal

specification, functional specification, interface specification, performance specification, requirements specification.

### specification language

A language, often a machine-processable combination of natural and formal language, used to specify the requirements, design, behavior, or other characteristics of a system or system component. (ANSI/IEEE)

### spooler

A system program that permits I/O transfers to be queued for an I/O device, thereby permitting the requesting task to continue executing even when it cannot immediately use the I/O device.

### spooling

An abbreviation for Simultaneous Peripheral Operation On-Line.

### stability

- a. The ability to continue unchanged despite disturbing or disruptive events. (ANSI/IEEE)
- b. The ability to return to an original state after disturbing or disruptive events. (ANSI/IEEE)

### stack

- a. A list that is accessed in a last-in, first-out manner. (ANSI/IEEE) Contrast with queue.
- b. An area of contiguous core store locations.

### standards

Specifications that refer to the method of development procedures, rules, conventions, and

guidelines used for prescribing all or any part of program design, coding, and testing. (RTCA)

#### standards enforcer

A software tool that determines whether prescribed development standards have been followed. The standards may include module size, module structure, commenting conventions, use of certain statement forms, and documentation conventions. (ANSI/IEEE)

#### state diagram

A directed graph in which nodes correspond to internal states of a system, and edges correspond to transitions; often used for describing a system in terms of state changes. (ANSI/IEEE) See also petri net.

#### statement testing

A test phase whose aim is to execute all program statements. The primary value of statement testing rests in the fact that conventional test-set production methods are inadequate and tend to result in up to 40% of any piece of software not being tested (executed at all. In practice, it will often be impossible to execute all statements. Generally, testing of more than 85% of statements is deemed acceptable.

#### static analysis

The process of evaluating a program without executing the program. See also desk checking, code audit, inspection, static analyzer, walk-through. (ANSI/IEEE) Contrast with dynamic analysis.

#### static analyzer

A software tool that aids in the evaluation of a computer program without executing the program. Examples include syntax checkers, compilers, cross-reference generators, standard enforcers, and flowcharters. (ANSI/IEEE) Contrast with dynamic analyzer.



**static binding**

Binding performed prior to execution of a program and not subject to change during execution. (ANSI/IEEE)  
Contrast with dynamic binding.

**static priority**

A form of scheduling in which a task's execution priority is fixed when the task is loaded into memory. This priority remains unchanged as the task executes.

**statistical test model**

A model that relates program faults to the input data set (or sets) which cause them to be encountered. The model also gives the probability that these faults will cause the program to fail. (ANSI/IEEE)

**status accounting**

The process of documenting the current approved status of the system, including a historical record of all approved changes. (RTCA)

**stepwise refinement**

A system development methodology in which data definitions and processing steps are defined broadly at first and then with increasing detail. (ANSI/IEEE)

**strength**

A relative measure of the strength of relationships among the internal components of a module insofar as they contribute to the variation in assumptions made by the outside program concerning the role the module plays in the program. Invariant assumptions about a module indicate high strength. (DAN 1153)

**stress testing**

The test phase in which the system is forced to operate at its maximum processing rate. The type of testing is particularly important for on-line, real-time,

transaction-driven systems. Stress tests designed to comfort programs with abnormal situations. Stress testing executes a system in a manner that demands resources in abnormal quantity, frequency, or volume. A variation of stress testing is a technique called sensitivity testing.

### string

A linear sequence of entities such as characters or physical elements. (ANSI)

### string processing

This includes components which perform operations on lists of characters. (SEL)

### strong typing

A programming language feature that requires the data type of each data object to be declared, and that precludes the application of operators to inappropriate data objects and, thereby, prevents the interaction of data objects of incomparable types. (ANSI/IEEE)

### structured design

A disciplined approach to software design that adheres to a specified set of rules based on principles such as top-down design, stepwise refinement, and data flow analysis. (ANSI/IEEE)

### structured program

A program constructed of a basic set of control structures, each one having one entry point and one exit. The set of control structures typically includes: sequence of two or more instructions, conditional selection of one of two or more instructions or sequences of instructions, and repetition of an instruction or a sequence of instructions. (ANSI/IEEE)

**structured programming**

A well-defined software development technique that incorporates top-down design and implementation and strict use of structured program control constructs. (ANSI/IEEE)

**structured programming language**

A programming language that provides the structured program constructs and that facilitates the development of structured programs. (ANSI/IEEE)

**structured walkthrough**

A generic name given to a set of techniques (i.e., design validation, design verification, and code verification), each with different objectives and each occurring at different times in the software development cycle. (DAN 154)

**structuredness**

Code possesses the characteristic structuredness to the extent that it possesses a definite pattern of organization of its interdependent parts. This implies that evolution of the program design has proceeded in an orderly and systematic manner, and that standard control structures have been followed in coding the program, etc. (DAN 239)

**stub**

- a. A dummy program module used during the development and testing of a higher-level module. (ANSI/IEEE)
- b. A program statement substituting for the body of a program unit and indicating that the unit is or will be defined elsewhere. (ANSI/IEEE)
- c. A program portion which is a substitute for a actual segment. It does not perform the functions of the actual segment, but it provides sufficiently similar input/output behavior to fool the higher-level segment that references it. In most cases, a stub is a

short, basic sequence of code that also provides an execution trace.

#### subprogram

A program unit that may be invoked by one or more other program units. (ANSI/IEEE) Examples are procedure, function, subroutine.

#### subroutine

A sequenced set of statements that may be used in one or more computer programs and at one or more points in a computer program. (ISO)

#### subscript

The numeric key which identifies a particular element in a set of table.

#### subsystem

A group of assemblies or components or both combined to perform a single function. (ANSI N45.2.10-1973)

#### subsystem testing

Subsystem testing is the next step after module testing in the overall testing process where module are put together to form subsystem. As modules cooperate and communicate, subsystem testing should concentrate on testing module interfaces

#### supervisory program

A computer program, usually part of an operating system, that controls the execution of other computer programs and regulates the flow of work in a data processing system. (ISO) Synonymous with executive program, supervisor.

**support software**

All programs used in the development and maintenance of the delivered operational programs and test/maintenance programs. Support programs include, but are not limited to:

- a. Compilers, assemblers, emulators, builders, and loaders required to generate machine code and to combine subprograms or components into a complete computer program.
- b. Debugging programs.
- c. Stimulation and simulation programs used in operator training sites.
- d. Data abstraction and reduction programs applicable to operational programs used for management control, configuration management or document generation and control during development. (DAN 384)

**support tools**

The set of software tools and procedures used as aids in software development. (DAN 1201)

**suspended**

A task state in which task is discontinued pending the occurrence of a event.

**swapping**

At intervals determined by the operating system, usually dictated by cpu-scheduling policies, copying jobs from main memory to a backing store and later copying back to main memory. This scheme allows more jobs to be run than can be fit into memory at one time. This is a feature of an operating system which permits suspended tasks to be moved to secondary storage in order to generate enough memory space so that the next task on the Ready List can be loaded into memory (if it is not already resident) and executed. The task will be swapped back into system memory when it is scheduled to resume execution. This movement to and from secondary storage

may occur many times to a given task before its execution is complete.

### symbolic execution

A verification technique in which program execution is simulated using symbols rather than actual values for input data, and program outputs are expressed as logical or mathematical expressions involving these symbols.  
(ANSI/IEEE)

### synchro

Any of several devices used for transmitting and receiving angular position or angular motion over wires, such as a synchro transmitter or synchro receiver.

### synchronization

The process of setting up a mechanism to perform operations or procedures in a time or sequence hierarchy.  
(DAN 1153)

### synchronous

- a. Occurring with a regular or predictable time relationship.
- b. The performance of a sequence of operations under the control of a cycle of equally spaced signals generated by a clock.

### syntactic

Concerning the rules governing the relations of words in language structures.

### syntactic analysis

During this stage, the source statements are recognized as language constructs described by the grammar being used. This process is a kind of building the parse tree for the statements being translated.

---

**syntax**

- a. The relationship among characters or groups of characters, independent of their meanings or the manner of their interpretation and use. (ISO)
- b. The structure of expressions in a language. (ANSI)
- c. The rules governing the structure of a language. (ANSI)

---

**system**

- a. An assembly of methods, procedures, or techniques united by regulated interaction to form an organized whole. (ADatP-2)
- b. An organized collection of men, machines, and methods required to accomplish a set of specific functions. (ADatP-2)

---

**system architecture**

The structure and relationship among the components of a system. The system architecture may also include the system's interface with its operational environment. (ANSI/IEEE)

---

**system design**

- a. The process of defining the hardware and software architectures, components, modules, interfaces, and data for a system to satisfy specified system requirements. (ANSI/IEEE)
- b. The result of the system design process. (ANSI/IEEE)

---

**system design language**

A computer program used to provide an understandable representation of the software design as it evolves. These programs allow designs to be constructed and

expanded in a hierarchical fashion. They document the design and the decisions that led to it. (DAN 134)

#### system design review

A review conducted to evaluate the optimization, traceability, correlation, completeness, and risk associated with the establishment of the allocated requirements (allocated configuration identification), including the corresponding test requirements. The objective is to assure they fulfil the system and/or system segment requirements (the functional configuration identification).

#### system documentation

Documentation conveying the requirements, design philosophy, design details, capabilities, limitations, and other characteristics of a system. (ANSI/IEEE)  
Contrast with user documentation.

#### system library

A controlled collection of system resident software that can be accessed for use or incorporated into other programs by reference, for example, a group of routines that a linkage editor can incorporate into a program as required. (ANSI/IEEE) See also software library.

#### system reliability

The probability that a system, including all hardware and software subsystems, will perform a required task or mission for a specified time in a specified environment. (ANSI/IEEE)

#### system requirement review

A review conducted when a significant portion of the system functional requirements have been established in order to determine the contractor's progress and responsiveness to the requirements of the system specification and the Statement of Work (SOW). Software aspects of this review are limited to the allocation of requirements to computer programs and the resulting functional interfaces.



**system software**

Software designed for a specific computer system or family of computer systems to facilitate the operation and maintenance of the computer system and associated programs, for example, operating systems, compilers, utilities. (ANSI/IEEE) Contrast with application software.

**system testing**

The process of testing an integrated hardware and software system to verify that the system meets its specified requirements. (ANSI/IEEE)

**systems analyst**

One who investigates the data processing of an organization to define how it may best be performed by a computer, and writes the specifications from which the computer programs are written.

**systems engineering**

The iterative technical management process which transforms an operational need into a description of parameters, and integrates these parameters amongst all elements to produce an optimally balanced design.

**systems integration**

The process of interconnecting functional elements to form an organized whole.

**systems programmer**

A programmer who writes generalized routines designed to be used in all the installations of a particular computer, such as compilers, utility routines, sort programs, etc.

**table**

- a. A collection of data in which each item is uniquely identified by a label, by its position relative to the other items, or by some other means. (ANSI)
- b. A set of data organized so that each element may be uniquely identified by a key held in the element, or by the position of the element in the table.

**tallying**

Counting, or keeping count.

**target language**

A language into which source statements are translated. (ANSI) Contrast with source language.

**target machine**

The computer on which a program is intended to operate. (ANSI/IEEE)

**task**

A software module in which code is executed in a sequential manner.

**task control block**

Another term for a Task Descriptor.

**task descriptor**

A block of task information that is manipulated by the system and often stored in system tables to increase scheduling efficiency. This information includes the task name, number, priority, and state, as well as other pertinent status data.

**task list**

A system data structure containing a list of tasks within the system.

**task state**

The status of a task; it may be undefined, ready to execute, executing, or suspended awaiting some event.

**terminal**

- a. A point in a system or communication network at which data can either enter or leave. (ADatP-2)
- b. In a data bus system, the electronic module necessary to connect the data bus with the sub-system. (Terminals may exist as separate line replaceable units (LRUs) or be contained with the elements of the sub-system.) See also remote terminal.

**termination proof**

In proof of correctness, the demonstration that a program will terminate under all specified input conditions. (ANSI/IEEE)

**ternary element**

A constituent element of data which may assume one of only three values.

**test bed**

A test environment containing the hardware, instrumentation tools, simulators, and other support software necessary for testing a system or system component. (ANSI/IEEE)

**test case**

A specific set of test data and associated procedures developed for a particular objective, such as to exercise a particular program path or to verify compliance with a specific requirement. (ANSI/IEEE)

**test data**

Data developed to test a system or system component. (ANSI/IEEE)

**test data generators**

An automated analysis system which assists a user in selecting test data that cause a program to behave in a particular fashion.

**test driver**

A driver that invokes the item under test and that may provide test inputs and report test results. (ANSI/IEEE)

**test log**

A chronological record of all relevant details of a testing activity. (ANSI/IEEE)

**test plan**

A document prescribing the approach to be taken for intended testing activities. The plan typically identifies the items to be tested, the testing to be performed, test schedules, personnel requirements, reporting requirements, evaluation criteria, and any risks requiring contingency planning. (ANSI/IEEE)

**test procedure**

Detailed instructions for the setup, operation, and evaluation of results for a given test. A set of associated procedures is often combined to form a test procedures document. (ANSI/IEEE)

**test repeatability**

An attribute of a test indicating whether the same results are produced each time the test is conducted.  
(ANSI/IEEE)

**test report**

A document describing the conduct and results of the testing carried out for a system or system component.  
(ANSI/IEEE)

**test validity**

The degree to which a test accomplishes its specified goal. (ANSI/IEEE)

**testability**

The extent to which software facilitates both the establishment of test criteria and the evaluation of the software with respect to those criteria. (ANSI/IEEE)

**testing**

- a. The primary goal of testing ought to be to uncover defects (read bugs) within the programs and system. That is, to find errors not known to exist in the program or system.
- b. A testing team consisting of the members of the developing team and outside people, such as members of the quality assurance team and the user department, is likely to be more productive in terms of uncovering the maximum number of defects, compared to a testing team consisting entirely of development team members. Developers will have a major role in the module testing phase, whereas the people representing the user department will have a major rule in the acceptance testing phase.
- c. Testing uncovers bugs harboured by software, in which sense they prove the presence of defects

within the program. Testing do not and cannot prove the absence of defects.

- d. Testing should not be considered as a means of ensuring program correctness and it should be optimal, avoiding both undertesting and overtesting.
- e. Testing establishes what the software (or system) can do and what it cannot do. It demonstrates that the software works from both the functional point of view and the performance point of view. Working from the functional point of view means that the software performs its function accurately and reliably. Working from the performance point of view means that the performance of the system is within the reasonable limits.
- f. Testing also demonstrates that interfaces between programs and between different systems function properly.
- g. The process of exercising or evaluating a system or system component by manual or automated means to verify that it satisfies specified requirements or to identify differences between expected and actual results. (ANSI/IEEE) Compare with debugging.

### throughput

The amount of data which is processed by a system in a given time expressed in terms meaningful to that system. (ADatP-2)

### time division multiplex (TDM)

The transmission of information from several signal sources through one communication channel with different signal samples phased in time.

### time sharing

- a. An operating technique of a computer system that provides for the interleaving in time of two or more processes in one processor. (ISO)

- b. Pertaining to the interleaved use of time on a computing system that enables two or more users to execute computer programs concurrently.  
(ANSI)

### time slice

A period of time, normally 1-50 milliseconds, allocated to each task in a time sharing system.

### timer

A hardware device that supplies interrupts on an interval of time-of-day basis to the operating system.

### timing

Estimate or measurement of the amount of computation time required or actually utilized. Usually compared with the available time for computation to determine timing margin. (RTCA)

### timing analyzer

A software tool that estimates or measures the execution time of a computer program or portions of a computer program either by summing the execution times of the instructions in each path, or by inserting probes at specific points in the program and measuring the execution time between probes. (ANSI/IEEE)

### tolerance

The ability of a system to provide continuity of operation under various abnormal conditions. (ANSI/IEEE)

### top-down

Pertaining to an approach that starts with the highest level component of a hierarchy and proceeds through progressively lower levels; for example, top-down design, top-down programming, top-down testing.  
(ANSI/IEEE) Contrast with bottom-up.

---

**top-down design**

The process of designing a system by identifying its major components, decomposing them into their lower level components, and iterating until the desired level of detail is achieved. (ANSI/IEEE) Contrast with bottom-up design.

---

**top-down testing**

- a. Implementation of the top-down approach to testing requires that program stubs, or dummy routines, be created for segments that are referred to by name in higher-level segments under test but are not yet coded.
- b. The process of checking out hierarchically organized programs, progressively, from top to bottom, using simulation of lower level components. (ANSI/IEEE)

---

**total correctness**

In proof of correctness, a designation indicating that a program's output assertions follow logically from its input assertions and processing steps, and that, in addition, the program terminates under all specified input conditions. (ANSI/IEEE) Contrast with partial correctness.

---

**trace (n)**

- a. A record of the execution of a computer program; it exhibits the sequences in which the instructions were executed. (ANSI)
- b. A record of all or certain classes of instructions or program events occurring during execution of a computer program. (ANSI/IEEE)

---

**trace (v)**

To produce a trace. (ANSI/IEEE)



**trace routine**

A diagnostic aid which monitors the progress of a program by pointing out the contents of specified areas, signalling all transfers of control and so on.

**traceability**

The ability to trace or associate each function between the various documentation levels which define the system and software requirements, test and design.  
(RTCA)

**tracer**

A software tool used to trace. (ANSI/IEEE)

**translator**

A program that transforms a sequence of statements in one language into an equivalent sequence of statements in another language. (ANSI/IEEE)

**trap**

A technique in which the logic flow of a program is interrupted for the purpose of setting aside interim results for test measurement or performing special test functions. (DAN 134)

**tree**

An abstract hierarchical structure consisting of nodes connected by branches, in which:

- a. each branch connects one node to a directly subsidiary node, and
- b. there is a unique node called the root that is not subsidiary to any other node, and
- c. every node besides the root is directly subsidiary to exactly one other node.  
(ANSI/IEEE)

**truncation**

The deletion of digits from one end or another of a number, e.g. because it is too big to hold in the storage location allocated to it.

**turn around time**

The interval between the time a program/job is submitted and the time it is completed.

**twisted shielded pair (TSP)**

A type of electronic cable which is a pair of insulated wires, uniformly twisted together. These are in turn covered by an electrostatic shield and the assembly in turn is covered by an appropriate insulating sheath.

**undefined**

A task state in which the task is unknown to the operating system. This means that the task descriptor information has not been entered into the system tables. A task is in this state before it has been created and after it has been deleted from the system.

**usability**

The effort required to learn, operate, prepare input, and interpret the output of a program.

**user documentation**

Documentation conveying to the end user of a system instructions for using the system to obtain desired results; for example, a user's manual. (ANSI/IEEE)  
Contrast with system documentation.

**utility software**

Computer programs or routines designed to perform some general support function required by other

application software, by the operating system, or by system users. (ANSI/IEEE)

### utilization

The fraction of time that a resource is in use.

### validation

The process of evaluating software at the end of the software development process to ensure compliance with software requirements. (ANSI/IEEE)

### variable

In programming, a character or group of characters that refers to a value and, in the execution of a computer program, corresponds to an address. (ANSI)

### verification

- a. The process of determining whether or not the products of a given phase of the software development cycle fulfil the requirements established during the previous phase. (ANSI/IEEE)
- b. The act of reviewing, inspecting, testing, checking, auditing, or otherwise establishing and documenting whether or not items, processes, services, or documents conform to specified requirements. (ANSI/ASQC A3-1978)

### version

An identified and documented body of software.

### very large scale integration (VLSI)

Technology by which tens of thousands of electronic devices are fabricated on a single semiconductor chip.

**virtual cockpit**

A cockpit display where physical displays and controls are replaced by sensory illusions based on the use of computer graphics, computer driven audio systems, and artificial tactile sensors.

**virtual machine**

A functional simulation of a computer and its associated devices. (ANSI/IEEE)

**virtual memory**

An apparent memory seen by the programmer which frees him from the size and field limitations of the physical memory and which uses a logical storage addressing scheme to link addresses between the apparent and physical memories.

**volatile storage**

A storage device in which stored data are lost when the applied power is removed, e.g., an acoustic delay line. (ADatP-2)

**wait List**

A list of tasks waiting for an event or message at a mailbox. Each mailbox has a separate wait list. If a message is present in the mailbox task. On the other hand, if no message is in the mailbox or if there are tasks currently in the wait list for the mailbox, the requesting task is suspended and placed at the end of the wait list.

**wakeup**

The act of making a task ready to run after a period of suspension.

**walk-through**

A review process in which a designer or programmer leads one or more other members of the development team

through a segment of design or code that he or she has written, while the other members ask questions and make comments about technique, style, possible errors, violation of development standards, and other problems. (ANSI/IEEE) Contrast with inspection.

word

See computer word.

word length

A measure of the size of a word, usually specified in units such as bytes or bits.

CHAPTER/CHAPITRE 2TERMES ET DEFINITIONSFRANCAIS FRENCH TERMS ANDDEFINITIONSabandonner

Mettre fin à un traitement avant qu'il ne soit terminé. (ANSI/IEEE)

abstraction de données

Résultat obtenu lorsqu'on extrait et retient uniquement les propriétés caractéristiques essentielles des données en définissant les types de données spécifiques et les caractéristiques fonctionnelles connexes, séparant et "cachant" ainsi les détails de représentation. (ANSI/IEEE)

accéder

Obtenir une information sur un support de rangement ou une ancienne mémoire à tores.

accès direct

- a. Recherche ou enregistrement d'une donnée en mémoire en fonction de son adresse et ne dépendant pas des données recherchées ou enregistrées auparavant. (ADatP-2)
- b. Se dit d'une mémoire dont le temps d'accès est indépendant de l'adresse. (ADatP-2)

accès direct mémoire (DMA)

Méthode utilisée pour alimenter en données les emplacements de mémoire ou pour extraire des données de ces emplacements sans intervention de l'unité centrale.

**accès sélectif**

Mode d'accès suivant lequel des enregistrements logiques donnés sont recherchés ou placés dans un fichier de mémoire de masse de manière non séquentielle. (ADatP-2)

**accès séquentiel**

- a. Qualifie une transmission séquentielle ou consécutive de données vers une mémoire ou à partir de celle-ci. (ADatP-2)
- b. Qualifie une méthode de rangement en mémoire ou de lecture d'une donnée dans laquelle le temps d'accès à cette donnée dépend de l'emplacement d'une autre donnée (généralement précédente). (ADatP-2)

**accessibilité**

Degré d'aptitude du logiciel à faciliter l'utilisation ou l'entretien sélectif de ses éléments. (ANSI/IEEE)

**accumulateur**

Registre où est formé le résultat d'une opération arithmétique ou logique.

**acronyme**

Abréviation, comme UNO, NATO, ALGOL, COBOL, formée des premières lettres d'un nom de plusieurs mots.

**Ada**

Langage de programmation de calculateur utilisé particulièrement pour des applications militaires.

**adaptabilité**

Facilité avec laquelle le logiciel tient compte des diverses contraintes des systèmes et répond aux besoins des utilisateurs. (ANSI/IEEE)

**adresse**

- a. Caractère, ou groupe de caractères, qui identifie un registre, une zone de mémoire précise ou toute autre source ou destination de données. (ISO)
- b. Adresser : Se reporter à un périphérique ou à une donnée élémentaire. (ISO)

**adresse de base**

Adresse donnée dont on tire une adresse absolue par combinaison avec une adresse relative. (ADatP-2)

**adresse logique**

Moyen d'adressage de données et/ou d'instructions de programme de façon qu'elles soient indépendantes de la structure physique de la mémoire d'une machine donnée. Quand le programme est exécuté, cette adresse logique est traduite sur la machine d'exécution en une adresse physique dans la structure de la machine.

**adresse physique**

Adresse d'un emplacement de mémoire d'un système existant.

**affectation de mémoire dynamique**

Affectation de mémoire, comme des zones de travail, à un programme uniquement en cas de besoin et non pas d'une façon permanente au cours de l'exécution du programme.



**affectation des ressources**

Fonction du système d'exploitation qui attribue les ressources aux tâches qui les demandent. Les ressources telles que les périphériques (imprimantes, terminaux) et la mémoire centrale sont affectées aux tâches en fonction de leur disponibilité.

**affectation dynamique**

Attribution d'une mémoire adressable et d'autres ressources à un programme durant son exécution.  
(ANSI/IEEE)

**affichage à cristaux liquides (LCD)**

Dispositif utilisant la technologie des cristaux liquides pour la visualisation des informations.

**affichage multifonction**

Affichage électro-optique qui peut fournir des informations que l'on sélectionne, généralement sous forme de trames, de lignes ou de tableaux.

**aide au diagnostic**

Programme qui aide à la mise au point d'un programme à l'essai, par exemple : imprimer le contenu de zones de mémoire définies lors de l'exécution du programme ou quand celui-ci est déficient.

**ajusteur**

Algorithme pour affectation de mémoire qui explore la liste des blocs de mémoire pour trouver le bloc qui est le plus proche en taille de celui qui est demandé pour la fonction.

**ALGOL**

Acronyme de Algorithmic Language, langage de programmation pour la définition de problèmes scientifiques et mathématiques.

**algorithme**

- a. Ensemble fini de règles déterminées servant à résoudre un problème au moyen d'un nombre fini d'opérations. Exemple : énoncé complet de la suite d'opérations arithmétiques permettant le calcul du sinus de x avec une précision donnée. (ADatP-2)
- b. Ensemble fini de règles déterminées servant à résoudre un problème au moyen d'un nombre fini d'opérations. Exemple : spécification complexe d'une séquence d'opérations arithmétiques pour calculer le sinus de x avec une précision donnée. (ISO)
- c. Ensemble fini de règles déterminées donnant une séquence d'opérations pour effectuer une tâche particulière. (ANSI/IEEE)
- d. Un programme ou un sous-programme peut être un algorithme, ou bien peut consister en un certain nombre d'algorithmes.

**alignement**

Décalage des opérandes impliqués dans une opération arithmétique, de telle façon qu'ils soient dans la même position relative au sein des mots machines.

**amorce**

- a. Court programme machine intégré dans l'ordinateur, ou qui peut y être chargé facilement, dont le déroulement permet l'introduction d'un programme plus important, tel qu'un système d'exploitation ou son chargeur, dans la mémoire. (ANSI/IEEE)
- b. Jeu d'instructions qui entraîne le chargement d'instructions additionnelles jusqu'à ce que l'ensemble du programme machine soit en mémoire. (ISO)
- c. Technique ou dispositif conçu pour se mettre lui-même dans l'état désiré. Il peut, par exemple, s'agir d'un programme machine dont les

premières instructions sont suffisantes pour lui permettre d'entraîner le reste du programme dans l'ordinateur à partir d'un périphérique d'entrée. (ANSI)

- d. Partie d'un programme machine utilisée pour élaborer une autre version de ce programme. (ANSI)

#### amorcer

Utiliser une amorce. (ISO)

#### analogique

Qualifie des données représentées au moyen d'une grandeur physique variant de façon continue. (ADatP-2)

#### analyse d'erreur

Examen d'un défaut du logiciel pour en découvrir la cause, la phase du processus de développement durant laquelle il s'est introduit, les méthodes qui auraient pu le prévenir ou permettre de le détecter plus tôt et la méthode qui a permis de le détecter. (ANSI/IEEE)

#### analyse d'un algorithme

Examen d'un algorithme pour déterminer son exactitude par rapport à l'usage que l'on veut en faire, pour déterminer ses caractéristiques opérationnelles ou pour mieux le comprendre afin de le modifier, de le simplifier ou de l'améliorer. (ANSI/IEEE)

#### analyse de code dynamique

Voir analyse dynamique.

#### analyse des branches

Analyse d'un programme permettant de relever toutes les branches possibles dans ce programme, de repérer les branches incomplètes ou de découvrir les parties du

programme qui ne se trouvent dans aucune branche.  
(ANSI/IEEE)

### analyse des exigences

Etude des besoins de l'utilisateur pour en arriver à une définition des exigences du système ou du logiciel.  
(ANSI/IEEE)

### analyse des performances

Evaluation technique d'un système ou d'un élément de système pour déterminer dans quelle mesure les objectifs d'exploitation ont été atteints. (ANSI/IEEE)

### analyse des vices cachés d'un logiciel

Technique appliquée au logiciel pour repérer les voies ou conditions logiques latentes qui pourraient empêcher l'exécution d'une opération voulue ou produire l'effet contraire. (ANSI/IEEE)

### analyse dynamique

Processus d'évaluation d'un programme fondé sur l'exécution de ce programme. (ANSI/IEEE) En contraste avec analyse statique.

### analyse statique

Evaluation d'un programme sans exécution du programme. Voir également contrôle de programmation, vérification de code, inspection, analyseur statique et revue de projet. (ANSI/IEEE) En contraste avec analyse dynamique.

### analyse syntaxique

A ce stade, les instructions sources sont identifiées comme des structures de langage décrites par la grammaire employée. Ce processus revient à reconstituer par l'analyse l'arbre des instructions en cours de traduction.

**analyser**

Déterminer la structure syntaxique d'une unité de langage artificielle ou naturelle en décomposant cette unité en sous-unités plus élémentaires et en établissant des relations entre elles; par exemple, des blocs, des instructions et des expressions doivent être décomposées en instructions, en expressions et en opérateurs et opérandes. (ANSI/IEEE)

**analyseur de conception**

Outil de conception automatisé qui reçoit les informations sur la conception d'un programme et produit des sorties telles que des schémas hiérarchiques modulaires, des représentations graphiques de structures de contrôle et de données et des listes de blocs de données sollicités. (ANSI/IEEE)

**analyseur de durée**

Outil logiciel qui évalue ou mesure la durée d'exécution d'un programme ou de parties d'un programme machine en additionnant les durées d'exécution des instructions dans chaque branche ou en insérant des sondes à des points déterminés du programme et en mesurant la durée d'exécution entre les sondes. (ANSI/IEEE)

**analyseur dynamique**

Outil logiciel qui aide à évaluer un programme machine en surveillant son exécution. Les instruments, les moniteurs logiciels et les programmes de traçage en sont quelques exemples. (ANSI/IEEE) En contraste avec analyseur statique.

**analyseur statique**

Outil logiciel qui facilite l'évaluation d'un programme machine sans exécution de programme. Exemples : contrôleurs de syntaxe, compilateurs, générateurs de correspondances, logiciels de contrôle des normes d'élaboration et traceurs d'organigrammes. (ANSI/IEEE) En contraste avec analyseur dynamique.

**analyste de système**

Personne qui étudie le traitement de données d'une organisation pour déterminer la meilleure manière de le faire effectuer par un ordinateur et rédige les spécifications à partir desquelles les programmes d'ordinateur sont écrits.

**anomalie**

Evénement qui entraîne la suspension de l'exécution normale d'un programme. (ANSI/IEEE)

**antémémoire**

Mémoire tampon de petite capacité mais très rapide, située entre le processeur et la mémoire centrale d'un système informatique. Elle fonctionne sur le principe que l'on sollicite très souvent certaines adresses de la mémoire (normalement pour les lectures). Ainsi, quand on lit une adresse de la mémoire centrale, celle-ci est stockée en même temps dans l'antémémoire. Toute autre référence de lecture à cette adresse est automatiquement dirigée vers l'antémémoire. Un point d'accès d'écriture figure généralement à la fois dans la mémoire centrale et dans l'antémémoire. Puisqu'une antémémoire représente plusieurs emplacements de mémoire centrale non continus, des registres à contenu adressable sont généralement utilisés pour déterminer quand une adresse de la mémoire centrale est actuellement stockée dans l'antémémoire.

**appareil terminal**

Dans un système de bus de données, tous les terminaux qui ne fonctionnent ni comme moniteur de bus, ni comme contrôleur de bus.

**appel sélectif**

Méthode de contrôle et de commande à partir d'un ordinateur central. Utilisée quand plusieurs terminaux à distance se partagent la même ligne de transmission, elle consiste à les interroger à tour de rôle en respectant les priorités établies. (ADatP-2)

**apt**

Acronyme de Automatically Programmed Tools, langage de programmation pour la commande numérique des machines-outils.

**arbre**

Structure hiérarchique abstraite formée de noeuds reliés par des branchements, dans laquelle :

- a. chaque branchement relie un noeud à un autre noeud qui est son fils immédiat, et
- b. il n'y a qu'un noeud, appelé la racine, qui ne soit pas le fils d'un autre noeud, et
- c. tout noeud, à part la racine, est le fils immédiat d'un autre noeud. (ANSI/IEEE)

**architecture**

En avionique, représentation des éléments matériels et logiciels d'un système et de leurs relations, en considérant le système dans son ensemble.

**architecture d'ordinateur**

Définition structurelle et fonctionnelle d'un ordinateur relativement à l'ensemble de ses instructions machine et de ses capacités entrée/sortie. (GLOS-1)

**architecture de jeu d'instructions (ISA)**

Attributs d'un calculateur numérique vu par un programmeur qui utilise un langage machine ou un langage d'assemblage. L'ISA comprend les unité centrale, entrée/sortie, jeux d'instructions et formats, codes d'exploitation, mode d'adressage, gestion et répartition de la mémoire (si elle est accessible au programmeur), vitesse des horloges accessibles, structure d'interruption, mode d'utilisation et format de tous les registres et emplacements de mémoire qui peuvent être directement manipulés ou testés. Cette définition exclut les durée ou vitesse de toute opération, découpage interne du calculateur, organisation physique et électrique, circuits ou composants du calculateur,

technologie de fabrication, organisation de la mémoire, cycle de base de la mémoire et largeur des bus de la mémoire.

### architecture de programme

Structure d'un programme machine et relations entre ses éléments. L'architecture du programme peut également comprendre l'interface du programme avec son cadre d'exploitation. (ANSI/IEEE)

### architecture du système

Structure du système et relation entre ses éléments. L'architecture du système peut également comprendre l'interface du système avec son cadre d'exploitation. (ANSI/IEEE)

### architecture fédérée

Architecture de système informatique où deux ordinateurs, ou plus, exécutent des tâches sous le contrôle souple d'un programme directeur centralisé.

### architecture répartie

Architecture de système selon laquelle les fonctions sont réparties en unités séparées physiquement.

### arrondi

Troncature de chiffres à l'extrémité la moins significative d'un nombre; en outre, lorsque l'on arrondit par excès, on augmente le dernier chiffre restant de un si le premier chiffre tronqué (décimale) est égal ou supérieur à 5. Ainsi, si le chiffre 476,58 est arrondi au chiffre fractionnel supérieur, il devient 476,6.

### ascendant

Terme se rapportant à une approche qui commence par les éléments logiciels du niveau le plus bas dans une hiérarchie et qui franchit progressivement les autres niveaux jusqu'à l'élément du niveau le plus élevé.



Exemple : programmation ascendante ou essai ascendant.  
(ANSI/IEEE) Antonyme de descendant.

### assembler

- a. Préparer un programme de langage machine à partir d'un programme de langage symbolique, en substituant des codes absolus à des codes d'opérations symboliques, des adresses absolues ou translatables à des adresses symboliques, y compris, dans le langage machine, des programmes qui peuvent être appelés d'une bibliothèque de programmes. (ADatP-2)
- b. Traduire en langage machine un programme écrit en langage d'assemblage et éventuellement relier les sous-programmes. L'assemblage est effectué habituellement en substituant le code d'opérations du langage machine au code d'opérations du langage d'assemblage et en substituant des adresses absolues, des adresses immédiates, des adresses translatables ou des adresses virtuelles ou des adresses virtuelles aux adresses symboliques. (ANSI/IEEE) Par contraste avec compiler, interpréter.

### assembleur

- a. Programme machine qui assemble. (ADatP-2)
- b. Programme machine destiné à assembler. (ISO)  
Par contraste avec compilateur et interpréteur.

### assembleur croisé

Assembleur exécuté sur un ordinateur, mais qui produit un code résultant pour un ordinateur différent.  
(ANSI/IEEE)

### assertion

Expression logique qui précise l'état d'un programme qui doit exister ou un groupe de conditions auxquelles les variables d'un programme doivent répondre à un moment précis durant l'exécution du programme. Par exemple, A est positif et plus grand que B. (ANSI/IEEE)

**assertion d'entrée**

Expression logique spécifiant une ou plusieurs conditions auxquelles les entrées d'un programme doivent satisfaire pour être valides. (ANSI/IEEE)

**assertion de sortie**

Expression logique spécifiant une ou plusieurs conditions auxquelles les sorties de programme doivent satisfaire pour que le programme soit correct. (ANSI/IEEE)

**association**

Attribution d'une valeur ou d'une référence à un identificateur. Il peut, par exemple, s'agir de l'attribution d'une valeur à un paramètre ou de l'attribution d'une adresse absolue, d'une adresse virtuelle ou d'un identificateur de périphérique à une adresse symbolique ou à un label dans un programme machine. (ANSI/IEEE)

**association dynamique**

Association effectuée durant l'exécution d'un programme. (ANSI/IEEE) Antonyme d'association statique.

**assurance de la qualité**

Modèle planifié et systématique de toutes les mesures nécessaires pour assurer que l'article ou le produit est conforme aux exigences techniques établies. (ANSI/IEEE STD 730-1981)

**asynchrone**

Se rapporte à un événement ou à un dispositif qui n'est pas contrôlé par une horloge commune.

**attente**

- a. Temps compris entre le moment où une adresse est reconnue et le moment où débute le transfert de l'information de son emplacement

en mémoire vers l'unité centrale. Ce temps comprend le délai nécessaire pour accéder aux mémoires extérieures (par exemple disques et tambours). (ADatP-2)

- b. En avionique, dans un système de bus de données, délai s'écoulant entre l'envoi de la commande de transmission de données émanant du contrôleur de bus et la réception de cette donnée par le terminal ou le sous-système approprié.

#### attribution

Service du système d'exploitation utilisé par une tâche afin de demander une zone de mémoire pour un message ou des tampons Entrée/Sortie.

#### autocode

Langage informatique qui peut être similaire à un langage naturel, comme l'Anglais ou l'algèbre, et est orienté vers le problème plutôt que vers la machine.

#### automate à nombre fini d'états

Modèle de calcul comportant un nombre fini d'états et des transitions entre ces états. (ANSI/IEEE)

#### autoroute de l'information

Voie principale par laquelle les signaux transitent depuis l'une des sources jusqu'à l'une des destinations.

#### avionique

Electronique appliquée à l'aviation.

#### avionique centrale

Portion d'un système d'avionique applicable à une large gamme de missions opérationnelles.

**avionique de mission**

Portion d'un système d'avionique qui se rapporte uniquement à une mission opérationnelle définie. Opposée à avionique centrale.

**banc d'essai**

- a. Cadre d'essai contenant le matériel, les outils d'essai, les simulateurs et d'autres logiciels de soutien nécessaires à l'essai d'un système ou d'un élément d'un système. (ANSI/IEEE)
- b. Essai conçu pour évaluer les performances d'un programme machine selon des critères définis.

**base de données**

- a. Ensemble de données, formant tout ou partie d'un autre ensemble de données, et constitué au moins par un fichier, défini pour une application ou un système de traitement de l'information déterminé. (ISO/ADatP-2)
- b. Groupe de données fondamentales pour un système. (ANSI/IEEE)
- c. Groupe de données fondamentales pour une entreprise. (ANSI/IEEE)
- d. Structure organisée de grandes quantités de données auxquelles on peut avoir accès par différents programmes utilisant différents sous-ensembles des données.

**base de données logicielles**

Fichier centralisé de définitions de données et de valeurs présentes de données communes à un système logiciel opérationnel et qui se trouvent dans un tel système. (ANSI/IEEE)

**base de produit**

Activité marquante qui définit le système de logiciel produit et fournit une base pour sa modification

en corrigeant les erreurs et en incluant les nouvelles exigences.

### base fonctionnelle

La base fonctionnelle est normalement établie par la spécification du segment/système authentifiée à la fin de la phase de conception.

### bibliothèque d'élaboration de logiciels

Bibliothèque de logiciels contenant des informations relatives à l'élaboration de logiciels et pouvant être lues par des ordinateurs et par des personnes. (ANSI/IEEE)

### bibliothèque de logiciels

Collection de logiciels et de documents connexes destinée à aider à l'élaboration, à l'utilisation ou à la maintenance de logiciels. Mentionnons, entre autres, la bibliothèque d'élaboration de logiciels, la bibliothèque générale, la bibliothèque de production, la bibliothèque de programmes et le dépôt de logiciels. (ANSI/IEEE)

### bibliothèque de production

Bibliothèque contenant des logiciels approuvés pour l'exploitation courante. (ANSI)

### bibliothèque des programmes

Groupe de programmes d'ordinateurs classés dans un ordre déterminé. Ces programmes doivent être assortis d'une documentation qui leur permette d'être utilisés par n'importe quel programmeur ou analyste (et pas seulement par le programmeur qui en est l'auteur). (ANSI)

### bibliothèque du système

Collection contrôlée de logiciels résidant dans le système qui peuvent être sollicités pour être utilisés dans d'autres programmes par renvoi; par exemple, un groupe de routines qu'un éditeur de liens peut intégrer

dans un programme au besoin. (ANSI/IEEE) Voir également bibliothèque de logiciels.

### bibliothèque principale

Bibliothèque de logiciels contenant de la documentation et des versions de logiciels officiellement diffusées. (ANSI/IEEE) En contraste avec bibliothèque de production.

### binaire

- a. Qualifie un objet, une condition ou une action pouvant présenter l'un quelconque de deux valeurs ou états distincts.
- b. Qualifie une numération à base fixe dont la base est deux. (AdatP-2)

### bit

Contraction de "binary digit".

### bit de parité

Bit de contrôle annexé à un groupe de chiffres binaires pour rendre la somme de tous les chiffres binaires, y compris le bit de contrôle, toujours paire ou impaire.

### bit de plus faible poids (LSB)

Dans un mot, bit représentant la puissance la plus faible de la base de numération.

### bit de plus fort poids

Dans un mot, bit représentant la puissance la plus élevée de la base de numérotation.

**bit de signe**

Dans un mot-machine, bit, placé en premier ou en dernier, qui, par son positionnement, indique si le contenu numérique du mot est positif ou négatif.

**bits utiles**

En transmission, cette expression désigne les bits qui proviennent d'une source de données et qui ne sont pas utilisés comme bits de contrôle par le système de transmission.

**bloc**

Unité physique de données sur un support de stockage magnétique, comme une bande ou un disque.

**bloc de commande de tâche**

Autre appellation du descripteur de tâche.

**bloc de programme**

Dans les langages orientés vers les problèmes, subdivision de programme machine qui sert à grouper des instructions connexes, à délimiter des programmes, à spécifier l'espace mémoire alloué et à décrire l'applicabilité des labels ou des segments de branche du programme machine à d'autres fins. (ANSI)

**bloc début-fin**

Suite d'instructions de conception ou de programmation délimitée par des bornes de début et de fin et caractérisée par une seule entrée et une seule sortie. (ANSI/IEEE)

**bloc fonctionnel**

Unité ou module individuel utilisé par les programmes ou modules de niveau plus élevé. (ANSI/IEEE)

**blocage**

Situation où toutes les tâches dans un système sont suspendues en attendant des ressources qui ont déjà été affectées à d'autres tâches également en attente de ressources supplémentaires. Ainsi, le système ne peut effectuer de travail utile à moins que les tâches ne soient supprimées et leurs ressources récupérées.

**boque**

- a. Faute ou mauvais fonctionnement. (ADatP-2)
- b. Repère sur un affichage.
- c. Voir défaut.

**boîte à lettres**

Structure de données d'un système chargée de la communication des tâches. Les tâches envoient des messages aux boîtes à lettres et en reçoivent de celles-ci.

**boîte noire**

- a. Terme générique utilisé pour décrire un dispositif non précisé qui effectue une fonction particulière ou dans lequel l'introduction de données connues fournit des résultats connus dans une relation fixée.
- b. Terme générique, pour toute unité ou dispositif intégré mais très souvent utilisé pour faire référence à des dispositifs électroniques.  
(Dictionnaire informatique)

**booléen**

Signifiait à l'origine "relatif au système algébrique inventé par le mathématicien britannique George Boole (1815-1864)". Utilisé comme nom en programmation, fait référence à un contact, pris comme un bit unique, qui peut avoir un ou deux états.



**boucle**

Suite d'instructions qui est exécutée de manière répétitive jusqu'à ce que soit vérifiée une condition de "sortie de boucle". (ISO)

**branchement**

Voir branch/branchement.

**bus**

Voir canal (highway). (ADatP-2)

**bus de données**

- a. En termes généraux, bus dont la fonction prévue est le transfert des informations d'un point à un autre.
- b. Dans un système conforme au STANAG 3838 AVS, ensemble de matériels comportant des paires torsadées blindées, des résistances isolantes, des transformateurs, etc, nécessaires pour assurer le cheminement des données entre la commande de bus et tous les terminaux éloignés associés.

**bus de données multiplex**

Technique de transmission de données utilisant la technique du multiplexage pour connecter et intégrer plusieurs sous-systèmes.

**C**

Langage de programmation que l'on peut décrire comme de niveau moyen, ayant des caractéristiques et capacités à la fois d'un langage évolué et d'un langage bas de gamme. Il a des capacités que l'on trouve dans un langage d'assemblage, mais rarement dans un langage évolué. Ses créateurs, Dennis Ritchie et Brian Kernigan, ont donné à C la vitesse et la souplesse d'un langage d'assemblage et des structures de données et de commandes (comme si-alors, pour, qui) d'un langage évolué.

**calcul d'adresse**

Calcul, au moyen d'une formule ou d'un algorithme appliqué à une clé d'enregistrement, de l'adresse physique de l'enregistrement sur une mémoire de grande capacité, comme un disque.

**calcul de la durée**

Estimation ou mesure du temps de calcul nécessaire ou réellement utilisé. Ce temps est habituellement comparé au temps disponible pour le calcul afin de déterminer la marge de temps. (RTCA)

**calculateur numérique**

- a. Calculateur dans lequel on utilise essentiellement des représentations discrète des données. (ADatP-2)
- b. Calculateur qui exploite des données discrètes en effectuant des opérations arithmétiques et logiques sur celles-ci. (ADatP-2)

**canal**

Système informatique spécialisé chargé de gérer les entrées/ sorties indépendamment de l'unité centrale. Un canal peut avoir accès directement à la mémoire centrale pour stocker ou extraire de l'information. Il y a trois types de canaux : sélecteur, multiplexeur et multiplexeur par bloc. La communication entre un canal et un processeur se fait ordinairement par appel sélectif ou par interruption.

**caractère structurel**

Le code possède le caractère structurel spécifique dans la mesure où il possède un modèle défini de l'organisation de ses parties interdépendantes. Cela suppose que l'évolution de la conception du programme s'est faite de façon ordonnée et systématique et que les structures de contrôle standard ont été respectées dans le codage du programme, etc. (DAN 239)

**catégorie d'erreur**

Catégorie, dans un ensemble de catégories, dans laquelle une erreur, un défaut ou une défaillance peuvent être classés. Les catégories peuvent être définies selon la cause, la gravité, l'effet, la phase du cycle de vie durant laquelle l'erreur a été introduite ou détectée ou d'autres caractéristiques de l'erreur, du défaut ou de la défaillance. (ANSI/IEEE)

**chaîne**

- a. Eléments d'information, e.g. dans une table; chacun des éléments est relié au suivant, et peut-être à d'autres parties de la chaîne, par un pointeur.
- b. Suite linéaire d'éléments tels que des caractères ou des éléments physiques. (ANSI)

**champ**

Elément dans un enregistrement ou partie de celui-ci.

**charger**

Lire des données, habituellement des instructions d'ordinateur, dans une mémoire ou des registres internes, en particulier pour les copier sur des registres. (ADatP-2)

**chargeur**

En avionique, programme qui provoque le chargement d'un autre programme dans une mémoire de calculateur.

**chargeur de liens**

Voir éditeur de liens.

**chargeur-amorce**

Sous-programme d'introduction dans lequel des opérations machine prédéfinies sont utilisées pour charger une amorce. (ISO)

**chef programmeur**

Chef d'une équipe de programmation structurée; programmeur de niveau supérieur dont les responsabilités comprennent la production de parties importantes du logiciel assigné à l'équipe, la coordination des activités de celle-ci et l'examen du travail des autres membres de l'équipe. Le chef programmeur doit de plus avoir une compréhension technique globale du logiciel en voie d'élaboration. (ANSI/IEEE)

**chiffre**

Caractère graphique représentant à lui seul un nombre entier. Exemple : l'un des caractères de 0 à 9. (ADatP-2)

**chiffre binaire (bit)**

L'un des chiffres 0 et 1, lorsqu'il est employé en numération binaire. Habituellement utilisé pour représenter deux caractéristiques distinctes, e.g. ouvert-fermé, oui-non. Généralement abrégé en bit. (ADatP-2)

**clé**

Un ou plusieurs caractères, à l'intérieur d'un ensemble de données, contenant des informations sur cet ensemble, notamment son identification. (ISO)

**co-programmes**

Deux ou plusieurs modules qui peuvent s'appeler l'un l'autre, mais qui ne sont pas dans la relation supérieur-inférieur. (ANSI/IEEE)

Cobol

Acronyme de "Common Business Oriented Language", langage de programmation évolué pour applications commerciales.

codage

Technique de conception et de programmation en vertu de laquelle les données sont stockées ou acheminées, au moyen d'une projection réversible, du domaine d'origine dans un autre domaine. (ABBOTT)

code

- a. Ensemble de règles précises spécifiant la manière dont les données peuvent être représentées sous une forme discrète. Synonyme de "coding scheme". (ADatP-2)
- b. Dans les télécommunications, ensemble de règles et conventions selon lesquelles les signaux représentant des données peuvent être formés, transmis, reçus et traités. (ADatP-2)
- c. Ensemble d'articles, comme des abréviations, représentant les membres d'un autre ensemble. (ADatP-2)

code d'ordinateur

Code machine pour un ordinateur particulier. (ANSI)

code machine

- a. Code opération reconnu par une machine conçue à cet effet.
- b. Chiffres absolus, noms ou symboles affectés par le concepteur de la machine à une partie quelconque de celle-ci.
- c. Identique à code opération. (Computer dictionary)

- d. Consiste en configurations binaires enregistrées comme impulsions magnétiques, chaque configuration causant l'exécution d'une fonction différente de l'ordinateur.

#### code machine absolu

Langage machine qui doit être chargé dans des emplacements mémoire fixes à chaque utilisation et qui ne peut être traduit. (ANSI/IEEE) Antonyme de code machine translatable.

#### code machine translatable

Langage machine qui exige que les adresses relatives soient traduites en adresses absolues avant l'exécution sur ordinateur. (ANSI/IEEE) Antonyme de code machine absolu.

#### code objet

- a. Code obtenu en sortie d'un compilateur ou d'un assembleur qui est un code machine exécutable ou est apte à être traité pour produire un code machine exécutable. (ADatP-2)
- b. Représentation de données ou d'un programme d'ordinateur sous forme symbolique qui est entièrement compilé ou assemblé et prêt à être chargé dans un ordinateur.

#### code objet exécutable

Forme binaire des instructions de code qui peut être utilisée directement par l'unité centrale (UC). (RTCA)

#### code rentrant

Code qui peut être exécuté simultanément par plusieurs tâches. Ainsi, le code ne peut se modifier lui-même et chaque tâche doit garder ses propres données.

**code source**

Code écrit en langage source (langage d'assemblage ou langage évolué) exploitable par une machine pour l'assembleur ou le compilateur. (RTCA)

**coder**

- a. Représenter des données ou un programme machine sous une forme symbolique recevable par une machine de traitement de l'information. (ADatP-2)
- b. Ecrire un programme. (ADatP-2)
- c. Préparer ou présenter les données sous une forme et un format adaptés à la saisie par un système de traitement automatisé. (ADatP-2)

**codeur**

Individu chargé de rédiger - mais non de définir - des programmes d'ordinateur.

**cohérence**

Utilisation de techniques de conception et de documentation uniformes pour tout le projet de développement d'un logiciel.

**cohésion**

Indique dans quelle mesure les tâches exécutées par un module de programme unique sont fonctionnellement reliées entre elles. (ANSI/IEEE) En contraste avec de couplage.

**commentaire**

Information intrinsèque d'un programme machine, ordre de gestion ou ensemble des données destinés à éclairer les utilisateurs. Les commentaires ne modifient par l'interprétation machine. (ANSI/IEEE)

**commission de contrôle de la configuration**

Autorité chargée d'évaluer, d'approuver ou de désapprouver les modifications techniques proposées et d'assurer la mise en oeuvre des modifications approuvées. (ANSI/IEEE)

**comparateur**

Outil logiciel utilisé pour comparer deux programmes machine, des fichiers ou des ensembles de données afin de relever les similitudes ou les différences. Les versions similaires du code source, du code objet, des fichiers de bases de données ou des résultats de tests constituent des objets types de comparaison. (ANSI/IEEE)

**compatibilité**

Etat qui existe entre deux ordinateurs, ou plus, qui peuvent exécuter le même répertoire d'instructions machine et actionner des périphériques identiques. (ADH WP)

**compilateur**

- a. Programme qui compile. (ADatP-2)
- b. Programme d'ordinateur utilisé pour compiler. (ISO) Différent de assembleur et interpréteur. (ANSI/IEEE)
- c. Logiciel d'ordinateur qui traduit d'autres programmes écrits en langage évolué comme les FORTRAN, PASCAL, C, etc, en code machine.

**compilateur croisé**

Compilateur exécuté sur un ordinateur, mais qui produit un code d'assemblage ou un code objet pour un ordinateur différent. (ANSI/IEEE)

**compilateur de base**

Compilateur dont la sortie est une représentation d'un programme de niveau intermédiaire et non lié à un



type de machine particulier. Un compilateur de base associé à un générateur de codes lié à un type de machine particulier constitue un compilateur complet.  
(ANSI/IEEE)

### compiler

- a. Préparer un programme de langage machine à partir d'un programme d'ordinateur écrit dans un autre langage de programmation, en utilisant la structure logique générale du programme ou en générant plusieurs instructions machine pour chaque instruction symbolique, ou en faisant ces deux opérations, et en assurant également la fonction d'assemblage. (ADatP-2)
- b. Traduire un programme de langage évolué en son équivalent en langage machine absolu ou translatable. (ANSI/IEEE) En contraste avec de assembler, interpréter.

### complexité

Degré de complication d'un système, ou d'un élément de système, déterminé par des facteurs tels que le nombre d'interfaces et leur complexité, le nombre de branchements conditionnels et leur complexité, le degré d'emboîtement, les types de structures de données et d'autres caractéristiques du système. (ANSI/IEEE)

### comptage

Action de compter ou de tenir un compte.

### conception architecturale

- a. Définition d'un groupe d'éléments matériels ou logiciels et de leurs interfaces afin d'établir un cadre pour l'élaboration d'un système information. (ANSI/IEEE)
- b. Résultat du processus de conception architecturale. (ANSI/IEEE)

---

**conception ascendante**

Conception d'un système commençant par les éléments les plus fondamentaux ou primitifs et allant vers les éléments de niveaux plus élevés qui utilisent ceux des niveaux inférieurs. (ANSI/IEEE) Antonyme de conception descendante.

---

**conception de système**

- a. Processus de définition de l'architecture du logiciel, des éléments, des modules, des interfaces, de la méthode d'essai et des données d'un système logiciel afin de satisfaire aux exigences spécifiées. (ANSI/IEEE)
- b. Résultat du processus de conception. (ANSI/IEEE)

---

**conception descendante**

Processus de conception d'un système qui consiste à en définir les principaux éléments, à les décomposer en éléments d'un niveau inférieur et à répéter l'opération jusqu'à ce que l'on ait atteint le niveau de précision voulu. (ANSI/IEEE) Antonyme de conception ascendante.

---

**conception détaillée**

- a. Processus d'épuration et d'extension de la conception préliminaire afin qu'elle comporte plus de descriptions détaillées de la logique du traitement ainsi que des structures et des définitions de données, de façon qu'elle soit suffisamment complète pour être mise en oeuvre. (ANSI/IEEE)
- b. Résultat du processus de conception détaillée. (ANSI/IEEE)

---

**conception dirigée par données**

Conception de logiciel tirée d'une analyse des données d'entrée et de sortie du système.

---

**conception du système**

- a. Définition des éléments, des modules, des interfaces et des architectures du matériel et du logiciel, ainsi que des données, de manière à satisfaire aux exigences spécifiées du système. (ANSI/IEEE)
- b. Résultat du processus d'élaboration du système. (ANSI/IEEE)

---

**conception fonctionnelle**

Spécification des relations fonctionnelles entre les parties d'un ensemble, en fonction de leurs actions caractéristiques. (ANSI)

---

**conception orientée vers l'objet**

Technique de conception dans laquelle un système est vu comme un rassemblement d'objets plutôt que comme des fonctions avec des messages passés d'un objet à un autre, chacun avec son propre ensemble d'opérations associées.

---

**conception structurée**

Approche méthodique de la conception des logiciels qui utilise un ensemble déterminé de règles fondées sur des principes tels que la conception descendante, la méthode des approximations successives et l'analyse des flux de données. (ANSI/IEEE)

---

**concision**

Compacité du programme en termes de lignes de code.

---

**condition de choix d'une branche**

Ensemble des conditions qui doivent être remplies pour que les opérations d'une branche particulière du programme soient exécutées. (ANSI/IEEE)

**configuration**

En avionique, disposition des parties d'un système; peut être pris dans le sens physique ou dans le sens logique.

**contrôle de la configuration**

Processus d'évaluation, d'approbation ou de désapprobation et de coordination des changements apportés aux éléments de la configuration après l'identification positive de la configuration.

**contrôle de programmation**

Simulation manuelle de l'exécution d'un programme pour détecter les erreurs de logique ou de syntaxe dans le code source au moyen d'un examen pas à pas.  
(ANSI/IEEE)

**contrôle des changements**

Processus en vertu duquel un changement est proposé, évalué, approuvé ou rejeté, planifié et suivi.  
(ANSI/IEEE)

**contrôleur de bus**

Dans un système de bus de données, terminal ayant pour fonction de déclencher les transferts d'informations sur le bus de données.

**conversationnel**

Se rapporte à un système interactif qui assure une interaction similaire à un dialogue humain entre l'utilisateur et le système. (ANSI/IEEE)

**conversion**

Modification d'un logiciel existant permettant à celui-ci de fonctionner avec une capacité semblable dans un cadre d'utilisation différent. Exemples : conversion d'un programme en langage FORTRAN en un programme en langage Ada, conversion d'un programme utilisé sur un

ordinateur de façon à ce qu'il puisse être utilisé sur un autre ordinateur. (ANSI/IEEE)

### conversion analogique/numérique (A/N)

Conversion de signaux analogiques en données numériques.

### correction

- a. Modification d'un programme objet par le remplacement d'une partie du code machine existant par le code machine modifié. (ANSI/IEEE)
- b. Corriger : modifier un programme objet sans recompiler le programme source. (ANSI/IEEE)

### couplage

- a. Mesure de l'interdépendance entre les modules dans un programme machine. (ANSI/IEEE)  
Antonyme de cohésion.
- b. Mesure de la complexité d'un programme ou d'un système. Décrit comment un changement bénin dans un module cause un problème totalement imprévu dans un autre module. Des modules fortement interdépendants sont dits avoir un couplage serré, ce qui est une caractéristique non souhaitable dans une conception structurée.

### croissance de la fiabilité

Amélioration de la fiabilité d'un logiciel par suite de la correction des défauts de ce logiciel. (ANSI/IEEE)

### cycle d'élaboration d'un logiciel

Période qui commence avec la décision d'élaborer un produit logiciel et se termine lorsque ce produit est livré. Ce cycle comprend généralement une phase d'établissement des exigences, une phase de conception, une phase de mise en oeuvre, une phase d'essai, et, parfois, une phase de mise en place et de vérification. (ANSI/IEEE) Différent de cycle de vie du logiciel.

**cycle de vie du logiciel**

Période qui commence avec la conception du produit logiciel et qui se termine lorsque ce produit n'est plus disponible. Le cycle de vie du logiciel comprend généralement une phase d'établissement des exigences, une phase de conception, une phase de mise en oeuvre, une phase d'essai, une phase de mise en place et de vérification, une phase d'exploitation et de maintenance et, parfois, une phase de mise au rancart. (ANSI/IEEE)  
En contraste avec cycle d'élaboration d'un logiciel.

**cycle machine**

Algorithme qui définit un ordre de succession des opérations exécutées par une machine abstraite en vue de déterminer quelles opérations supplémentaires doivent être exécutées et à quelles données elles doivent être appliquées. Si une machine abstraite a, par définition, un cycle machine, ce cycle machine est exécuté régulièrement et toutes les fois que la machine abstraite a terminé l'exécution de l'opération appelée par le cycle machine antérieur. (ABBOTT)

**débit**

Quantité de données traitées par un système en un temps donné exprimée en des termes significatifs pour ce système.

(ADatP-2)

**débit binaire**

Nombre de bits par unité de temps qui passent à un point donné par un canal donné ou qu'un système peut recevoir (entrée) ou délivrer (sortie). (ADatP-2)

**début d'exécution**

Moment où commence l'exécution d'un programme.  
(ANSI/IEEE) Voir également durée d'exécution.

**décalage**

Action de déplacer les valeurs des bits dans un mot-machine d'une position à une autre.

**décomposition fonctionnelle**

Méthode d'élaboration d'un système qui consiste à décomposer celui-ci en ses éléments de manière à ce que ceux-ci correspondent directement aux fonctions et aux sous-fonctions du système. (ANSI/IEEE)

**décomposition hiérarchique**

Méthode d'élaboration d'un système qui consiste à le décomposer en ses éléments en effectuant une série de simplifications descendantes. (ANSI/IEEE)

**décomposition modulaire**

Méthode de conception d'un système par sa décomposition en modules. (ANSI/IEEE) Voir également décomposition hiérarchique.

**découpage**

La fonction "A" est extraite de la fonction "B" si aucune action de la fonction "B" ne peut causer de défaillance dans la fonction "A". (RTCA)

**défaillance**

- a. Impossibilité, pour un élément, d'exécuter la fonction requise. (ISO)
- b. Incapacité, pour un système, ou pour un élément de système, d'exécuter une fonction requise dans des limites spécifiées. Une défaillance peut se produire lorsqu'un défaut se manifeste. (ANSI/IEEE)
- c. L'exécution du programme ne se déroule pas comme prévu. (ANSI/IEEE)

**défaut**

- a. Condition accidentelle qui empêche une unité d'exécuter la fonction requise. (ISO)
- b. Manifestation d'une erreur dans le logiciel. Un défaut peut causer une défaillance. (ANSI/IEEE) Synonyme d'erreur.

**dépannage**

Maintenance exécutée spécifiquement pour corriger les défaillances. (ISO)

**dépôt de logiciels**

Bibliothèque de logiciels où s'effectue l'archivage permanent des logiciels et des documents connexes. (ANSI/IEEE)

**dérivation**

En programmation, branchement sur une instruction autre que celle qui suit immédiatement l'instruction de dérivation écrite dans le programme. Les dérivations peuvent être conditionnelles, en fonction du résultat du test, ou non.

**déroutement**

Technique selon laquelle le déroulement logique d'un programme est interrompu afin de mettre de côté les résultats intérimaires en vue d'effectuer des mesures d'essai ou d'autres fonctions d'essai spéciales. (DAN 134)

**descendant**

Terme se rapportant à une approche qui commence par l'élément du niveau le plus élevé dans une hiérarchie et qui passe progressivement aux niveaux inférieurs; par exemple, la conception descendante, la programmation descendante et la vérification descendante. (ANSI/IEEE) Antonyme d'ascendant.



**descripteur de tâche**

Bloc d'informations relatives aux tâches manipulé par le système et souvent stocké dans les tables de ce dernier afin d'accroître l'efficacité de l'ordonnancement. Ces informations comprennent le nom, le numéro, la priorité et l'état de la tâche, ainsi que d'autres informations pertinentes sur sa situation.

**désempilage**

Terme utilisé pour récupérer de l'information dans une pile.

**désynchroniseur des entrées/sorties**

Programme d'exploitation permettant de mettre les transferts entrée/sortie dans la file d'attente d'une unité d'entrée/sortie, autorisant de ce fait la poursuite de l'exécution de la tâche demanderesse même lorsque celle-ci ne peut utiliser immédiatement l'unité d'entrée/sortie.

**diagnostic**

Opération consistant à détecter et à isoler les causes d'une panne ou d'un mauvais fonctionnement.  
(ADatP-2)

**diagramme d'état**

Graphe orienté dans lequel les noeuds correspondent aux états internes d'un système et les barres aux transitions; souvent utilisé pour décrire un système en fonction des changements d'état. (ANSI/IEEE) Voir également réseau de Pétri.

**dictionnaire de données**

- a. Répertoire des noms de toutes les données élémentaires utilisées dans un système logiciel et de leurs propriétés pertinentes. Il peut s'agir, par exemple, de la longueur de la donnée élémentaire, de sa représentation, etc.  
(ANSI/IEEE)

- b. Ensemble de définitions de flux de données, d'éléments d'information, de fichiers, de bases de données et de traitements mentionnés dans un organigramme de données. (ANSI/IEEE)

### diffusion

Dans un système de bus de données, la mise en oeuvre du système faite de telle façon que l'information transmise par la commande de bus ou un terminal éloigné soit adressée à plusieurs terminaux éloignés reliés au bus de données.

### diode électroluminescente

Diode semi-conductrice utilisée comme source lumineuse.

### disponibilité

Rapport des heures effectives de fonctionnement sur le nombre des heures prévues de fonctionnement, exprimé en pourcentage.

### dispositif

Unité de périphérique ou partie de logiciel d'un système généralement utilisé pour les entrées et/ou sorties.

### dispositif de contrôle d'accès

Caractéristiques du matériel ou du logiciel, procédures d'exploitation ou procédures de gestion élaborées pour permettre l'accès autorisé à un système informatique et pour empêcher l'accès non autorisé à ce système. (ANSI/IEEE)

### dispositif logique

Référence à un dispositif entrée/sortie par un nom ou un chiffre, sans tenir compte de la nature particulière de ce dispositif. Au moment de l'exécution, un opérateur, ou une autre tâche affecte un dispositif

physique au nom ou chiffre du dispositif logique. Le sous-système entrée/sortie fait alors en sorte que les programmes du dispositif approprié soient utilisés pour effectuer les transferts entrée/sortie.

### dispositif physique

Dispositif périphérique matériel existant, comme une imprimante ligne par ligne, un terminal, etc., qui est relié à un système informatique.

### disque

Dispositif de stockage magnétique portant des pistes circulaires concentriques sur lesquelles des données sont enregistrées. Une unité de disques contient un ou plusieurs disques qui peuvent être soit interchangeables, quand ils peuvent être enlevés et remplacés, soit fixes.

### document de base attribué

Etape marquant l'achèvement de la phase d'analyse du logiciel. Le document de base attribué établit une correspondance entre l'architecture et la configuration logicielle.

### documentation

- a. Création, collecte, organisation, stockage, appel et dissémination de documents ou d'informations enregistrées sur les documents. (ANSI)
- b. Ensemble de documents ou d'informations sur un sujet donné. (ANSI)
- c. Une documentation typique peut comprendre : analyse du problème par un analyste de système, organigrammes du programme écrit pour le résoudre, descriptions écrites du programme et listages informatiques imprimés.

### documentation de l'utilisateur

Documentation qui fournit à l'utilisateur final d'un système les instructions sur la façon d'utiliser ce

système pour en obtenir les résultats désirés; par exemple, un guide d'utilisateur. (ANSI/IEEE) Antonyme de documentation sur le système.

### documentation sur l'état de la configuration

Enregistrement et déclaration de l'information nécessaire à la gestion efficace d'une configuration, comprenant un listage de l'identification de la configuration approuvée, l'état des changements proposés pour la configuration et l'état de la mise en oeuvre des changements approuvés. (MIL-STD-973)

### documentation sur le système

Documentation contenant les exigences, les principes et les détails de conception, les possibilités, les limites et d'autres caractéristiques d'un système. (ANSI/IEEE) En contraste avec documentation de l'utilisateur.

### documentation technique du logiciel

Données ou renseignements techniques, comprenant les listages d'ordinateur, pouvant être lus par des personnes, qui décrivent ou spécifient la conception ou certains détails, expliquent les possibilités ou fournissent les consignes d'exploitation en vue d'obtenir les résultats désirés d'un système logiciel. (ANSI/IEEE)

### domaine des données

Méthode permettant d'obtenir une évaluation de la fiabilité opérationnelle. En principe, si tous les ensembles de valeurs de données d'entrée qui doivent être traités par le programme machine sont identifiés, on pourra obtenir une évaluation de la fiabilité de ce programme en procédant à l'exécution de celui-ci pour chacun de ces ensembles. En pratique, on utilise une méthodologie pour sélectionner quelques ensembles de données d'essai représentatifs de l'usage opérationnel désiré, et le programme est exécuté pour ces ensembles uniquement. Les résultats obtenus sont utilisés pour évaluer la fiabilité du programme suivant le cadre opérationnel désiré. (DAN-238)

**donnée de contrôle**

Donnée utilisée pour influencer sur le déroulement d'un programme (en le sélectionnant ou en le modifiant).  
(ADatP-2)

**données**

- a. Fait, notion ou instruction représentés sous une forme conventionnelle convenant à une communication, une interprétation ou un traitement soit par l'homme, soit par des moyens automatiques. (ADatP-2)
- b. Représentation quelconque par caractère ou par quantités analogiques et qui s'accommode d'un sens particulier. (ADatP-2)

**données d'essai**

Données élaborées pour essayer un système ou un élément de système. (ANSI/IEEE)

**données de fiabilité**

Information nécessaire pour évaluer la fiabilité d'un logiciel à des points déterminés du cycle de vie de ce logiciel. A titre d'exemples, mentionnons les données d'erreur et les données de temps pour les modèles de fiabilité, les attributs de programme tels que la complexité, et les caractéristiques de programmation telles que les techniques de développement utilisées et l'expérience du programmeur. (ANSI/IEEE)

**données de référence sur un logiciel**

Données relatives à l'élaboration ou à l'utilisation d'un logiciel qui pourraient être utiles dans la création de modèles, les prédictions de fiabilité ou d'autres descriptions quantitatives du logiciel. (ANSI/IEEE)

**données informatiques**

Données disponibles pour assurer la communication entre les matériels informatiques ou à l'intérieur de ceux-ci. Ces données peuvent être externes (sous forme

exploitable par un ordinateur) ou résidentes dans le matériel informatique et peuvent revêtir la forme de signaux analogiques ou numériques. (ANSI/IEEE)

### données sur les erreurs

Terme communément (mais non précisément) utilisé pour désigner les informations décrivant les problèmes, les défauts, les défaillances et les changements relatifs au logiciel ainsi que leurs caractéristiques et les conditions dans lesquelles ils apparaissent ou sont corrigés. (ANSI/IEEE)

### double programmation

Technique d'élaboration suivant laquelle deux versions fonctionnellement identiques d'un programme sont créées par différents programmeurs ou par différentes équipes de programmation à partir de la même spécification. Le code source résultant sera dans un même langage ou dans un langage différent. Le but de la double programmation est d'assurer la détection des erreurs, d'augmenter la fiabilité, de fournir de la documentation additionnelle ou de réduire la probabilité d'erreurs systématiques de programmation ou d'erreurs du compilateur exerçant une influence sur le résultat final. (ANSI/IEEE)

### drapeau

- a. Un des types d'indicateurs utilisé pour l'identification, par exemple : drapeau mot. (ANSI)
- b. Caractère indiquant l'apparition d'une condition quelconque, comme la fin d'un mot. (ANSI)

### durée d'exécution

Temps que prend l'exécution d'un programme. Bien que la durée d'exécution corresponde habituellement à la durée de fonctionnement du processeur central, elle peut également comprendre la durée du traitement par les périphériques et le temps d'accès à ces derniers; par exemple, une durée d'exécution de cinq heures. (ANSI/IEEE)

**durée de fiabilité**

Probabilité qu'un article exécutera une fonction requise dans des conditions déterminées pendant une période de temps déterminée. (ANSI/ASQC A3-1978)

**éditeur**

Programme machine qui permet la révision sélective de données stockées en mémoire. (ANSI/IEEE)

**éditeur de liens**

Programme qui produit un module chargeable en transformant des modules résultants en un format acceptable pour un calculateur.

**effet secondaire**

Traitement, activité ou résultat dérivés de la fonction primaire d'un programme, d'un sous-programme ou d'une opération. (ANSI/IEEE)

**efficacité**

Indique dans quelle mesure un logiciel exécute les fonctions pour lesquelles il a été conçu avec une consommation minimale des ressources informatiques. (ANSI/IEEE)

**électro-optique**

Concerne la translation du rayonnement électromagnétique dans les portions visible et infrarouge du spectre dans un format adapté aux applications électroniques et vice versa.

**élément**

Partie fondamentale d'un système ou d'un programme. (ANSI/IEEE)

---

**élément binaire**

Elément constitutif d'une donnée pouvant représenter l'une ou l'autre de deux valeurs ou états distincts.  
(ADatP-2)

---

**élément de la configuration**

Ensemble d'éléments matériels ou logiciels considérés comme une unité aux fins de gestion de la configuration. (ANSI/IEEE)

---

**élément de remplacement temporaire**

- a. Instruction de programme remplaçant le corps d'une unité de programme et indiquant que cette unité est ou sera définie ailleurs.  
(ANSI/IEEE)
- b. Module de programme fictif utilisé durant l'élaboration et l'essai d'un module d'un niveau supérieur. (ANSI/IEEE)
- c. Partie de programme remplaçant un véritable segment. Il ne remplit pas les fonctions du véritable segment, mais a un comportement suffisamment similaire en ce qui concerne les entrées et les sorties pour que le segment du niveau supérieur qui y fait référence s'y trompe. Dans la plupart des cas, l'élément de remplacement temporaire est une courte séquence de code de base qui fournit également une trace d'exécution.

---

**élément ternaire**

Elément constitutif d'une donnée pouvant représenter l'une ou l'autre de trois valeurs données.

---

**emboîter**

Disposer des programmes ou des données à l'intérieur d'autres programmes ou d'autres données. Les programmes ainsi emboîtés se trouvent à un niveau hiérarchique différent et ces différents niveaux peuvent être exécutés ou obtenus de façon récursive. (ANSI)



**émission**

Service système utilisé pour la communication et la synchronisation des tâches qui place un message dans la boîte à lettres appropriée. Si une tâche attend dans la boîte à lettres, elle est placée sur la liste de réserve.

**empiler**

Terme utilisé pour placer une information sur une pile.

**émulateur**

Logiciel exécuté sur un ordinateur central qui accepte les mêmes données d'entrée, exécute les mêmes programmes et produit les mêmes sorties que l'ordinateur d'exécution. (RTCA)

**émulation**

Imitation d'un système informatique ou d'une partie de celui-ci par un autre système, surtout par le matériel, afin que ce système "imitateur" accepte les mêmes données, exécute les mêmes programmes et obtienne les mêmes résultats que le système imité. (ISO)

**en mode duplex**

Le mode duplex (ou full duplex) caractérise un mode de transmission qui autorise des communications simultanées entre deux points. (ADatP-2)

**en mode semi-duplex**

Caractérise le mode d'utilisation d'un circuit de transmission qui, bien qu'adapté au duplex, n'est utilisé qu'alternativement par les terminaux, en raison de leur nature. (ADatP-2)

**en suspens**

Etat d'un tâche interrompue en attendant la survenance d'un événement.

**en temps partagé**

S'applique à l'utilisation imbriquée du temps dans un système informatique qui permet à deux ou plusieurs utilisateurs de faire exécuter simultanément des programmes machine. (ANSI)

**en temps réel**

- a. Qualifie la période effective au cours de laquelle se déroule un processus physique.
- b. Qualifie l'utilisation d'un ordinateur pendant le déroulement effectif du phénomène constaté. De plus, les résultats des calculs effectués au cours du processus doivent être fournis en temps voulu pour pouvoir corriger le processus en cause. (ADatP-2)

**enregistrement**

Ensemble de données ou de mots connexes traités comme une unité. (ISO)

**enregistrement logique**

Ensemble d'articles considéré sans tenir compte de leurs caractéristiques physiques. Les différentes parties d'un tel enregistrement peuvent être situées dans des enregistrements physiques différents. (ANSI)

**ensemble de puces**

Groupe de puces reliées entre elles accomplissant une (ou des) fonction(s) intégrée(s).

**ensemble des aides à la programmation**

Ensemble intégré d'outils sollicités au moyen d'un seul langage de gestion afin d'assurer des fonctions de soutien à la programmation durant tout le cycle de vie du logiciel. Cet ensemble comprend généralement des outils pour la conception, l'édition, la compilation, le

chargement, l'essai de la gestion de la configuration et la gestion du projet. (ANSI/IEEE)

### entrée dans le système

Mécanisme par lequel un utilisateur de système informatique fournit son identification et accède au système.

### entrée-traitement-sortie hiérarchique (HIPO)

Technique de conception et de documentation basée sur le développement d'une hiérarchie descendante d'éléments fonctionnels d'un système et de diagrammes entrée-traitement-sortie (IPO) pour chacun de ces éléments. La technique HIPO fournit une organisation hiérarchique du système et une description des caractéristiques des éléments du logiciel qui constituent le système.

### environnement/cadre/contexte d'utilisation

Milieu dans lequel le système doit fonctionner. Tous les éléments sur lesquels le concepteur n'a pas de contrôle et qui affectent le système ou ses entrées et sorties.

### environnement du soutien de programmation ADA (APSE)

Ensemble d'outils logiciels venant en soutien des applications ADA.

### équipe de programmation structurée

Groupe d'élaboration de logiciels formé d'un chef programmeur, d'un programmeur-assistant, d'un secrétaire bibliothécaire ainsi que de programmeurs et de spécialistes additionnels, au besoin. De plus, l'équipe de programmation structurée utilise des procédures de soutien conçues pour accroître la communication entre les membres du groupe et pour exploiter au maximum les capacités de chacun d'entre eux. (ANSI/IEEE)

**équipement d'essai automatique (ATE)**

Équipement d'essai à but général et particulier utilisé pour appliquer une suite programmée d'essais dans le but de vérifier le bon fonctionnement d'un système ou sous-système, ou bien d'en identifier et localiser les défaillances.

**équipement de test intégré (BITE)**

Dispositif faisant partie intégrante d'un équipement ou d'un système, utilisé dans le but exclusif de tester cet équipement ou ce système.

**erreur**

Ecart entre une grandeur calculée, observée ou mesurée et la valeur vraie, prescrite ou théorique correspondante. (ANSI)

**erreur du programme**

Erreur dans un programme machine qui n'a pas été insérée dans le cadre du processus d'implantation d'erreurs. (ANSI/IEEE)

**espace adresses**

Gamme des adresses accessibles à un programme machine. (ANSI/IEEE)

**espace interbloc**

Espace entre blocs de données enregistrées, par exemple sur une bande magnétique.

**essai**

- a. Le but principal de l'essai devrait être de découvrir les défauts (éliminer les imperfections) des programmes et des systèmes, c'est-à-dire de trouver, dans le programme ou le système, des erreurs dont on ne connaissait pas l'existence.

- b. Une équipe d'essai composée de membres de l'équipe d'élaboration et de personnes extérieures, telles que des membres de l'équipe d'assurance de la qualité et du département utilisateur, sera vraisemblablement plus productive, lorsqu'il s'agira de découvrir le plus grand nombre possible de défauts, qu'une équipe d'essai entièrement constituée de membres de l'équipe d'élaboration. Les personnels chargés de l'élaboration auront un rôle important lors de la phase d'essai des modules, tandis que les représentants du département utilisateur auront un rôle important lors de la phase d'essai de recette.
- c. L'essai met à jour les imperfections du logiciel et, ce faisant, prouve la présence de défauts dans le programme. L'essai ne prouve pas et ne peut pas prouver l'absence de défauts.
- d. L'essai ne doit pas être considéré comme un moyen de garantir l'exactitude du programme; il doit être équilibré, ni excessif, ni insuffisant.
- e. L'essai détermine ce que le logiciel (ou le système) peut et ne peut pas faire. Il démontre que le logiciel est opérationnel tant du point de vue fonctionnel que du point de vue des performances. Etre opérationnel du point de vue fonctionnel signifie que le logiciel assume sa fonction avec exactitude et fiabilité. Etre opérationnel du point de vue des performances signifie que les performances du système se situent dans des limites raisonnables.
- f. L'essai démontre également que les interfaces entre les programmes et entre les différents systèmes fonctionnent convenablement.
- g. Contrôle ou évaluation manuels ou automatiques d'un système ou d'un élément de système afin de vérifier si ce système ou cet élément de système est conforme aux exigences spécifiées ou pour relever les différences entre les résultats prévus et les résultats réels. (ANSI/IEEE) Comparer cette notion avec la mise au point.

### essai à pleine charge

Phase d'essai pendant laquelle on oblige le système à fonctionner à sa vitesse maximale de traitement. Ce type d'essai est particulièrement important pour les systèmes de traitement par transactions en temps réel en ligne. Cet essai est destiné à mettre le système en face de situations anormales. L'essai à pleine charge fait fonctionner le système d'une façon qui exige une quantité et un volume de ressources anormaux à une fréquence anormale. La technique appelée essai de sensibilité est une variante de l'essai à pleine charge.

### essai alpha

Phase d'essai menée par le client chez le concepteur. Ces essais sont menés dans un environnement contrôlé.

### essai ascendant

Opposé à essai descendant. Premièrement, les fonctions constituant un module sont testées une par une. Elles sont ensuite intégrées pour former un module qui est testé. Après que chaque module a été testé, ceux-ci sont intégrés et, finalement, le sous-système est testé. Cette sorte d'essai comprend d'abord le test des modules au niveau le plus bas de la hiérarchie, puis on passe au niveau supérieur jusqu'à ce que le dernier module soit testé. Les programmes de gestion doivent être faits pour les modules du niveau le plus bas afin de leur fournir les informations d'entrée appropriées. L'essai ascendant a l'inconvénient de ne pas démontrer la disponibilité du programme de travail avant que le tout dernier module ait été testé.

### essai d'instruction

Phase d'essai dont le but est d'exécuter toutes les instructions du programme. L'intérêt principal de l'essai d'instruction réside dans le fait que les méthodes traditionnelles de production d'ensembles d'essai ne sont pas adaptées et ont tendance à aboutir à ce que jusqu'à 40 % de n'importe quel logiciel ne soit pas essayés (exécutés) du tout. En pratique, il sera souvent impossible d'exécuter toutes les instructions. En général, on estime qu'essayer 85 % des instructions est acceptable.

---

**essai d'intégration**

- a. Progression ordonnée de l'essai suivant laquelle les éléments logiciels, les éléments matériels ou les deux sont combinés et essayés jusqu'à ce que le système entier ait été intégré. (ANSI/IEEE)
- b. Vérification des interfaces des parties d'un système (modules, composants et sous-systèmes). Les progrès de l'essai d'intégration sont souvent mesurés par le nombre d'instructions codées et essayées, ou le nombre de fonctions ou de modules mis en oeuvre et essayés. La mesure des progrès devrait être renforcée par l'utilisation de modèles de fiabilité qui évaluent et projettent le temps moyen entre pannes (MTBF). L'essai d'intégration est suivi soit par un essai de simulation, soit par un essai clientèle initial, qui peut être vu comme l'ultime essai d'intégration. L'essai d'intégration est une technique systématique de construction de la structure d'un programme et, en même temps, une recherche des erreurs liées à l'interface. Le but est de prendre des modules individuellement et de construire une structure de programme dictée par la conception. En menant un essai d'intégration, l'expérimentateur devrait identifier les modules critiques.

---

**essai d'interface**

Essai effectué pour s'assurer que les éléments du programme ou du système s'échangent correctement les informations ou les ordres. (ANSI/IEEE)

---

**essai de branchement**

La mesure la plus largement utilisée pour s'assurer de l'état complet de l'essai est la vérification de branchement. Un ensemble de tests est considéré comme complet si chaque branchement du programme est parcouru au moins une fois lors des tests. Un branchement est un transfert de commande d'une instruction de programme à la suivante, soit par l'exécution normale des instructions soit par l'exécution d'un "go-to". L'essai de branchement complet est un critère nécessaire de justesse, dans le sens où il est essentiel de tester tous les branchements afin de trouver toutes les erreurs dans le programme. L'utilisation de l'essai de branchement

exige que l'on dispose d'un outil de test de programme que l'on puisse utiliser pour repérer les branchements qui sont parcourus lors d'une séquence de tests.

### essai de mise en place

Validation de chaque installation particulière du système dans le but de repérer toute erreur faite lors de la mise en place du système.

### essai de module

Après l'essai de chaque fonction individuelle, il est nécessaire de tester la coopération de ces fonctions quand elles sont réunies en un module. Il devrait être possible de tester un module comme une unité indépendante, sans la présence d'autres modules du système.

### essai de performance

Phase d'essai conçue pour tester la performance en temps réel d'un logiciel dans le contexte d'un système intégré. L'essai de performance est fait au cours de toutes les étapes du processus de test. Même au niveau de l'unité, la performance d'un module individuel peut être évaluée lors des essais de "boîte blanche". Les essais de performance sont souvent couplés aux tests avec une charge élevée et exigent souvent des outils matériels et logiciels. C'est-à-dire qu'il est souvent nécessaire de mesurer l'utilisation de la ressource (par exemple, les cycles du processeur) d'une façon très précise.

### essai de réception

Essai formel effectué pour déterminer si un système répond aux critères de réception et pour permettre au client de voir si le système lui convient. (ANSI/IEEE)  
Voir également test de performance et essai de système.

### essai de récupération

Essai de système par lequel on rend le logiciel défaillant de différentes façons et on vérifie que la récupération est correctement effectuée. Si la récupération est automatique (effectuée par le système lui-même), la nouvelle initialisation, les mécanismes des



points de reprise, la récupération des données et le redémarrage sont tous examinés pour vérifier qu'ils sont corrects. Si la récupération exige une intervention humaine, le temps moyen de réparation (MTTR) est évalué pour vérifier qu'il est dans des limites acceptables.

### essai de sécurité

Phase d'essai au cours de laquelle on s'efforce de s'assurer que les mécanismes de protection intégrés à un système le mettront effectivement à l'abri de toute intrusion malhonnête. Pendant l'essai de sécurité, l'essayeur joue le rôle d'un individu qui cherche à s'introduire dans le système. Il peut tenter de se procurer les mots de passe par des moyens extérieurs (employés de bureau, par exemple), il peut attaquer le système avec un logiciel spécial conçu pour déjouer toutes les protections installées, il peut saturer le système, empêchant ainsi les autres de s'en servir, il peut commettre des erreurs délibérées, en espérant pouvoir entrer dans le système pendant la récupération, ou encore parcourir les données publiques dans l'espoir d'y trouver la clef de l'accès au système.

### essai de simulation

Action de faire fonctionner le logiciel avec un programme de simulation et, souvent, quelques périphériques reproduisant le plus exactement possible le cadre d'exploitation réel. L'ordinateur employé pour faire passer le logiciel et le programme de simulation en machine peut être celui qui sera utilisé sur le terrain ou un ordinateur ayant servi à l'élaboration.

### essai de sous-système

Etape qui suit l'essai de module dans le processus global d'essai où les modules sont assemblés pour former des sous-systèmes. Etant donné que les modules coopèrent et communiquent, ce travail devrait consister, pour l'essentiel, à essayer les interfaces de modules.

### essai de système

Essai d'un système matériel et logiciel intégré pour vérifier qu'il répond aux exigences spécifiées.  
(ANSI/IEEE)

### essai des branches

La méthode de l'essai des branches ne peut être appliquée avec succès qu'aux programmes qui ont été conçus d'une manière descendante en utilisant une programmation structurée. L'essai des branches est alors utilisé pour tester séparément chacun des modules du programme. Chaque "branche logique" dans un programme devrait être testée au moins une fois.

### essai fonctionnel

Dans la grande majorité des cas, le premier type d'essai géré par ordinateur qui est effectué est l'essai fonctionnel. Le but de cet essai est de vérifier que le programme fait ce qu'il est supposé faire. Normalement, la personne qui a créé un programme en fait l'essai fonctionnel. Dans un essai fonctionnel, les jeux d'essai sont extraits par l'analyse de la spécification du logiciel. L'essai fonctionnel est essentiellement l'approche traditionnelle utilisée pour faire un essai d'une façon stricte et systématique.

### essai formel

Action d'effectuer des essais et de rendre compte des résultats conformément à un plan d'essai approuvé.  
(ANSI/IEEE)

### essai pilote

Phase du test menée chez un ou plusieurs clients par l'utilisateur final du logiciel. Au contraire de l'essai en laboratoire, le concepteur n'est généralement pas présent. L'essai pilote est donc une application "en grandeur nature" du logiciel dans un environnement que le concepteur ne peut contrôler. Le client signale tous les problèmes (réels ou imaginaires) rencontrés au cours de cette phase et en informe le concepteur à intervalles réguliers. A la suite des problèmes signalés lors de l'essai pilote, le concepteur du logiciel y apporte des modifications et prépare sa diffusion à tous les clients.

### état de la tâche

Situation d'une tâche; elle peut être non définie, prête à exécuter, en cours d'exécution ou suspendue en attendant un événement.

### étude préliminaire

- a. Analyse des diverses conceptions possibles et définition de l'architecture du logiciel.  
L'étude préliminaire comprend généralement la définition et la structuration des éléments et des données du programme machine, la définition des interfaces et la préparation des évaluations de temps et de capacité.  
(ANSI/IEEE)
- b. Résultat de l'étude préliminaire. (ANSI/IEEE)

### évaluation

Estimation de la capacité de mémoire interne ou du nombre de lignes sources nécessaires à un système ou à un élément de système. (ANSI/IEEE)

### évaluation de la conception

- a. Evaluation de la conception du système pour déterminer l'exactitude quant aux exigences établies, la conformité aux normes de conception, l'efficacité du système et d'autres critères. (ANSI/IEEE)
- b. Evaluation de différentes approches de conception. (ANSI/IEEE) Voir également étude préliminaire.

### évaluation de la fiabilité

Processus visant à déterminer le niveau de fiabilité d'un système ou d'un élément de système existant.  
(ANSI/IEEE)

**exactitude**

Degré d'exemption d'erreur ou degré de conformité à une valeur vraie ou à une règle. Il y a une différence entre exactitude et précision. Par exemple, un nombre à quatre chiffres est moins précis qu'un nombre à 6, cependant, un nombre à quatre chiffres correctement calculé peut être plus exact qu'un nombre à six chiffres mal calculé. (ANSI)

**exactitude partielle**

Dans une preuve d'exactitude, expression indiquant que les assertions de sortie d'un programme proviennent logiquement de ses assertions d'entrée et des étapes du traitement. (ANSI/IEEE) Antonyme d'exactitude totale.

**exactitude totale**

Dans la preuve d'exactitude, expression indiquant que les assertions de sortie d'un programme découlent logiquement des assertions d'entrée et des étapes du traitement et, en outre, que l'exécution du programme se termine conformément à toutes les conditions d'entrée spécifiées. (ANSI/IEEE) En contraste avec exactitude partielle.

**examen critique de la conception (CDR)**

Examen en règle de la conception détaillée du fournisseur, avant de déterminer si celle-ci prend en compte d'une façon adéquate tous les besoins de performance/interface de l'élément de la configuration (CI). Le CDR devrait être un examen technique formel de la conception exprimée par des spécifications et organigrammes suffisamment détaillés pour permettre au programmeur de coder, compiler, mettre au point un programme informatique, afin de s'assurer que les spécifications de la conception ont été satisfaites avant de commencer le codage. Cela a normalement lieu avant l'autorisation de fabrication/production.

**NOTE :** Le bon achèvement du CDR ne constitue pas l'accord du client selon lequel la conception remplira tous les besoins des spécifications d'un programme informatique.

### examen de la conception

Réunion officielle durant laquelle la conception préliminaire ou détaillée d'un système est présentée à l'utilisateur, au client ou à toute autre partie intéressée afin de connaître leurs commentaires et d'obtenir leur approbation. (ANSI/IEEE)

### examen de la conception du système

Examen effectué pour évaluer l'optimalisation, la capacité de traçage, la corrélation et l'exhaustivité de la définition des exigences affectées (identification de la configuration affectée), ainsi que les risques qui y sont associés, y compris les exigences correspondantes en matière d'essai. L'objectif est de s'assurer qu'elle répond aux exigences du système et/ou des segments du système (identification de la configuration fonctionnelle).

### examen des exigences du système

Examen effectué lorsqu'une partie importante des exigences fonctionnelles du système ont été définies afin de déterminer l'état d'avancement des travaux du contractant et sa réponse aux exigences de la spécification du système et du cahier des charges (SOW). En ce qui concerne le logiciel, ce travail se limite à l'affectation des exigences aux programmes machines et aux interfaces fonctionnelles qui en résultent.

### examen formel de qualification

Examen formel, normalement effectué par tranches chez le fournisseur, de comptes rendus d'essais et de données d'essai obtenus lors de la qualification formelle d'un nouvel élément de la configuration (CI), pour s'assurer que tous les essais prévus pour le contrôle de la qualité de la (des) spécification(s) de développement ont été effectués et que le CI fonctionne comme prévu par les stipulations de la (des) spécification(s) de développement.

### exécution

- a. Etat d'une tâche qui est exécutée par un processeur.

- b. Exécution d'une instruction ou des instructions d'un programme machine sur un ordinateur. (ISO)

### exécution symbolique

Technique de vérification selon laquelle l'exécution du programme est simulée au moyen de symboles plutôt que de valeurs réelles pour les données d'entrée; les résultats produits par le programme sont présentés sous forme d'expressions logiques ou mathématiques utilisant ces symboles. (ANSI/IEEE)

### exigence

- a. Condition ou aptitude dont un utilisateur a besoin pour résoudre un problème ou atteindre un objectif. (ANSI/IEEE)
- b. Condition que doit remplir ou aptitude que doit posséder un système ou un élément de système pour être conforme à une norme, à une spécification, à un contrat ou à tout autre document formellement imposé. L'ensemble de toutes les exigences constitue la base du développement ultérieur du système ou de l'élément de système. (ANSI/IEEE) Voir également analyse des exigences, phase d'établissement des exigences, spécification des exigences.

### exigence d'interface

Exigence qui spécifie le matériel, le logiciel ou l'élément de base de données auquel un système ou un élément de système doit être relié, ou qui expose les contraintes, causées par cette interface, qui s'exercent sur les formats, sur la synchronisation ou sur d'autres facteurs. (ANSI/IEEE)

### exigence de performance

Exigence qui spécifie une caractéristique de performance que doit posséder un système ou un élément de système; par exemple, la vitesse, l'exactitude, la fréquence. (ANSI/IEEE)

**exigence fonctionnelle**

Exigence qui spécifie une fonction qu'un système ou un élément de système doit être capable d'exécuter.  
(ANSI/IEEE)

**exigence physique**

Exigence qui spécifie une caractéristique physique que doit posséder un système ou un élément de système; par exemple, la matière, la forme, la taille ou le poids.  
(ANSI/IEEE)

**exigences de conception**

Toute exigence qui a un impact ou qui exerce une influence sur la conception d'un système logiciel ou d'un élément de système logiciel. Il peut s'agir, par exemple, d'exigences fonctionnelles, d'exigences physiques, d'exigences de rendement ainsi que de normes d'élaboration et d'assurance qualité du logiciel.  
(ANSI/IEEE)

**exigences de la mise en oeuvre**

Toute exigence qui a un impact ou qui exerce une contrainte sur la mise en application de la conception d'un logiciel. Il peut s'agir, par exemple, de descriptions de conception, de normes d'élaboration d'un logiciel, d'exigences du langage de programmation et de normes d'assurance de la qualité d'un logiciel.  
(ANSI/IEEE)

**exploitabilité**

Effort nécessaire pour apprendre et faire fonctionner un programme, pour en préparer les données d'entrée et pour en interpréter les données de sortie.

**exploitation en mode simplex**

Méthode d'exploitation selon laquelle la communication entre deux postes s'effectue dans un seul sens à la fois.

---

**extensibilité**

Degré jusqu'où une conception d'une architecture, d'une procédure ou de données peut être étendue.

---

**extension de programme**

Augmentation du logiciel existant pour accroître ses possibilités. (ANSI/IEEE)

---

**facteur de groupage**

Nombre maximum d'enregistrements contenus dans un bloc sur un support de mémoire auxiliaire, pour un fichier particulier.

---

**fiabilité**

Probabilité de bon fonctionnement d'un système pendant un temps déterminé (en service ou en stock). (ANSI)

---

**fiabilité d'un logiciel**

- a. Probabilité qu'un logiciel ne cause pas la défaillance d'un système dans des conditions d'exploitation définies et dans une période de temps limitée. La probabilité est une fonction des données introduites dans le système et de l'utilisation de celui-ci, de même qu'elle est une fonction de l'existence de défauts dans le logiciel. Les données introduites dans le système indiquent l'existence de défauts, s'il y a lieu. (ANSI/IEEE)
- b. Aptitude d'un programme à exécuter une fonction donnée dans des conditions d'exploitation définies et dans une période de temps limitée. (ANSI/IEEE)



**fiabilité du système**

Probabilité qu'un système comprenant tous les sous-systèmes matériels et logiciels exécutera une tâche ou une mission donnée dans une période de temps déterminée dans des conditions d'exploitation définies. (ANSI/IEEE)

**fiabilité opérationnelle**

Fiabilité d'un système ou d'un sous-système logiciel dans son contexte d'utilisation réel. La fiabilité opérationnelle peut différer considérablement de la fiabilité dans un contexte déterminé ou dans un contexte d'essai. (ANSI/IEEE)

**fibre optique**

- a. Fibres de verre (ou d'un autre matériau) par l'intermédiaire desquelles est transmise l'énergie lumineuse.
- b. Technologie de l'emploi de fibres de verre (ou d'un autre matériau) pour la transmission de l'énergie lumineuse.

**fichier**

- a. Ensemble d'enregistrements. Le fichier du personnel, par exemple, contiendrait des informations sur les individus de l'établissement. Les informations sur chaque individu constitueraient un enregistrement propre à celui-ci. Les informations pourraient comprendre nom, âge, situation de famille, salaire, échelon, etc., chacun de ces points constituerait une zone dans son enregistrement.
- b. Ensemble d'enregistrements apparentés traité comme un tout. (ISO)

**fichier de projet**

Dépôt central de documents tels que des mémos, des plans, des rapports techniques, etc., relatifs à un projet. (ANSI/IEEE)

**fichier logique**

Fichier indépendant de son contexte physique. Les parties d'un même fichier logique peuvent se trouver dans des fichiers physiques différents tandis que plusieurs fichiers logiques ou parties de fichiers logiques peuvent se trouver dans le même fichier physique. (ANSI/IEEE)

**fichier maître**

Fichier de données de base, actuelles, périodiquement mis à jour et utilisé dans l'intervalle, par exemple, pour l'extraction de données pour des rapports.

**file d'attente**

Liste qui est sollicitée selon la méthode "premier entré, premier sorti". Antonyme de pile. (ANSI/IEEE)

**file d'attente de réinjection multiniveau**

Mesure d'ordonnancement de processus par élimination qui est particulièrement utile dans les systèmes avec des combinaisons de travaux diversifiés.

**fonction**

- a. But précis ou caractéristique d'une entité.  
(ANSI)
- b. Sous-programme appelé durant l'évaluation d'une expression dans laquelle son nom apparaît et qui retourne une valeur au point d'appel.  
(ANSI/IEEE)
- c. Fonction critique pour le vol. Fonction pour laquelle toute défaillance ou toute erreur de conception empêcherait la poursuite du vol et l'atterrissage en sécurité de l'aéronef.
- d. Fonction critique pour la mission. Fonction pour laquelle toute défaillance ou toute erreur de conception empêcherait l'équipage d'un aéronef de commencer, continuer ou terminer la mission donnée.

- e. Fonction essentielle pour le vol. Fonction pour laquelle toute défaillance ou toute erreur de conception réduirait les capacités de l'aéronef, ou les possibilités de l'équipage à faire face à des conditions de fonctionnement défavorables.
- f. Fonction essentielle pour la mission. Fonction pour laquelle toute défaillance ou toute erreur de conception réduirait les capacités de l'aéronef ou limiterait les choix laissés à l'équipage pour commencer, continuer ou terminer la mission donnée.
- g. Autre. Fonction pour laquelle des défaillances ou erreurs de conception ne réduiraient pas de façon importante les capacités de l'aéronef ni les choix laissés à l'équipage, quelles que soient les circonstances.

#### fonctionnement multitâche

Accomplissement concurrent de plusieurs tâches par un ordinateur. (ANSI)

#### fond de panier

Structure qui assure l'interface physique des modules/cartes que comprend un système. L'interface consiste en connecteurs de modules/cartes dans lesquels ceux-ci sont insérés pour maintenir la continuité de la connexion dans la structure de fond de panier.

#### force

Mesure relative de la force des relations entre les éléments internes d'un module dans la mesure où elles contribuent aux fluctuations dans les hypothèses faites par le programme externe concernant le rôle que joue le module dans le programme. L'absence de fluctuations dans les hypothèses sur un module indique une grande force. (DAN 1153)

**fortement typé**

Caractéristique d'un langage de programmation qui exige que le type de données de chaque objet de données soit déclaré et qui empêche l'application d'opérateurs à des objets de données inappropriés; cela prévient donc l'interaction des objets de données de type différents. (ANSI/IEEE)

**FORTTRAN**

Acronyme de "formula translation", langage de programmation pour la définition et la solution de problèmes mathématiques et scientifiques.

**fragmentation**

Sections restantes d'une mémoire qui sont trop petites pour satisfaire à une demande d'affectation. Division de zone de mémoire contiguë telle que mémoire principale ou secondaire, d'une façon qui cause la perte de certaines zones. Par exemple, dans la plupart des systèmes organisés en pages, les pages entières sont toujours affectées, même quand un programme ou un segment de données a une taille inférieure à celle de la page. Dans les systèmes segmentés ou dans un système avec affectation de mémoire dynamique, le chargement et l'échange constants de nouvelles tâches auront toujours pour effet de rendre de petites zones de mémoire inutilisables.

**générateur**

Un générateur produit des données d'essai ou des jeux d'essai pour tester le système prototype. Dans ce cas, le générateur est différent du simulateur, car il crée des données d'essai qui utilisent des intégrateurs numériques, des générateurs de nombres aléatoires, etc. Lorsque les données sont produites par le générateur, un simulateur peut être requis pour acheminer ces données au système. Les générateurs sont utiles pour tester les systèmes là où des données réelles ne sont pas disponibles. Les bandes magnétiques, qui peuvent être utilisées avec un dispositif de relecture pour établir les jeux d'essai standard, représentent des sorties utiles d'un générateur de données. (DAN 134)

### générateur d'essais automatisé

Outil logiciel qui accepte un programme machine et des critères d'essai comme entrée, qui génère des données d'entrée d'essai qui répondent à ces critères et qui détermine parfois les résultats attendus. (ANSI/IEEE)

### générateur de code

Programme ou fonction du programme, souvent compris dans un compilateur, qui fait passer un programme machine d'un niveau de représentation intermédiaire quelconque (souvent la sortie d'un analyseur syntaxique) à un niveau de représentation inférieur, tel que le code d'assemblage ou le code machine. (ANSI/IEEE)

### générateur de compilateurs

Traducteur ou interpréteur utilisé pour créer des compilateurs. (ISO) Synonyme de métacompilateur.

### générateurs de données d'essai

Système d'analyse automatique qui aide un utilisateur à choisir des données d'essai entraînant un programme à se comporter d'une certaine façon.

### génération de code

Après l'analyse de la syntaxe d'un programme, la dernière tâche de compilation est la génération de code objet. Quand l'analyseur syntaxique reconnaît une portion d'un programme source en fonction d'une règle grammaticale, le programme correspondant est exécuté. Ces programmes sont souvent qualifiés de programmes sémantiques; ils génèrent directement un code objet, aussi est-il habituel de s'y référer comme des programmes de génération de code. Avec des compilateurs plus complexes, les programmes sémantiques peuvent générer une forme intermédiaire de programme qui sera analysé plus en détail pour essayer de générer un code plus efficace.

### g nie logiciel

Approche syst matique de l' laboration, de l'exploitation, de la maintenance et de la mise au rancart des logiciels. (ANSI/IEEE)

### gestion de la configuration

Processus d'identification et de d finition des  l ments de la configuration dans un syst me, de contr le de la diffusion et de la modification de ces  l ments durant tout le cycle de vie du syst me, d'enregistrement et de d claration de l' tat des  l ments de la configuration et des demandes de changement, et de v rification de la compl tude et de l'exactitude de ces  l ments. Voir  galement contr le des changements, identification de la configuration, contr le de la configuration, documentation sur l' tat de la configuration et v rification de la configuration. (ANSI/IEEE)

### gestion de produit de base

Application de directives techniques et administratives pour d signer les documents et les corrections qui y sont apport es, qui identifient, d signent et contr lent officiellement l' l ment de configuration   des moments particuliers de son cycle de vie.

### gestion des ressources

Une t che demande une ressource pour sa propre ex cution; si la ressource n cessaire n'est pas disponible, la t che attend qu'elle le devienne. La ressource est alors affect e exclusivement   la t che demanderesse qui commence   l'utiliser en fonction de ses besoins. Lorsque la t che n'a plus besoin de la ressource, le service de r int gration est employ  pour renvoyer la ressource dans le syst me d'exploitation qui l'affecte alors   la t che demanderesse suivante.

### gestion m moire

Syst me de contr le adaptatif qui attribue des espaces m moire et organise les op rations du processeur central de fa on   en maximiser le rendement. (DAN 595)

**graphe**

Modèle comprenant un ensemble fini de noeuds ayant des connexions appelées arêtes ou arcs. (ANSI/IEEE)

**graphe orienté**

Graphe dont les côtés sont unidirectionnels.  
(ANSI/IEEE)

**gravité**

Classification d'une erreur ou d'une défaillance du logiciel, basée sur l'évaluation du degré d'impact de cette erreur ou de cette défaillance sur l'élaboration ou le fonctionnement d'un système (terme souvent employé pour déterminer si, ou à quel moment, la défaillance doit être corrigée). (ANSI/IEEE)

**grouper**

Assembler les enregistrements en blocs en vue de leur traitement. (ADatP-2)

**guide d'élaboration d'un logiciel**

Ensemble du matériel relatif à l'élaboration d'un module logiciel donné. Comprend généralement les exigences, la conception, les rapports techniques, les listes de codes, les plans d'essai, les résultats d'essai, les rapports sur les problèmes, les programmes, les notes, etc. relatifs au module. (ANSI/IEEE)

**habitable virtuel**

Système de visualisation placé dans l'habitable dans lequel les commandes et les affichages réels sont remplacés par des illusions sensorielles faisant appel à l'infographie, aux systèmes audio à gestion informatique et à des capteurs tactiles artificiels.

**hiérarchie**

Structure dont les éléments sont classés à divers niveaux de subordination suivant un ensemble de règles précis. (ANSI/IEEE)

**homologation**

Processus dont l'objet est de confirmer qu'un système, un sous-système logiciel ou un programme machine peuvent répondre à des exigences déterminées dans un cadre d'exploitation donné. L'homologation s'effectue habituellement sur place, dans des conditions réelles, et elle sert à évaluer non seulement le logiciel lui-même, mais aussi ses spécifications d'élaboration. De plus, l'homologation permet de vérifier et de valider le cadre d'exploitation réel ou simulé. (ANSI/IEEE)

**identificateur**

Symbole qui identifie, qui indique ou qui donne un nom à une donnée ou à un groupe de données. (ANSI)

**identification de la configuration**

Processus de désignation des éléments de la configuration dans un système et d'enregistrement de leurs caractéristiques. (ANSI/IEEE)

**imbrication**

Insertion séquentielle de segments d'un programme dans un autre programme, de telle façon que les deux programmes puissent effectivement être exécutés simultanément. (Computer dictionary)

**implantation d'erreurs**

Addition volontaire d'un nombre connu d'erreurs à celles déjà présentes dans le programme machine afin d'évaluer le nombre d'erreurs propres au programme. (ANSI/IEEE)



**indépendant d'une machine**

Qualifie des procédures ou des programmes dont la définition ne tient pas compte de la machine sur laquelle ils seront exécutés. (ADatP-2)

**indicateur d'assiette électronique**

Affichage fournissant des informations sur l'assiette par l'intermédiaire d'un tube cathodique (CRT) ou d'une matrice luminescente. D'autres informations comme altitude, angle d'incidence, vitesse propre, etc. peuvent aussi être affichées. Divers modes, comme décollage, croisière, largage d'armes, atterrissage, etc. peuvent généralement être choisis.

**indicateur/Pointeur**

- a. Donnée élémentaire qui indique l'emplacement d'une autre donnée élémentaire. (ANSI/IEEE)
- b. Emplacement de mémoire, utilisé comme un opérande, contenant l'adresse d'un autre emplacement de mémoire.

**indice inférieur**

Clé numérique qui identifie un élément particulier dans un ensemble d'une table.

**informatique répartie**

- a. Mode d'utilisation des ordinateurs dans lequel les programmes sont mis en oeuvre localement sur les ordinateurs satellites et seules les données nécessaires à l'échelon central sont transférées vers l'ordinateur central. (ADatP-2)
- b. En avionique, calcul accompli par un système dans lequel les fonctions de calcul sont réparties entre plusieurs éléments du système.

---

**ingénierie des systèmes**

Processus itératif de gestion technique qui transforme un besoin opérationnel en description de paramètres et intègre ces paramètres parmi tous les autres éléments afin de déboucher sur une conception aussi équilibrée que possible.

---

**insensibilité aux défauts**

Capacité intégrée d'un système d'assurer une bonne exécution continue en présence d'un nombre limité de défauts logiciels ou matériels. (ANSI/IEEE)

---

**inspection**

- a. Technique d'évaluation formelle suivant laquelle les exigences, la conception ou le code d'un logiciel sont examinés en détail par une personne ou par un groupe autre que celui qui les a créés pour détecter les défauts, la violation des normes d'élaboration et d'autres problèmes. (ANSI/IEEE) En contraste avec revue de projet.
- b. Activité qui implique l'examen de documents particuliers.

---

**instruction**

Spécifie une opération et identifie les opérandes. (ADatP-2) Voir aussi instruction machine.

---

**instruction d'affectation**

Instruction utilisée pour décrire une suite d'opérations ou pour attribuer des opérandes à des variables déterminées, ou à des symboles, ou aux deux. (ANSI)

---

**instruction de cas**

Instruction conditionnelle à branchement multiple qui permet l'exécution sélective de groupes liés d'instructions de programmes en fonction de la valeur

d'une expression de commande. (ANSI/IEEE) Voir également structure de contrôle.

### instruction de choix d'une branche

Expression logique indiquant les conditions d'entrée qui doivent être remplies pour que les opérations d'une branche particulière du programme soient exécutées. (ANSI/IEEE)

### instruction de contrôle

Instruction du langage de programmation qui détermine l'ordre dans lequel les opérations sont exécutées. (ANSI/IEEE)

### instruction logique

Instruction qui exploite des chiffres binaires, en utilisant la logique Booleenne.

### instruction machine

- a. Instruction exécutable par un ordinateur déterminé. (ADatP-2)
- b. mInstruction pouvant être reconnue et exécutée par le calculateur pour lequel elle est conçue. (ADatP-2)

### instruction privilégiée

Instruction qui ne peut être utilisée que par un superviseur. (ISO)

### intégration

Voir également "system integration/intégration de système".

**intégration à grande échelle (LSI)**

Technologie permettant de fabriquer sur une seule puce à semi-conducteurs des milliers de dispositifs électroniques (voir également VLSI, MSI, VHSIC).

**intégration à moyenne échelle (MSI)**

Technologie avec laquelle de 10 à 1.000 dispositifs électroniques environ sont fabriqués sur une seule puce à semi-conducteur.

**intégration à très grande échelle (VLSI)**

Technologie permettant de fabriquer sur une seule puce à semi-conducteurs des dizaines de milliers de dispositifs électroniques.

**intégration système**

Processus d'interconnexion d'éléments fonctionnels pour former un tout organisé.

**intégrité**

Indique dans quelle mesure l'accès non autorisé au logiciel ou aux données, ou la modification de ceux-ci, peut être contrôlé dans un système informatique.  
(ANSI/IEEE)

**interactif**

Relatif à un système qui fournit une réponse à chaque entrée de l'utilisateur. (ANSI/IEEE)

**interface**

- a. Frontière ou point commun de deux entités, ou plus, par lequel l'information circule.  
(ADatP-2)
- b. En avionique, moyen par lequel des éléments fonctionnels sont connectés. Ces moyens peuvent être physiques, électriques, mécaniques

ou fonctionnels et peuvent être utilisés pour transmettre l'information.

- c. Frontière commune. Une interface peut consister en un composant matériel reliant deux dispositifs ou en une portion de mémoire ou de registre utilisée par deux programmes, ou plus. (ANSI)
- d. Interaction ou communication avec un composant d'un autre système.
- e. Initialement, terme faisant référence au circuit de contrôle entre deux unités matérielles. Maintenant s'applique aussi au logiciel, par exemple pour faire référence aux paramètres qui circulent entre un programme et ses sous-programmes.

### interpréter

Traduire et exécuter chaque instruction en langage source d'un programme machine avant de traduire et d'exécuter l'instruction suivante. En contraste avec assembler et compiler. (ISO)

### interpréteur

- a. Programme qui traduit et exécute chaque instruction d'un langage source avant de traduire et d'exécuter la suivante. (ADatP-2)
- b. Dispositif qui imprime en clair sur une carte les caractères perforés dans cette carte. (ADatP-2)

### interrompre

Arrêter un processus de façon à pouvoir le reprendre.

### interruption

Événement qui modifie la séquence selon laquelle un processeur exécute des instructions. Elle est produite

par le matériel du système informatique. Quand une interruption se produit, le système d'exploitation :

- a. Prend le contrôle.
- b. Sauvegarde l'état du processus interrompu. Dans beaucoup de systèmes, cette information est stockée dans le bloc de commande de processus (PCB) du processus interrompu.
- c. Analyse l'interruption et passe le contrôle au programme approprié pour la traiter. Il y a six types d'interruptions :
  - (1) Interruption d'appel au superviseur (SVC).
  - (2) Interruption d'entrée/sortie.
  - (3) Interruption externe.
  - (4) Interruption de redémarrage.
  - (5) Interruption de contrôle de programme.
  - (6) Interruption d'horloge et de contrôle machine.

### itération

- a. Exécution répétitive d'une suite d'instructions en langage de programmation jusqu'à ce qu'une condition donnée soit remplie ou tant qu'elle est vraie. (ANSI/IEEE)
- b. Exécution unique d'une boucle. (ANSI/IEEE)

### jalón

Evénement planifié dont un participant au projet ou un chef de projet est tenu pour responsable et qui sert à mesurer les progrès réalisés; par exemple, une revue officielle, l'émission d'une spécification, la livraison d'un produit. (ANSI/IEEE)

JCL

Abréviation de langage de contrôle de travaux ("Job Control Language"). Langage de contrôle informatique spécial conçu pour utilisation dans les systèmes par lots afin d'informer le logiciel des systèmes et l'opérateur du calculateur des besoins propres au déroulement d'un programme.

jeu d'essai

Ensemble précis de données d'essai et de procédures connexes élaborées en fonction d'un objectif particulier, notamment pour essayer un branchement de programme particulier ou vérifier la conformité à une exigence spécifique. (ANSI/IEEE)

jeu d'instructions

Répertoire d'instructions pour un calculateur donné.

journal

Enregistrement d'une phase de traitement, souvent sous la forme d'une séquence de messages par une machine à écrire de pupitre.

journal d'essai

Enregistrement chronologique de toutes les données détaillées pertinentes relatives à un essai. (ANSI/IEEE)

justesse

- a. Indique dans quelle mesure le logiciel est exempt de défauts de conception et de programmation; c'est-à-dire dans quelle mesure il est mis au point. (ANSI/IEEE)
- b. Indique dans quelle mesure le logiciel est conforme aux exigences spécifiées. (ANSI/IEEE)
- c. Indique dans quelle mesure le logiciel répond aux désirs de l'utilisateur. (ANSI/IEEE)

**kilo (K)**

Utilisé pour exprimer le nombre  $2^{10}$  (1024) en terminologie numérique (exemple : 16 K mots de mémoire = 16.384 mots).

**label**

- a. Un ou plusieurs caractères, à l'intérieur d'un ensemble de données, ou rattachés à celui-ci, contenant des informations sur cet ensemble, notamment son identification. (ISO)
- b. Enregistrement d'identification pour une bande ou un fichier-disque. (ANSI/IEEE)
- c. En programmation, identificateur d'une instruction. (ISO)

**langage à structure de blocs**

Langage de conception ou de programmation dont les suites d'instructions sont habituellement délimitées par des bornes de début et de fin. (ANSI/IEEE)

**langage cible**

Langage dans lequel les instructions source sont traduites. (ANSI) En contraste avec langage source.

**langage d'assemblage**

Langage de programmation symbolique à partir duquel on assemble un programme machine correspondant. Les instructions dans un programme de langage machine sont habituellement en correspondance biunivoque avec celles du programme de langage d'assemblage. (ADatP-2)

**langage d'assemblage ou code d'assemblage**

Langage de programmation dans lequel la forme numérique d'une instruction de code machine est représentée par un code mnémonique, par exemple "ADX" pour un code numérique signifiant "mettre le contenu de l'emplacement de mémoire dans l'accumulateur".



**langage de commande**

Ensemble d'opérateurs de traitement, avec syntaxe connexe, utilisés pour indiquer les fonctions qui doivent être exécutées par le système d'exploitation. (ISO)  
Synonyme d'ordres de gestion.

**langage de conception**

Langage comportant des structures spéciales et, parfois, des protocoles de vérification utilisés pour élaborer, analyser et documenter une conception. (ANSI/IEEE)

**langage de conception de programme (PDL)**

Technique d'utilisation d'instructions en langue anglaise pour décrire le processus logique d'un élément fonctionnel du système. Le PDL est un outil pour le développement du programme et une partie essentielle de la documentation du système.

**langage de conception de système**

Programme machine utilisé pour fournir une représentation compréhensible de la conception du logiciel durant son évolution. Ce programme permet l'élaboration et l'expansion de la conception de façon hiérarchique. Il documente la conception et les décisions connexes. (DAN 134)

**langage de programmation**

Langage destiné à l'établissement de programmes d'ordinateur. (Langage évolué et langage d'assemblage). (ADatP-2)

**langage de programmation structuré**

Langage de programmation qui fournit les structures de programmes structurés et qui facilite l'élaboration de programmes structurés. (ANSI/IEEE)

### langage de spécification

Langage, qui est souvent la combinaison exploitable par une machine du langage naturel et d'un langage formel, utilisé pour spécifier les exigences, la conception, le comportement ou d'autres caractéristiques d'un système ou d'un élément de système. (ANSI/IEEE)

### langage de spécification des exigences

Langage formel possédant des protocoles de vérification et des structures spéciales pour spécifier, vérifier et documenter les exigences. (ANSI/IEEE)

### langage évolué

- a. Langage qui permet la représentation des algorithmes et des structures de données sous une forme symbolique indépendante d'une machine avec une teneur élevée en informations (et non en données) et qui, par conséquent, est compris plus facilement par les humains que le langage machine ou le langage assembleur.
- b. Langage de programmation qui comprend habituellement des éléments comme les expressions emboîtées, types de données définies par l'utilisateur et transfert de paramètres que l'on ne trouve normalement pas dans des langages moins évolués, qui ne reflète pas la structure d'un ordinateur ou d'une classe d'ordinateurs donnés et qui peut être utilisé pour écrire des programmes source non liés à une machine particulière. Une instruction unique de langage évolué peut représenter des opérations machine multiples. Différent de langage machine, langage d'assemblage. (ANSI/IEEE)

### langage formel

Langage dont les règles sont explicitement établies avant son utilisation. Synonyme de langage artificiel. Les langages de programmation, tels le FORTRAN et l'Ada, et les langages mathématiques ou logiques, tel le calcul des prédicats, représentent quelques exemples du langage formel. (ANSI/IEEE) En contraste avec langage naturel.

**langage machine**

Langage utilisé directement par une machine.  
(ADatP-2)

**langage naturel**

Langage dont les règles sont fondées sur l'usage courant sans être établies explicitement. (ISO)  
L'Anglais, le Chinois, le Français et le Swahili en sont des exemples. Antonyme de langage formel.

**langage orienté d'application**

- a. Langage orienté vers la machine dont les ressources ou les notations peuvent tout d'abord être associées à une zone de mono-application; par exemple, un langage pour l'analyse statistique ou pour la conception d'une machine. (ANSI/IEEE)
- b. Langage orienté vers un problème et dont les instructions contiennent la terminologie de l'occupation ou de la profession de l'utilisateur ou une terminologie équivalente. (ANSI)

**langage orienté vers les problèmes**

Langage informatique conçu pour faciliter la programmation des problèmes dans un domaine particulier, plutôt que de tirer profit des possibilités d'une machine particulière. Opposé à langage orienté vers la machine.

**langage source**

- a. Programme machine rédigé dans un langage source. (ADatP-2)
- b. Langage utilisé pour rédiger des programmes sources. (ANSI/IEEE)
- c. Langage à partir duquel les instructions sont traduites. (ISO) Antonyme de langage objet. (ANSI/IEEE)

**les points critiques d'abord**

Expression se rapportant à une approche de création de logiciel qui vise tout d'abord à mettre en place les aspects les plus critiques d'un système logiciel. Les points critiques peuvent être définis en fonction des services fournis, du degré de risque, des difficultés ou d'autres critères. (ANSI/IEEE)

**lexical**

Se rapporte aux mots d'un langage; opposé à syntaxique ou grammatical qui signifie "se rapportant aux relations entre les mots dans les structures autorisées d'un langage". L'analyse lexicale porte sur l'identification de mots isolés dans une instruction et constitue la première fonction exécutée par un compilateur.

**libération**

Service de système qui informe le système d'exploitation qu'une tâche n'exige plus l'utilisation d'une ressource.

**lien**

Instruction de branchement à la fin d'un sous-programme pour remettre la commande à un point dans le programme principal à partir duquel le sous-programme est appelé.

**linker**

Voir "link editor" (éditeur de liens).

**listage**

Sortie d'ordinateur sous forme de liste pouvant être lue par l'utilisateur. (ANSI/IEEE)

**listage source**

Format du code source que peut lire l'utilisateur et qui est annoté pour décrire les modules, les fonctions et le déroulement du programme. (RTCA)

**liste**

- a. Ensemble ordonné d'éléments d'information. (ISO)
- b. Impression ou affichage des éléments d'information qui répondent à des critères précis. (ANSI)

**liste chaînée**

Liste dans laquelle les articles peuvent être dispersés mais où chaque article contient un identificateur pour l'emplacement physique du prochain article à considérer. (ANSI)

**liste d'attente**

Liste de tâches attendant un événement ou un message dans une boîte à lettres. Chaque boîte à lettres a une liste d'attente distincte, s'il y a un message ou une tâche dans la "boîte à lettres". En revanche, s'il n'y a pas de messages dans la boîte à lettres ou s'il y a, au moment considéré, des tâches en attente dans la liste d'attente de la boîte à lettres, la tâche demanderesse est suspendue et placée à la fin de la liste d'attente.

**liste de liens**

Structure de données où chaque élément contient un indicateur sur son prédécesseur ou son successeur (liste à lien unique) ou sur les deux (liste à lien double).

**liste de réserve**

Liste de tâches qui sont prêtes, c'est-à-dire tâches qui sont prêtes à être exécutées sur un processeur.

**liste de tâches**

Structure des données d'un système contenant une liste des tâches se trouvant dans le système.

**liste libre**

Liste d'emplacements de mémoire qui sont actuellement inutilisés et peuvent être affectés par la gestion de mémoire aux tâches en attente. Les listes libres sont habituellement organisées en listes reliées de blocs de mémoire, où chaque bloc contient la taille du bloc et un indicateur sur le prochain bloc de la liste.

**livraison**

- a. Moment, dans le cycle d'élaboration d'un logiciel, où le produit est remis à l'utilisateur pour lequel il a été conçu en vue de son utilisation opérationnelle. (ANSI/IEEE)
- b. Moment, dans le cycle d'élaboration d'un logiciel, où l'utilisateur accepte le produit qui a été conçu pour lui. (ANSI/IEEE)

**logiciel**

Ensemble des programmes machines, des procédures, des règles et, éventuellement, de la documentation s'y rapportant, ayant trait à l'exploitation d'un système informatique. (ADatP-2)

**logiciel d'application**

Logiciel écrit pour effectuer une tâche fonctionnelle voulue et qui est distinct des systèmes d'exploitation, d'exécution et des programmes superviseurs généralement.

**logiciel de base**

Logiciel conçu pour un système informatique ou un groupe de systèmes informatiques précis en vue de faciliter l'exploitation et la maintenance du système informatique et des programmes connexes; par exemple, les systèmes d'exploitation, les compilateurs et les

programmes utilitaires. (ANSI/IEEE) En contraste avec logiciel d'application.

### logiciel de contrôle des normes d'élaboration

Outil logiciel qui détermine si les normes d'élaboration prescrites ont été respectées. Parmi les normes, mentionnons la taille et la structure du module, les conventions qui régissent les observations et la documentation et l'utilisation de certaines formes d'instructions. (ANSI/IEEE)

### logiciel de soutien

Tous les programmes utilisés dans l'élaboration et la maintenance des programmes opérationnels livrés et des programmes d'essai/de maintenance. Les programmes de soutien comprennent, entre autres :

- a. Les compilateurs, les assembleurs, les émulateurs, les générateurs et les chargeurs nécessaires pour produire le code machine et pour combiner les routines ou les éléments dans un programme machine complet.
- b. Les programmes de mise au point.
- c. Les programmes de stimulation et de simulation utilisés dans les endroits où l'on forme des opérateurs.
- d. Les programmes d'abstraction et de réduction de données applicables aux programmes opérationnels utilisés pour le contrôle de la gestion, la gestion de la configuration ou la production et le contrôle de documents durant l'élaboration. (DAN 384)

### logiciel intégré

Logiciel d'un système informatique intégré.  
(ANSI/IEEE)

**logiciel opérationnel (intégré)**

Partie d'un programme machine, comprenant des programmes d'application et de supervision en temps réel, qui est utilisée directement dans le pilotage des avions et la commande du matériel électronique embarqué. (GLOS-1)

**logiciel utilitaire**

Programmes ou routines machine conçus pour exécuter des fonctions de soutien général requises par un autre logiciel d'application, par le système d'exploitation ou par les utilisateurs du système. (ANSI/IEEE)

**logique booléenne**

Système de déduction se rapportant aux classes, propositions et éléments de circuit à deux états, représentés par des symboles et termes comme ET, OU, NON-ET.

**longueur de mot**

Mesure de la taille des mots, généralement indiquée en unités telles que l'octet ou le bit.

**lot**

Type de système d'exploitation où les travaux sont effectués un par un. Un travail doit être terminé avant qu'un autre ne soit commencé.

**machine**

Terme souvent utilisé pour désigner un ordinateur.

**machine abstraite**

- a. Représentation des caractéristiques d'un traitement ou d'une machine. (ANSI/IEEE)
- b. Module qui traite les entrées comme s'il était une machine. (ANSI/IEEE)



**machine cible**

Ordinateur sur lequel un programme doit être exécuté. (ANSI/IEEE)

**machine virtuelle**

Simulation fonctionnelle d'un ordinateur et de ses périphériques. (ANSI/IEEE)

**macro-instruction**

Instruction en langage source qui doit être remplacée par une suite définie d'instructions dans le même langage. La macro-instruction peut également préciser des valeurs pour les paramètres des instructions qui doivent la remplacer. (ISO)

**macroprocesseur**

La partie de certains assembleurs et compilateurs qui permet à un programmeur de définir et d'utiliser des macro-instructions. (ANSI/IEEE)

**maintenabilité**

- a. Facilité avec laquelle un logiciel peut être maintenu en état de bon fonctionnement. (ANSI/IEEE)
- b. Facilité avec laquelle la maintenance d'une unité fonctionnelle peut être effectuée conformément aux exigences établies. (ISO)
- c. Aptitude d'un article, dans des conditions d'utilisation déterminées, à être conservé ou rétabli, pendant une période de temps donnée, dans un état dans lequel il puisse exécuter ses fonctions lorsque la maintenance est effectuée dans les conditions spécifiées et lorsqu'il utilise les procédures et les ressources prévues. (ANSI/ASQC A3-1978)

---

**maintenance adaptative**

Maintenance effectuée pour permettre l'utilisation d'un produit logiciel dans un cadre différent.  
(ANSI/IEEE)

---

**maintenance du logiciel**

Modification d'un produit logiciel après sa livraison en vue d'en corriger les défauts, d'améliorer ses performances ou d'autres attributs ou de l'adapter à un contexte d'utilisation différent. (ANSI/IEEE) Voir également maintenance adaptative et dépannage.

---

**masquage de l'information**

Technique d'encapsulation des décisions sur la conception de logiciels dans des modules de sorte que les interfaces de ces modules révèlent le moins de choses possible sur le fonctionnement interne de ceux-ci. Ainsi, chaque module est une "boîte noire" pour les autres modules du système. Le masquage de l'information interdit l'utilisation d'informations sur un modèle qui n'est pas dans la spécification de l'interface du module.  
(ANSI/IEEE)

---

**matériel**

- a. En avionique, équipement physique, par opposition au programme informatique ou au mode d'emploi, par exemple : dispositifs mécaniques, magnétiques, électriques ou électroniques.
- b. Appareillage physique d'une installation informatique.

---

**matériel de servitude au sol (GSE)**

Matériel nécessaire à la maintenance d'un aéronef et des équipements associés.

---

**matrice**

Pour le matériel, une trame, par exemple de tores magnétiques; en mathématique, une suite de nombres; pour le logiciel, souvent synonyme de tables.

**mémoire**

Dispositif capable d'enregistrer des programmes machine et des données connexes exploitables par une machine. Ce peut être une puce à circuits intégrés, une carte à circuits contenant des circuits intégrés, une mémoire à tores, un disque, une bande magnétique, etc. (RTCA)

**mémoire à accès sélectif (RAM)**

Mémoire dont les mots peuvent être lus ou écrits dans tout ordre voulu.

**mémoire à liste inversée**

Mémoire qui manipule les données de telle sorte que la prochaine donnée qui en sera extraite est la dernière entrée, selon la méthode dite "dernier entré, premier sorti" (LIFO). (ISO)

**mémoire à tores**

Mémoire d'une unité centrale consistant dans la plupart des machines actuelles en tores de ferrite magnétiques.

**mémoire associative**

Même traduction pour "associative memory" et "content addressable memory".

**mémoire auxiliaire**

Egalement appelées "mémoire de masse" (bulk store ou mass store), ces mémoires sont extérieures à l'ordinateur auquel elles sont cependant directement reliées et par lequel elles sont commandées (exemple : disques ou tambours magnétiques en ligne).

**mémoire centrale**

Mémoire de calculateur qui fait partie de l'unité centrale et qui est utilisée pour contenir les programmes ou parties de programmes en cours d'exécution. Synonyme de mémoire principale. (ADatP-2)

**mémoire d'archivage**

Fonctionnellement similaire à la mémoire auxiliaire, sauf que l'on utilise des supports d'information autonomes.

**mémoire de grande capacité**

Voir mémoire auxiliaire ("secondary memory").

**mémoire de masse**

- a. Mémoire auxiliaire de grande capacité, comme un disque magnétique.
- b. Voir mémoire secondaire.

**mémoire morte (ROM)**

Mémoire qui ne peut être modifiée au cours du fonctionnement de l'ordinateur.

**mémoire morte programmable (PROM)**

Mémoire qui autorise uniquement la lecture des données enregistrées dans des conditions d'exploitation.

**mémoire organisée en pages**

Mémoire divisée en portions de longueurs fixes.

**mémoire primaire**

Voir mémoire centrale.

**mémoire rémanente**

Mémoire qui n'a pas besoin d'être sous tension pour conserver les données enregistrées. (RTCA)

**mémoire transitoire**

Dispositif de mémoire dans lequel les informations qui s'y trouvent stockées sont perdues lorsque le courant d'alimentation est coupé. Exemple : une ligne à retard acoustique. (ADatP-2)

**mémoire virtuelle**

Mémoire apparente vue par le programmeur, qui libère celui-ci des contraintes de dimension et de zone de la mémoire physique et qui utilise un plan d'adresses de mémoire logique pour relier les adresses entre la mémoire apparente et la mémoire physique.

**message**

Unité de communication entre des tâches. Dans un système typique, un message contient habituellement un en-tête avec des informations sur la taille du message, la tâche d'origine, etc.

**message guide-opérateur**

- a. Message informant l'utilisateur que le système est prêt à recevoir l'ordre suivant, le message suivant ou une autre demande de l'utilisateur. (ANSI/IEEE)
- b. Guider : informer l'utilisateur que le système est prêt à recevoir l'ordre suivant, l'élément suivant ou une autre donnée. (ANSI/IEEE)

**mesure de la complexité**

Facteur quantitatif d'un programme d'ordinateur qui mesure son poids logique. Exprimé mathématiquement par la formule  $v(g)=e-n+2$  où  $e$  est le nombre d'arcs et  $n$  le nombre de noeuds.

**mesure de la qualité**

Mesure quantitative qui indique jusqu'à quel point un logiciel possède un attribut donné qui influe sur sa qualité. (ANSI/IEEE)

**mesures de secours**

Dispositions prises pour récupérer des fichiers de données ou des logiciels, pour relancer un traitement ou pour utiliser du matériel informatique de remplacement après une panne du système ou un autre problème. (ANSI/IEEE)

**métalangage**

Langage utilisé pour spécifier un ou plusieurs langages. (ANSI/IEEE)

**méthode d'assertion inductive**

Technique de la preuve d'exactitude dans laquelle les assertions écrites décrivent les entrées, les sorties et les conditions intermédiaires d'un programme; un ensemble de théorèmes est élaboré, qui établit un rapport entre la satisfaction des assertions d'entrée et la satisfaction des assertions de sortie, et la validité de ces théorèmes est démontrée. (ANSI/IEEE)

**méthode des approximations successives**

Méthodologie d'élaboration de système dans laquelle les définitions des données et les étapes du traitement sont définies d'abord dans les grandes lignes, puis de façon de plus en plus détaillée. (ANSI/IEEE)

**méthode heuristique**

Méthode exploratoire de solution de problèmes dans laquelle on fait une évaluation des progrès vers une solution finale acceptable en utilisant une série de résultats approximatifs.

**méthodologie d'élaboration**

Approche systématique relative à la création d'un logiciel qui définit les phases de développement et qui précise les activités, les produits, les procédures de vérification et les critères d'achèvement pour chaque phase. (ANSI/IEEE)

**méthodologie de conception**

Méthode systématique d'élaboration d'une conception, comportant l'application ordonnée d'un ensemble déterminé d'outils, de techniques et de lignes directrices. (ANSI/IEEE)

**métrologie**

Indique dans quelle mesure le logiciel possède ou manifeste une caractéristique, une propriété qualitative ou un attribut donnés. (GLOS-1)

**mettre au point**

Ensemble d'opérations consistant à déceler, identifier et corriger les erreurs d'un programme ou les pannes d'un calculateur.

**micro-ordinateur**

Microprocesseur comportant d'autres composants tels que des mémoires, des horloges et différents dispositifs d'interface qui fonctionnent collectivement comme un ordinateur (les micro-ordinateurs peuvent être réalisés sous forme de circuits intégrés, d'ensembles de circuits intégrés, de cartes mono-circuit ou de cartes à circuits multiples).

**microcode**

Instructions élémentaires utilisées pour le contrôle du fonctionnement d'éléments du circuit; utilisé notamment pour mettre en oeuvre l'ISA.

**microprocesseur**

Puce ou ensemble de puces assurant des fonctions arithmétiques et logiques.

**microprogrammation**

Information introduite dans une mémoire morte, qui peut être modifiée en reprogrammant physiquement ou en remplaçant le dispositif, mais qui ne peut être modifiée dans le cadre du fonctionnement normal du calculateur.

**microprogramme**

Programme établi en microcode.

**mise à l'essai d'une fonction externe**

Vérification d'un système externe, comme il a été défini dans les spécifications externes.

**mise à l'essai de programmation**

Mise en fonction simulée : mise à l'essai d'un programme sans utiliser l'ordinateur, en appliquant des données d'essai et en vérifiant le résultat de chaque instruction exécutée.

**mise à l'essai du domaine**

Méthode de détermination fiable d'un ensemble de données d'essai qui mettront en lumière les erreurs de "domaine" et de limite. Elle comprend une analyse du programme par des "méthodes mathématiques" impliquant l'hypergéométrie.

**mise au point**

- a. Essai de programmes en cours de développement identifiant et corrigeant les erreurs qu'ils contiennent.



- b. Identification, analyse et correction des défauts suspectés. (IEEE) Comparer avec mise à l'essai.

### mise au point en clientèle

Fonctionnement initial du système réel logiciel-matériel chez un client en mode "essai" (à pleine capacité ou à capacité limitée) pour détecter autant d'erreurs existantes que possible.

### mise au point imparfaite

Dans la modélisation de fiabilité, hypothèse selon laquelle les essais pour corriger ou pour éliminer un défaut détecté ne sont pas toujours fructueux.  
(ANSI/IEEE)

### mise en oeuvre

- a. Réalisation d'une abstraction en termes plus concrets; en particulier, en termes de matériel, de logiciel, ou les deux.  
(ANSI/IEEE)
- b. Type de programme pouvant être exécuté sur machine ou pouvant être traduit automatiquement sous une forme exécutable sur machine.  
(ANSI/IEEE)
- c. Traduction d'une conception en un code et mise au point de ce code. (ANSI/IEEE)

### modèle

Représentation d'un processus, d'un dispositif ou d'un concept du monde réel. Voir également modèle analytique, modèle de disponibilité, modèle d'erreur, modèle de fiabilité, modèle d'essai statistique.  
(ANSI/IEEE)

**modèle analytique**

Représentation d'un traitement ou d'un phénomène par une série d'équations solubles, par opposition à la simulation. (ANSI/IEEE)

**modèle d'erreur**

Modèle mathématique utilisé pour prédire ou évaluer le nombre de défauts non corrigés, la fiabilité, le temps d'essai requis ou d'autres caractéristiques similaires d'un système logiciel. (ANSI/IEEE)

**modèle d'essai statistique**

Modèle qui établit un rapport entre les défauts du programme et l'ensemble (ou les ensembles) de données d'entrée qui révèlent ces défauts. Le modèle indique également la probabilité que ces défauts provoquent la défaillance du programme. (ANSI/IEEE)

**modèle de disponibilité**

Modèle utilisé pour prédire ou pour évaluer la disponibilité. (ANSI/IEEE)

**modèle de fiabilité**

Modèle utilisé pour prédire, estimer ou évaluer la fiabilité. (ANSI/IEEE)

**modification de programme**

- a. Modification intentionnelle d'une version de programme afin de déterminer la valeur des essais de programme pour déceler les modifications. (ANSI/IEEE)
- b. Opération consistant à modifier un programme en vue de déterminer la valeur des données d'essai de programme. (ANSI/IEEE)

**modularité**

Indique dans quelle mesure un logiciel est composé d'éléments discrets de telle sorte qu'une modification apportée à un élément a peu d'influence sur les autres éléments. (ANSI/IEEE)

**modulation par impulsions codées (PCM)**

Type de modulation de signal où l'onde de modulation est échantillonnée à intervalles réguliers, quantifiée en phases discrètes et transmise comme une combinaison d'impulsions.

**module**

- a. Unité de programme discontinue et identifiable pour assemblage, combinaisons avec d'autres unités de programme et chargement, par exemple programme d'entrée ou de sortie, assembleur, compilateur, éditeur de liens ou routine d'exécution. (ADatP-2)
- b. Unité fonctionnelle et préconditionnée de matériel, conçue pour être utilisée avec d'autres composants. (ADatP-2)
- c. Unité fonctionnelle, de logiciel ou de matériel, qui peut être combinée à d'autres unités de différentes façons pour former des ensembles plus grands.

**module critique**

Module ayant une ou plusieurs des caractéristiques suivantes :

- a. Satisfait plusieurs besoins.
- b. A un haut niveau de commande (réside relativement haut dans la structure du programme).
- c. Sans possibilité d'erreur.
- d. A des spécifications de performance bien déterminées. Les modules critiques devraient

être testés aussi tôt que possible. En outre, l'essai de régression devrait mettre l'accent sur la fonction du module critique.

#### module de chargement

- a. Résultat de traitement de l'éditeur de liens.  
(ADatP-2)
- b. Programme mis en forme en vue de son chargement en mémoire centrale pour être exécuté. (ADatP-2)

#### module objet

Module fourni en sortie par un assembleur ou un compilateur et repris en entrée par un éditeur de liens.

#### moniteur

Un moniteur détermine laquelle de deux ou plusieurs opérations a la priorité pour la gestion de l'exécution. Il passe la main à l'opération prioritaire et met l'autre opération ou les autres opérations en file d'attente.  
(DAN 922)

#### moniteur de bus

Dans les systèmes de bus de données, terminal ayant pour fonction de recevoir le trafic du bus et d'en extraire des informations sélectionnées en vue d'une utilisation ultérieure.

#### moniteur logiciel

Outil logiciel qui travaille en même temps qu'un autre programme machine et qui fournit des renseignements détaillés sur l'exécution de l'autre programme.  
(ANSI/IEEE)

**mot-machine**

- a. Mot rangé dans un emplacement de mémoire d'un ordinateur et pouvant être traité comme un tout. (ADatP-2)
- b. Voir "computer word".

**multiplet (octet)**

- a. Chaîne d'éléments binaires traitée comme un tout et généralement plus petite qu'un mot-machine. (ADatP-2)
- b. Groupe de chiffres binaires (généralement huit) (octet) adressable comme une seule unité.

**multiplexage à partage du temps (TDM)**

Transmission d'informations à partir de plusieurs sources de signaux par l'intermédiaire d'une seule voie de transmission, les différents éléments du signal étant échelonnés dans le temps.

**multiplexer**

- a. Emboîter ou transmettre simultanément plusieurs messages sur une voie unique. (ADatP-2)
- b. Utilisé parfois à la place de "multiple" (par opposition à "simplex") dans le contexte de la redondance.

**multiprocesseur**

Ordinateur comprenant deux ou plusieurs unités centrales ayant accès à une mémoire centrale commune. (ADatP-2)

**multiprogrammation**

Mode d'exploitation qui permet l'exécution simultanée de deux ou plusieurs programmes machine par un seul processeur. (ISO)

**MUX**

Abréviation de multiplex.

**N-aire**

Caractérisé par une sélection, un choix ou une condition qui ne peut avoir des valeurs ou des états différents. (ISO)

**niveau de documentation**

Description de la documentation requise indiquant son champ d'application, son contenu, son format et sa qualité. La sélection du niveau peut être fondée sur le coût du projet, l'usage qu'on veut en faire, l'effort requis ou d'autres facteurs. (ANSI/IEEE)

**noeud**

- a. Position dans une structure arborescente définie par des indicateurs, soit sur le noeud au-dessus ou au-dessous d'elle dans la hiérarchie, soit sur les deux.
- b. Point d'un organigramme qui indique un état ou un événement. (ANSI)

**non définie**

Etat d'une tâche inconnue du système d'exploitation. Cela signifie que les informations relatives au descripteur de tâche n'ont pas été introduites dans les tables du système. Se dit d'une tâche qui n'a pas encore été créée ou qui a été annulée.

**normes**

Spécifications relatives à la méthode de développement, aux procédures, aux règles, aux conventions et aux lignes directrices utilisées pour prescrire, totalement ou partiellement, la conception, le codage et l'essai d'un programme. (RTCA)

### normes du génie logiciel

Ensemble de lignes directrices et (ou) de contraintes permettant d'assurer la cohérence durant la période d'élaboration; elles peuvent s'appliquer à des domaines tels que la nomenclature, l'affectation de noms, l'écriture de labels, le codage, l'utilisation de l'ordinateur, la documentation, la procédure et la mise au point. (DAN 300)

### notation pondérée

Système de numération assurant la représentation d'un nombre par une suite de caractères à chacun desquels on attribue une valeur dépendant à la fois de sa nature et de sa position dans la suite.

### noyau

- a. Pièce maîtresse ou coeur, tout comme le noyau d'un système d'exploitation. (ANSI/IEEE)
- b. Encapsulation d'une fonction élémentaire. Les noyaux peuvent être combinés pour former une partie, ou l'ensemble, d'un système d'exploitation ou un ensemble de microprogrammes. (ANSI/IEEE)
- c. Modèle utilisé dans les études sur la sélection d'un ordinateur pour en évaluer la performance. (ANSI/IEEE)

### noyau de sécurité

Petit ensemble autonome d'instructions clefs relatives à la sécurité qui fonctionne comme une partie privilégiée d'un système d'exploitation. On doit répondre à tous les critères spécifiés pour avoir accès à un programme ou aux données. (ANSI/IEEE)

### numérique

Se rapporte aux chiffres ou aux représentations de données ou de quantités physiques par des chiffres. (ADatP-2)

---

**numérique à analogique (N/A)**

Conversion de données numériques en signal analogique.

---

**occultation**

- a. Action d'empêcher une modification, une utilisation, une destruction ou une diffusion de données non autorisées durant un accès autorisé. (ANSI/IEEE) Voir également intégrité.
- b. Restriction appliquée à des programmes et à des traitements pour les empêcher d'avoir accès à des données, à des programmes ou à des opérations ou d'exercer une influence sur eux s'ils n'ont reçu aucune autorisation spéciale d'agir ainsi. (ANSI/IEEE)

---

**opérande**

Elément auquel s'applique une opération. Un opérande est habituellement défini par une plage d'adresses d'une instruction. (ANSI)

---

**opérateur**

- a. Dans la description d'un processus, symbole représentant le type d'action à effectuer sur des opérandes. (ISO)
- b. Personne qui fait fonctionner un appareil. (ANSI)

---

**opérations exécutées en simultanéité**

Opérations pouvant être exécutées parallèlement sur plusieurs processeurs ou de façon asynchrone sur un seul processeur. Les opérations exécutées en simultanéité peuvent agir réciproquement les unes sur les autres, et une opération peut interrompre temporairement l'exécution d'un travail durant la réception d'informations provenant d'un autre traitement ou à l'occasion d'un événement



externe quelconque. (ANSI/IEEE) En contraste avec de processus séquentiels.

### optimisation

Modifications apportées au code source pour améliorer le rendement du programme; par exemple, une exécution plus rapide ou l'utilisation d'un espace plus restreint. L'optimisation n'est pas une correction d'erreur; toutefois, le mot "erreur" est justifié si on effectue un changement visant à occuper moins d'espace en vue de se conformer aux limites d'espace spécifiées. Remarque : l'efficacité est une caractéristique qualitative; l'optimisation peut être une opération qui accroît l'efficacité. (GLOS-1)

### ordinateur/calculateur

Machine de traitement de l'information capable d'effectuer des calculs importants, comportant de nombreuses opérations arithmétiques ou logiques, sans intervention humaine en cours d'exécution. (ADatP-2)

### ordinateur central

Ordinateur utilisé pour préparer des programmes qui seront employés sur un autre ordinateur.

### ordinateur universel

Ordinateur conçu pour traiter une grande variété de problèmes.

### ordonnancement

- a. Module de système d'exploitation qui choisit le prochain travail à admettre dans le système et le prochain processus à faire passer en machine.
- b. Dans les systèmes d'exploitation, désigne l'attribution des travaux rangés dans une file d'attente de manière à assurer une exécution efficace ou suivant un ordre de priorité préétabli. (GLOS-1)

### ordonnancement à priorités

Algorithme d'ordonnancement où chaque tâche reçoit une priorité, et la prochaine tâche à exécuter est celle qui a la plus haute priorité sur la liste de réserve. Les priorités peuvent être statiques ou dynamiques. Avec les priorités statiques, il est possible que quelques tâches de faible priorité ne soient jamais exécutées à cause de la fréquence et du délai d'exécution des tâches de priorités plus élevées. Ceci est malsain dans un système à temps partagé où il est nécessaire de garantir que toutes les tâches bénéficieront d'un temps du processeur sur une base régulière. Pour ces systèmes, la priorité de la tâche est souvent changée en fonction du temps, de telle façon que plus longtemps une tâche attend d'être exécutée, plus sa priorité devient élevée.

### ordonnancement à tour de rôle

Moyen d'ordonnancement grâce auquel les tâches figurant dans la liste de réserve sont exécutées dans l'ordre et les entrées dans la liste de réserve sont toujours mises à la fin de la liste. Généralement, un certain délai est alloué pour l'exécution de chaque tâche. Si, passé ce délai, la tâche n'est pas terminée, elle est remise à la fin de la liste de réserve et la tâche suivante est exécutée.

### ordonnancement modifiable

Système d'ordonnancement où une tâche de priorité élevée empêchera l'exécution d'une tâche de priorité plus basse dès que la tâche de priorité élevée sera prête à être exécutée, habituellement après une interruption. Pour mettre en oeuvre cet algorithme d'ordonnancement, chaque tâche reçoit un algorithme d'exécution. Généralement, plus la priorité de la tâche est élevée, plus il est important qu'elle soit exécutée. L'ordonnancement modifiable ne peut être mis en oeuvre que dans les systèmes multiprogrammation où les tâches peuvent être suspendues et redémarrées. Le type d'ordonnancement le plus simple et l'ordonnancement circulaire (ou ordonnancement à tour de rôle).

### ordonnancement non modifiable

Sorte d'ordonnancement qui exige qu'une tâche aille jusqu'à son terme une fois qu'elle est commencée. L'ordonnancement doit prévoir les besoins des tâches et s'efforcer de les exécuter d'une manière permettant d'en

limiter les délais. Puisque l'ordonnanceur ne peut arrêter une tâche une fois qu'elle est lancée, les ressources (temps maximum du processeur, mémoire et dispositifs périphériques) nécessaires à celle-ci doivent être précisées par l'utilisateur avant sa mise en oeuvre. Le type le plus simple d'algorithme d'ordonnancement non modifiable est l'ordonnancement FOF0. Les techniques d'ordonnancement non modifiables s'efforcent de prendre les tâches à exécuter en ayant un temps d'exécution moyen minimum. Cependant, une fois qu'une longue tâche est lancée, toutes les autres doivent attendre son achèvement, quelles que soient leurs priorités.

### ordre d'exécution

Séquence des opérations effectuées dans l'exécution d'un algorithme. (ANSI/IEEE)

### organigramme

Représentation graphique d'un problème, de son analyse et de sa résolution à l'aide de symboles représentant des opérations, des données, des liaisons, des appareils ou des organes. (ANSI) En contraste avec schéma fonctionnel.

### organigramme des données

Organigramme de traitement de l'information représentant le cheminement des données intervenant dans la solution d'un problème, en définissant les grandes phases de traitement ainsi que les différents supports d'information employés. (ANSI)

### outil d'essai

Outil logiciel qui produit et insère des compteurs ou d'autres sondes à des endroits stratégiques dans un autre programme pour fournir des statistiques sur l'exécution du programme, notamment pour indiquer dans quelle mesure le code du programme est appliqué. (ANSI/IEEE)

### outil de conception automatisée

Outil logiciel utile pour la synthèse, l'analyse, la modélisation ou la documentation de la conception d'un

logiciel. Il peut s'agir, notamment, de simulateurs, d'aides analytiques, de processeurs de représentation conceptuelle et de générateurs de documentation.  
(ANSI/IEEE)

### outil de vérification automatisée

Catégorie d'outils logiciels utilisés pour évaluer les produits du processus d'élaboration de logiciels. Ces outils contribuent à la vérification de caractéristiques d'exactitude, de complétude, de cohérence, de possibilité de traçage, de possibilité d'essai et de conformité aux normes. Les analyseurs de projets, les systèmes automatisés de vérification, les analyseurs statiques, les analyseurs dynamiques et les logiciels de contrôle des normes de développement en sont quelques exemples. (ANSI/IEEE)

### outil logiciel

Programme machine utilisé pour faciliter l'élaboration, l'essai, l'analyse ou la maintenance d'un autre programme machine ou de sa documentation; mentionnons, par exemple, l'aide à la conception automatisée, le compilateur, l'outil d'essai et l'outil de maintenance. (ANSI/IEEE)

### outils de soutien

Ensemble des procédures et des outils logiciels utilisés comme aides dans l'élaboration du logiciel.  
(DAN 1201)

### pagination

- a. Méthode d'organisation des programmes et de leur mise en mémoire par petits sous-ensembles (ou pages) rendant possible leur exécution dans un volume restreint de mémoire centrale disponible. (ADatP-2)
- b. Division d'une mémoire centrale en blocs physiques de longueur fixe (structures de pages). Programmes et données sont divisés en blocs (pages) de la même taille. Ainsi, une page d'informations peut être chargée dans une structure de page. La longueur des pages est

déterminée par les caractéristiques du matériel plutôt que par la logique du programme.

### pagination à la demande

Algorithme de pagination dans un système de stockage virtuel où les pages ne sont pas chargées en mémoire jusqu'à ce qu'une référence y soit faite.

### paire torsadée blindée (TSP)

Type de câble électronique composé d'une paire de fils isolés torsadés uniformément et recouverts ensuite d'un blindage électrostatique. L'ensemble est recouvert d'une gaine isolante appropriée.

### palier arrière

Partie d'un signal vidéo composite suivant l'impulsion de synchronisation horizontale et s'étendant jusqu'au flanc arrière de l'impulsion de suppression correspondante.

### palier avant

Partie d'un signal vidéo composite située entre le front d'une impulsion de suppression de ligne et le front de synchronisation correspondant.

### paramètre

Variable à laquelle on assigne une valeur constante déterminée pour chaque cas particulier.

### paramètre effectif

Argument ou expression utilisé au moment de l'appel d'un sous-programme pour préciser quelles données ou quels éléments d'un programme doivent être transmis au sous-programme. (ANSI/IEEE) Par contraste avec paramètre formel.

paramètre formel

Variable utilisée dans un sous-programme pour représenter les données ou les éléments d'un programme qui doivent être transmis au sous-programme par un programme d'appel. Synonyme de paramètre fictif. (ANSI/IEEE) En contraste avec paramètre effectif.

partage

Moyen de desservir de nombreux utilisateurs avec une capacité de mémoire limitée. Impose généralement d'avoir recours à la pagination ou à la segmentation.

partage de temps

Technique d'exploitation d'un système informatique qui permet l'imbrication dans le temps de deux ou plusieurs processus dans un processeur. (ISO)

pascal

Langage de programmation structuré évolué baptisé en l'honneur du physicien français Blaise Pascal et utilisé particulièrement pour les applications scientifiques et techniques. Le Pascal est connu pour ses règles strictes. Les structures de données sont basées sur un langage apparenté au Pascal, comme le langage C.

permutation

Action de copier, à des intervalles déterminés par le système d'exploitation, généralement imposés par les politiques d'ordonnancement de l'unité centrale, des travaux de la mémoire principale dans une mémoire auxiliaire, puis de les recopier ultérieurement dans la mémoire principale. Cela permet d'exécuter plus de travaux que la mémoire ne peut en accepter simultanément. C'est une caractéristique des systèmes d'exploitation permettant de transférer les tâches en suspens dans une mémoire secondaire de manière à libérer suffisamment d'espace pour que la tâche suivante sur la liste de réserve puisse être chargée dans la mémoire (si elle n'est pas déjà résidente) et exécutée. La tâche est retransférée dans la mémoire du système lorsque la reprise de son exécution est programmée. Ce mouvement de transfert aller-retour dans une mémoire secondaire peut

se produire de nombreuses fois pour une tâche donnée avant que son exécution ne soit terminée.

### phase d'établissement des exigences

Période du cycle de vie d'un logiciel durant laquelle les exigences d'un produit logiciel, telles que les possibilités fonctionnelles et de performance, sont définies et documentées. (ANSI/IEEE)

### phase de conception

Période de temps dans le cycle de vie d'un logiciel durant laquelle la conception de l'architecture, des éléments du logiciel, des interfaces et des données est élaborée, documentée et vérifiée pour satisfaire aux exigences. (ANSI/IEEE)

### phase de déclin

Période du cycle de vie d'un logiciel durant laquelle prend fin le soutien apporté à un produit logiciel. (ANSI/IEEE)

### phase de mise en oeuvre

Période du cycle de vie du logiciel durant laquelle un produit logiciel est créé à partir de la documentation de conception et est mis au point. (ANSI/IEEE)

### phase de mise en place et de vérification

Période dans le cycle de vie du logiciel durant laquelle un produit logiciel est intégré dans son cadre d'utilisation et y est essayé pour que l'utilisateur s'assure qu'il fonctionne comme prévu. (ANSI/IEEE)

### pile

- a. Liste qui est sollicitée suivant la méthode "dernier entré, premier sorti". (ANSI/IEEE)  
Antonyme de file d'attente.

- b. Zone d'emplacements de mémoire à tores contigus.

### PL/1

Acronyme de "Programming Language 1". Langage de programmation conçu pour combiner les moyens des langages de programmation commerciaux et scientifiques comme le COBOL et le FORTRAN.

### placeur

Algorithme pour affectation de mémoire qui explore la liste libre seulement le temps nécessaire pour trouver un bloc de mémoire assez grand pour l'exécution de la tâche voulue.

### plan d'élaboration d'un logiciel

Plan de projet pour l'élaboration d'un produit logiciel. (ANSI/IEEE) Synonyme de plan d'élaboration d'un programme machine.

### plan d'essai

Document indiquant la méthode qui devrait être utilisée dans les essais projetés. Le plan indique généralement les articles à essayer, les essais à effectuer, les échéanciers des essais, les exigences relatives au personnel et aux rapports, les critères d'évaluation et tous les risques nécessitant un plan de secours. (ANSI/IEEE)

### plan d'implantation

Liste établie par ordinateur qui désigne l'emplacement ou la taille de toutes les parties d'un programme machine résidant en mémoire, ou de certaines d'entre elles, ou des données résidant en mémoire. (ANSI/IEEE)



**plan de projet**

Document de gestion décrivant l'approche qui sera utilisée pour un projet. Le plan décrit généralement le travail à effectuer, les ressources nécessaires, les méthodes à employer, les procédures à suivre relativement à la gestion de la configuration et à l'assurance de la qualité, les échéanciers, l'organisation du projet, etc. (ANSI/IEEE)

**point d'entrée**

Première instruction définie par le programmeur qui est exécutée dans un programme quand il est activé. Il peut y avoir plusieurs points d'entrée pour couvrir différents modes de fonctionnement du programme.

**portabilité**

Facilité avec laquelle un logiciel peut être transféré d'un environnement ou système informatique à un autre. (ANSI/IEEE)

**possibilité d'échange d'informations**

Capacité de deux systèmes ou plus d'échanger des informations et d'utiliser mutuellement ces informations. (ANSI/IEEE)

**possibilité de répétition d'un essai**

Attribut d'un essai indiquant si les mêmes résultats sont obtenus chaque fois qu'un essai est effectué. (ANSI/IEEE)

**possibilité de réutilisation**

Indique dans quelle mesure un module peut être utilisé dans de nombreuses applications. (ANSI/IEEE)

**possibilité de traçage**

Possibilité de tracer ou d'associer chaque fonction entre les divers niveaux de documentation qui définissent

la configuration, l'essai et la conception du système et du logiciel. (RTCA)

### poste de commande et d'affichage

En avionique, poste utilisé comme interface entre un sous-système ou un système donné et un membre de l'équipage. Le poste affiche des informations et est utilisé pour introduire des commandes et des données dans le sous-système ou le système.

### précision

Degré de discrimination utilisé pour la représentation d'une quantité. Par exemple, un nombre à trois chiffres permet de faire la discrimination entre 1.000 possibilités. (ANSI)

### précompilateur

Programme machine qui prétraite un code source, dont une partie ne pourrait pas être acceptée par un compilateur, afin de produire un code équivalent qui pourrait être accepté par le compilateur; par exemple, un préprocesseur convertit un programme en langage FORTRAN structuré en un programme en FORTRAN standard ANSI. (ANSI/IEEE)

### prédicat

Assertion ou proposition logique concernant l'état d'un programme à un point donné, ayant une valeur vraie ou fausse. Pour ce qui a trait à l'exactitude du programme, toutes ces assertions doivent être des axiomes ou être démontrées vraies. (DAN 1153)

### prédiction des erreurs

Enoncé sur le nombre prévu ou la nature des problèmes, des défauts ou des défaillances dans un système logiciel. (ANSI/IEEE)

**premier plan/arrière-plan**

Type de système d'exploitation avec une combinaison de traitements en temps réel et soit de traitements par lots, soit de traitements multitâches. Ce type de système est utilisé pour permettre à des programmes prioritaires de fonctionner en premier plan et de se dérouler avec une priorité élevée alors que les assemblages, modifications d'arrière-plan, etc. se font aussi avec une priorité beaucoup plus basse. Les programmes de premier plan ont toujours la priorité sur le travail d'arrière-plan. Ce type de système d'exploitation a généralement la possibilité de protéger le premier plan de l'arrière-plan et est utilisé dans les applications comme la conduite de processus.

**préprocesseur**

Programme qui effectue un calcul ou une organisation préliminaire. (ISO)

**présentation tête basse (HDD)**

Visualisation d'informations de vol, de navigation, d'attaque ou autres, située dans l'habitacle et qui oblige l'équipage à regarder à l'intérieur de celui-ci.

**présentation tête haute (HUD)**

Visualisation par superposition optique dans le champ de vision normal du pilote de différentes informations relatives au vol, à la navigation, à l'armement, etc. (AAP-6)

**prêt**

Etat dans lequel une tâche donnée est prête à être exécutée.

**preuve d'accomplissement normal**

Dans une preuve d'exactitude, démonstration qu'un programme se terminera selon toutes les conditions d'entrée spécifiées. (ANSI/IEEE)

---

**preuve d'exactitude**

- a. Technique formelle utilisée pour prouver mathématiquement qu'un programme est conforme à ses spécifications. Voir également exactitude partielle, exactitude totale. (ANSI/IEEE)
- b. Preuve d'un programme qui résulte de l'application de cette technique. (ANSI/IEEE)

---

**priorité dynamique**

Forme d'organisation de la priorité d'exécution d'une tâche qui varie en fonction de l'environnement du système. Par exemple, il est courant pour les systèmes en temps partagé de donner initialement aux tâches étendues une faible priorité, mais d'en élever périodiquement le niveau quand la tâche est en attente d'exécution. Ceci permet que les tâches à faible priorité ne soient pas bloquées au moment de l'exécution par des tâches limitées à priorité élevée.

---

**priorité statique**

Forme d'ordonnancement dans laquelle la priorité d'exécution d'une tâche est fixée lorsque la tâche est chargée dans la mémoire. Cette priorité ne change pas pendant l'exécution de la tâche.

---

**procédure**

- a. Partie d'un programme machine à laquelle on donne un nom et qui effectue une tâche précise. Comparer cette notion avec sous-programme, fonction, module. (ANSI/IEEE)
- b. Démarche suivie pour résoudre un problème. (ISO)
- c. Description de la démarche suivie pour résoudre un problème. (ANSI)

---

**procédure d'essai**

Instructions détaillées sur l'établissement, l'exploitation et l'évaluation des résultats d'un essai donné. On réunit souvent des procédures connexes pour

former un document sur les procédures d'essai.  
(ANSI/IEEE)

### procédures pour erreurs

Procédures dans un programme qui sont suivies quand une erreur, comme des données erronées, est détectée par le programme.

### processeur

- a. En matériel, machine de traitement de l'information.
- b. En logiciel, programme de calculateur comportant la compilation, l'assemblage, la traduction et les fonctions connexes pour un langage de programmation donné. (ADatP-2)

### processeur de langage

Programme machine qui exécute des fonctions de traduction, d'interprétation et d'autres tâches requises pour traiter un langage de programmation précis. Par exemple, un compilateur FORTRAN ou un compilateur COBOL. (ISO)

### processus

Séquence systématique d'actions en vue d'obtenir un résultat spécifié. (ADatP-2)

### processus d'élaboration d'un logiciel

Processus suivant lequel les besoins de l'utilisateur sont traduits sous forme de configuration logicielle, elle-même incorporée dans la conception, laquelle est mise en oeuvre sous la forme d'un code, qui est testé, documenté et certifié pour exploitation. (ANSI/IEEE)

**processus séquentiels**

Processus exécutés de telle manière que l'un doit être terminé avant que le suivant ne commence.  
(ANSI/IEEE) En contraste avec opérations exécutées en simultanéité.

**produit de base**

- a. Produit revu et accepté officiellement et qui sert par la suite de base pour les créations additionnelles. Le produit de base peut être modifié uniquement au moyen de procédures officielles de contrôle des changements.  
(ANSI/IEEE)
- b. Document, ou ensemble de documents, d'identification de configuration, dûment créé et appliqué à un moment précis durant un cycle de vie d'un élément de configuration. Les documents de base, ainsi que les changements approuvés s'y rapportant, constituent l'identification de la configuration courante. Il existe trois documents de base pour la gestion de la configuration :
  - (1) Document de base fonctionnel : configuration fonctionnelle approuvée initialement.
  - (2) Document de base attribué : configuration approuvée initialement.
  - (3) Document de base du produit : identification de la configuration du produit du document approuvée initialement ou conditionnellement. (MIL-STD-973)

**produit logiciel**

Entité logicielle qui doit être livrée à un utilisateur. (ANSI/IEEE)

**progiciel d'essai et de vérification**

Progiciel d'essai et de vérification automatisés composé de mots de commande d'exécution d'essai, de

données d'essai et de scénarios de situations servant à lancer et à tester un programme objet. (DAN 1201)

### programmable

Qualifie un dispositif, un composant, un sous-système ou un système conçu de telle façon que sa fonction et/ou ses caractéristiques physiques peuvent être modifiées.

### programmation

Ensemble des travaux de conception, d'écriture et d'essai d'un programme. (ADatP-2)

### programmation collective

Approche d'élaboration de logiciel fondée sur le concept de la responsabilité d'équipe pour l'élaboration d'un programme. Le but de la programmation collective est d'empêcher le programmeur de s'identifier trop étroitement à son produit, de façon à ne pas fausser l'évaluation objective. (ANSI/IEEE)

### programmation modulaire

Technique d'élaboration d'un système ou d'un programme comme un ensemble de modules. (ANSI/IEEE)

### programmation orientée vers l'objet

Technique de programmation basée sur la notion d'objet qui permet la mise en oeuvre directe des conceptions orientées vers l'objet. Exige l'utilisation d'un langage orienté vers l'objet.

### programmation structurée

Technique d'élaboration de logiciels bien définie qui incorpore la conception et la mise en oeuvre descendantes et l'utilisation rigoureuse de structures de contrôle de programme structuré. (ANSI/IEEE)

programme de construction

Version opérationnelle d'un produit logiciel qui comprend un sous-ensemble déterminé des capacités qui seront incorporées dans le produit final. (ANSI/IEEE)

programme de gestion (logiciel) ou  
programme de commande (matériel).

programme de gestion

- a. Module de logiciel d'un système qui commande directement le transfert de données vers et depuis les entrées/ sorties périphériques.
- b. Module de logiciel utilisé dans les essais ascendants pour préparer les modules au test.
- c. Programme qui teste un système ou un élément de système en simulant l'activité d'un élément de niveau supérieur. (ANSI/IEEE)
- d. (Handler = Driver.)

programme de gestion d'essai

Programme de gestion qui appelle l'article faisant l'objet d'un essai et qui peut fournir les paramètres d'essai et exposer les résultats de l'essai. (ANSI/IEEE)

programme de projection

Fonction d'un compilateur ou d'un assembleur qui produit un plan d'implantation. (ANSI/IEEE)

programme de traçage

Outil logiciel utilisé pour tracer. (ANSI/IEEE)

programme de vol opérationnel (OFP)

Programme chargé dans un ordinateur d'avionique qui effectue les calculs, fonctions et algorithmes



nécessaires aux sous-systèmes pour l'accomplissement d'une mission particulière.

#### programme directeur

Partie de programme qui contrôle l'exécution d'autres programmes. Synonyme de programme superviseur. (ADatP-2)

#### programme machine

Programme constitué par une suite d'instructions et d'énoncés mis sous une forme acceptable par un ordinateur. (ADatP-2)

#### programme objet

Programme entièrement compilé ou assemblé qui est prêt à être chargé dans l'ordinateur. En contraste avec programme source. (ADatP-2)

#### programme récurrent

Programme qui peut être utilisé comme son propre sous-programme, qui peut s'appeler lui-même ou être appelé par un autre sous-programme, qu'il a lui-même appelé. L'utilisation d'un programme récurrent exige habituellement la tenue de registres sur l'état des utilisations non terminées du programme, notamment dans une liste inversée. (ANSI/IEEE)

#### programme structuré

Programme constitué d'un ensemble fondamental de structures de contrôle, chacune ayant un point d'entrée et une sortie. L'ensemble de structures de contrôle comprend généralement : une suite de deux ou plusieurs instructions, la sélection conditionnelle d'une instruction ou d'une série d'instructions parmi deux ou plusieurs et la répétition d'une instruction ou d'une suite d'instructions. (ANSI/IEEE)

**programme superviseur**

Programme, faisant habituellement partie d'un système d'exploitation, qui contrôle l'exécution d'autres programmes et qui règle le déroulement des opérations dans un système de traitement de l'information. (ISO)  
Synonyme de programme directeur et de superviseur.

**programme-source**

Programme écrit en langage source. (ADatP-2)

**programmer**

- a. Concevoir, écrire et essayer un programme.  
(ADatP-2)
- b. D'une manière générale, écrire un programme.

**programmeur**

Personne qui conçoit, écrit et essaie les programmes d'ordinateur. (ADatP-2)

**programmeur de système**

Programmeur qui rédige des routines générales conçues pour être utilisées dans toutes les installations comportant un ordinateur donné, telles que compilateurs, routines utilitaires, programmes de tri, etc.

**programmeur-assistant**

Chef adjoint d'une équipe de programmation structurée; programmeur de niveau supérieur dont les responsabilités comportent un apport important dans l'élaboration des logiciels par l'équipe, l'aide apportée au chef programmeur dans l'examen du travail des autres membres de l'équipe et le remplacement de celui-ci au besoin. De plus, le programmeur-assistant doit avoir une compréhension technique globale des logiciels créés.  
(ANSI/IEEE)

---

**PROM**

ROM programmable. Semi-conducteur où l'information stockée peut être modifiée, mais pas pendant une opération. La mémoire peut être effacée par une application prolongée d'ultra-violets et réécrite avec un dispositif électronique de puissance relativement élevée.

---

**protection**

Dispositif visant à restreindre l'accès à l'ensemble, ou à une partie, d'un système informatique ou l'utilisation de l'ensemble ou d'une partie de ce système. (ISO)

---

**protection de programme**

Application de contrôles internes ou externes pour empêcher tout accès non autorisé à un programme machine ou toute modification non autorisée d'un tel programme. (ANSI/IEEE)

---

**protocole**

- a. Ensemble de conventions ou de règles qui régissent les interactions des processus ou des applications dans un système ou un réseau informatique. (ANSI/IEEE)
- b. Ensemble de règles qui régissent l'utilisation d'unités fonctionnelles en vue d'établir la communication. (ISO)

---

**pseudo-code**

- a. Combinaison du langage de programmation et du langage naturel pour la conception d'un programme machine. (ANSI/IEEE)
- b. Méthode de documentation simple représentant la logique de programmation structurée.

---

**pseudonyme**

- a. Synonyme d'article. (ANSI/IEEE)

- b. Label de remplacement. Par exemple, on peut utiliser un label et un ou plusieurs pseudonymes pour désigner le même élément d'information ou un programme machine. (ANSI)

#### puce

Elément de matériau semi-conducteur qui accomplit une (ou des) fonction(s) spécifique(s).

#### qualité

Ensemble des caractéristiques d'un produit ou d'un service qui permettent de répondre à certains besoins. (ANSI/ASQCA3-1978)

#### qualité d'un logiciel

Ensemble des caractéristiques d'un produit logiciel influant sur la satisfaction de certains besoins; la conformité aux spécifications, par exemple. (ANSI/IEEE)

#### rapport d'essai

Document décrivant le déroulement de l'essai et ses résultats pour un système ou un élément de système donné. (ANSI/IEEE)

#### rapport sur l'état

Documentation sur l'état actuel approuvé du système incluant un dossier historique de tous les changements approuvés. (RTCA)

#### rapport sur un problème logiciel (SPR)

Rapport sur un problème constaté, à savoir que le logiciel n'est pas conforme aux exigences définies ou ne remplit pas les fonctions prévues. (RTCA)

**reconfiguration**

Modification des composants ou de l'organisation des composants d'un système

**redondance**

Inclusion d'éléments de système en double pour améliorer la fiabilité opérationnelle en assurant le fonctionnement continu en cas de défaillance d'un élément primaire. (ANSI/IEEE)

**registre**

- a. Dispositif capable de stocker un nombre spécifié de données comme un mot. (ADatP-2)
- b. Dispositif pour le stockage temporaire d'un ou de plusieurs mots afin de faciliter des opérations arithmétiques, logiques ou de transfert. Exemples : système de prise en charge, adresse, index, instruction et registres M-Q.
- c. Matériel pour le stockage d'un ou de plusieurs mots de calculateur. (Computer dictionary)

**registre d'index**

Registre dont le contenu peut être ajouté à une adresse d'opérande ou retiré de celle-ci, avant ou pendant l'exécution d'une instruction.

**registre de base**

Emplacement de mémoire rapide qui contient l'adresse du premier mot d'information ou instruction d'un segment. Toutes les adresses dans le segment sont désignées au début du segment, et, lors de l'exécution, l'adresse physique de chaque référence de mémoire est calculée en ajoutant simplement le contenu du registre de base aux adresses relatives figurant dans le segment. Ainsi la translation est aisément effectuée en transférant simplement un segment en mémoire et en mettant la nouvelle adresse dans le registre de base.

**registre de longueur**

Emplacement de mémoire rapide utilisé avec un registre de base dans les mises en oeuvre de sécurité de tâches. Le registre de base contient l'adresse physique inférieure du segment; le contenu du registre de base plus celui du registre de position terminale constituent l'adresse physique supérieure du segment.

**reliure statique**

Reiure effectuée avant l'exécution d'un programme ne faisant pas l'objet de modifications pendant l'exécution. (ANSI/IEEE) En contrast avec reliure dynamique.

**rendez-vous**

Interaction qui se produit entre deux tâches parallèles lorsqu'une tâche appelle une valeur de l'autre tâche et qu'une instruction d'acceptation est exécutée par celle-ci au profit de la première. (ANSI/IEEE)

**renvoi**

Référence dans une partie d'un programme à l'identificateur d'une autre partie (c'est-à-dire quelque chose que l'on trouve ailleurs). (ANSI/IEEE)

**répartiteur**

Autre terme pour échéancier.

**réponse à une commande**

Mode de fonctionnement d'un système de bus de données tel que les terminaux éloignés ne reçoivent et transmettent des données que lorsque la commande de bus le leur demande.

**reprise sur incident**

Retour d'un système à un état de fonctionnement fiable après un incident. (ANSI/IEEE)

**réseau**

Groupe de noeuds interconnectés ou interreliés.  
(ANSI/IEEE)

**réseau d'ordinateurs**

Système interconnecté de plusieurs ordinateurs.  
(ADatP-2)

**réseau de Pétri**

Modèle abstrait et formel d'un flux informationnel indiquant les propriétés statiques et dynamiques d'un système. Un réseau de Pétri est habituellement présenté sous la forme d'un graphe ayant deux types de noeuds (appelés places et transitions) reliés par des arcs, et des marqueurs (appelés jetons) indiquant les propriétés dynamiques. (ANSI/IEEE)

**responsable de la bibliothèque des logiciels**

Personne chargée de la mise sur pied, de la gestion et de l'actualisation d'une bibliothèque de logiciels.  
(ANSI/IEEE)

**ressource**

Moyens d'un système informatique que le système d'exploitation peut utiliser ou affecter à des tâches pour en permettre l'exécution. Des moyens tels que mémoires, espaces disques, imprimantes, terminaux et mémoire centrale sont affectés aux tâches en fonction de leur disponibilité.

**ressources informatiques**

Ensemble du matériel informatique disponible et utile, des programmes, de la documentation, des services, des fournitures et du personnel. Au niveau de l'attribution, les "ressources" sont plus précisément orientées vers la mémoire disponible et vers le temps d'exécution admissible pour réaliser un travail ou un projet précis. (GLOS-1)

---

**restauration**

Programme utilitaire du système qui recharge un fichier dans la mémoire secondaire à partir d'un volume d'archivage tel qu'une bande magnétique.

---

**restructuration dynamique**

- a. Action de changer les éléments ou la structure d'un logiciel pendant que le système fonctionne. (ANSI/IEEE)
- b. Action de restructurer une base de données ou une structure de données durant l'exécution du programme. (ANSI/IEEE)

---

**résumé analytique**

- a. Vue d'un problème qui permet d'extraire des informations essentielles en rapport avec un but précis et d'ignorer le reste d'entre elles. (ANSI/IEEE)
- b. Processus d'élaboration d'un résumé analytique. (ANSI/IEEE)

---

**résumé de programme machine**

Brève description d'un programme machine fournissant suffisamment d'informations aux utilisateurs éventuels pour leur permettre de déterminer l'opportunité du programme machine en fonction de leurs besoins et de leurs ressources. (ANSI/IEEE)

---

**réveil**

Action consistant à mettre une tâche en état d'être exécutée après une période de suspension.

---

**revue de projet**

Revue dans laquelle un concepteur ou un programmeur dirige un ou plusieurs membres de l'équipe de



développement dans l'examen d'un segment du projet ou du code qu'il a écrit, tandis que les autres membres posent des questions et font des commentaires sur la technique, le style, les erreurs possibles, la violation des normes d'élaboration et d'autres problèmes. (ANSI/IEEE) En contraste avec inspection.

### revue de projet structurée

Nom générique donné à un ensemble de techniques (à savoir, la validation de projet, la vérification de projet et la vérification de code), chacune ayant des objectifs différents et se produisant à des moments différents du cycle d'élaboration du logiciel. (DAN 154)

### robustesse

Indique dans quelle mesure un logiciel peut continuer à fonctionner correctement malgré l'introduction de données invalides. (ANSI/IEEE)

### routine

Programme machine, ou partie de programme machine, qui peut avoir un emploi général ou répété. (ADatP-2)

### routine de traçage

Aide au diagnostic qui contrôle le déroulement d'un programme en appelant l'attention sur le contenu de certaines zones, en signalant les branchements, etc.

### rythmeur

Matériel fournissant, à certains moments de la journée, des interruptions au système d'exploitation.

### schéma fonctionnel

Schéma représentant un ensemble, un appareil ou un ordinateur par ses principales parties, à l'aide de figures géométriques annotées de manière à indiquer les fonctions remplies et les relations fonctionnelles entre ces parties. Différent d'organigramme. (ANSI)

---

**section critique**

L'un des deux segments de code à exécution mutuellement exclusive appelés tous deux sections critiques. Les exécutions des segments de code doivent être mutuellement exclusives si les segments doivent tous les deux utiliser une même ressource informatique ou une même donnée élémentaire. (ANSI/IEEE)

---

**sécurité**

Protection du matériel et du logiciel contre tout accès ou toute utilisation, modification, destruction ou divulgation accidentels ou malveillants. S'applique également au personnel, à la transmission des données et à la protection physique des installations informatiques. (ANSI/IEEE)

---

**sécurité multiniveau**

Mode d'exploitation qui permet de mémoriser et de traiter concurremment des données à divers niveaux de sécurité dans un système informatique lorsque quelques utilisateurs au moins n'ont pas besoin d'avoir accès à toutes les données du système ou ne sont pas autorisés à le faire. (ANSI/IEEE)

---

**segment**

- a. Partie autonome d'un programme machine qui peut être exécutée sans qu'il soit nécessaire de conserver tout le programme machine dans la mémoire interne à quelque moment que ce soit. (ISO) Voir également élément, module, sous-programme.
- b. Suite d'instructions de programme machine entre deux points de branchement consécutifs. (ANSI/IEEE) Voir également analyse des branches.

---

**segment de recouvrement**

Technique utilisée pour exécuter des programmes plus grands que la capacité de mémoire disponible dans des systèmes sans possibilités de pagination ou de

segmentation. Pour utiliser cette méthode, un programme doit être divisé manuellement en un certain nombre de groupes de modules de logiciels s'excluant mutuellement. Tout logiciel commun à plus d'un segment de base doit rester en permanence dans la mémoire du système; les autres segments sont chargés et exécutés un par un dans une zone de la mémoire désignée. Ce chargement est normalement effectué par le système d'exploitation à la demande du segment de recouvrement de base.

### segmentation

Découpage de l'espace d'une adresse de programme en blocs (segments) ou en modules de taille variable. La segmentation impose de morceler un programme en unités logiquement séparables. Par exemple, un segment peut se composer d'une sous-routine, un autre d'une zone de données. Le déplacement des segments trop gros vers la mémoire principale (mémoire centrale) ou à partir de cette dernière prend trop de temps. Les segments trop petits imposent un trop grand nombre de déplacements.

### segmenter

Diviser un programme machine en segments. (ISO)

### sémantique

Relations entre les symboles et leurs significations. (ANSI)

### sémaphore

Variable partagée utilisée pour synchroniser le fonctionnement simultané de plusieurs processeurs en indiquant si une action est terminée ou si un événement s'est produit. (ANSI/IEEE)

### service libre

Service de système d'exploitation qui retourne les blocs de mémoire affectés à la gestion de mémoire pour réutilisation.

**signal discret**

Signal qui ne nécessite que deux états pour représenter toute fonction ou information.

**simulateur**

Dispositif, système ou programme machine représentant certains aspects du comportement d'un système physique ou abstrait. (ANSI)

**simulation**

Représentation de certains aspects du comportement d'un système physique ou abstrait par le comportement d'un autre système, par exemple, représentation de phénomènes physiques au moyen d'opérations effectuées par un ordinateur ou représentation d'opérations d'un ordinateur par celles d'un autre ordinateur. (ANSI)

**sondage de programme**

- a. Sondes, telles que des instructions ou des assertions, insérées dans un programme machine pour faciliter la surveillance de l'exécution et le contrôle de l'exactitude et des ressources ou d'autres activités. (ANSI/IEEE)
- b. Opération consistant à préparer et à insérer des sondes dans un programme machine. (ANSI/IEEE)

**sortie**

Toute instruction d'un programme machine ou d'un sous-programme dont l'exécution, une fois terminée, empêche ce programme machine de continuer à exercer un contrôle. (ISO)

**sous-programme**

Unité de programme qui peut être appelée par une ou plusieurs unités de programme. (ANSI/IEEE) La procédure, la fonction et la sous-routine en sont des exemples.

**sous-routine**

Ensemble ordonné d'instructions pouvant être utilisé dans un ou plusieurs programmes machines et à un ou plusieurs points d'un programme machine. (ISO)

**sous-système**

Groupe d'ensembles ou d'éléments, ou les deux combinés, destiné à exécuter une seule fonction. (ANSI N45.2.10-1973)

**spécification**

Document qui prescrit, d'une manière complète, précise et vérifiable, les exigences, la conception, le comportement ou d'autres caractéristiques d'un système ou d'un élément de système. (ANSI/IEEE) Voir également spécification de conception, spécification formelle, spécification fonctionnelle, spécification d'interface, spécification de performance et spécification des exigences.

**spécification d'interface**

Spécification qui expose les exigences d'interface d'un système ou d'un élément de système. (ANSI/IEEE)

**spécification de conception**

Spécification qui documente la conception d'un système ou d'un élément de système; par exemple, un élément de la configuration d'un logiciel. Comprend généralement les algorithmes du système ou de l'élément, la logique de commande, les structures de données, les renseignements sur l'utilisation d'un ensemble de données, les formats d'entrée/ sortie et les descriptions des interfaces. (ANSI/IEEE) Voir également spécification des exigences.

---

**spécification de performance**

Spécification qui établit les exigences de performance d'un système ou d'un élément de système. (ANSI/IEEE)

---

**spécification de programme**

Spécification d'un programme machine. Voir spécification du projet, spécification fonctionnelle, spécification de performance, spécification des exigences. (ANSI/IEEE)

---

**spécification des exigences**

Spécification qui expose les exigences d'un système ou d'un élément de système, par exemple, un élément de configuration de logiciel. Cela comprend généralement les exigences fonctionnelles, les exigences de performance, les exigences d'interface, les exigences du projet et les normes d'élaboration. (ANSI/IEEE)

---

**spécification fonctionnelle**

Spécification qui définit les fonctions qui doivent être effectuées par un système ou par un élément de système. (ANSI/IEEE)

---

**spécification formelle**

- a. Spécification écrite et approuvée conformément à des normes établies. (ANSI/IEEE)
- b. Dans une preuve d'exactitude, description dans un langage formel du comportement visible externe d'un système ou d'un élément de système. (ANSI/IEEE)

---

**stabilité**

- a. Aptitude à continuer sans modification malgré les événements gênants ou perturbateurs. (ANSI/IEEE)

- b. Aptitude à revenir à l'état original après que se sont produits des événements gênants ou perturbateurs. (ANSI/IEEE)

### structure de contrôle

Structure qui détermine l'ordre d'exécution dans un programme machine. (ANSI/IEEE)

### structure de contrôle conditionnel

Structure de contrôle de la programmation qui permet un ordre d'exécution des opérations donné dans un programme si des conditions déterminées telles que si ... alors ... autrement ... ont été remplies. (ANSI/IEEE)

### structure de données

Représentation formalisée des relations de rangement et d'accessibilité des données élémentaires, ne tenant pas compte de leur configuration de stockage réelle. (ANSI/IEEE)

### support

Dispositif ou matériel qui porte le logiciel sous forme d'objet ou de source; en voici des exemples : liste de programme, bande magnétique, disque magnétique, cartouche, cassette, bande perforée, carte perforée, etc. (RTCA)

### symbole mnémonique

Symbole dont la forme est conçue pour aider la mémoire humaine. Exemple : "ad", abréviation pour "addition". (ANSI)

### synchro

L'un quelconque des dispositifs utilisées pour la transmission et la réception par fil d'une position angulaire ou d'un mouvement angulaire (exemple : transmetteur synchro ou récepteur synchro).

**synchrone**

- a. Qui se produit avec un rapport temps régulier ou prévisible.
- b. Réalisation d'une suite d'opérations sous le contrôle d'un cycle de signaux générés par une horloge et espacés régulièrement.

**synchronisation**

Réglage d'un mécanisme en vue d'effectuer des opérations ou des procédures suivant un ordre temporel ou séquentiel. (DAN 1153)

**syntaxe**

- a. Relation entre des caractères ou des groupes de caractères indépendamment de leur signification ou de la façon dont ils sont interprétés et utilisés. (ISO)
- b. Structure des expressions dans un langage. (ANSI)
- c. Règles régissant la structure d'un langage. (ANSI)

**syntaxique**

Qui concerne les règles régissant les relations des mots dans les structures de langage.

**synthèse de programme**

Utilisation d'outils logiciels pour aider à la transformation d'une spécification de programme en un programme qui réalise cette spécification. (ANSI/IEEE)



---

**système**

- a. Réunion de méthodes, procédures ou techniques organisées de façon à coordonner leurs effets en traitement de l'information. (ADatP-2)
- b. Ensemble organisé de personnes, de machines et de méthodes nécessaire pour remplir un ensemble de fonctions. (ADatP-2)

---

**système automatisé de vérification**

Outil logiciel qui accepte, comme entrée, un programme machine et une représentation de ses caractéristiques et qui produit, éventuellement avec l'aide d'un utilisateur, une preuve d'exactitude ou d'inexactitude du programme. (ANSI/IEEE)

---

**système d'exploitation (SE)**

- a. Logiciel destiné à commander l'exécution des programmes d'un ordinateur et pouvant assurer l'enchaînement des travaux, la mise au point des programmes, la commande des entrées et sorties, la comptabilité d'exploitation, la compilation, l'attribution de mémoire, la gestion des données et tous services associés. (ADatP-2)
- b. Logiciel qui commande l'exécution des programmes. Un système d'exploitation peut fournir des services comme l'ordonnancement de l'attribution des ressources, la commande des entrées et sorties et la gestion des données. Bien que le système d'exploitation concerne principalement le logiciel, des mises en oeuvre partielles ou complètes de matériel sont possibles. (ISO)
- c. Un système d'exploitation fournit un soutien en un point unique plutôt que de forcer chaque programme à s'intéresser au contrôle du matériel. Voir "system software" = logiciel de système. (ANSI/IEEE)

---

**système d'exploitation à utilisateurs multiples**

Système d'exploitation qui permet à plusieurs utilisateurs d'exploiter le système informatique d'une façon telle qu'il semble que chaque utilisateur a le contrôle exclusif de la machine. Ceci est fait en laissant chaque utilisateur désigné exécuter son programme pendant une période de temps fixée. Ces systèmes sont connus comme des systèmes à temps partagé (TSS).

#### système de commande et de contrôle

Logiciel d'application utilisé soit pour générer des instructions soit pour les transmettre à partir du centre de contrôle.

#### système de désynchronisation des entrées-sorties

Le terme anglais est l'abréviation de l'expression "Simultaneous Peripheral Operation On Line", c'est-à-dire "fonctionnement simultané des périphériques en ligne".

#### système dernier entré-premier sorti (LIFO)

Structure de données dans laquelle les données sont extraites séquentiellement et dans laquelle on accède en premier lieu à l'entrée la plus récente.

#### système expert

Programme qui a une large base de connaissances dans un domaine limité et utilise un raisonnement d'inférences complexe pour effectuer des tâches qu'un homme expert pourrait effectuer.

#### système informatique

Unité fonctionnelle constituée d'un ou plusieurs ordinateurs et des logiciels connexes, qui utilise une mémoire commune pour un programme entier ou une partie de celui-ci de même que pour l'ensemble des données nécessaires à l'exécution du programme, ou une partie de celles-ci. Le système informatique exécute des programmes écrits ou indiqués par l'utilisateur, manipule des données indiquées par l'utilisateur, incluant des opérations arithmétiques et logiques, et peut exécuter des programmes qui se modifient eux-mêmes durant leur exécution. Un système informatique peut être une unité

autonome ou peut comporter plusieurs unités interconnectées. (ISO) Synonyme de système de traitement automatique de l'information et système de calcul.

### système informatique intégré

Système informatique intégré dans un plus gros système dont le but premier n'a pas trait aux calculs. Il peut s'agir, par exemple, d'un système informatique dans une arme, un avion, un système de commande et de contrôle ou un réseau de transport rapide. (ANSI/IEEE)

### système ouvert

Un système qui implémente suffisamment de spécifications ouvertes en matière d'interfaces, de services et de grilles de saisie/formats de soutien pour permettre à des composantes bien conçues soit d'être utilisées sans trop de changements dans de nombreux réseaux ou avec d'autres composantes dans des réseaux locaux ou éloignés, soit d'assurer l'interaction avec les utilisateurs de manière à faciliter la portabilité du système. Un système ouvert se caractérise par :

- a. des interfaces et des protocoles bien définis, non exclusifs et utilisés à grande échelle;
- b. le respect de normes développées et adoptées par des organismes de normalisation reconnus par l'industrie;
- c. la définition de tous les aspects de ses interfaces afin de permettre d'ajouter des réseaux ou d'augmenter la puissance des réseaux existants, de manière à offrir une vaste gamme d'applications;
- d. des clauses explicites qui permettent d'étendre le système ou de le mettre à niveau en y ajoutant des composantes ou en y installant des composantes plus performantes, mais en veillant à ce que ces opérations aient le moins d'impact possible sur le système.

(IEEE POSIX 1003.0/D15 tel que modifié selon le groupe de travail - systèmes interarmées ouverts)

---

**système premier entré, premier sorti (FIFO)**

Structure de données dans laquelle on ne peut accéder qu'à l'entrée la moins récente.

---

**table**

- a. Groupe de données dont chaque article est identifiable sans ambiguïté par un label, par sa position par rapport aux autres articles ou par un autre moyen. (ANSI)
- b. Ensemble de données organisé de manière que chaque élément puisse être identifié sans ambiguïté par une clé se trouvant dans l'élément ou par la position de l'élément dans la table.

---

**table de décision**

- a. Représentation sous forme de tableau d'une série d'essais à appliquer aux données. Les tables de décisions ne sont pas seulement un substitut pratique aux organigrammes dans de nombreux cas, mais peuvent aussi être transformées en instructions de langage de programmation par des programmes de préprocesseur spéciaux.
- b. Tableau présentant toutes les éventualités à envisager dans la description d'un problème et les démarches à suivre dans les différents cas. Les tables de décision sont parfois utilisées au lieu d'organigrammes pour les descriptions de problèmes et la documentation. (ANSI)

---

**table de Karnaugh**

Représentation symbolique des valeurs prises par une fonction logique à l'aide d'un tableau rectangulaire subdivisé lui-même en cases, de manière que chacun des domaines communs à plusieurs de ces cases représente une combinaison unique des valeurs des variables et que toute combinaison possible de ces valeurs soit représentée dans le tableau.

**tâche**

Module de logiciel dans lequel le code est exécuté de manière séquentielle.

**tampon**

- a. Programme ou mémoire utilisés pour compenser une différence de débit des informations ou un décalage de l'arrivée des signaux, lorsque les informations sont transférées d'un organe à un autre. (ADatP-2)
- b. En avionique, un circuit isolant utilisé pour que le circuit d'excitation ne soit pas influencé par le circuit excité.
- c. Terme se rapportant à la fois au logiciel et au matériel. Emplacement intermédiaire où les données, e.g. sous forme d'enregistrements, sont stockées, jusqu'à ce qu'elles soient acceptées par une autre unité (comme une unité centrale) ou par une procédure dans un programme.

**taux de défaillance**

Rapport entre le nombre de défaillances et une certaine unité de mesure. Il peut, par exemple, s'agir de défaillances par unité de temps, de défaillances par nombre de transactions ou de défaillances par nombre de phases de traitement. (ANSI/IEEE)

**temps d'accès**

- a. Délai compris entre le moment où une information rangée en mémoire est demandée et le moment où cette fourniture commence.
- b. Délai compris entre le moment où est demandé le rangement d'une information en mémoire et le moment où ce rangement commence.

**temps d'exécution**

Temps nécessaire au traitement d'une instruction, d'une routine ou d'un programme. (ADatP-2)

**temps de réponse**

Délai qui s'écoule entre l'introduction d'une commande et la fin de son exécution.

**temps de rotation**

Intervalle entre le moment où un programme/travail est pris en charge et le moment où il est terminé.

**temps système**

Temps de traitement demandé par un système d'exploitation pour effectuer des tâches de servitude comme la pagination, les permutations et l'ordonnancement. Ce temps est généralement exprimé en pourcentage du temps total disponible. Si le temps système est de 20 %, les programmes de l'utilisateur peuvent utiliser 48 secondes (80 %) de chaque minute du temps d'exécution du processeur.

**terminal**

- a. Dans un système ou un réseau de transmissions, point auquel les données peuvent soit entrer, soit sortir. (ADatP-2)
- b. Dans un système de bus de données, module électronique nécessaire à la connexion du bus de données et du sous-système. (Un terminal peut être une unité remplaçable en piste (LRU) ou intégrée aux éléments du sous-système.). (Voir également appareil terminal.).

**test d'aptitude**

Test conçu pour savoir si un candidat a l'aptitude de base pour un emploi particulier. Dans le cas de la programmation, le test consistera habituellement en problèmes logiques et sémantiques.

**test de performance**

Test officiel, effectué habituellement par un réalisateur à l'intention du client, pour démontrer que le logiciel est conforme aux exigences spécifiées. (ANSI/IEEE)

**test de régression**

Revérification sélective pour déceler les défauts qui se seraient introduits durant la modification d'un système ou d'un élément de système, pour vérifier que les modifications n'ont pas causé d'effets défavorables, ou encore pour vérifier qu'un système ou un élément de système modifié demeure conforme aux exigences spécifiées. (ANSI/IEEE)

**test intégré (BIT)**

Méthode de test mettant en oeuvre un équipement de test intégré, un matériel ou un logiciel à test automatique afin de tester tout ou partie d'une unité.

**testabilité**

Indique dans quelle mesure le logiciel facilite l'établissement de critères d'essai et l'évaluation du logiciel en fonction de ces critères. (ANSI/IEEE)

**tête**

Partie d'un dispositif d'enregistrement magnétique qui lit la configuration binaire des champs magnétiques, sur une bande, un disque ou un tambour, par exemple, ou bien qui enregistre ces champs. La plupart des têtes peuvent lire et écrire (lire et enregistrer).

**théorie de la durée d'exécution**

Théorie qui utilise un temps d'exécution cumulatif comme base d'évaluation de la fiabilité d'un logiciel. (ANSI/IEEE)

**tolérance**

Aptitude d'un système à assurer la continuité de l'exploitation dans diverses conditions anormales.  
(ANSI/IEEE)

**trace**

- a. Enregistrement de l'exécution d'un programme machine; il indique l'ordre dans lequel les instructions ont été exécutées. (ANSI)
- b. Enregistrement de toutes les catégories d'instructions ou d'événements ou de certaines d'entre elles, qui apparaissent durant l'exécution d'un programme machine. (ANSI/IEEE)

**tracer**

Produire une trace.

**traducteur**

Programme qui sert à la traduction d'une suite d'instructions dans un langage en une suite d'instructions équivalentes dans un autre langage.  
(ANSI/IEEE)

**traitement de liste**

Méthode de traitement des données présentées sous forme de listes. En traitement de liste on emploie les listes chaînées de façon que l'ordre des articles puisse être changé sans en modifier l'emplacement physique.  
(ISO)

**traitement de signal**

Manipulation de signaux numériques ou analogiques en vue d'obtenir l'effet souhaité.

**traitement par chaînes**

Cela comprend les éléments qui effectuent des opérations sur des listes de caractères. (SEL)



**traitement réduit**

Résultat de la conception d'un système permettant de garder des capacités de traitement, bien que diminuées, face à la défaillance progressive de composants.

**tranche de temps**

Durée - normalement 1 à 50 millisecondes - allouée à chaque tâche dans un système en temps partagé.

**translation**

Processus consistant à changer un programme de place à l'intérieur de la mémoire du système. La translation peut être statique ou dynamique. Il y a translation statique lorsque toutes les adresses de programme sont attribuées et fixées avant que le programme ne soit chargé en mémoire. La translation dynamique est une technique qui exige que toutes les adresses de programme soient relatives à une adresse de base, ce qui permet de traduire les programmes à n'importe quel moment en déplaçant le code du programme et en changeant l'adresse de base.

**travail**

Ensemble de tâches, groupées et exécutées ensemble pour effectuer une fonction particulière.

**troncature**

Suppression de chiffres au début ou à la fin d'un nombre, par exemple, parce qu'il est trop long pour rentrer dans l'emplacement de mémoire qui lui est affecté.

**type de données**

Catégorie de données caractérisée par les membres qui en font partie et par les opérations qui peuvent leur être appliquées; par exemple, un nombre entier, un nombre réel, une entité logique. (ANSI/IEEE)

**unité centrale (UC)**

- a. Partie de l'ordinateur qui dirige l'interprétation et l'exécution des instructions. (RTCA)
- b. Coeur d'une installation informatique consistant en unités logiques, arithmétiques et de commande, incorporées dans un circuit électronique.

**unité de calcul**

Unité d'un système informatique qui contient les circuits effectuant les opérations arithmétiques.

**unité fonctionnelle**

Entité de matériel, de logiciel, ou les deux, capable d'atteindre un but précis. (ISO)

**unité remplaçable sur place (LRU)**

Toute unité qui peut être remplacée par le service de maintenance sur piste.

**usine à logiciels**

Ensemble formé d'outils, de méthodologies et d'une base commune de données qui fournit une méthode basée sur des procédures permettant de réaliser avec succès des projets de logiciels. (DAN 207)

**utilisation**

Fraction du temps pendant laquelle une ressource est utilisée.

**validation**

Evaluation d'un logiciel à la fin de son élaboration visant à s'assurer qu'il est conforme aux exigences y afférentes. (ANSI/IEEE)

**validité de l'essai**

Indique dans quelle mesure un essai atteint l'objectif spécifié. (ANSI/IEEE)

**variable**

En programmation, caractère ou groupe de caractères qui correspond à une valeur et, dans l'exécution d'un programme machine, qui correspond à une adresse. (ANSI)

**vérification**

- a. Opération qui consiste à déterminer si les produits d'une phase donnée du cycle d'élaboration d'un logiciel satisfont aux exigences établies durant la phase antérieure. (ANSI/IEEE)
- b. Opération qui consiste à revoir, à inspecter, à essayer, à contrôler, à analyser ou à établir de quelque autre façon, à l'aide de documents, si les articles, les processus, les services ou les documents sont conformes aux exigences spécifiées. (ANSI/ASQC A3-1978)
- c. Révision indépendante visant à évaluer dans quelle mesure on a satisfait aux exigences du logiciel, aux spécifications, aux produits de base, aux normes, aux procédures, aux instructions, aux codes et aux conditions contractuelles et d'autorisation. (ANSI/IEEE)

**vérification de code**

Examen indépendant du code source effectué par une personne, une équipe ou un outil pour vérifier sa conformité avec la documentation sur la conception de logiciel et avec les normes de programmation. L'exactitude et l'efficacité peuvent également être évaluées. (ANSI/IEEE)

**vérification de la configuration**

Action de vérifier si tous les éléments de la configuration requis ont été produits, si la version courante est conforme aux exigences spécifiées, si la

documentation technique décrit entièrement et fidèlement les éléments de la configuration et si on a répondu à toutes les demandes de changement. (ANSI/IEEE)

### **vérification des exigences**

Vérification des exigences du système ou du logiciel. (ANSI/IEEE)

### **vérification descendante**

- a. Application de l'approche descendante aux essais; exige que des éléments de remplacement ou des routines fictives soient créés pour les segments nommément mentionnés dans les segments supérieurs soumis à l'essai mais pas encore codés.
- b. Contrôle des programmes organisés de façon hiérarchique, allant progressivement de haut en bas et ayant recours à la simulation des éléments de niveau inférieur. (ANSI/IEEE)

### **vérification et validation par un tiers**

Vérification et validation d'un produit logiciel par des individus ou des groupes autres que ceux qui l'ont tout d'abord conçu, mais qui peuvent appartenir à la même organisation. Le degré d'indépendance doit être en fonction de l'importance du logiciel. (ANSI/IEEE)

### **version**

Logiciel identifié et documenté.

### **vidage**

- a. Données qui ont été vidées. (ISO)
- b. Action d'écrire le contenu d'une mémoire, ou d'une partie de mémoire, généralement à partir d'une mémoire interne vers un support externe, dans un but précis, notamment pour permettre une autre utilisation de la mémoire comme protection contre les défaillances ou les

erreurs ou pour faciliter la mise au point.  
(ISO)

- c. Transfert périodique d'informations de la mémoire de l'ordinateur vers un support périphérique, ou bien d'un support vers un autre. Il préserve l'état actuel d'un programme et de ses données.

### violation de protection

Tentative d'une tâche d'accéder à une ressource protégée, ou d'effectuer une opération non autorisée sur une ressource affectée. Par exemple, essayer d'écrire sur un fichier à mémoire morte ou d'exécuter une instruction privilégiée constituera une violation de protection.

### virgule flottante

Forme d'arithmétique où un nombre est exprimé par un exposant et une mantisse et où la position de la virgule du nombre est déterminée en élevant la mantisse à la puissance représentée par l'exposant.

### viseur de casque

Visualisation d'informations simples de visée sur une unité fixée au casque des membres d'équipage.

### visuel de casque

Visualisation d'informations de vol, de navigation, d'attaque ou autres sur une unité fixée au casque des membres d'équipage.

### word

Voir mot-machine.

### zone commune

Zone utilisée par plus d'un segment d'un programme, ou par plus d'un programme.

CHAPTER/CHAPITRE 3ENGLISH TO FRENCH CROSS-REFERENCERENOI ANGLAIS A FRANCAIS

ENGLISH TERM	TERME FRANCAIS
abort	abandonner
absolute machine code	code machine absolu
abstract machine	machine abstraite
abstraction	résumé analytique
acceptance testing	essai de réception
access	accéder
access control mechanism	dispositif de contrôle d'accès
access time	temps d'accès
accessibility	accessibilité
accumulator	accumulateur
accuracy	exactitude
acronym	acronyme
actual parameter	paramètre effectif
Ada	Ada
Ada programming support environment (APSE)	environnement du soutien de programmation ADA (APSE)
adaptability	adaptabilité
adaptive maintenance	maintenance adaptative
address	adresse
address generation	calcul d'adresse
address space	espace adresses
ALGOL	ALGOL
algorithm	algorithme
algorithm analysis	analyse d'un algorithme
alias	pseudonyme
alignment	alignement
allocate	attribution

ENGLISH TERM	TERME FRANCAIS
allocated baseline	document de base attribué
alpha test	essai alpha
analog	analogique
analogue-to-digital (A/D)	conversion analogique/numérique (A/N)
analytical model	modèle analytique
application(s) software	logiciel d'application
application-oriented language	langage orienté d'application
apt	apt
aptitude test	test d'aptitude
architectural design	conception architecturale
architecture	architecture
archival memory	mémoire d'archivage
arithmetic unit	unité de calcul
assemble	assembler
assembler	assembleur
assembly language	langage d'assemblage
assembly language or assembly code	langage d'assemblage ou code d'assemblage
assertion	assertion
assignment statement	instruction d'affectation
associative memory	mémoire associative
asynchronous	asynchrone
audit	vérification
autocode	autocode
automated design tool	outil de conception automatisée
automated test generator	générateur d'essais automatisé
automated verification system	système automatisé de vérification
automated verification tools	outil de vérification automatisée
automatic test equipment (ATE)	équipement d'essai automatique (ATE)
availability	disponibilité
availability model	modèle de disponibilité

ENGLISH TERM	TERME FRANCAIS
avionics	avionique
back porch	palier arrière
back-up	mesures de secours
back-up programmer	programmeur-assistant
backplane	fond de panier
base address	adresse de base
base registers	registre de base
baseline	produit de base
baseline management	gestion de produit de base
batch	lot
begin-end block	bloc début-fin
benchmark	banc d'essai
best fit	ajusteur
beta testing	essai pilote
binary	binaire
binary digit (bit)	chiffre binaire (bit)
binary element	élément binaire
binding	association
bit	bit
bit rate	débit binaire
black box	boîte noire
block (n)	bloc
block (v)	grouper
block diagram	schéma fonctionnel
block-structured language	langage à structure de blocs
blocking factor	facteur de groupage
boolean	booléen
boolean logic	logique booléenne
boot	amorcer
bootstrap	amorce
bootstrap loader	chargeur-amorce



ENGLISH TERM	TERME FRANCAIS
bottom-up	ascendant
bottom-up design	conception ascendante
bottom-up testing	essai ascendant
branch	dérivation
branch testing	essai de branchement
broadcast	diffusion
buffer	tampon
bug	bogue
build	programme de construction
building block	bloc fonctionnel
built-in test equipment (BITE)	équipement de test intégré (BITE)
built-in-test (BIT)	test intégré (BIT)
bulk storage	mémoire de masse
bus	bus
bus controller	contrôleur de bus
bus monitor	moniteur de bus
byte	multiplet (octet)
C	C
cache	antémémoire
case	instruction de cas
central processing unit (CPU)	unité centrale (UC)
certification	homologation
chain	chaîne
chained list	liste chaînée
change control	contrôle des changements
channel	canal
chief programmer	chef programmeur
chief programmer team	équipe de programmation structurée
chip	puce
chip set	ensemble de puces
cobol	Cobol

ENGLISH TERM	TERME FRANCAIS
code (n)	code
code (v)	coder
code audit	vérification de code
code generation	génération de code
code generator	générateur de code
coder	codeur
cohesion	cohésion
command and control systems	système de commande et de contrôle
command language	langage de commande
command response	réponse à une commande
comment	commentaire
common area	zone commune
comparator	comparateur
compatibility	compatibilité
compile	compiler
compiler	compilateur
compiler generator	générateur de compilateurs
complexity	complexité
complexity measure	mesure de la complexité
component	élément
computer	ordinateur/calculateur
computer architecture	architecture d'ordinateur
computer code	code d'ordinateur
computer data	données informatiques
computer instruction	instruction machine
computer network	réseau d'ordinateurs
computer program abstract	résumé de programme machine
computer programme	programme machine
computer resources	ressources informatiques
computer system	système informatique
computer word	mot-machine

ENGLISH TERM	TERME FRANCAIS
conciseness	concision
concurrent processes	opérations exécutées en simultanéité
conditional control structure	structure de contrôle conditionnel
configuration	configuration
configuration audit	vérification de la configuration
configuration control	contrôle de la configuration
configuration control board	commission de contrôle de la configuration
configuration identification	identification de la configuration
configuration item	élément de la configuration
configuration management	gestion de la configuration
configuration status accounting	documentation sur l'état de la configuration
confinement	occultation
connection	renvoi
consistency	cohérence
control data	donnée de contrôle
control display unit	poste de commande et d'affichage
control statement	instruction de contrôle
control structure	structure de contrôle
conversational	conversationnel
conversion	conversion
core avionics	avionique centrale
core store	mémoire à tores
coroutines	co-programmes
corrective maintenance	dépannage
correctness	justesse
coupling	couplage
critical design review	examen critique de la conception (CDR)
critical module	module critique
critical piece first	les points critiques d'abord

ENGLISH TERM	TERME FRANCAIS
critical section	section critique
criticality	gravité
cross-assembler	assembleur croisé
cross-compiler	compilateur croisé
data	données
data abstraction	abstraction de données
data base	base de données
data bus	bus de données
data dictionary	dictionnaire de données
data domain	domaine des données
data driven design	conception dirigée par données
data flowchart	organigramme des données
data structure	structure de données
data type	type de données
deadlock	blocage
debug	mettre au point
debugging	mise au point
decision table	table de décision
delivery	livraison
demand paging	pagination à la demande
design	conception de système
design analysis	évaluation de la conception
design analyzer	analyseur de conception
design language	langage de conception
design methodology	méthodologie de conception
design phase	phase de conception
design requirement	exigences de conception
design review	examen de la conception
design specification	spécification de conception
desk checking	contrôle de programmation
desk-checking	mise à l'essai de programmation

ENGLISH TERM	TERME FRANCAIS
detailed design	conception détaillée
development methodology	méthodologie d'élaboration
device	dispositif
device driver	programme de gestion (logiciel)
diagnostic	diagnostic
diagnostic aid	aide au diagnostic
digit	chiffre
digital	numérique
digital computer	calculateur numérique
digital-to-analogue (D/A)	numérique à analogique (N/A)
direct access	accès direct
direct memory access (DMA)	accès direct mémoire (DMA)
directed graph	graphe orienté
discrete signal	signal discret
disk	disque
dispatcher	répartiteur
distributed architecture	architecture répartie
distributed processing	informatique répartie
documentation	documentation
domain testing	mise à l'essai du domaine
driver	programme de gestion
dual coding	double programmation
dump	vidage
duplex operation	en mode duplex
dynamic allocation	affectation dynamique
dynamic analysis	analyse dynamique
dynamic analyzer	analyseur dynamique
dynamic binding	association dynamique
dynamic code analysis	analyse de code dynamique
dynamic priority	priorité dynamique
dynamic restructuring	restructuration dynamique

ENGLISH TERM	TERME FRANCAIS
dynamic storage allocation	affectation de mémoire dynamique
editor	éditeur
efficiency	efficacité
egoless programming	programmation collective
electro-optical	électro-optique
electronic attitude director indicator	indicateur d'assiette électronique
embedded computer system	système informatique intégré
embedded software	logiciel intégré
emulation	émulation
emulator	émulateur
encoding	codage
entry point	point d'entrée
environment	environnement/cadre/contexte d'utilisation
error	erreur
error analysis	analyse d'erreur
error category	catégorie d'erreur
error data	données sur les erreurs
error model	modèle d'erreur
error prediction	prédiction des erreurs
error procedures	procédures pour erreurs
exception	anomalie
executable object code	code objet exécutable
execution	exécution
execution time	temps d'exécution
execution time theory	théorie de la durée d'exécution
executive routine	programme directeur
exit	sortie
expendability	extensibilité
expert system	système expert
external function testing	mise à l'essai d'une fonction externe

ENGLISH TERM	TERME FRANCAIS
failure	défaillance
failure rate	taux de défaillance
failure recovery	reprise sur incident
fault	défaut
fault seeding	implantation d'erreurs
fault tolerance	insensibilité aux défauts
federated architecture	architecture fédérée
fibre optics	fibre optique
field	champ
field testing	mise au point en clientèle
file	fichier
finite state machine	automate à nombre fini d'états
firmware	microprogrammation
first fit	placeur
first in-first out (FIFO)	"système premier entré, premier sorti (FIFO)"
flag	drapeau
floating-point	virgule flottante
flow of control	ordre d'exécution
flowchart	organigramme
foreground/background	premier plan/arrière-plan
formal language	langage formel
formal parameter	paramètre formel
formal qualification review	examen formel de qualification
formal specification	spécification formelle
formal testing	essai formel
fortran	FORTRAN
fragmentation	fragmentation
free	service libre
free list	liste libre
front porch	palier avant
function	fonction

ENGLISH TERM	TERME FRANCAIS
functional baseline	base fonctionnelle
functional decomposition	décomposition fonctionnelle
functional design	conception fonctionnelle
functional requirement	exigence fonctionnelle
functional specification	spécification fonctionnelle
functional test	essai fonctionnel
functional unit	unité fonctionnelle
general purpose computer	ordinateur universel
generator	générateur
graceful degradation	traitement réduit
graph	graphe
ground support equipment (GSE)	matériel de servitude au sol (GSE)
half duplex operation	en mode semi-duplex
handler	programme de gestion
hardware	matériel
head	tête
head down display	présentation tête basse (HDD)
head-up display	présentation tête haute (HUD)
helmet mounted display	visuel de casque
helmet mounted sight	viseur de casque
heuristic method	méthode heuristique
hierarchical decomposition	décomposition hiérarchique
hierarchical input-process-output	entrée-traitement-sortie hiérarchique (HIPO)
hierarchy	hiérarchie
high(er) order language	langage évolué
highway	autoroute de l'information
host computer	ordinateur central
identifier	identificateur
imperfect debugging	mise au point imparfaite
implementation	mise en oeuvre
implementation phase	phase de mise en oeuvre



ENGLISH TERM	TERME FRANCAIS
implementation requirement	exigences de la mise en oeuvre
independent verification and validation	vérification et validation par un tiers
index register	registre d'index
indigenous fault	erreur du programme
inductive assertion method	méthode d'assertion inductive
information bits	bits utiles
information hiding	masquage de l'information
input assertion	assertion d'entrée
inspection	inspection
installation and check-out phase	phase de mise en place et de vérification
installation testing	essai de mise en place
instruction	instruction
instruction set	jeu d'instructions
instruction set architecture (ISA)	architecture de jeu d'instructions (ISA)
instrumentation tool	outil d'essai
integration	intégration
integration testing	essai d'intégration
integrity	intégrité
interactive	interactif
interblock gap	espace interbloc
interface	interface
interface requirement	exigence d'interface
interface specification	spécification d'interface
interface testing	essai d'interface
interleaving	imbrication
interoperability	possibilité d'échange d'informations
interpret	interpréter
interpreter	interpréteur
interrupt (n)	interruption
interrupt (v)	interrompre

ENGLISH TERM	TERME FRANCAIS
iteration	itération
JCL	JCL
job	travail
jump	branchement
karnaugh map	table de Karnaugh
kernel	noyau
key	clé
kilo (K)	kilo (K)
label	label
language processor	processeur de langage
large scale integration	intégration à grande échelle (LSI)
last in-first out	système dernier entré-premier sorti (LIFO)
latency	attente
least significant bit (LSB)	bit de plus faible poids (LSB)
length register	registre de longueur
level of documentation	niveau de documentation
lexical	lexical
light emitting diode (LED)	diode électroluminescente
line replaceable unit (LRU)	unité remplaçable sur place (LRU)
link	lien
link list	liste de liens
link(age) editor	éditeur de liens
link(age) loader	chargeur de liens
linker	linker
liquid crystal display	affichage à cristaux liquides (LCD)
list	liste
list processing	traitement de liste
listing	listage
load	charger
load map	plan d'implantation

ENGLISH TERM	TERME FRANCAIS
load module	module de chargement
loader	chargeur
log	journal
log on	entrée dans le système
logical address	adresse logique
logical device	dispositif logique
logical file	fichier logique
logical instruction	instruction logique
logical record	enregistrement logique
loop	boucle
machine	machine
machine code	code machine
machine cycle	cycle machine
machine instruction	instruction machine
machine language	langage machine
machine word	mot-machine
machine-independent	indépendant d'une machine
macroinstruction	macro-instruction
macroprocessor	macroprocesseur
mailbox	boîte à lettres
main storage	mémoire centrale
maintainability	maintenabilité
map program	programme de projection
mass storage	mémoire de masse
mass store	mémoire de grande capacité
master file	fichier maître
master library	bibliothèque principale
matrix	matrice
media	support
medium scale integration (MSI)	intégration à moyenne échelle (MSI)
memory device	mémoire

ENGLISH TERM	TERME FRANCAIS
memory management	gestion mémoire
message	message
metalanguage	métalangage
metric	métrologie
microcode	microcode
microcomputer	micro-ordinateur
microprocessor	microprocesseur
microprogramme	microprogramme
milestone	jalón
mission avionics	avionique de mission
mnemonic symbol	symbole mnémonique
model	modèle
modular decomposition	décomposition modulaire
modular programming	programmation modulaire
modularity	modularité
module	module
module testing	essai de module
monitor	moniteur
most significant bit	bit de plus fort poids
multi-level-feedback queue	file d'attente de réinjection multiniveau
multi-tasking	fonctionnement multitâche
multi-user operating system	système d'exploitation à utilisateurs multiples
multifunction display	affichage multifonction
multilevel security	sécurité multiniveau
multiplex	multiplexer
multiplex data bus	bus de données multiplex
multiprocessor	multiprocesseur
multiprogramming	multiprogrammation
MUX	MUX
N-ary	N-aire
natural language	langage naturel

ENGLISH TERM	TERME FRANCAIS
nest	emboîter
network	réseau
node	noeud
non-preemptive scheduling	ordonnancement non modifiable
non-volatile memory	mémoire rémanente
numerical reliability	durée de fiabilité
object code	code objet
object module	module objet
object oriented design	conception orientée vers l'objet
object oriented programming	programmation orientée vers l'objet
object programme	programme objet
open system	système ouvert
operand	opérande
operating system (OS)	système d'exploitation (SE)
operational flight programme (OFP)	programme de vol opérationnel (OFP)
operational reliability	fiabilité opérationnelle
operational software (embedded)	logiciel opérationnel (intégré)
operator	opérateur
optical fibre	fibre optique
optimization	optimisation
output assertion	assertion de sortie
overhead	temps système
overlay	segment de recouvrement
paged-store	mémoire organisée en pages
paging	pagination
parameter	paramètre
parity bit	bit de parité
parse	analyser
partial correctness	exactitude partielle
partitioning	découpage

ENGLISH TERM	TERME FRANCAIS
pascal	pascal
patch	correction
path analysis	analyse des branches
path condition	condition de choix d'une branche
path expression	instruction de choix d'une branche
path testing	essai des branches
performance evaluation	analyse des performances
performance requirement	exigence de performance
performance specification	spécification de performance
performance testing	essai de performance
petri net	réseau de Pétri
physical address	adresse physique
physical device	dispositif physique
physical requirement	exigence physique
PL/1	PL/1
pointer	indicateur/Pointeur
polling	appel sélectif
pop	désempilage
portability	portabilité
positional notation	notation pondérée
precision	précision
precompiler	précompilateur
predicate	prédicat
preemptive scheduling	ordonnancement modifiable
preliminary design	étude préliminaire
preprocessor	préprocesseur
primary memory	mémoire primaire
priority scheduling	ordonnancement à priorités
privileged instruction	instruction privilégiée
problem-oriented	langage orienté vers les problèmes

ENGLISH TERM	TERME FRANCAIS
procedure	procédure
process	processus
processor	processeur
product baseline	base de produit
production library	bibliothèque de production
program architecture	architecture de programme
program block	bloc de programme
program design language	langage de conception de programme (PDL)
program extension	extension de programme
program instrumentation	sondage de programme
program library	bibliothèque des programmes
program mutation	modification de programme
program protection	protection de programme
program specification	spécification de programme
program synthesis	synthèse de programme
programmable	programmable
programmable read-only memory (PROM)	mémoire morte programmable (PROM)
programme	programmer
programmer	programmeur
programming	programmation
programming language	langage de programmation
programming support environment	ensemble des aides à la programmation
project notebook	fichier de projet
project plan	plan de projet
prom	PROM
prompt	message guide-opérateur
proof of correctness	preuve d'exactitude
protection	protection
protection violation	violation de protection
protocol	protocole

ENGLISH TERM	TERME FRANCAIS
pseudo code	pseudo-code
pulse code modulation (PCM)	modulation par impulsions codées (PCM)
push	empiler
pushdown storage	mémoire à liste inversée
qualification testing	test de performance
quality	qualité
quality assurance	assurance de la qualité
quality metric	mesure de la qualité
queue	file d'attente
random access	accès sélectif
random access memory (RAM)	mémoire à accès sélectif (RAM)
read only memory (ROM)	mémoire morte (ROM)
ready	prêt
ready list	liste de réserve
real time	en temps réel
reconfiguration	reconfiguration
record	enregistrement
recovery testing	essai de récupération
recursive routine	programme récurrent
redundancy	redondance
reentrant code	code rentrant
register	registre
regression testing	test de régression
release	libération
reliability	fiabilité
reliability assessment	évaluation de la fiabilité
reliability data	données de fiabilité
reliability growth	croissance de la fiabilité
reliability model	modèle de fiabilité
relocatable machine code	code machine translatable
relocation	translation



ENGLISH TERM	TERME FRANCAIS
remote terminal	appareil terminal
rendez-vous	rendez-vous
requirement	exigence
requirements analysis	vérification des exigences
	analyse des exigences
requirements phase	phase d'établissement des exigences
requirements specification	spécification des exigences
requirements specification language	langage de spécification des exigences
resource	ressource
resource allocation	affectation des ressources
resource management	gestion des ressources
response time	temps de réponse
restore	restauration
retirement phase	phase de déclin
reusability	possibilité de réutilisation
robustness	robustesse
root compiler	compilateur de base
round robin scheduling	ordonnancement à tour de rôle
rounding	arrondi
routine	routine
run time	durée d'exécution
	début d'exécution
running	exécution
scenario	progiciel d'essai et de vérification
scheduling	ordonnancement
secondary memory	mémoire auxiliaire
security	sécurité
security kernel	noyau de sécurité
security testing	essai de sécurité
segment (n)	segment

ENGLISH TERM	TERME FRANCAIS
segment (v)	segmenter
segmentation	segmentation
semantics	sémantique
semaphore	sémaphore
send	émission
sequential processes	processus séquentiels
serial access	accès séquentiel
sharing	partage
shift	décalage
side-effect	effet secondaire
sign bit	bit de signe
signal processing	traitement de signal
simplex operation	exploitation en mode simplex
simulation	simulation
simulation testing	essai de simulation
simulator	simulateur
sizing	évaluation
software	logiciel
software data base	base de données logicielles
software development cycle	cycle d'élaboration d'un logiciel
software development library	bibliothèque d'élaboration de logiciels
software development notebook	guide d'élaboration d'un logiciel
software development plan	plan d'élaboration d'un logiciel
software development process	processus d'élaboration d'un logiciel
software documentation	documentation technique du logiciel
software engineering	génie logiciel
software engineering standards	normes du génie logiciel
software experience data	données de référence sur un logiciel
software factory	usine à logiciels

ENGLISH TERM	TERME FRANCAIS
software librarian	responsable de la bibliothèque des logiciels
software library	bibliothèque de logiciels
software life cycle	cycle de vie du logiciel
software maintenance	maintenance du logiciel
software monitor	moniteur logiciel
software problem report (SPR)	rapport sur un problème logiciel (SPR)
software product	produit logiciel
software quality	qualité d'un logiciel
software reliability	fiabilité d'un logiciel
software repository	dépôt de logiciels
software sneak analysis	analyse des vices cachés d'un logiciel
software tool	outil logiciel
source code	code source
source language	langage source
source listing	listage source
source programme	programme-source
specification	spécification
specification language	langage de spécification
spooler	désynchroniseur des entrées/sorties
spooling	système de désynchronisation des entrées-sorties
stability	stabilité
stack	pile
standards	normes
standards enforcer	logiciel de contrôle des normes d'élaboration
state diagram	diagramme d'état
statement testing	essai d'instruction
static analysis	analyse statique
static analyzer	analyseur statique
static binding	reliure statique
static priority	priorité statique

ENGLISH TERM	TERME FRANCAIS
statistical test model	modèle d'essai statistique
status accounting	rapport sur l'état
stepwise refinement	méthode des approximations successives
strength	force
stress testing	essai à pleine charge
string	chaîne
string processing	traitement par chaînes
strong typing	fortement typé
structured design	conception structurée
structured program	programme structuré
structured programming	programmation structurée
structured programming language	langage de programmation structuré
structured walkthrough	revue de projet structurée
structuredness	caractère structurel
stub	élément de remplacement temporaire
subprogram	sous-programme
subroutine	sous-routine
subscript	indice inférieur
subsystem	sous-système
subsystem testing	essai de sous-système
supervisory program	programme superviseur
support software	logiciel de soutien
support tools	outils de soutien
suspended	en suspens
swapping	permutation
symbolic execution	exécution symbolique
synchro	synchro
synchronization	synchronisation
synchronous	synchrone
syntactic	syntaxique

ENGLISH TERM	TERME FRANCAIS
syntactic analysis	analyse syntaxique
syntax	syntaxe
system	système
system architecture	architecture du système
system design	conception du système
system design language	langage de conception de système
system design review	examen de la conception du système
system documentation	documentation sur le système
system library	bibliothèque du système
system reliability	fiabilité du système
system requirement review	examen des exigences du système
system software	logiciel de base
system testing	essai de système
systems analyst	analyste de système
systems engineering	ingénierie des systèmes
systems integration	intégration système
systems programmer	programmeur de système
table	table
tallying	comptage
target language	langage cible
target machine	machine cible
task	tâche
task control block	bloc de commande de tâche
task descriptor	descripteur de tâche
task list	liste de tâches
task state	état de la tâche
terminal	terminal
termination proof	preuve d'accomplissement normal
ternary element	élément ternaire
test bed	banc d'essai
test case	jeu d'essai

ENGLISH TERM	TERME FRANCAIS
test data	données d'essai
test data generators	générateurs de données d'essai
test driver	programme de gestion d'essai
test log	journal d'essai
test plan	plan d'essai
test procedure	procédure d'essai
test repeatability	possibilité de répétition d'un essai
test report	rapport d'essai
test validity	validité de l'essai
testability	testabilité
testing	essai
throughput	débit
time division multiplex (TDM)	multiplexage à partage du temps (TDM)
time sharing	partage de temps
	en temps partagé
time Slice	tranche de temps
timer	rythmeur
timing	calcul de la durée
timing analyzer	analyseur de durée
tolerance	tolérance
top-down	descendant
top-down design	conception descendante
top-down testing	vérification descendante
total correctness	exactitude totale
trace (n)	trace
trace (v)	tracer
trace routine	routine de traçage
traceability	possibilité de traçage
tracer	programme de traçage
translator	traducteur
trap	déroutement

ENGLISH TERM	TERME FRANCAIS
tree	arbre
truncation	troncature
turn around time	temps de rotation
twisted shielded pair (TSP)	paire torsadée blindée (TSP)
undefined	non définie
usability	exploitabilité
user documentation	documentation de l'utilisateur
utility software	logiciel utilitaire
utilization	utilisation
validation	validation
variable	variable
verification	vérification
version	version
very large scale integration (VLSI)	intégration à très grande échelle (VLSI)
virtual cockpit	habitacle virtuel
virtual machine	machine virtuelle
virtual memory	mémoire virtuelle
volatile storage	mémoire transitoire
wait List	liste d'attente
wakeup	réveil
walk-through	revue de projet
word	word
word length	longueur de mot

CHAPTER/CHAPITRE 4RENOI FRANCAIS A ANGLAISFRENCH TO ENGLISH CROSS-REFERENCE

TERME FRANCAIS	ENGLISH TERM
abandonner	abort
abstraction de données	data abstraction
accéder	access
accès direct	direct access
accès direct mémoire (DMA)	direct memory access (DMA)
accès sélectif	random access
accès séquentiel	serial access
accessibilité	accessibility
accumulateur	accumulator
acronyme	acronym
Ada	Ada
adaptabilité	adaptability
adresse	address
adresse de base	base address
adresse logique	logical address
adresse physique	physical address
affectation de mémoire dynamique	dynamic storage allocation
affectation des ressources	resource allocation
affectation dynamique	dynamic allocation
affichage à cristaux liquides (LCD)	liquid crystal display
affichage multifonction	multifunction display
aide au diagnostic	diagnostic aid
ajusteur	best fit
ALGOL	ALGOL
algorithme	algorithm
alignement	alignment



TERME FRANCAIS	ENGLISH TERM
amorcer	bootstrap
amorcer	boot
analogique	analog
analyse d'erreur	error analysis
analyse d'un algorithme	algorithm analysis
analyse de code dynamique	dynamic code analysis
analyse des branches	path analysis
analyse des exigences	requirements analysis
analyse des performances	performance evaluation
analyse des vices cachés d'un logiciel	software sneak analysis
analyse dynamique	dynamic analysis
analyse statique	static analysis
analyse syntaxique	syntactic analysis
analyser	parse
analyseur de conception	design analyzer
analyseur de durée	timing analyzer
analyseur dynamique	dynamic analyzer
analyseur statique	static analyzer
analyste de système	systems analyst
anomalie	exception
antémémoire	cache
appareil terminal	remote terminal
appel sélectif	polling
apt	apt
arbre	tree
architecture	architecture
architecture d'ordinateur	computer architecture
architecture de jeu d'instructions (ISA)	instruction set architecture (ISA)
architecture de programme	program architecture
architecture du système	system architecture

TERME FRANCAIS	ENGLISH TERM
architecture fédérée	federated architecture
architecture répartie	distributed architecture
arrondi	rounding
ascendant	bottom-up
assembler	assemble
assembleur	assembler
assembleur croisé	cross-assembler
assertion	assertion
assertion d'entrée	input assertion
assertion de sortie	output assertion
association	binding
association dynamique	dynamic binding
assurance de la qualité	quality assurance
asynchrone	asynchronous
attente	latency
attribution	allocate
autocode	autocode
automate à nombre fini d'états	finite state machine
autoroute de l'information	highway
avionique	avionics
avionique centrale	core avionics
avionique de mission	mission avionics
banc d'essai	benchmark
	test bed
base de données	data base
base de données logicielles	software data base
base de produit	product baseline
base fonctionnelle	functional baseline
bibliothèque d'élaboration de logiciels	software development library
bibliothèque de logiciels	software library

TERME FRANCAIS	ENGLISH TERM
bibliothèque de production	production library
bibliothèque des programmes	program library
bibliothèque du système	system library
bibliothèque principale	master library
binaire	binary
bit	bit
bit de parité	parity bit
bit de plus faible poids (LSB)	least significant bit (LSB)
bit de plus fort poids	most significant bit
bit de signe	sign bit
bits utiles	information bits
bloc	block (n)
bloc de commande de tâche	task control block
bloc de programme	program block
bloc début-fin	begin-end block
bloc fonctionnel	building block
blocage	deadlock
bogue	bug
boîte à lettres	mailbox
boîte noire	black box
booléen	boolean
boucle	loop
branchement	jump
bus	bus
bus de données	data bus
bus de données multiplex	multiplex data bus
C	C
calcul d'adresse	address generation
calcul de la durée	timing
calculateur numérique	digital computer
canal	channel

TERME FRANCAIS	ENGLISH TERM
caractère structurel	structuredness
catégorie d'erreur	error category
chaîne	chain
	string
champ	field
charger	load
chargeur	loader
chargeur-amorce	bootstrap loader
chargeur de liens	link(age) loader
chef programmeur	chief programmer
chiffre	digit
chiffre binaire (bit)	binary digit (bit)
clé	key
co-programmes	coroutines
Cobol	cobol
codage	encoding
code	code (n)
code d'ordinateur	computer code
code machine	machine code
code machine absolu	absolute machine code
code machine translatable	relocatable machine code
code objet	object code
code objet exécutable	executable object code
code rentrant	reentrant code
code source	source code
coder	code (v)
codeur	coder
cohérence	consistency
cohésion	cohesion
commentaire	comment

TERME FRANCAIS	ENGLISH TERM
commission de contrôle de la configuration	configuration control board
comparateur	comparator
compatibilité	compatibility
compilateur	compiler
compilateur croisé	cross-compiler
compilateur de base	root compiler
compiler	compile
complexité	complexity
comptage	tallying
conception architecturale	architectural design
conception ascendante	bottom-up design
conception de système	design
conception descendante	top-down design
conception détaillée	detailed design
conception dirigée par données	data driven design
conception du système	system design
conception fonctionnelle	functional design
conception orientée vers l'objet	object oriented design
conception structurée	structured design
concision	conciseness
condition de choix d'une branche	path condition
configuration	configuration
contrôle de la configuration	configuration control
contrôle de programmation	desk checking
contrôle des changements	change control
contrôleur de bus	bus controller
conversationnel	conversational
conversion	conversion
conversion analogique/numérique (A/N)	analogue-to-digital (A/D)
correction	patch

TERME FRANCAIS	ENGLISH TERM
couplage	coupling
croissance de la fiabilité	reliability growth
cycle d'élaboration d'un logiciel	software development cycle
cycle de vie du logiciel	software life cycle
cycle machine	machine cycle
débit	throughput
débit binaire	bit rate
début d'exécution	run time
décalage	shift
décomposition fonctionnelle	functional decomposition
décomposition hiérarchique	hierarchical decomposition
décomposition modulaire	modular decomposition
découpage	partitioning
défaillance	failure
défaut	fault
dépannage	corrective maintenance
dépôt de logiciels	software repository
dérivation	branch
déroutement	trap
descendant	top-down
descripteur de tâche	task descriptor
désempilage	pop
désynchroniseur des entrées/sorties	spooler
diagnostic	diagnostic
diagramme d'état	state diagram
dictionnaire de données	data dictionary
diffusion	broadcast
diode électroluminescente	light emitting diode (LED)
disponibilité	availability
dispositif	device

TERME FRANCAIS	ENGLISH TERM
dispositif de contrôle d'accès	access control mechanism
dispositif logique	logical device
dispositif physique	physical device
disque	disk
document de base attribué	allocated baseline
documentation	documentation
documentation de l'utilisateur	user documentation
documentation sur l'état de la configuration	configuration status accounting
documentation sur le système	system documentation
documentation technique du logiciel	software documentation
domaine des données	data domain
donnée de contrôle	control data
données	data
données d'essai	test data
données de fiabilité	reliability data
données de référence sur un logiciel	software experience data
données informatiques	computer data
données sur les erreurs	error data
double programmation	dual coding
drapeau	flag
durée d'exécution	run time
durée de fiabilité	numerical reliability
éditeur	editor
éditeur de liens	link(age) editor
effet secondaire	side-effect
efficacité	efficiency
électro-optique	electro-optical
élément	component
élément binaire	binary element

TERME FRANCAIS	ENGLISH TERM
élément de la configuration	configuration item
élément de remplacement temporaire	stub
élément ternaire	ternary element
emboîter	nest
émission	send
empiler	push
émulateur	emulator
émulation	emulation
en mode duplex	duplex operation
en mode semi-duplex	half duplex operation
en suspens	suspended
en temps partagé	time sharing
en temps réel	real time
enregistrement	record
enregistrement logique	logical record
ensemble de puces	chip set
ensemble des aides à la programmation	programming support environment
entrée dans le système	log on
entrée-traitement-sortie hiérarchique (HIPO)	hierarchical input-process-output
environnement/cadre/contexte d'utilisation	environment
environnement du soutien de programmation ADA	Ada programming support environment
équipe de programmation structurée	chief programmer team
équipement d'essai automatique (ATE)	automatic test equipment (ATE)
équipement de test intégré (BITE)	built-in test equipment (BITE)
erreur	error
erreur du programme	indigenous fault
espace adresses	address space



TERME FRANCAIS	ENGLISH TERM
espace interbloc	interblock gap
essai	testing
essai à pleine charge	stress testing
essai alpha	alpha test
essai ascendant	bottom-up testing
essai d'instruction	statement testing
essai d'intégration	integration testing
essai d'interface	interface testing
essai de branchement	branch testing
essai de mise en place	installation testing
essai de module	module testing
essai de performance	performance testing
essai de réception	acceptance testing
essai de récupération	recovery testing
essai de sécurité	security testing
essai de simulation	simulation testing
essai de sous-système	subsystem testing
essai de système	system testing
essai des branches	path testing
essai fonctionnel	functional test
essai formel	formal testing
essai pilote	beta testing
état de la tâche	task state
étude préliminaire	preliminary design
évaluation	sizing
évaluation de la conception	design analysis
évaluation de la fiabilité	reliability assessment
exactitude	accuracy
exactitude partielle	partial correctness
exactitude totale	total correctness

TERME FRANCAIS	ENGLISH TERM
examen critique de la conception (CDR)	critical design review
examen de la conception	design review
examen de la conception du système	system design review
examen des exigences du système	system requirement review
examen formel de qualification	formal qualification review
exécution	execution
	running
exécution symbolique	symbolic execution
exigence	requirement
exigence d'interface	interface requirement
exigence de performance	performance requirement
exigence fonctionnelle	functional requirement
exigence physique	physical requirement
exigences de conception	design requirement
exigences de la mise en oeuvre	implementation requirement
exploitabilité	usability
exploitation en mode simplex	simplex operation
extensibilité	expendability
extension de programme	program extension
facteur de groupage	blocking factor
fiabilité	reliability
fiabilité d'un logiciel	software reliability
fiabilité du système	system reliability
fiabilité opérationnelle	operational reliability
fibre optique	fibre optics
	optical fibre
fichier	file
fichier de projet	project notebook
fichier logique	logical file
fichier maître	master file

TERME FRANCAIS	ENGLISH TERM
file d'attente	queue
file d'attente de réinjection multiniveau	multi-level-feedback queue
fonction	function
fonctionnement multitâche	multi-tasking
fond de panier	backplane
force	strength
fortement typé	strong typing
FORTTRAN	fortran
fragmentation	fragmentation
générateur	generator
générateur d'essais automatisé	automated test generator
générateur de code	code generator
générateur de compilateurs	compiler generator
générateurs de données d'essai	test data generators
génération de code	code generation
génie logiciel	software engineering
gestion de la configuration	configuration management
gestion de produit de base	baseline management
gestion des ressources	resource management
gestion mémoire	memory management
graphe	graph
graphe orienté	directed graph
gravité	criticality
grouper	block (v)
guide d'élaboration d'un logiciel	software development notebook
habitacle virtuel	virtual cockpit
hiérarchie	hierarchy
homologation	certification
identificateur	identifier

TERME FRANCAIS	ENGLISH TERM
identification de la configuration	configuration identification
imbrication	interleaving
implantation d'erreurs	fault seeding
indépendant d'une machine	machine-independent
indicateur d'assiette électronique	electronic attitude director indicator
indicateur/Pointeur	pointer
indice inférieur	subscript
informatique répartie	distributed processing
ingénierie des systèmes	systems engineering
insensibilité aux défauts	fault tolerance
inspection	inspection
instruction	instruction
instruction d'affectation	assignment statement
instruction de cas	case
instruction de choix d'une branche	path expression
instruction de contrôle	control statement
instruction logique	logical instruction
instruction machine	computer instruction
	machine instruction
instruction privilégiée	privileged instruction
intégration	integration
intégration à grande échelle (LSI)	large scale integration
intégration à moyenne échelle (MSI)	medium scale integration (MSI)
intégration à très grande échelle (VLSI)	very large scale integration (VLSI)
intégration système	systems integration
intégrité	integrity
interactif	interactive
interface	interface

TERME FRANCAIS	ENGLISH TERM
interpréter	interpret
interpréteur	interpreter
interrompre	interrupt (v)
interruption	interrupt (n)
itération	iteration
jalón	milestone
JCL	JCL
jeu d'essai	test case
jeu d'instructions	instruction set
journal	log
journal d'essai	test log
justesse	correctness
kilo (K)	kilo (K)
label	label
langage à structure de blocs	block-structured language
langage cible	target language
langage d'assemblage	assembly language
langage d'assemblage ou code d'assemblage	assembly language or assembly code
langage de commande	command language
langage de conception	design language
langage de conception de programme (PDL)	program design language
langage de conception de système	system design language
langage de programmation	programming language
langage de programmation structuré	structured programming language
langage de spécification	specification language
langage de spécification des exigences	requirements specification language
langage évolué	high(er) order language
langage formel	formal language
langage machine	machine language

TERME FRANCAIS	ENGLISH TERM
langage naturel	natural language
langage orienté d'application	application-oriented language
langage orienté vers les problèmes	problem-oriented
langage source	source language
les points critiques d'abord	critical piece first
lexical	lexical
libération	release
lien	link
linker	linker
listage	listing
listage source	source listing
liste	list
liste chaînée	chained list
liste d'attente	wait List
liste de liens	link list
liste de réserve	ready list
liste de tâches	task list
liste libre	free list
livraison	delivery
logiciel	software
logiciel d'application	application(s) software
logiciel de base	system software
logiciel de contrôle des normes d'élaboration	standards enforcer
logiciel de soutien	support software
logiciel intégré	embedded software
logiciel opérationnel (intégré)	operational software (embedded)
logiciel utilitaire	utility software
logique booléenne	boolean logic
longueur de mot	word length
lot	batch

TERME FRANCAIS	ENGLISH TERM
machine	machine
machine abstraite	abstract machine
machine cible	target machine
machine virtuelle	virtual machine
macro-instruction	macroinstruction
macroprocesseur	macroprocessor
maintenabilité	maintainability
maintenance adaptative	adaptive maintenance
maintenance du logiciel	software maintenance
masquage de l'information	information hiding
matériel	hardware
matériel de servitude au sol (GSE)	ground support equipment (GSE)
matrice	matrix
mémoire	memory device
mémoire à accès sélectif (RAM)	random access memory (RAM)
mémoire à liste inversée	pushdown storage
mémoire à tores	core store
mémoire associative	associative memory
mémoire auxiliaire	secondary memory
mémoire centrale	main storage
mémoire d'archivage	archival memory
mémoire de grande capacité	mass store
mémoire de masse	bulk storage
	mass storage
mémoire morte (ROM)	read only memory (ROM)
mémoire morte programmable (PROM)	programmable read-only memory (PROM)
mémoire organisée en pages	paged-store
mémoire primaire	primary memory
mémoire rémanente	non-volatile memory
mémoire transitoire	volatile storage

TERME FRANCAIS	ENGLISH TERM
mémoire virtuelle	virtual memory
message	message
message guide-opérateur	prompt
mesure de la complexité	complexity measure
mesure de la qualité	quality metric
mesures de secours	back-up
métalangage	metalanguage
méthode d'assertion inductive	inductive assertion method
méthode des approximations successives	stepwise refinement
méthode heuristique	heuristic method
méthodologie d'élaboration	development methodology
méthodologie de conception	design methodology
métrologie	metric
mettre au point	debug
micro-ordinateur	microcomputer
microcode	microcode
microprocesseur	microprocessor
microprogrammation	firmware
microprogramme	microprogramme
mise à l'essai d'une fonction externe	external function testing
mise à l'essai de programmation	desk-checking
mise à l'essai du domaine	domain testing
mise au point	debugging
mise au point en clientèle	field testing
mise au point imparfaite	imperfect debugging
mise en oeuvre	implementation
modèle	model
modèle analytique	analytical model
modèle d'erreur	error model
modèle d'essai statistique	statistical test model



TERME FRANCAIS	ENGLISH TERM
modèle de disponibilité	availability model
modèle de fiabilité	reliability model
modification de programme	program mutation
modularité	modularity
modulation par impulsions codées (PCM)	pulse code modulation (PCM)
module	module
module critique	critical module
module de chargement	load module
module objet	object module
moniteur	monitor
moniteur de bus	bus monitor
moniteur logiciel	software monitor
mot-machine	computer word
	machine word
multiplet (octet)	byte
multiplexage à partage du temps (TDM)	time division multiplex (TDM)
multiplexer	multiplex
multiprocesseur	multiprocessor
multiprogrammation	multiprogramming
MUX	MUX
N-aire	N-ary
niveau de documentation	level of documentation
noeud	node
non définie	undefined
normes	standards
normes du génie logiciel	software engineering standards
notation pondérée	positional notation
noyau	kernel
noyau de sécurité	security kernel
numérique	digital

TERME FRANCAIS	ENGLISH TERM
numérique à analogique (N/A)	digital-to-analogue (D/A)
occultation	confinement
opérande	operand
opérateur	operator
opérations exécutées en simultanéité	concurrent processes
optimisation	optimization
ordinateur/calculateur	computer
ordinateur central	host computer
ordinateur universel	general purpose computer
ordonnancement	scheduling
ordonnancement à priorités	priority scheduling
ordonnancement à tour de rôle	round robin scheduling
ordonnancement modifiable	preemptive scheduling
ordonnancement non modifiable	non-preemptive scheduling
ordre d'exécution	flow of control
organigramme	flowchart
organigramme des données	data flowchart
outil d'essai	instrumentation tool
outil de conception automatisée	automated design tool
outil de vérification automatisée	automated verification tools
outil logiciel	software tool
outils de soutien	support tools
pagination	paging
pagination à la demande	demand paging
paire torsadée blindée (TSP)	twisted shielded pair (TSP)
palier arrière	back porch
palier avant	front porch
paramètre	parameter
paramètre effectif	actual parameter
paramètre formel	formal parameter

TERME FRANCAIS	ENGLISH TERM
partage	sharing
partage de temps	time sharing
pascal	pascal
permutation	swapping
phase d'établissement des exigences	requirements phase
phase de conception	design phase
phase de déclin	retirement phase
phase de mise en oeuvre	implementation phase
phase de mise en place et de vérification	installation and check-out phase
pile	stack
PL/1	PL/1
placeur	first fit
plan d'élaboration d'un logiciel	software development plan
plan d'essai	test plan
plan d'implantation	load map
plan de projet	project plan
point d'entrée	entry point
portabilité	portability
possibilité d'échange d'informations	interoperability
possibilité de répétition d'un essai	test repeatability
possibilité de réutilisation	reusability
possibilité de traçage	traceability
poste de commande et d'affichage	control display unit
précision	precision
précompilateur	precompiler
prédicat	predicate
prédiction des erreurs	error prediction
premier plan/arrière-plan	foreground/background
préprocesseur	preprocessor

TERME FRANCAIS	ENGLISH TERM
présentation tête basse (HDD)	head down display
présentation tête haute (HUD)	head-up display
prêt	ready
preuve d'accomplissement normal	termination proof
preuve d'exactitude	proof of correctness
priorité dynamique	dynamic priority
priorité statique	static priority
procédure	procedure
procédure d'essai	test procedure
procédures pour erreurs	error procedures
processeur	processor
processeur de langage	language processor
processus	process
processus d'élaboration d'un logiciel	software development process
processus séquentiels	sequential processes
produit de base	baseline
produit logiciel	software product
progiciel d'essai et de vérification	scenario
programmable	programmable
programmation	programming
programmation collective	egoless programming
programmation modulaire	modular programming
programmation orientée vers l'objet	object oriented programming
programmation structurée	structured programming
programme de construction	build
programme de gestion	driver
	handler
programme de gestion (logiciel)	device driver
programme de gestion d'essai	test driver

TERME FRANCAIS	ENGLISH TERM
programme de projection	map program
programme de traçage	tracer
programme de vol opérationnel (OFP)	operational flight programme (OFP)
programme directeur	executive routine
programme machine	computer programme
programme objet	object programme
programme récurrent	recursive routine
programme-source	source programme
programme structuré	structured program
programme superviseur	supervisory program
programmer	programme
programmeur	programmer
programmeur-assistant	back-up programmer
programmeur de système	systems programmer
PROM	prom
protection	protection
protection de programme	program protection
protocole	protocol
pseudo-code	pseudo code
pseudonyme	alias
puce	chip
qualité	quality
qualité d'un logiciel	software quality
rapport d'essai	test report
rapport sur l'état	status accounting
rapport sur un problème logiciel (SPR)	software problem report (SPR)
reconfiguration	reconfiguration
redondance	redundancy
registre	register
registre d'index	index register

TERME FRANCAIS	ENGLISH TERM
registre de base	base registers
registre de longueur	length register
reliure statique	static binding
rendez-vous	rendez-vous
renvoi	connection
répartiteur	dispatcher
réponse à une commande	command response
reprise sur incident	failure recovery
réseau	network
réseau d'ordinateurs	computer network
réseau de Pétri	petri net
responsable de la bibliothèque des logiciels	software librarian
ressource	resource
ressources informatiques	computer resources
restauration	restore
restructuration dynamique	dynamic restructuring
résumé analytique	abstraction
résumé de programme machine	computer program abstract
réveil	wakeup
revue de projet	walk-through
revue de projet structurée	structured walkthrough
robustesse	robustness
routine	routine
routine de traçage	trace routine
rythmeur	timer
schéma fonctionnel	block diagram
section critique	critical section
sécurité	security
sécurité multiniveau	multilevel security
segment	segment (n)

TERME FRANCAIS	ENGLISH TERM
segment de recouvrement	overlay
segmentation	segmentation
segmenter	segment (v)
sémantique	semantics
sémaphore	semaphore
service libre	free
signal discret	discrete signal
simulateur	simulator
simulation	simulation
sondage de programme	program instrumentation
sortie	exit
sous-programme	subprogram
sous-routine	subroutine
sous-système	subsystem
spécification	specification
spécification d'interface	interface specification
spécification de conception	design specification
spécification de performance	performance specification
spécification de programme	program specification
spécification des exigences	requirements specification
spécification fonctionnelle	functional specification
spécification formelle	formal specification
stabilité	stability
structure de contrôle	control structure
structure de contrôle conditionnel	conditional control structure
structure de données	data structure
support	media
symbole mnémonique	mnemonic symbol
synchro	synchro
synchrone	synchronous

TERME FRANCAIS	ENGLISH TERM
synchronisation	synchronization
syntaxe	syntax
syntaxique	syntactic
synthèse de programme	program synthesis
système	system
système automatisé de vérification	automated verification system
système d'exploitation (SE)	operating system (OS)
système d'exploitation à utilisateurs multiples	multi-user operating system
système de commande et de contrôle	command and control systems
système de désynchronisation des entrées-sorties	spooling
système dernier entré-premier sorti (LIFO)	last in-first out
système expert	expert system
système informatique	computer system
système informatique intégré	embedded computer system
système ouvert	open system
système premier entré, premier sorti (FIFO)	first in-first out (FIFO)
table	table
table de décision	decision table
table de Karnaugh	karnaugh map
tâche	task
tampon	buffer
taux de défaillance	failure rate
temps d'accès	access time
temps d'exécution	execution time
temps de réponse	response time
temps de rotation	turn around time
temps système	overhead
terminal	terminal



TERME FRANCAIS	ENGLISH TERM
test d'aptitude	aptitude test
test de performance	qualification testing
test de régression	regression testing
test intégré (BIT)	built-in-test (BIT)
testabilité	testability
tête	head
théorie de la durée d'exécution	execution time theory
tolérance	tolerance
trace	trace (n)
tracer	trace (v)
traducteur	translator
traitement de liste	list processing
traitement de signal	signal processing
traitement par chaînes	string processing
traitement réduit	graceful degradation
tranche de temps	time Slice
translation	relocation
travail	job
troncature	truncation
type de données	data type
unité centrale (UC)	central processing unit (CPU)
unité de calcul	arithmetic unit
unité fonctionnelle	functional unit
unité remplaçable sur place (LRU)	line replaceable unit (LRU)
usine à logiciels	software factory
utilisation	utilization
validation	validation
validité de l'essai	test validity
variable	variable
vérification	audit

TERME FRANCAIS	ENGLISH TERM
	verification
vérification de code	code audit
vérification de la configuration	configuration audit
vérification des exigences	requirements analysis
vérification descendante	top-down testing
vérification et validation par un tiers	independent verification and validation
version	version
vidage	dump
violation de protection	protection violation
virgule flottante	floating-point
viseur de casque	helmet mounted sight
visuel de casque	helmet mounted display
word	word
zone commune	common area

CHAPTER/CHAPITRE 5LIST OF ABBREVIATIONSLISTE DES ABBREVIATIONS

ENGLISH/ ANGLAIS	MEANING/SIGNIFICATION	MEANING/SIGNIFICATION	FRENCH/ ANGLAIS
A/D	Analogue to digital	Analogue à numérique	A/N
ATE	Automatic test equipment	Equippement de test automatique	ATE
ALU	Arithmetic logic unit	Unité arithmétique et logique	UAL
BIT	Built-in test	Test intégré	BIT
BITE	Built-in test equipment	Equipement de test intégré	BITE
CRT	Cathode ray tube	Tube cathodique	CRT
CDU	Control display unit	Post de commande et d'affichage	CDU
CPU	Central processing unit	Unité centrale	U.C.
D/A	Digital to analogue	Numérique à analaogique	N/A
DMA	Direct memory access	Accès direct mémoire	ADM
FIFO	First in - first out	Premier entré - premier sorti	FIFO
GSE	Ground support equipment	Matériel de servitude au sol	MSS
HDD	Head down display	Visualisation tête basse	VTS
HMD	Helmet mounted display	Visuel de casque	
HMS	Helmet mounted sight	Viseur de casque	
HOL	High order language	Language évolué	IHN
HUD	Head up display	Visualisation tête haute	VTH
ISA	Instruction set architecture	Machine virtuelle	
I/O	Input/Output	Entrée - Sortie	E/S
LCD	Liquid crystal display	Visualisation à cristeaux liquides	LCD
LED	Light emitting diode	Diode électroluminescente	LED
LIFO	Last in - first out	Dernier entré - Premier sorti	LIFO

ENGLISH/ ANGLAIS	MEANING/SIGNIFICATION	MEANING/SIGNIFICATION	FRENCH/ ANGLAIS
LSP	Least significant bit	Bit de poids faible	LSB
LSI	Large scale integration	Intégration à grande échelle	LSI
LRU	Line replaceable unit	Unité remplaçable en piste	URP
MSB	Most significant bit	Bit de poids fort	MSB
MSI	Medium scale integration	Intégration à moyenne échelle	MSI
OFP	Operational flight program	Programme embarqué	
OS	Operating system	Système d'exploitation	CS
PCM	Pulse code modulation	Modulation par impulsion codée	MIC
PROM	Programmable read only memory	Mémoire morte programmable	PROM
RAM	Random access memory	Mémoire à accès aléatoire	RAM
ROM	Read only memory	Mémoire morte	ROM
TDM	Time division multiplex	Multiplexage à partage du temps	
TSP	Twisted-shielded pair	Paire torsadée blindée	
VHSIC	Very high speed integration circuit	Circuit intégré à très grande vitesse	VHSIC
VLSI	Very large scale integration	Intégration à très grande échelle	