|  |  |
| --- | --- |
| *voenmeh* | МИНОБРНАУКИ РОССИИ  федеральное государственное бюджетное образовательное учреждение  высшего профессионального образования  **«Балтийский государственный технический университет «ВОЕНМЕХ» им. Д.Ф. Устинова»**  **(БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова»)** |
| БГТУ.СМК-Ф-4.2-К5-01 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Факультет |  | И | |  |  | Информационные и управляющие системы |
|  |  | шифр | |  |  | наименование |
| Кафедра |  | И9 | |  |  | Систем управления и компьютерных технологий |
|  |  | шифр | |  |  | наименование |
|  |  |  |  | | | |

|  |
| --- |
| Отчет |
| Выбор метода создания и структурирования |
| электронной обучающей системы |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Выполнил студент группы | | | |  | | И9М31 |
| Разомазов А.В | | | | | | |
| Фамилия И.О. | | | | | | |
| **РУКОВОДИТЕЛЬ** | | | | | | |
| Черкасов О.Ф. | |  |  | | | |
| Фамилия И.О. Подпись | | | | | | |
| Оценка |  | | | |  | |
| «\_\_\_\_\_» |  | | | | 2018 г. | |

САНКТ-ПЕТЕРБУРГ

2018 г.

Оглавление

[Введение 3](#_Toc517855533)

[Выбор метода построения ЭОС и построение общей структуры системы 6](#_Toc517855534)

[Обоснование выбора языка программирования 8](#_Toc517855535)

[Обоснование выбора инструментальных возможностей программной реализации задачи 11](#_Toc517855536)

[Разработка общего алгоритма 14](#_Toc517855537)

[Список используемой литературы 19](#_Toc517855538)

# Введение

Автоматизированные обучающие системы представляют собой программно-технический комплекс, включающие в себя методическую, учебную и организационную поддержку процесса обучения, проводимого на базе информационных технологий. Автоматизированные обучающие системы своим появлением способствовали расширению наших представлений о возможностях электронно-вычислительных машин (ЭВМ) прежде всего потому, что открыли новые перспективы в деле совершенствования образования.

Применение ЭВМ в системе образования явилось следствием появления новых направлений как в использовании технических средств обучения, так и в методах и формах самого процесса обучения. В настоящее время обучающие системы используются во многих образовательных учреждениях.

Рассматривая электронные обучающие системы (ЭОС) как средства автоматизации обучения, повышения качества образования, а также возможность получения знаний без контакта обучающегося и преподавателя, стоит сразу же отметить: универсальных систем обучения не существует. Проблема заключается в том, что каждый обучаемый человек индивидуален, имеет значение специфика направления учебного курса и сам принцип обучения. Проектируя модель ЭОС необходимо сразу определить задачи, такие как:

* уяснение задачи, в результате чего определяется проблемная ситуация, устанавливается структура данных и неизвестных;
* разработка математической модели ситуации;
* составление плана решения (в виде текста, блок-схемы, графа и т.п.);
* выбор (разработка) алгоритма, наиболее эффективного для применения компьютера;
* составление программы в соответствии с алгоритмом;
* отладка и тестирование программы;
* проверка пригодности модели;
* оценка эффективности принятого плана решения;
* организация взаимодействия с компьютером.

При разработке ЭОС требуется сразу определить цели, которые данная система будет решать в рамках образовательного процесса. Важно отметить, что лучшие обучающие системы не насыщенны сложными данными, формулами, схемами, совмещены с основным обучением, а сам процесс нередко дополняют забавные фразы, шутки, игры. Анализ и практика показывают, что положительные эффекты обучения наиболее отчетливо проявляются при:

* изучении базиса дисциплины, ее сложных закономерностей и алгоритмов, динамических процессов;
* реализации игр и имитаций;
* организации исследовательских и тренирующих процессов;
* автоматизации самоконтроля, контроля, оценки обучения;

Стоит отметить, что существуют различные, по типу работы и поставленной задаче, системы обучения. Все эти критерии показывают, что разработка ЭОС включает в себя не только технические методы реализации, выбор типа и структуры системы, но и методы предъявления учебной информации. Усложняется разработка тем, что стандартов на построение подобных систем нет.

Целью данной работы является разработка практических рекомендаций по формированию автоматизированных обучающих систем, выработка основных методов по созданию и организации структуры ЭОС, в которой разработчик имел бы больше возможностей по управлению внутренней структуры и имел бы больше возможностей влиять на качество системы.

В ходе работы решаются следующие задачи:

* Проведение сравнительного анализа и классификации основных существующих методов разработки ЭОС. Практическое освоение перспективных методов разработки, моделей систем, способов реализации и вспомогательных средств.
* Анализ полученных данных и изучение лучших методов разработки и структурирования обучающих систем. Выбор программных средств для реализации ЭОС.
* Разработка типовой структуры ЭОС.
* Создание конкретной ЭОС с целью отладки правильности принятых решений на конкретных примерах.
* Анализ результатов, выработка рекомендаций на основе результатов, предложение по усовершенствованию системы.

Объектом исследования являются методы создания и структурирования автоматизированных обучающих систем, выработка практических рекомендаций по разработке.

В качестве методов исследования были применены научные методы: классификация, анализ, моделирование, системный анализ.

Прикладная ценность и новизна исследования заключаются в том, что не существует единых стандартов по разработке и структуризации ЭОС, а также критериев качества подобных программных продуктов. Знание технологии создания ЭОС поможет сократить временные затраты и оптимизировать усилия разработчиков данного вида программных продуктов.

# Выбор метода построения ЭОС и построение общей структуры системы

Ввиду возложения на ЭОС выполнения таких функций, как создание и модификация учебных курсов, включая сопутствующие материалы, файлы и информацию; организация и контроль процесса обучения; проведение обучения; анализ результатов обучения (выполненных тестов), для построения системы выбрана структура тестирующей замкнутой ОС (тестирование с закрытыми вопросами). В качестве алгоритма выбран нелинейный циклический алгоритм с обратной связью. Обоснование выбора объясняется требованиями технического задания.

Таким образом, учебный процесс предполагается организовывать следующим образом: преподаватель организует работу обучаемых с учебно-методическими средствами, ЭОС контролирует работу обучаемого с учебным материалом.

Обучаемый – основной пользователь учебной системы. Пользуясь средствами автоматизации, может вызывать необходимую информацию, тренироваться в решении задач, производить вычисления, контролировать свои знания. Учебно-методическое обеспечение представляет собой совокупность учебного материала, методических указаний, библиотеки курсов.

Предполагается, что разработка будет внедряться в традиционный процесс обучения. Учебная система предполагает наличие обратной связи в триаде «Педагог – Обучающая программа – Обучаемый» (рисунок 1.10).

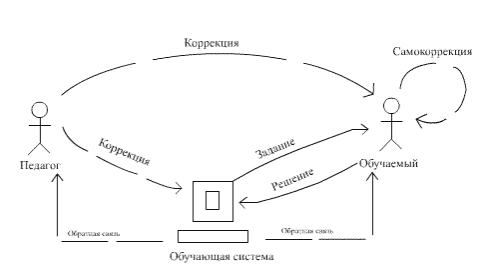
****

Рисунок 1.10 – Схема взаимодействия объектов«Педагог – Обучающая программа – Обучаемый»

Внутренняя обратная связь - это информация, которая поступает от обучающей программы к ученику в ответ на его действия при выполнении упражнений. Она предназначена для самокоррекции учеником своей учебной деятельности. Внутренняя обратная связь дает возможность ученику сделать осознанный вывод об успешности или ошибочности учебной деятельности. Она побуждает ученика к рефлексии, является стимулом к дальнейшим действиям, помогает оценить и скорректировать результаты учебной деятельности.

Консультативная обратная связь может быть разной: помощь, разъяснение, подсказка, наталкивание со стороны преподавателя.

Результативная обратная связь также может быть различной: от «верно – неверно» до демонстрации правильного результата или способа действия.

Такая схема работы триады «Педагог - Обучающая программа - Обучаемый» обеспечивает каждому студенту индивидуальный, наиболее удобный график его работы, а преподавателю – уменьшение времени на проверку работ студентов (этот процесс автоматизирован) и улучшение контроля над действиями каждого обучаемого.

В настоящем случае, ОС поэтапно выполняет расчетную часть курсового проекта, при этом для перехода пользователя на каждый следующий этап необходимо выполнение теста. Для обеспечения обратной связи с обучаемым, учебная система автоматизированного проектирования содержит информационные поля и основные расчетные формулы

**Обоснование выбора языка программирования**

В качестве языка программирования для реализации данного дипломного проекта был выбран С++ — компилируемый статически типизированный язык программирования общего назначения. Именно этот язык наиболее подходит для автоматизации информационной системы учёта работы строительной фирмы, так как С++ успешно используется во многих областях приложения, далеко выходящих за указанные рамки и позволяет создавать как простые приложения и утилиты, так и коммерческие системы.

Поддерживает такие парадигмы программирования как процедурное программирование, объектно-ориентированное программирование, обобщённое программирование, обеспечивает модульность, раздельную компиляцию, обработку исключений, абстракцию данных, объявление типов (классов) объектов, виртуальные функции. Стандартная библиотека включает, в том числе, общеупотребительные контейнеры и алгоритмы. C++ сочетает свойства как высокоуровневых, так и низкоуровневых языков.В сравнении с его предшественником — языком C, — наибольшее внимание уделено поддержке объектно-ориентированного и обобщённого программирования.

C++ широко используется для разработки программного обеспечения, являясь одним из самых популярных языков программирования. Область его применения включает создание операционных систем, разнообразных прикладных программ, драйверов устройств, приложений для встраиваемых систем, высокопроизводительных серверов, а также развлекательных приложений (игр). Существует множество реализаций языка C++, как бесплатных, так и коммерческих и для различных платформ. Например, на платформе x86 это GCC, Visual C++, Intel C++ Compiler, Embarcadero (Borland) C++ Builder и другие. Синтаксис C++ унаследован от языка C. Одним из принципов разработки было сохранение совместимости с C. Тем не менее, C++ не является в строгом смысле надмножеством C; множество программ, которые могут одинаково успешно транслироваться как компиляторами C, так и компиляторами C++, довольно велико, но не включает все возможные программы на C.

Также C++ — чрезвычайно мощный язык, содержащий средства создания эффективных программ практически любого назначения, от низкоуровневых утилит и драйверов до сложных программных комплексов самого различного назначения.

В частности, поддерживаются различные стили и технологии программирования, включая традиционное директивное программирование, ООП; имеется возможность работы на низком уровне с памятью, адресами, портами; возможность создания обобщённых алгоритмов для разных типов данных, их специализация и вычисления на этапе компиляции, используя шаблоны.

С++ является ISO-стандартизированным языком программирования.

Ключевым понятием С++ является класс. Класс - это определяемый пользователем тип. Классы обеспечивают упрятывание данных, их инициализацию, неявное преобразование пользовательских типов, динамическое задание типов, контролируемое пользователем управление памятью и средства для перегрузки операций. В языке С++ концепции контроля типов и модульного построения программ реализованы более полно, чем в С. Кроме того, С++ содержит усовершенствования, прямо с классами не связанные: символические константы, функции-подстановки, стандартные значения параметров функций, перегрузка имен функций, операции управления свободной памятью и ссылочный тип. В С++ сохранены все возможности С эффективной работы с основными объектами, отражающими аппаратную "реальность" (разряды, байты, слова, адреса и т.д.). Это позволяет достаточно эффективно реализовывать пользовательские типы.

Язык C++ явился мощным и стремительным рывком в развитии программирования. C++ и по сей день занимает господствующее положение среди языков программирования в мире. Огромное множество профессиональных программистов использует именно его при разработке разного рода проектов. Также С++ тесно связан со средой разработки Microsoft Visual C++, именно этот язык позволяет создавать простые и понятные приложения и компоненты.

С++ - это также универсальный язык программирования, задуманный так, чтобы сделать программирование более приятным для серьезного программиста. За исключением второстепенных деталей С++ является надмножеством языка программирования C. Помимо возможностей, которые дает C , С++ предоставляет гибкие и эффективные средства определения новых типов. Используя определения новых типов, точно отвечающих концепциям приложения, программист может разделять разрабатываемую программу на легко поддающиеся контролю части. Такой метод построения программ часто называют абстракцией данных. Информация о типах содержится в некоторых объектах типов, определенных пользователем. Такие объекты просты и надежны в использовании в тех ситуациях, когда их тип нельзя установить на стадии компиляции. Программирование с применением таких объектов часто называют объектно-ориентированным. При правильном использовании этот метод дает более короткие, проще понимаемые и легче контролируемые программы.

C++ компилируется непосредственно в машинный код, что позволяет ему быть одним из самых быстрых в мире языков. Также C++ подразумевает, что программист знает, что делает, и позволяет невероятное количество возможностей, ограниченных только лишь фантазией. Таким образом, проверка типов данных может выполняться во время компиляции или во время выполнения. И это ещё раз доказывает гибкость С++.

# Обоснование выбора инструментальных возможностей программной реализации задачи

Для реализации поставленной задачи мною была выбрана среда разработки Microsoft Visual Studio 2013 , так как она наиболее подходит для создания данного проекта, а именно автоматизированной информационной системы, имеет понятный интерфейс и большой набор функций,инструментов.

Microsoft Visual Studio — линейка продуктов компании Майкрософт, включающих интегрированную среду разработки программного обеспечения и ряд других инструментальных средств. Данные продукты позволяют разрабатывать как консольные приложения, так и приложения с графическим интерфейсом, в том числе с поддержкой технологии Windows Forms, а также веб-сайты, веб-приложения, веб-службы как в родном, так и вуправляемом кодах для всех платформ, поддерживаемых Microsoft Windows, Windows Mobile, Windows CE, .NET Framework, Xbox, Windows Phone .NET Compact Framework и Microsoft Silverlight.

Visual Studio включает в себя редактор исходного кодас поддержкой технологииIntelliSenseи возможностью простейшегорефакторинга кода. Встроенныйотладчикможет работать как отладчик уровня исходного кода, так и как отладчик машинного уровня. Остальные встраиваемые инструменты включают в себя редактор форм для упрощения создания графического интерфейса приложения, веб-редактор, дизайнерклассови дизайнерсхемы базы данных. Visual Studio позволяет создавать и подключать сторонние дополнения (плагины) для расширения функциональности практически на каждом уровне, включая добавление поддержки системконтроля версий исходного кода (как например,SubversionиVisual SourceSafe), добавление новых наборов инструментов (например, для редактирования и визуального проектирования кода напредметно-ориентированных языках программирования или инструментов для прочих аспектов процесса разработки программного обеспечения.

Каждая новая версия программы состоит из новейших инструментов и технологий, позволяющих разрабатывать приложения с учетом особенностей и положительных моментов современных платформ. Например, Visual Studio 2012 может поддерживать более ранние версии, в том числе Windows XP и Windows Server 2003. При этом разработчикам открыта дорога к созданию новых и модернизации уже существующих приложений, предназначенных для ранних версий ОС Windows. Стоит отметить, что в процессе использования поддерживаемых системой вариантов исходные файлы, проекты и решения в программе Visual Studio будут работоспособными, но исходный код может нуждаться в изменениях.

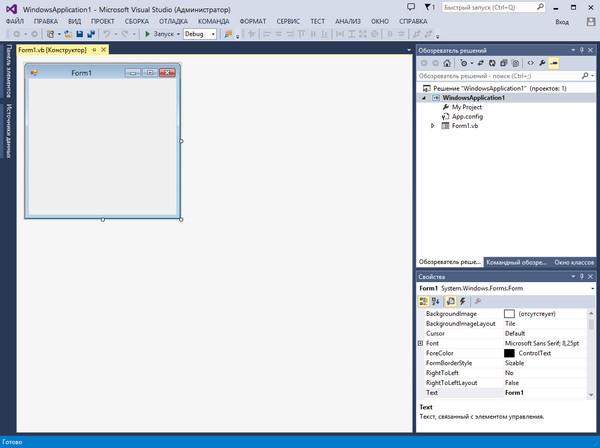


Рисунок 1 - Среда разработки MS Visual Studio 2013

Средства, входящие в состав Visual Studio для Windows, можно использовать для создания привлекательных инновационных приложений для Магазина Windows в среде Windows 8.1. Эти средства включают полнофункциональный редактор кода, мощный отладчик, специальный профилировщик и широкие возможности языковой поддержки, которые позволяют выполнять сборку приложений, написанных на языках HTML5/JavaScript, C++, C# и Visual Basic. В состав Visual Studio для Windows также входит имитатор устройств, который можно использовать для тестирования приложений Магазина Windows на устройствах различных видов.

Выбрана данная среду программирования, потому что она позволяет работать с данными различного типа. Имеет понятный интерфейс, широкий набор инструментов для разработки приложений который расширяется. Интеграция со многими языками программирования и программами, быстро компилирует программный код и выявляет ошибки.

# Разработка общего алгоритма

Следуя хорошему стилю программирования, программа будет разделена на несколько модулей, каждый из которых будет выполнять свою задачу.

С помощью главного модуля программы пользователь будет осуществлять навигацию по обучающей системе.

Каждый этап главного модуля должен содержать информационные подсказки пользователю, расчетные формулы, полученные результаты расчетов, и тесты на общую проверку знаний по дисциплине. Так же главный модуль программы будет содержать алгоритм проверки на количество запущенных одновременно копий обучающей системы, что не позволит запускать больше одной копии на одном компьютере. После завершения работы программы, введенный пользователем вариант исходных данных будет сохранен и загружен при следующем запуске программы, что удобно для обучающихся. Электродвигатели, будут загружаться из базы данных двигателей.

Дополнительно к этому в главном модуле будут отдельными пунктами представлены:

* принципиальная электрическая схема следящей системы, которую можно будет посмотреть с помощью программы просмотра изображений *Windows*,что очень удобно из-за больших размеров схемы;
* конспект лекций по дисциплине и инструкция пользователя в формате *html*.

На этапе ввода начальных данных пользователь сможет выбрать свой вариант данных из базы вариантов или ввести собственные начальные данные. Этот алгоритм будет находиться в отдельном модуле.

Иной модуль будет выполнять иную задачу, в этом модуле будет содержаться алгоритм тестирования.

По десяткам дисциплин студенты во время самостоятельной работы выполняют задания и тесты на ЭВМ. Информация о работе студентов является важной составной частью оценки успеваемости студентов. Преподаватель любой дисциплины может самостоятельно создавать и модифицировать задания и тесты, организовывать доступ студентов к ним и анализировать информацию о работе студентов.

Предлагаемые тесты традиционно построены на основе закрытых режимов диалога, где превалирует тип диалога «меню» (простой выборочный ответ). Благодаря простоте технической реализации этот тип диалога являетсяидеальной основой для разработки контрольно-обучающих подсистем, осуществляющих проверку минимального объема знаний студентов по дисциплине. Известно также, что этот тип контрольно обучающих подсистем имеет ряд существенных недостатков:

* высокий уровень подсказки;
* ограниченность применения по ряду учебных фрагментов;
* возможность «вычисления» правильного ответа логическим путем без твердого знания учебного материала;
* культивирование специфического мыслительного процесса угадывания ответов.

Другой вид тестов заключается в применении открытых вопросов, то есть таких, на которые обучающийся должен сам формулировать ответ без каких-либо синтаксических ограничений. В этом случае система должна уметь распознавать смысл свободно конструируемого ответа и формировать соответствующие реплики и разъяснения по результатам анализа.

В настоящей САПР, контрольно-обучающей подсистемой является модуль тестирования, построенный по принципу закрытых вопросов. Использование в системе открытых вопросов не является необходимостью, так как САПР снабжен информационными полями, содержащими информацию необходимую для расчета и проектирования маломощных следящих систем, и результатом работы с САПР является курсовой проект по дисциплине. Для подробного изучения дисциплины, обучаемый должен обратиться к курсу лекций по «Проектированию маломощных электромеханических следящих. Данный курс лекций включен в разрабатываемую учебную САПР.

Модуль тестирования разработан с целью проведения тестов на знание теории в процессе выполнения курсового проекта в ОС.

Упрощенная блок-схема модуля тестирования представлена на рисунке 1.14

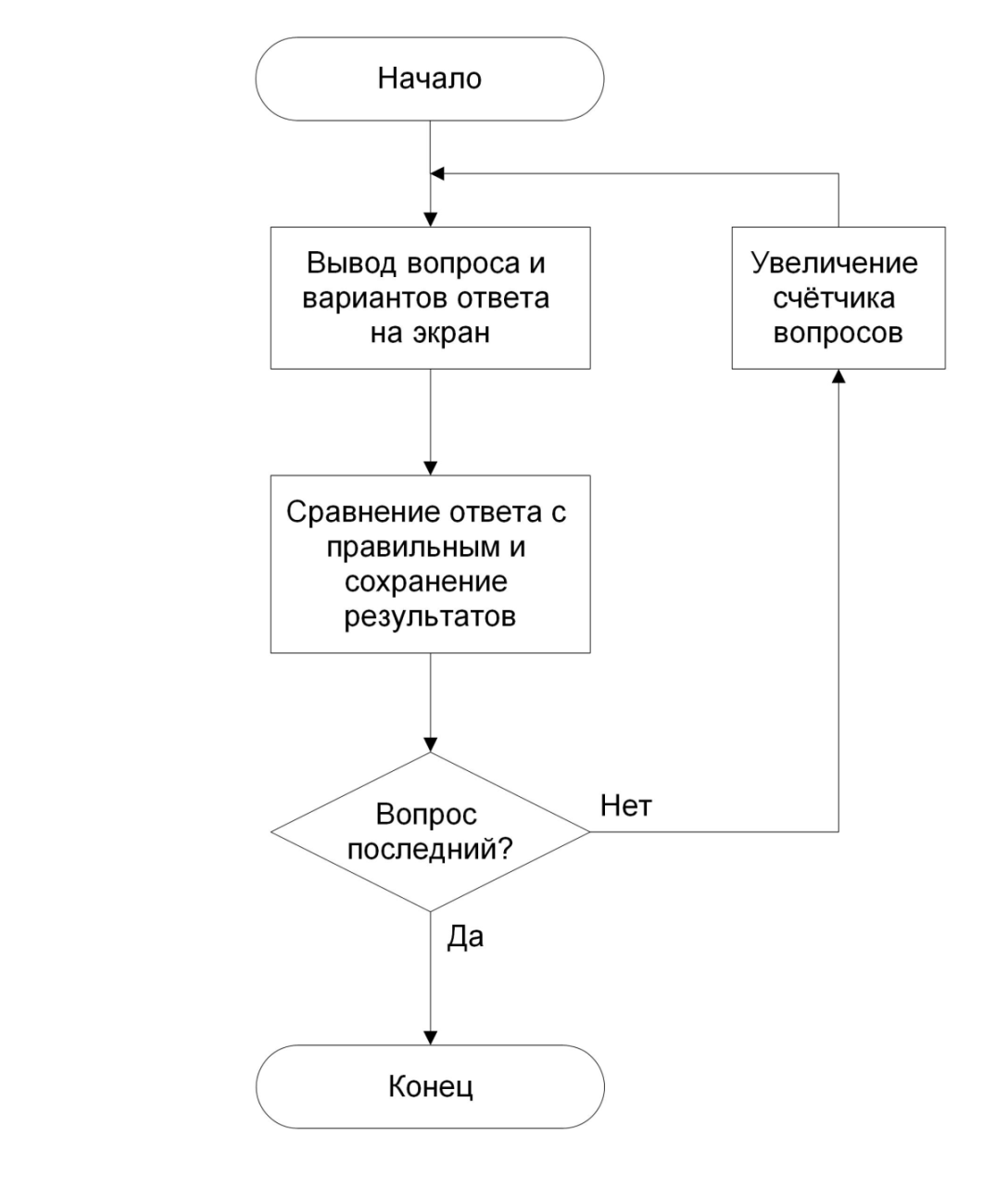


Рисунок 1.14 – Упрощенная блок-схема модуля тестирования

Благодаря наличию практически на каждом предприятии обширной локальной сети решено использовать отдельный сервер и два приложения для работы с ним – редактор и сама ЭОС. Программа будет разделена на несколько модулей, каждый из которых будет выполнять свою задачу. С помощью главного модуля программы пользователь будет осуществлять навигацию по обучающей системе. Каждый этап главного модуля будет содержать информационные подсказки пользователю, полученные результаты, и тесты на общую проверку знаний. Так же главный модуль программы будет содержать алгоритм проверки на количество запущенных одновременно копий обучающей системы, что не позволит запускать больше одной копии на одном компьютере. После завершения работы программы, введенный пользователем вариант исходных данных будет сохранен и загружен при следующем запуске программы, что удобно для обучающихся. Редактор реализован отдельным приложением, имеет простые функции «добавить/удалить». Учебный материал добавляется в виде .docx файлов. Тестирующий модуль представлен в виде тестов с принципом закрытых вопросов, работающий с .mdb. Вся информация добавляется в папку «study», в которой уже редактором создаются папки по темам. Запускаются программы исполняемым приложением .exe. Последний этап разработки – проверка функциональности и отладка ПО.

По мере выполнения работы были выработаны собственные рекомендации по разработке структуры обучающей системы и реализации некоторых технических решений, благодаря которым сама программа обладает полным функционалом как обучающее ПО, при этом имея простой и понятый интерфейс. Так как разработка обучающей системы до сих пор ведется, принято решение сформировать стандарт на построение подобных систем. Улучшение модуля тестирования, редактирования, а также представления информации поставлены как дальнейшие цели по повышению качества обучающей системы. Создание тестов по принципу открытых вопросов, то есть таких, на которые обучающийся должен сам формулировать ответ без каких-либо синтаксических ограничений прорабатывается на данный момент.

**Список используемой литературы**

1. Рассел, Стюарт, Норвиг, Питер. Искусственный интеллект: современный подход: пер. с англ. – 2-е изд. – М.: ИД «Вильямс», 2007. – 1408 с.
2. Талызина Н.Ф. Теоретические проблемы программированного обучения. – М.: Изд-во МГУ, 1969. – 133 с.
3. Загорулько Ю.А. Методологические проблемы построения онтологий для портала научных знаний // Когнитивные исследования: сб. науч. тр. / РАН, Ин-т психологии, Казан. гос. ун-т им., Ассоц. когнитивных исслед.; отв. ред. В.Д. Соловьев, Т.В. Черниговская. – М. Ин-т психологии РАН, 2006.
4. Голенков В.В., Гулякина Н.А. Открытый проект, направленный на создание технологии компонентного проектирования интеллектуальных систем // В кн.: Междунар. научн.-техн. конф. «Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2013)»: материалы III Междунар. научн.-техн. конф. (Минск, 21–23 февраля 2013 г.). – Минск: БГУИР, 2013.
5. Рыбина Г.В., Паронджанов С.С. Моделирование процессов взаимодействия интеллектуальных агентов в многоагентных системах // Искусственный интеллект и принятие решений. – 2012. – № 1.
6. Голенков В.В., Емельянов В.В., Тарасов В.Б. Виртуальные кафедры и интеллектуальные обучающие системы // Новости искусственного интеллекта. – 2001. – № 4
7. Трембач В.М. Системы управления базами эволюционирующих знаний для решения задач непрерывного образования: монография. – М.: МЭСИ, 2013.