**Введение**

На курсовое проектирование была поставлена задача, на тему: Разработка электронного средства обучения «AutoCad».

Цель курсового проекта заключается в изучении тем по «AutoCad».

Создаваемое приложение будет рассчитано для учащихся которые изучают «AutoCad», а также для проверки знаний по этой теме в виде заданий и теста.

Приведем краткое описание разделов пояснительной записки.

Первый раздел носит название “Анализ задачи”. В нем вы сможете ознакомиться с постановкой задачи, которая включает в себя: исследование предметной области поставленной задачи, определение ее организационно-экономической сущности. Также в этом разделе вы сможете узнать о том, как данная задача решается в настоящее время. Все входные и выходные данные тоже будут описаны в первом разделе. В подразделе “Инструменты разработки” будет рассмотрена среда, в которой создается данный курсовой проект. Здесь также будут установлены минимальные и оптимальные требования к аппаратным характеристикам, обеспечивающим правильное функционирование поставленной задачи.

В разделе “Проектирование задачи” будут рассмотрены основные аспекты разработки программного продукта. Здесь можно будет узнать об организации данных в контексте среды разработки. В данном разделе будет четко описан пользовательский интерфейс, составлены алгоритмы процесса обработки информации, описана разработка системы справочной информации.

“Реализация задачи” – это третий раздел пояснительной записки, в котором описываются все элементы и объекты, которые будут использованы при реализации данного приложения. В этом разделе будут четко описаны функции пользователя и их структура. Здесь можно будет найти таблицу, в которой будет представлена полная аннотация файлов используемых в данном проекте.

Четвертый раздел – “Тестирование”. В нем будет описано полное и функциональное тестирование данной программы, т.е. будет оттестирован каждый пункт меню, каждая операция, которая выполняется приложением. Будут смоделированы все возможные действия пользователя при работе с программой, начиная от запуска до выхода.

В разделе “Применение” будет описано назначение, область применения, среда функционирования курсовой программы. Также в нем будет описано использование справочной системы.

“Заключение” будет содержать краткую формулировку задачи, результаты проделанной работы, описание использованных методов и средств, описание степени автоматизации процессов на различных этапах разработки.

В “Списке используемых источников” будет приведен список используемой литературы, нормативно- техническую и другую документацию.

В приложениях к пояснительной записке будет приведен листинг программы с необходимыми комментариями.

Схема работы системы будет представлена в графической части.

**1. Анализ задачи**

**1.1 Постановка задачи**

Темой данного курсового проекта является Разработка электронного средства обучения «AutoCad».

Периодичность использования данного программного продукта неограниченна. Преподаватель может в любое время выдать ученикам приложение для изучения теории по темам «AutoCad», а также для проверки знаний учеников по этим темам.

Целью данного курсового проекта является разработка программного продукта, который будет содержать в себе теорию «AutoCad» и задания по этим темам, а также тест.

За основу программы возьмем из соответствующих источников (сайтов, статей) наиболее полную, достоверную и точную информацию. При реализации проекта необходимо его сделать простым и максимально доступным для пользователей.

Приложение должно располагать своим интерфейсом и функционалом к пользователю, чтобы ему было комфортно.

Данный проект должен стать общедоступным для всех пользователей. В поставленной задаче необходимо реализовать максимально простой пользовательский интерфейс, позволяющий использовать проект пользователю, не обладающему дополнительными знаниями электронно- вычислительной техники.

В данном проекте необходимо реализовать максимально просто пользовательский интерфейс, позволяющий использовать проект пользователю с максимальным комфортом и без затруднений.

Разрабатываемый программный продукт должен позволять выполнять следующие действия:

- обеспечить быстрый поиск необходимой информации;

- прохождение теста для закрепления знаний;

- просмотр практических заданий;

В разрабатываемой программе будут использоваться три вида данных: входная, выходная и постоянная информация.

К входной информации можно отнести вводимые пользователем значения,

например, ответы на вопросы при прохождении теста. К выходной – результат прохождения всех заданий.

Постоянной информацией в проекте будут являться текстовые файлы, картинки и др.

**1.2 Инструменты разработки**

Для разработки данного проекта будет выбрана среда Delphi 11, так как

это самое удобная и доступная среда разработки на данный момент. Delphi 11-язык –программирования, относящийся к классу RAD- (Rapid Application Development – «Средство быстрой разработки приложений») средств CASE –тeхнологии. Delphi 11 сделал разработку приложений дляWindows быстрым иприятным процессом. Теперь разрабатывать сложные и интересные проекты можно только одним человеком, использующим Delphi 11

Интерфейс Windows обеспечивает полное перенесение CASE-технологии в интегральную систему поддержки работ по созданию прикладной системы на всех фазах жизненного цикла работы и проектирования системы.

Delphi 11 обладает широким набором возможностей, начиная от проектировщика форм и заканчивая поддержкой всех форматов популярных баз данных. Среда устраняет необходимость программировать такие компоненты Windows общего назначения, как метки, программы и даже диалоговые панели. Работая в Windows, можно видеть одинаковые «объекты» во многих разнообразных приложениях. Диалоговые панели (например, Choose File и Save File) являются примерами многократно-используемых компонентов, встроенных непосредственно в Delphi 11, который позволяет приспособить эти компоненты к имеющийся задаче, чтобы они работали именно так, как требуется создаваемому приложению. Также здесь имеются предварительно-определенные визуальные и не визуальные объекты.

Три основные части разработки интерфейса следующие: проектирование панели, проектирование диалога и представление окон. Для общего пользовательского доступа также должны учитываться условия применения архитектуры прикладных систем.

Сегодня появилась реальная возможность с помощью моделирования на современных многофункциональных средствах обработки и отображения информации таких как Delphi 11 конкретизировать тип и характеристики используемых информационных моделей, выявить основные особенности будущей деятельности операторов, сформулировать требования к параметрам аппаратно-программных средств интерфейса взаимодействия и т.д. Delphi 11 позволяет создать различные виды программ: консольные приложения, оконные приложения, приложения для работы с Интернетом и базами данных. То есть, Delphi 11 является не только средствами для работы с языком программирования Паскаль, но дополнительные инструменты, призванные для максимального упрощения и ускорения создание приложений.

К дополнительным инструментам можно отнести визуальный редактор форм, благодаря которому можно с легкостью создать полноценную программу, и другие визуальные составляющие разработки программного обеспечения. С Delphi вам не нужно вручную просчитывать расположение каждого элемента интерфейса пользователя, поэтому при разработке программы значительно экономится время.

Выгоды от проектирования в среде Windows с помощью Delphi 11:

* устраняется необходимость в повторном вводе данных;
* обеспечивается согласованность проекта и его реализации;
* увеличивается производительность разработки и переносимость программ.

Ни одно серьезное программное обеспечение не обходится без модуля справочной информации и руководства пользователя. Это придает программе законченный вид и показывает заботу о пользователе. DrExplain – легкий в использовании и функциональный инструмент, упрощающий создание справочных файлов Windows, печать справочных руководств и документации в целом. Программа имеет интуитивно понятный интерфейс. Все созданные проекты можно сохранить в различных форматах:HTML Help, Winhelp и MS Help 2.0 / Visual Studio Help, Browser-based Help, PDF и Word RTF, а также печатной документации при использовании одного и того же проекта. В основном окне программы содержатся оглавление (в виде древовидного списка) и текстовый редактор. Это дает возможность легко ориентироваться в оглавлении, редактировать или перемещать разделы справки без каких-либо проблем. Также утилита позволяет конвертировать help-файлы из одного формата в другой. Кроме приложений для работы с текстом в данном продукте содержатся утилиты для создания скриншотов и редактирования графических файлов.

Для создания инсталлятора будет использоваться мощное и удобное средство - Smart Install Maker. Программа обладает удобным и интуитивно понятным интерфейсом, а также полным набором необходимых функций для создания профессиональных инсталляторов с минимальным размером, высокой степенью сжатия файлов и приятным интерфейсом.

Помимо стандартного минимума, Smart Install Maker позволяет редактировать системный реестр и INI-файлы, создавать программные ярлыки, запускать ассоциируемые и исполняемые файлы, регистрировать новые шрифты и ActiveX компоненты, отображать тексты информации и лицензионного соглашения. Также, с помощью этой утилиты, можно создать мультиязыковые инсталляторы с поддержкой более 20-ти популярных языков мира.

Easy Paint tool SAI - это популярная среди художников программа для рисования (в частности для проработки интерфейса и игры «одевалка»). Удобна своей простотой и в то же время большим функционалом. Программа очень сильно распространена в использовании во всем мире и предпочитается рисующими людьми как один из самых удобных инструментов для создания рисунков. В нем можно сохранить файлы картинок во множестве вариантов (.png, .jpg, .bmp и так далее).

Microsoft Word 2016 – редактор текста для написания документации.

Разработка ведется на ноутбуке Asus. У данного ноутбука следующие

параметры:

* процессор Ryzen 7;
* объем ОЗУ 16 гб;
* объем места на HDD – 512 гб ;
* видеоподсистема 1024х768 точек с глубиной цвета 16 Bit;
* ОС – Windows 10 домашняя.

Как видно разрабатываемое приложение не очень требовательно к аппаратным ресурсам, что, является большим плюсом.

**1.3 Требования к приложению**

На этапе исследования предметной области был установлен целый ряд требований, которые предъявляются к разрабатываемой программе.

По этой причине особенно актуальной становится разработка программного электронного продукта, который способствовал бы и давал возможность изучить и повторить информацию из теоретического курса. Чтобы эта информация лучше усваивалась необходимо подобрать правильный интерфейс.

Требования к интерфейсу: в связи с частым использованием программы она должна быть с приятной цветовой гаммой и понятной для пользователя. Следовательно, каждое окно должно иметь ясную визуальную иерархию своих элементов. Фрагменты текста должны располагаться на экране так, чтобы пользователя было просто и понятно принимать информацию.

Пользователь не должен испытывать какого-либо дискомфорта в плане восприятия информация, отображённой на экране. Объекты (рисунки и символы) не должны быть слишком мелкие. Все окна приложения по возможности должны помещаться на экран полностью, так как использование в процессе работы полос прокруток достаточно неудобно.

На одной форме нельзя допускать избытка и нагромождения данных. Формы должны быть эффектно оформлены согласно тематике разрабатываемого проекта.

Требования к надежности: специальных требований к надежности не предъявляется. Время восстановления после отказа, вызванного сбоем электропитания технических средств (иными внешними факторами), не фатальным сбое операционной системы, не должно превышать 30-ти минут при условии соблюдения условий эксплуатации технических и программных средств.

Время восстановления после отказа, вызванного неисправностью технических средств, фатальным сбоем операционной системы, не должно превышать времени, требуемого на устранение неисправностей технических средств и переустановки программных средств.

Отказы программы вследствие некорректных действий пользователя при взаимодействии с программой через графический интерфейс не должны влиять на конечный результат.

**2. Проектирование задачи**

**2.1 Организация данных**

Проектирование задачи – это очень важный и ответственный этап в разработке любого приложения.

Важным является он вследствие того, что методы, по средствам которых пользователь управляет формами, построены на высокой степени специализации каждого из компонентов.

Необходимым условием при разработке данного приложения является описание организации данных, т.е. логическая и физическая структура данных

в контексте среды разработки. В разрабатываемой программе будут использоваться три вида данных.

Первым видом являются данные, которые будут введены разработчиком на этапе реализации задачи. Сюда можно отнести изображения (иконки), описание, исходные коды сортировок.

Вторым видом данных, используемых в программе, является вводимая пользователем информация. Входной информацией в разрабатываемой программе будут являться данные. Подходящей структурой данных для хранения элементов является: динамический одномерный массив.

Третьим видом данных является результат программы –отсортированные пользовательские данные. Его также относят к отдельному виду, так как ни пользователь, ни разработчик его не вводят, а программа сама получает его в результате выполнения определенных действий.

Таким образом, организация данных является важной задачей при разработке данной и любой программы.

**2.2 Процессы**

Согласно всем перечисленным требованиям и указаниям, которые были рассмотрены в разделе «Анализ задачи», было определено, чем конкретно

должна заниматься разрабатываемая приложение. Главной задачей будет являться: повторение изученного материала и подготовка к квалификационному экзамену.

Для реализации задач будут использоваться процедуры. С помощью процедуры будет осуществляться переход на текстовый файл в формате .docx.

Материал и тест будет генерироваться на основе выбора пункта в объекте TRadioGroup. Далее после того, как будет прочитан теоретический материал и выполнена практическая часть, можно будет пройти тест для проверки своих знаний и получить оценку своих знаний. Ниже на рисунке 1 представлена система меню и организация навигации между окнами программы:

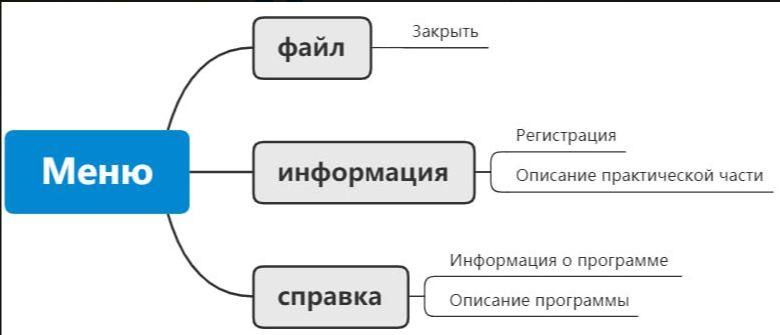


Рисунок 1 – Схема навигация между окнами программы

**2.3 Описание внешнего пользовательского интерфейса**

Важным при выполнении курсового проекта является организация диалога между, пользователем и самой программой. Во многом это зависит от того, как программист разработает данную программу, какие компоненты будут использованы и какие методы будут автоматизированы. Во-первых, особое внимание следует уделить интерфейсу. Разработчик должен так организовать внешний вид своей программы, чтобы пользователь понял, что от него требуется.

Для организации эффективной работы пользователя нужно создать целостное приложение данной предметной области, в которой все компоненты приложения будут сгруппированы по функциональному назначению. При этом необходимо обеспечить удобный графический интерфейс пользователя. Приложение должно позволить пользователю решать задачи, затрачивая значительно меньше усилий, чем при работе с разрозненными объектами.

Ниже на рисунке 1 представлена система меню и организация навигации между окнами программы:

**3. Реализация**

**3.1 Реализация проекта**

Данный курсовой проект содержит 7 модулей. Далее рассмотрим назначение каждого модуля:

Unit 1- является заставкой в курсовом проекте;

Unit 4- содержит весь теоретический материал с переходом на текстовый файл;

Unit 6- описание практической части;

Unit 7- регистрация;

Unit 8- информация о программе;

Unit 9-задания;

Unit 12-тестовая часть.

**3.1.1 Структура и описание процедур и функций пользователя**

Описание разработанных процедур находятся в таблице 1.

Таблица 1 – Процедуры и функции

|  |  |  |  |
| --- | --- | --- | --- |
| Имя процедуры (функции) | В каком модуле находится | За каким компонентом закреплена | Назначение |
| 1 | 2 | 3 | 4 |
| 1 procedure TForm4.Button1Click(Sender: TObject) | Unit 4 | Button | Переход на предыдущую форму |
| 2 procedure TForm4.Button2Click(Sender: TObject) | Unit 4 | Button | Переход на следующую форму |
| 3 procedure TForm4.Label1Click(Sender: TObject) | Unit 4 | Label | Осуществляет переход на текстовый файл в формате .docx |
| 4 procedure TForm4.Label2Click(Sender: TObject) | Unit 4 | Label | Осуществляет переход на текстовый файл в формате .docx |
| 5 procedure TForm4.Label3Click(Sender: TObject) | Unit 4 | Label | Осуществляет переход на текстовый файл в формате .docx |
| 6 procedure TForm4.Label4Click(Sender: TObject) | Unit 4 | Label | Осуществляет переход на текстовый файл в формате .docx |

Продолжение таблицы 1

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 2 | 3 | 4 |
| 7 procedure TForm4.Label5Click(Sender: TObject) | Unit 4 | Label | Осуществляет переход на текстовый файл в формате .docx |
| 8 procedure TForm4.Label6Click(Sender: TObject) | Unit 4 | Label | Осуществляет переход на текстовый файл в формате .docx |
| 9 procedure TForm4.Label7Click(Sender: TObject) | Unit 4 | Label | Осуществляет переход на текстовый файл в формате .docx |
| 10 procedure TForm4.Label8Click(Sender: TObject) | Unit 4 | Label | Осуществляет переход на текстовый файл в формате .docx |
| 11 procedure TForm4.Label9Click(Sender: TObject) | Unit 4 | Label | Осуществляет переход на текстовый файл в формате .docx |
| 12 procedure TForm4.Label10Click(Sender: TObject) | Unit 4 | Label | Осуществляет переход на текстовый файл в формате .docx |
| 13 procedure TForm4.N2Click(Sender: TObject) | Unit 4 | Main | Закрывает программу |
| 14 procedure TForm4.N4Click(Sender: TObject) | Unit 4 | Main | Переход на форму с регистрацией |
| 15 procedure TForm4.N5Click(Sender: TObject) | Unit 4 | Main | Переход на форму с описанием практической части |
| 16 procedure TForm4.N7Click(Sender: TObject) | Unit 4 | Main | Переход на форму «Информация о программе» |
| 17 procedure TForm6.Button1Click(Sender: TObject) | Unit 6 | Button | Переход на предыдущую форму |
| 18 procedure TForm6.Button2Click(Sender: TObject) | Unit 6 | Button | Переход на следующую форму |
| 19 procedure TForm6.Button3Click(Sender: TObject) | Unit 6 | Button | Генерирует теория от 1 до 10 |
| 20 procedure TForm6.N2Click(Sender: TObject) | Unit 6 | Main | Закрывает программу |
| 21 procedure TForm6.N4Click(Sender: TObject) | Unit 6 | Main | Переход на главную страницу |
| 22 procedure TForm6.N5Click | Unit 6 | Main | Переход на форму с регистрацией |

Продолжение таблицы 1

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 2 | 3 | 4 |
| 23 procedure TForm6.N7Click(Sender: TObject) | Unit 6 | Main | Переход на форму «Информация о программе» |
| 24 procedure TForm7.Button1Click(Sender: TObject) | Unit 7 | Button | Переход на предыдущую форму |
| 25 procedure TForm7.Button2Click(Sender: TObject) | Unit 7 | Button | Переход на форму с заданиями по темам |
| 26 procedure TForm7.N2Click(Sender: TObject) | Unit 7 | Main | Закрывает программу |
| 27 procedure TForm7.N4Click(Sender: TObject) | Unit 7 | Main | Переход на главную страницу |
| 28 procedure TForm7.N5Click(Sender: TObject) | Unit 7 | Main | Переход на форму с описанием практической части |
| 29 procedure TForm7.N7Click(Sender: TObject) | Unit 7 | Main | Переход на форму «Информация о программе» |
| 30 procedure TForm8.Button1Click(Sender: TObject) | Unit 8 | Button | Переход на предыдущую форму |
| 31 procedure TForm8.N2Click(Sender: TObject) | Unit 8 | Main | Закрывает программу |
| 32 procedure TForm8.N4Click(Sender: TObject) | Unit 8 | Main | Переход на главную страницу |
| 33 procedure TForm8.N5Click(Sender: TObject) | Unit 8 | Main | Переход на форму с описанием практической части |
| 34 procedure TForm8.N6Click(Sender: TObject) | Unit 8 | Main | Переход на форму с регистрацией |
| 35 procedure TForm9.Button12Click(Sender: TObject) | Unit 9 | Button | Переход на форму с тестом |
| 36 procedure TForm9.Button1Click(Sender: TObject) | Unit 9 | Button | Генерирует задания по «Геометрическим примитивам» от 1 до 16 |
| 37 procedure TForm9.Button2Click(Sender: TObject) | Unit 9 | Button | Генерирует задания по «Изометрии» от 1 до 16 |

Продолжение таблицы 1

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 2 | 3 | 4 |
| 38 procedure TForm9.Button3Click | Unit 9 | Button | Генерирует задания по «массивам» от 1 до 16 |
| 39 procedure TForm9.Button4Click | Unit 9 | Button | Генерирует задания по «плану» от 1 до 16 |
| 40 procedure TForm9.Button5Click | Unit 9 | Button | Генерирует задания по «полилинии» от 1 до 16 |
| 41 procedure TForm9.Button6Click | Unit 9 | Button | Генерирует задания по «примитивам» от 1 до 16 |
| 42 procedure TForm9.Button7Click | Unit 9 | Button | Генерирует задания по «слоям» от 1 до 16 |
| 43 procedure TForm9.Button8Click | Unit 9 | Button | Генерирует задания по «сопряжениям» от 1 до 16 |
| 44 procedure TForm9.Button9Click | Unit 9 | Button | Генерирует задания по «тексту» от 1 до 16 |
| 45 procedure TForm9.Button10Click | Unit 9 | Button | Генерирует задания по «штриховке» от 1 до 16 |
| 46 procedure TForm9.Button11Click | Unit 9 | Button | Показывает образец выполнения заданий по теме: «текст» |
| 47 procedure TForm9.Button12Click(Sender: TObject); | Unit 9 | Button | Переход на форму с тестом |
| 48 procedure TForm12.Button2Click(Sender: TObject); | Unit 12 | Button | При нажатии начинается тест |
| 49 procedure TForm12.Button1Click(Sender: TObject); | Unit 12 | Button | Переход на следующий вопрос |

**3.1.2 Описание использованных компонентов**

Описание использованных для разработки приложения компонентов приводится в таблице 2.

Таблица 2-Использованные компоненты

|  |  |  |
| --- | --- | --- |
| Компонент | На какой форме расположен | Назначение |
| 1 | 2 | 3 |
| TMainMenu | Form4, Form6, Form7, Form8 | Используется для создания главного меню проекта |

Продолжение таблицы 2

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| TImage | Form2, Form3, Form4, Form5, Form6, Form7, Form9, Form12 | Используется как фон |
| TLabel | Form1, Form4, Form6, Form7, Form9, Form12 | Отображение надписей на форме |
| TButton | Form1, Form4, Form6, Form7, Form8, Form9, Form12 | Используется для взаимодействия пользователя с программой: для начала теста, закрытие программы, перехода на другую формы и т.д. |
| TEdit | Form7 | Используется для ввода информации пользователя |
| TRadioGroup | Form12 | Используется для выбора варианта ответа в тесте |
| TProgressbar | Form12 | Используется для полосы времени |
| Ttimes | Form12 | Используется, как таймер |

**3.2 Спецификация программы**

Точное название проекта и его состав приводится в таблице 3.

Таблица 3 – Спецификация программы

|  |  |
| --- | --- |
| Имя файла | Назначение |
| 1 | 2 |
| Project1.exe | Исполняемый файл проекта, используется для запуска программы на выполнение. |
| Project1.dproj | Файл проекта, связывает все файлы из которых состоит приложение. |
| Unit1.pas | Файл программного модуля главной формы |
| Unit4.pas | Файл программного модуля теоретической части проекта |
| Unit6.pas | Файл программного модуля описания практической части проекта |
| Unit7.pas | Файл программного модуля регистрации части проекта |
| Unit8.pas | Файл программного модуля «Информация о программе» части проекта |
| Unit9.pas | Файл программного модуля «Задания» части проекта |
| Unit12.pas | Файл программного модуля тестовая часть проекта |
| Unit1.dfm | главная форма |
| Unit4.dfm | Форма с теоретической частью |
| Unit6.dfm | Форма с описанием практической части |
| Unit7.dfm | Форма с регистрацией |
| Unit8.dfm | Форма «Информация о программе» |
| Unit9.dfm | Форма с заданиями |
| Unit12.dfm | Форма с тестом |

Продолжение таблицы 3

|  |  |
| --- | --- |
| 1 | 2 |
| 1.txt | Тест по AutoCad |
| 2.txt | Тест по AutoCad |
| 3.txt | Тест по AutoCad |
| 4.txt | Тест по AutoCad |
| 5.txt | Тест по AutoCad |
| 1.png | Изображение |
| 2.png | Изображение |
| 3.png | Изображение |
| 4.png | Изображение |
| 5.png | Изображение |
| 6.png | Изображение |
| 7.png | Изображение |
| 8.png | Изображение |
| 9.png | Изображение |
| 10.png | Изображение |
| 11.png | Изображение |
| 12.png | Изображение |
| 13.png | Изображение |
| 14.png | Изображение |
| 15.jpg | Изображение |

**4. Тестирование**

При разработке данной программы многие возникающие ошибки и недоработки были исправлены на этапе реализации проекта. После завершения этапа написания программы было проведено тщательное функциональное тестирование. Функциональное тестирование должно гарантировать работу всех элементов программы в автономном режиме.

Отчет о результатах тестирования представлен в таблице 4

Таблица 4- Отчет о результатах тестирования

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Идентификатр теста | Тест | Ожидаемый результат | Физический результат | Результат тестирование |
| 1 | 2 | 3 | 4 | 5 |
| 1 | Проверка входа  пользователя вприложение | Открытие главного окна | Открытие главного  окна | Выполнено |
| 2 | Проверка кнопки «>>>» | Открытие формы с  теоретическим материалом | Открытие формы с  теоретическим материалом | Выполнено |
| 3 | Проверка перехода на текстовый файл в формате .docx | Открытие текстового файла в формате .docx | Открытие текстового файла в формате .docx | Выполнено |
| 4 | Проверка главного меню на кнопку «Закрыть» | Закрытие приложения | Закрытие приложения | Выполнено |
| 5 | Проверка главного меню на кнопку «Теоретическая часть» | Открытие формы с теоретическим материалом | Открытие формы с теоретическим материалом | Выполнено |
| 6 | Проверка главного меню на кнопку «Описание практической части» | Открытие формы с описанием практической части | Открытие формы с описанием практической части | Выполнено |
| 7 | Проверка главного меню на кнопку «Информация о программе» | Открытие формы «Информация о программе» | Открытие формы «Информация о программе» | Выполнено |
| 8 | Проверка кнопки «Далее» | Открытие формы с описанием практической части | Открытие формы с описанием практической части | Выполнено |
| 9 | Проверка кнопки «Сгенерировать» | Генерация чисел | Генерация чисел | Выполнено |

Продолжение таблицы 4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| 10 | Проверка кнопки «Далее» | Открытие формы с регистрацией | Открытие формы с регистрацией | Выполнено |
| 11 | Проверка перехода на текстовый файл в формате .docx | Открытие текстового файла в формате .docx | Открытие текстового файла в формате .docx | Выполнено |
| 12 | Проверка кнопки  «Начать» | Открытие формы с заданиями | Открытие формы с заданиями | Выполнено |
| 13 | Проверка кнопки  «Назад» | Открытие формы с регистрацией | Открытие формы с регистрацией | Выполнено |
| 14 | Проверка кнопки  «Тест» | Переход на форму с тестом | Переход на форму с тестом | Выполнено |
| 15 | Проверка кнопки  «Начать » | Открытие теста | Открытие теста | Выполнено |

Элементы программы были проверены, и было установлено, что все они работают правильно и выполняют задачи, указанные в процедурах.

При использовании данного приложения пользователь не столкнётся с ошибками системы.

**5 Руководство пользователя**

**5.1 Общие сведения о программном продукте**

Цель данного проекта заключается в обобщении знаний и подготовки к квалификационному экзамену.

Создаваемое приложение будет рассчитано на учащихся, проходивших подготовку рабочих по профессии связанной с «AutoCad ».

Быстродействие любой программы во многом зависит от характеристик выбранного персонального компьютера: рабочей частоты процессора, объема оперативной памяти и т.д. Несмотря на все реализованные в ней задачи, она легко запускается и функционирует на любых машинах.

Программа разработана на ПК со следующими характеристиками:

* процессор AMD Ryzen 7 4700U with Radeon Graphics, 2000 МГц, ядер: 8, логических процессоров: 8
* объем ОЗУ 32Гб;
* графический адаптер AMD Radeon Graphics Processor (0x1636), Advanced Micro Devices, Inc.-совместимый;
* операционная система Windows 10.

**5.2 Инсталляция**

Для того, чтобы установить программу необходимо запустить файл Установщик.exe. Появится окно установки приложения “Разработка электронного средства обучения «AutoCad»”.

Затем достаточно следовать приведенной инструкции установки приложения.

**5.3 Выполнение программы**

**5.3.1 Запуск программы**

Данную программу можно запустить различными способами. Первым из них является запуск с помощью ярлыка на рабочем столе. Необходимо дважды щелкнуть левой кнопкой мыши на ярлыке с названием “ Разработка электронного средства обучения «AutoCad».”

Вторым способом является запуск из каталога, в который устанавливалось приложение (по умолчанию C:\Program Files (x86)\ Разработка электронного средства обучения «AutoCad»).

По подготовленным тестам будет осуществляться функциональное и полное тестирование программного продукта. Отчет о результатах тестирования будет представлен в 4 разделе пояснительной записки.

**5.3.2 Инструкции по работе с программой**

После запуска приложения на экране нас встречает заставка на рисунок 2.



Рисунок 2 – Заставка

При нажатии на кнопку “>>>” будет осуществлен переход на форму с теоретическим материалом, представлен на рисунке 3, где по нажатию на тему можно перейти на текстовый файл в формате .docx, представлен на рисунке 4.

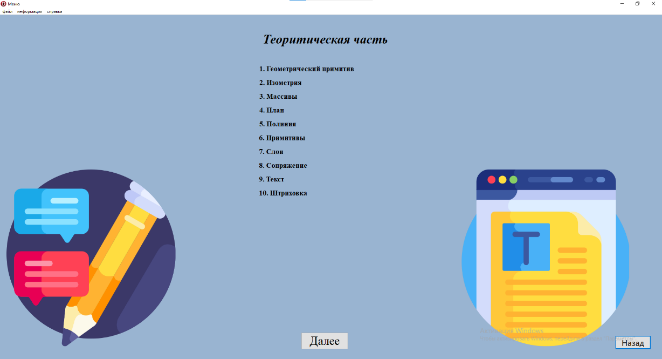


Рисунок 3-Теоретический материал

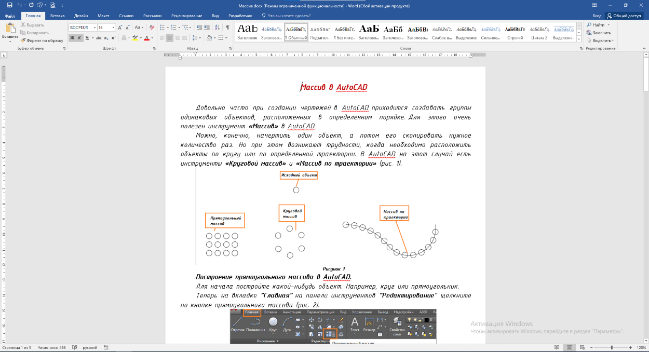
****

Рисунок 4- Текстовый файл в формате .docx

По нажатию на кнопку “Информация о программе” в главном меню можно выбрать описание программы, перейдя на новую форму рисунок 5.

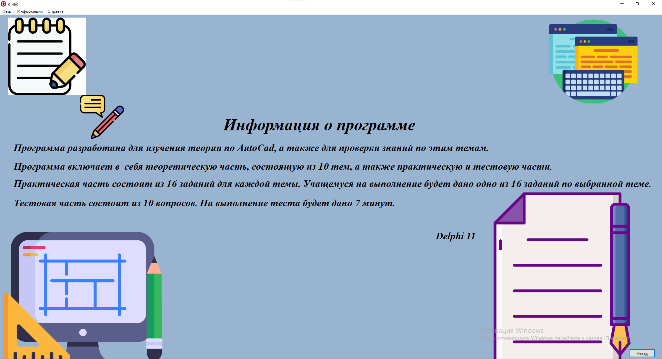


Рисунок 5-Описание программы

Вернуться назад можно через главное меню. Вернувшись назад на форму с теоретическим материалом, нажав на кнопку «Далее». Перед нами форма с описанием практической части, представлена на рисунке 6.

****

Рисунок 6-Описание практической части

Нажав на кнопку «Сгенерировать», нам случайным выпадает число с теорией, по которому можно перейти на текстовый файл в формате .docx на рисунке 7.

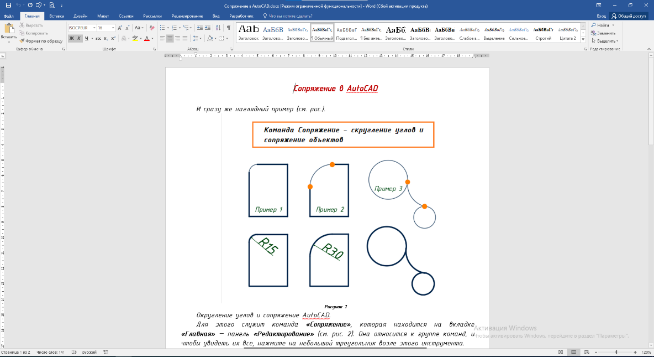


Рисунок 7-Теория в текстовом файле в формате .docx

Нажав на кнопку «Пройти тест», переходим на форму с регистрацией, представлена на рисунке 8.

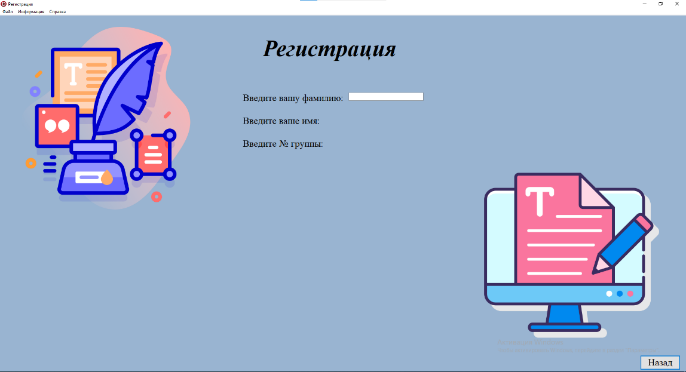


Рисунок 8-Регистрация.

Начать выполнять задания можно по кнопке «Начать», после чего мы оказываемся на форме с заданиями. Можно посмотреть на рисунке 9.

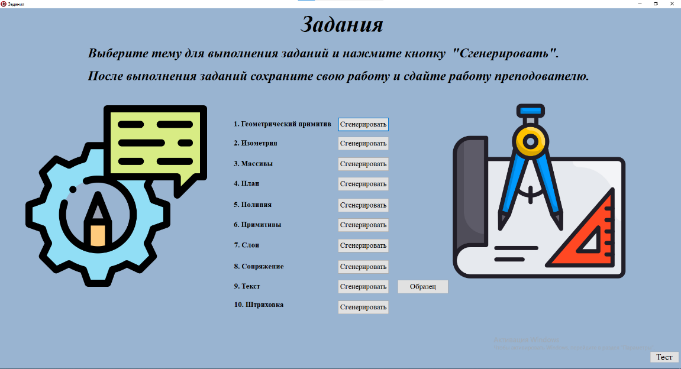
****

Рисунок 9-Задания

Нажав на кнопку «Сгенерировать», нам случайным выпадает число с заданием, по которому можно перейти на файл в формате .wmf, .png. Можно посмотреть на рисунке 10.

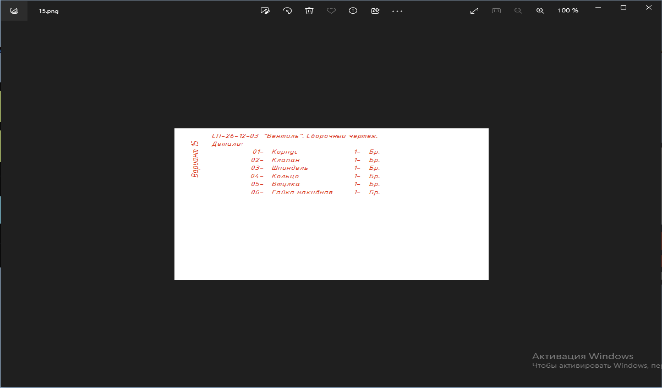


Рисунок 10- Задания в файле в формате .wmf, .png.

Начать тест можно по кнопке «Тест», после чего мы оказываемся на форме с тестом. Начав тест перед нами вопрос и ответы, можно посмотреть на рисунке 11.

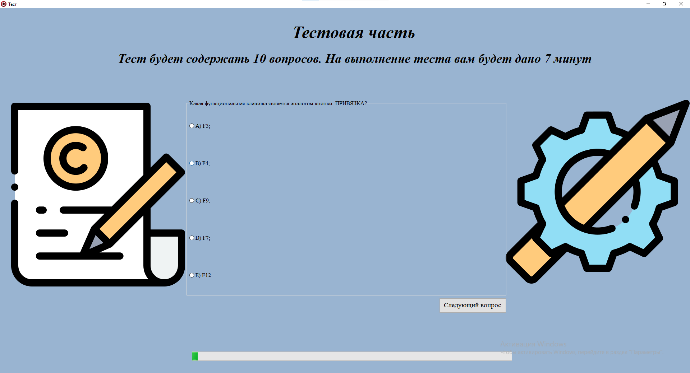


Рисунок 11-Тест

После завершение теста перед нами буду выведены результаты, представленные на рисунке 12.

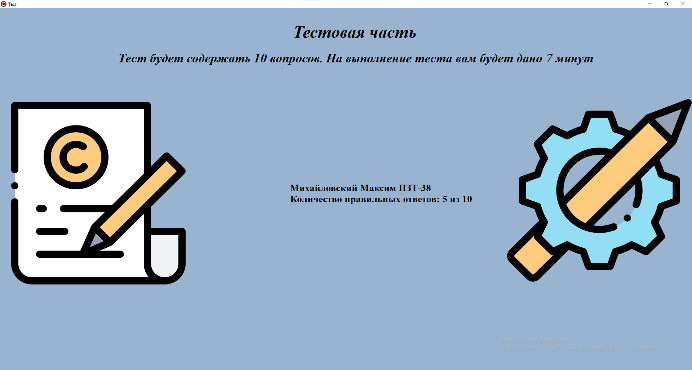


Рисунок 12-Результаты выполнения теста

**Заключение**

Целью данного проекта заключалось обобщение знаний и подготовка к квалификационному экзамену.

В поставленной задаче был реализован простой и понятный пользовательский интерфейс.

В ходе тестирования все исключительные ситуации были обработаны. Проект работает без сбоев и ошибок.

В процессе разработки программного продукта я научился создавать динамические компоненты на форме, проработал их взаимодействие, закрепил умение создания собственных процедур, научился обрабатывать все исключительные ситуации.

Исходя из этого, можно сделать вывод, что программа реализована успешно.

**Список использованных источников**

1. Симонович С.В., Евсеев Г.А., Алексеев А.Г. Специальная информатика: Учебное пособие. - М.: АСТ-ПРЕСС: Инфорком-Пресс, 2001. - 480 с.
2. Архангельский, А. Я. Delphi 7. Справочное пособие. - Москва: Бином-Пресс, 2014. - 1024 с.
3. Вощинская, Г.Э. Разработка компонентов в DELPHI. - Воронеж: ИПЦ ВГУ, 2007. - 57 с.
4. Культин, Н. Основы программирования в Delphi 7. - Санкт-Петербург: Питер, 2009. - 640 с.
5. Архангельский, А. Я. Delphi 7. Справочное пособие. - Москва: Бином-Пресс, 2014. - 1024 с.

Культин, Н. Delphi 6. Программирование на Object Pascal / Н. Культин. - М.: БХВ-Петербург, 2012. - 528 c.

**Приложение А**

Листинг программы

unit Unit1;

interface

uses

Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes, Vcl.Graphics,

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.Imaging.jpeg, Vcl.ExtCtrls,

Vcl.StdCtrls;

type

TForm1 = class(TForm)

Image2: TImage;

StaticText1: TStaticText;

StaticText2: TStaticText;

Button1: TButton;

StaticText3: TStaticText;

procedure Button1Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form1: TForm1;

implementation

uses

unit2, Unit4;

{$R \*.dfm}

procedure TForm1.Button1Click(Sender: TObject);

begin

Form1.Hide;

Form4.Show;

end;

end.

unit Unit4;

interface

uses

Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes, Vcl.Graphics,

Vcl.Controls, Vcl.Forms, Vcl.ExtCtrls, Vcl.Menus, Vcl.StdCtrls, Vcl.CheckLst, ShellApi,

Vcl.Imaging.pngimage;

type

TForm4 = class(TForm)

Button1: TButton;

MainMenu1: TMainMenu;

N1: TMenuItem;

N2: TMenuItem;

N3: TMenuItem;

N4: TMenuItem;

N5: TMenuItem;

N6: TMenuItem;

N7: TMenuItem;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

Label4: TLabel;

Label5: TLabel;

Label6: TLabel;

Label7: TLabel;

Label8: TLabel;

Label9: TLabel;

Label10: TLabel;

Image1: TImage;

Image2: TImage;

Label11: TLabel;

Button2: TButton;

procedure N2Click(Sender: TObject);

procedure N5Click(Sender: TObject);

procedure N7Click(Sender: TObject);

procedure N4Click(Sender: TObject);

procedure Button1Click(Sender: TObject);

procedure Label1Click(Sender: TObject);

procedure Label2Click(Sender: TObject);

procedure Label3Click(Sender: TObject);

procedure Label4Click(Sender: TObject);

procedure Label5Click(Sender: TObject);

procedure Label6Click(Sender: TObject);

procedure Label7Click(Sender: TObject);

procedure Label8Click(Sender: TObject);

procedure Label9Click(Sender: TObject);

procedure Label10Click(Sender: TObject);

procedure Button2Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form4: TForm4;

implementation

{$R \*.dfm}

uses Unit5, Unit6, Unit7, Unit8, Unit1;

procedure TForm4.Button1Clier: TObject);

begin

Form4.Hide;

Form1.Show;

end;

procedure TForm4.Button2Click(Sender: TObject);

begin

Form4.Hide;

Form6.Show;

end;

procedure TForm4.Label10Click(Sender: TObject);

begin

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Штриховка\Пособие\Штриховка в AutoCAD.docx'), nil, nil, SW\_SHOWNORMAL);

end;

procedure TForm4.Label1Click(Sender: TObject);

begin

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Геометріческій прімітів\пособие\Геометрические примитивы1.docx'), nil, nil, SW\_SHOWNORMAL);

end;

procedure TForm4.Label2Click(Sender: TObject);

begin

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Изометрия\Пособие\Установка изометрического стиля сетки.docx'), nil, nil, SW\_SHOWNORMAL);

end;

procedure TForm4.Label3Click(Sender: TObject);

begin

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Массивы\Пособие\Массив.docx'), nil, nil, SW\_SHOWNORMAL);

end;

procedure TForm4.Label4Click(Sender: TObject);

begin

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\План\Пособие\Инструкция выполнения коттеджа.docx'), nil, nil, SW\_SHOWNORMAL);

end;

procedure TForm4.Label5Click(Sender: TObject);

begin

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Полилиния\Пособие\Полилиния.docx'), nil, nil, SW\_SHOWNORMAL);

end;

procedure TForm4.Label6Click(Sender: TObject);

begin

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Примитивы\Примитивы.pdf'), nil, nil, SW\_SHOWNORMAL);

end;

procedure TForm4.Label7Click(Sender: TObject);

begin

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Слои\Пособие\Слои.docx'), nil, nil, SW\_SHOWNORMAL);

end;

procedure TForm4.Label8Click(Sender: TObject);

begin

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Сопряжения\Пособие\Сопряжение в AutoCAD.docx'), nil, nil, SW\_SHOWNORMAL);

end;

procedure TForm4.Label9Click(Sender: TObject);

begin

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\текст\Пособие\Создание таблиц с помощью настройки стилей таблиц в AutoCAD.docx'), nil, nil, SW\_SHOWNORMAL);

end;

procedure TForm4.N2Click(Sender: TObject);

begin

Close;

end;

procedure TForm4.N4Click(Sender: TObject);

begin

Form4.Hide;

Form7.Show;

end;

procedure TForm4.N5Click(Sender: TObject);

begin

Form4.Hide;

Form6.Show;

end;

procedure TForm4.N7Click(Sender: TObject);

begin

Form4.Hide;

Form8.Show;

end;

end.

unit Unit6;

interface

uses

Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes, Vcl.Graphics,

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.StdCtrls, Vcl.Imaging.pngimage,

Vcl.ExtCtrls, Vcl.Menus, ShellApi;

type

TForm6 = class(TForm)

Button1: TButton;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

Label4: TLabel;

Image1: TImage;

Image2: TImage;

Button2: TButton;

Label5: TLabel;

Button3: TButton;

MainMenu1: TMainMenu;

N1: TMenuItem;

N2: TMenuItem;

N3: TMenuItem;

N4: TMenuItem;

N5: TMenuItem;

N6: TMenuItem;

N7: TMenuItem;

Label7: TLabel;

Label8: TLabel;

Label9: TLabel;

procedure Button1Click(Sender: TObject);

procedure Button2Click(Sender: TObject);

procedure N2Click(Sender: TObject);

procedure N4Click(Sender: TObject);

procedure N5Click(Sender: TObject);

procedure N7Click(Sender: TObject);

procedure Button3Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form6: TForm6;

implementation

{$R \*.dfm}

uses Unit4, Unit1, Unit10, Unit5, Unit7, Unit8, Unit9;

procedure TForm6.Button1Click(Sender: TObject);

begin

Form6.Hide;

Form4.Show;

end;

procedure TForm6.Button2Click(Sender: TObject);

begin

Form6.Hide;

Form7.Show;

end;

procedure TForm6.Button3Click(Sender: TObject);

var

randomNumber: Integer;

begin

// Генерируем случайное число от 1 до 100

randomNumber := Random(10) + 1;

// Используйте полученное случайное число по вашему усмотрению

// Например, выведите его на метку (Label)

Button3.Caption := IntToStr(randomNumber);

if randomNumber=1 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Геометріческій прімітів\пособие\Геометрические примитивы1.docx'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=2 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Изометрия\Пособие\Установка изометрического стиля сетки.docx'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=3 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Массивы\Пособие\Массив.docx'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=4 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\План\Пособие\Инструкция выполнения коттеджа.docx'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=5 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Полилиния\Пособие\Полилиния.docx'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=6 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Примитивы\Примитивы.pdf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=7 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Слои\Пособие\Слои.docx'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=8 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Сопряжения\Пособие\Сопряжение в AutoCAD.docx'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=9 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\текст\Пособие\Создание таблиц с помощью настройки стилей таблиц в AutoCAD.docx'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=10 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Штриховка\Пособие\Штриховка в AutoCAD.docx'), nil, nil, SW\_SHOWNORMAL);

Button3.Caption := IntToStr(randomNumber);

end;

procedure TForm6.N2Click(Sender: TObject);

begin

Close;

end;

procedure TForm6.N4Click(Sender: TObject);

begin

Form6.Hide;

Form4.Show;

end;

procedure TForm6.N5Click(Sender: TObject);

begin

Form6.Hide;

Form7.Show;

end;

procedure TForm6.N7Click(Sender: TObject);

begin

Form6.Hide;

Form8.Show;

end;

end.

unit Unit7;

interface

uses

Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes, Vcl.Graphics,

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.StdCtrls, Vcl.Imaging.jpeg,

Vcl.Imaging.pngimage, Vcl.ExtCtrls, Vcl.Menus;

type

TForm7 = class(TForm)

Button1: TButton;

Label1: TLabel;

Label3: TLabel;

Label4: TLabel;

Edit1: TEdit;

Edit3: TEdit;

Image1: TImage;

Image2: TImage;

Button2: TButton;

MainMenu1: TMainMenu;

N1: TMenuItem;

N2: TMenuItem;

N3: TMenuItem;

N4: TMenuItem;

N5: TMenuItem;

N6: TMenuItem;

N7: TMenuItem;

Label2: TLabel;

Edit2: TEdit;

procedure Button1Click(Sender: TObject);

procedure Button2Click(Sender: TObject);

procedure N2Click(Sender: TObject);

procedure N4Click(Sender: TObject);

procedure N5Click(Sender: TObject);

procedure N7Click(Sender: TObject);

procedure Edit1Change(Sender: TObject);

procedure Edit2Change(Sender: TObject);

procedure Edit3Change(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form7: TForm7;

implementation

{$R \*.dfm}

uses Unit4, Unit1, Unit10, Unit5, Unit6, Unit8, Unit9;

procedure TForm7.Button1Click(Sender: TObject);

begin

Form7.Hide;

Form6.Show;

end;

procedure TForm7.Button2Click(Sender: TObject);

begin

Form7.Hide;

Form9.Show;

end;

procedure TForm7.Edit1Change(Sender: TObject);

begin

Edit2.Visible:=true;

end;

procedure TForm7.Edit2Change(Sender: TObject);

begin

Edit3.Visible:=true;

end;

procedure TForm7.Edit3Change(Sender: TObject);

begin

Button2.Visible:=true;

end;

procedure TForm7.N2Click(Sender: TObject);

begin

Close;

end;

procedure TForm7.N4Click(Sender: TObject);

begin

Form7.Hide;

Form4.Show;

end;

procedure TForm7.N5Click(Sender: TObject);

begin

Form7.Hide;

Form6.Show;

end;

procedure TForm7.N7Click(Sender: TObject);

begin

Form7.Hide;

Form8.Show;

end;

end.

unit Unit8;

interface

uses

Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes, Vcl.Graphics,

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.StdCtrls, Vcl.Menus,

Vcl.Imaging.jpeg, Vcl.ExtCtrls, Vcl.Imaging.pngimage;

type

TForm8 = class(TForm)

Button1: TButton;

MainMenu1: TMainMenu;

N1: TMenuItem;

N2: TMenuItem;

N3: TMenuItem;

N4: TMenuItem;

N5: TMenuItem;

N6: TMenuItem;

N7: TMenuItem;

N8: TMenuItem;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

Label4: TLabel;

Label5: TLabel;

Image1: TImage;

Image2: TImage

Image3: TImage;

Image4: TImage;

Image5: TImage;

Label6: TLabel;

procedure Button1Click(Sender: TObject);

procedure N2Click(Sender: TObject);

procedure N4Click(Sender: TObject);

procedure N5Click(Sender: TObject);

procedure N6Click(Sender: TObject);

procedure N8Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form8: TForm8;

implementation

{$R \*.dfm}

uses Unit4, Unit1, Unit10, Unit5, Unit6, Unit7, Unit9;

procedure TForm8.Button1Click(Sender: TObject);

begin

Form8.Hide;

Form4.Show;

end;

procedure TForm8.N2Click(Sender: TObject);

begin

Close;

end;

procedure TForm8.N4Click(Sender: TObject);

begin

Form8.Hide;

Form4.Show;

end;

procedure TForm8.N5Click(Sender: TObject);

begin

Form8.Hide;

Form6.Show;

end;

procedure TForm8.N6Click(Sender: TObject);

begin

Form8.Hide;

Form7.Show;

end;

procedure TForm8.N8Click(Sender: TObject);

begin

Form8.Hide;

Form4.Show;

end;

end.

unit Unit9;

interface

uses

Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes, Vcl.Graphics,

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.StdCtrls, ShellApi,

Vcl.Imaging.jpeg, Vcl.ExtCtrls, Vcl.Imaging.pngimage;

type

TForm9 = class(TForm)

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

Label4: TLabel;

Label5: TLabel;

Label6: TLabel;

Label7: TLabel;

Label8: TLabel;

Label9: TLabel;

Label10: TLabel

Button1: TButton;

Button2: TButton;

Button3: TButton;

Button5: TButton;

Button6: TButton;

Button7: TButton;

Button8: TButton;

Button9: TButton;

Button10: TButton;

Button4: TButton;

Button11: TButton;

Label11: TLabel;

Label12: TLabel;

Label13: TLabel;

Image2: TImage;

Image1: TImage;

Button12: TButton;

procedure Button1Click(Sender: TObject);

procedure Button2Click(Sender: TObject);

procedure Button3Click(Sender: TObject);

procedure Button5Click(Sender: TObject);

procedure Button6Click(Sender: TObject);

procedure Button7Click(Sender: TObject);

procedure Button8Click(Sender: TObject);

procedure Button9Click(Sender: TObject);

procedure Button10Click(Sender: TObject);

procedure Button4Click(Sender: TObject);

procedure Button11Click(Sender: TObject);

procedure Button12Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form9: TForm9;

implementation

{$R \*.dfm}

uses Unit1, Unit10, Unit4, Unit5, Unit6, Unit7, Unit8, Unit12;

procedure TForm9.Button10Click(Sender: TObject);

begin

var

randomNumber: Integer;

begin

// Генерируем случайное число от 1 до 100

randomNumber := Random(16) + 1;

// Используйте полученное случайное число по вашему усмотрению

// Например, выведите его на метку (Label)

Button10.Caption := IntToStr(randomNumber);

if randomNumber=1 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Штриховка\1.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=2 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Штриховка\2.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=3 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Штриховка\3.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=4 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Штриховка\4.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=5 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Штриховка\5.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=6 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Штриховка\6.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=7 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Штриховка\7.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=8 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Штриховка\8.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=9 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Штриховка\9.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=10 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Штриховка\10.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=11 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Штриховка\11.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=12 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Штриховка\12.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=13 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Штриховка\13.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=14 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Штриховка\14.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=15 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Штриховка\15.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=16 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Штриховка\16.wmf'), nil, nil, SW\_SHOWNORMAL);

Button10.Caption := IntToStr(randomNumber);

end;

end;

procedure TForm9.Button11Click(Sender: TObject);

begin

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\текст\png\Образец.png'), nil, nil, SW\_SHOWNORMAL);

end;

procedure TForm9.Button12Click(Sender: TObject);

begin

Form9.Hide;

Form12.Show;

end;

procedure TForm9.Button1Click(Sender: TObject);

begin

var

randomNumber: Integer;

begin

// Генерируем случайное число от 1 до 100

randomNumber := Random(16) + 1;

// Используйте полученное случайное число по вашему усмотрению

// Например, выведите его на метку (Label)

Button1.Caption := IntToStr(randomNumber);

if randomNumber=1 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Геометріческій прімітів\Новая папка\1.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=2 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Геометріческій прімітів\Новая папка\2.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=3 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Геометріческій прімітів\Новая папка\3.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=4 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Геометріческій прімітів\Новая папка\4.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=5 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Геометріческій прімітів\Новая папка\5.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=6 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Геометріческій прімітів\Новая папка\6.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=7 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Геометріческій прімітів\Новая папка\7.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=8 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Геометріческій прімітів\Новая папка\8.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=9 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Геометріческій прімітів\Новая папка\9.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=10 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Геометріческій прімітів\Новая папка\10.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=11 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Геометріческій прімітів\Новая папка\11.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=12 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Геометріческій прімітів\Новая папка\12.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=13 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Геометріческій прімітів\Новая папка\13.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=14 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Геометріческій прімітів\Новая папка\14.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=15 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Геометріческій прімітів\Новая папка\15.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=16 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Геометріческій прімітів\Новая папка\16.png'), nil, nil, SW\_SHOWNORMAL);

Button1.Caption := IntToStr(randomNumber);

end;

end;

procedure TForm9.Button2Click(Sender: TObject);

begin

var

randomNumber: Integer;

begin

// Генерируем случайное число от 1 до 100

randomNumber := Random(16) + 1;

// Используйте полученное случайное число по вашему усмотрению

// Например, выведите его на метку (Label)

Button2.Caption := IntToStr(randomNumber);

if randomNumber=1 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Изометрия\Варианты\1.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=2 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Изометрия\Варианты\2.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=3 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Изометрия\Варианты\3.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=4 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Изометрия\Варианты\4.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=5 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Изометрия\Варианты\5.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=6 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Изометрия\Варианты\6.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=7 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Изометрия\Варианты\7.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=8 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Изометрия\Варианты\8.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=9 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Изометрия\Варианты\9.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=10 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Изометрия\Варианты\10.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=11 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Изометрия\Варианты\11.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=12 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Изометрия\Варианты\12.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=13 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Изометрия\Варианты\13.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=14 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Изометрия\Варианты\14.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=15 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Изометрия\Варианты\15.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=16 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Изометрия\Варианты\16.wmf'), nil, nil, SW\_SHOWNORMAL);

Button2.Caption := IntToStr(randomNumber);

end;

end;

procedure TForm9.Button3Click(Sender: TObject);

begin

var

randomNumber: Integer;

begin

// Генерируем случайное число от 1 до 100

randomNumber := Random(16) + 1;

// Используйте полученное случайное число по вашему усмотрению

// Например, выведите его на метку (Label)

Button3.Caption := IntToStr(randomNumber);

if randomNumber=1 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Массивы\1.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=2 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Массивы\2.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=3 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Массивы\3.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=4 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Массивы\4.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=5 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Массивы\5.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=6 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Массивы\6.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=7 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Массивы\7.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=8 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Массивы\8.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=9 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Массивы\9.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=10 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Массивы\10.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=11 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Массивы\11.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=12 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Массивы\12.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=13 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Массивы\13.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=14 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Массивы\14.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=15 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Массивы\15.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=16 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Массивы\16.wmf'), nil, nil, SW\_SHOWNORMAL);

Button3.Caption := IntToStr(randomNumber);

end;

end;

procedure TForm9.Button4Click(Sender: TObject);

begin

var

randomNumber: Integer;

begin

// Генерируем случайное число от 1 до 100

randomNumber := Random(16) + 1;

// Используйте полученное случайное число по вашему усмотрению

// Например, выведите его на метку (Label)

Button4.Caption := IntToStr(randomNumber);

if randomNumber=1 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\План\1.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=2 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\План\2.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=3 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\План\3.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=4 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\План\4.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=5 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\План\5.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=6 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\План\6.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=7 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\План\7.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=8 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\План\8.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=9 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\План\9.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=10 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\План\10.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=11 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\План\11.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=12 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\План\12.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=13 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\План\13.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=14 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\План\14.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=15 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\План\15.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=16 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\План\16.wmf'), nil, nil, SW\_SHOWNORMAL);

Button4.Caption := IntToStr(randomNumber);

end;

end;

procedure TForm9.Button5Click(Sender: TObject);

begin

var

randomNumber: Integer;

begin

// Генерируем случайное число от 1 до 100

randomNumber := Random(16) + 1;

// Используйте полученное случайное число по вашему усмотрению

// Например, выведите его на метку (Label)

Button5.Caption := IntToStr(randomNumber);

if randomNumber=1 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Полилиния\Новая папка\1.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=2 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Полилиния\Новая папка\2.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=3 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Полилиния\Новая папка\3.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=4 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Полилиния\Новая папка\4.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=5 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Полилиния\Новая папка\5.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=6 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Полилиния\Новая папка\6.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=7 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Полилиния\Новая папка\7.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=8 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Полилиния\Новая папка\8.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=9 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Полилиния\Новая папка\9.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=10 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Полилиния\Новая папка\10.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=11 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Полилиния\Новая папка\11.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=12 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Полилиния\Новая папка\12.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=13 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Полилиния\Новая папка\13.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=14 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Полилиния\Новая папка\14.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=15 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Полилиния\Новая папка\15.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=16 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Полилиния\Новая папка\16.png'), nil, nil, SW\_SHOWNORMAL);

Button5.Caption := IntToStr(randomNumber);

end;

end;

procedure TForm9.Button6Click(Sender: TObject);

begin

var

randomNumber: Integer;

begin

// Генерируем случайное число от 1 до 100

randomNumber := Random(16) + 1;

// Используйте полученное случайное число по вашему усмотрению

// Например, выведите его на метку (Label)

Button6.Caption := IntToStr(randomNumber);

if randomNumber=1 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Примитивы\1.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=2 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Примитивы\2.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=3 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Примитивы\3.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=4 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Примитивы\4.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=5 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Примитивы\5.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=6 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Примитивы\6.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=7 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Примитивы\7.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=8 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Примитивы\8.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=9 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Примитивы\9.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=10 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Примитивы\10.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=11 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Примитивы\11.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=12 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Примитивы\12.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=13 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Примитивы\13.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=14 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Примитивы\14.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=15 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Примитивы\15.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=16 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Примитивы\16.wmf'), nil, nil, SW\_SHOWNORMAL);

Button6.Caption := IntToStr(randomNumber);

end;

end;

procedure TForm9.Button7Click(Sender: TObject);

begin

var

randomNumber: Integer;

begin

// Генерируем случайное число от 1 до 100

randomNumber := Random(16) + 1;

// Используйте полученное случайное число по вашему усмотрению

// Например, выведите его на метку (Label)

Button7.Caption := IntToStr(randomNumber);

if randomNumber=1 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Слои\1.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=2 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Слои\2.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=3 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Слои\3.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=4 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Слои\4.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=5 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Слои\5.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=6 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Слои\6.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=7 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Слои\7.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=8 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Слои\8.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=9 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Слои\9.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=10 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Слои\10.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=11 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Слои\11.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=12 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Слои\12.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=13 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Слои\13.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=14 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Слои\14.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=15 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Слои\15.wmf'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=16 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Слои\16.wmf'), nil, nil, SW\_SHOWNORMAL);

Button7.Caption := IntToStr(randomNumber);

end;

end;

procedure TForm9.Button8Click(Sender: TObject);

begin

var

randomNumber: Integer;

begin

// Генерируем случайное число от 1 до 100

randomNumber := Random(16) + 1;

// Используйте полученное случайное число по вашему усмотрению

// Например, выведите его на метку (Label)

Button8.Caption := IntToStr(randomNumber);

if randomNumber=1 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Сопряжения\Электронный\png\1.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=2 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Сопряжения\Электронный\png\2.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=3 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Сопряжения\Электронный\png\3.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=4 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Сопряжения\Электронный\png\4.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=5 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Сопряжения\Электронный\png\5.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=6 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Сопряжения\Электронный\png\6.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=7 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Сопряжения\Электронный\png\7.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=8 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Сопряжения\Электронный\png\8.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=9 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Сопряжения\Электронный\png\9.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=10 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Сопряжения\Электронный\png\10.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=11 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Сопряжения\Электронный\png\11.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=12 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Сопряжения\Электронный\png\12.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=13 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Сопряжения\Электронный\png\13.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=14 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Сопряжения\Электронный\png\14.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=15 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Сопряжения\Электронный\png\15.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=16 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Сопряжения\Электронный\png\16.png'), nil, nil, SW\_SHOWNORMAL);

Button8.Caption := IntToStr(randomNumber);

end;

end;

procedure TForm9.Button9Click(Sender: TObject);

begin

var

randomNumber: Integer;

begin

// Генерируем случайное число от 1 до 100

randomNumber := Random(16) + 1;

// Используйте полученное случайное число по вашему усмотрению

// Например, выведите его на метку (Label)

Button9.Caption := IntToStr(randomNumber);

if randomNumber=1 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Текст\png\1.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=2 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Текст\png\2.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=3 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Текст\png\3.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=4 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Текст\png\4.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=5 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Текст\png\5.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=6 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Текст\png\6.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=7 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Текст\png\7.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=8 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Текст\png\8.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=9 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Текст\png\9.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=10 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Текст\png\10.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=11 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Текст\png\11.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=12 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Текст\png\12.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=13 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Текст\png\13.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=14 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Текст\png\14.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=15 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Текст\png\15.png'), nil, nil, SW\_SHOWNORMAL);

if randomNumber=16 then

ShellExecute(Application.Handle, nil, PChar('D:\2 курс\курсач\информация к курсачу\Текст\png\16.png'), nil, nil, SW\_SHOWNORMAL);

Button9.Caption := IntToStr(randomNumber);

end;

end;

end.

unit Unit12;

interface

uses

Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes, Vcl.Graphics,

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.StdCtrls, Vcl.ExtCtrls, Vcl.ComCtrls,

Vcl.Imaging.pngimage;

type

TForm12 = class(TForm)

RadioGroup1: TRadioGroup;

Button1: TButton;

Label1: TLabel;

ProgressBar1: TProgressBar;

Timer1: TTimer;

Label2: TLabel;

Button2: TButton;

Label3: TLabel;

Image1: TImage;

Image2: TImage;

procedure Timer1Timer(Sender: TObject);

procedure Button2Click(Sender: TObject);

procedure Button1Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form12: TForm12;

f: text; // файловая переменная

s: string; // вспомогательная переменная для считывания строк из текстов. файла

Nvern, ball: integer; // номер верного ответа и количествого набираемых баллов за верные ответы

implementation

{$R \*.dfm}

uses Unit7;

procedure TForm12.Button1Click(Sender: TObject);

begin

//Если выбран вариант ответа и не достигнут конец файла

if (RadioGroup1.ItemIndex>-1) and (not Eof(f)) then begin

if RadioGroup1.ItemIndex = Nvern-1 then ball:=ball+1; //Если выбранный вариант соответствует

RadioGroup1.Items.Clear; //номеру верного ответа то балл прибавляется

Repeat //и очищается поле для следующего вопроса

if (s[1]='-') then begin

delete(s,1,1);

RadioGroup1.Caption:=s;

end

else if s[1]='\*' then begin

delete(s,1,1);

Nvern:=StrToInt(s);

end

else RadioGroup1.Items.Add(s);

readln(f,s);

until (s[1]='-') or (Eof(f));

end

//Если конец файла достигнут, значит вопросы закончились

Else if Eof(f) then begin

delete(s,1,1);

Nvern:=StrToInt(s);

if RadioGroup1.ItemIndex = Nvern-1 then ball:=ball+1;

Label1.Caption:=(form7.Edit1.Text+' '+form7.Edit2.Text+' '+form7.Edit3.Text+#13+'Количество правильных ответов: '+IntToStr(ball)+' из 10'); //Вывод количества баллов

CloseFile(f);

RadioGroup1.Visible:=false;

Button1.Visible:=False;

Button2.Visible:=False;

Button1.Enabled:=False; //кнопка становится недоступной в завершении проекта

progressbar1.Visible:=false;

progressbar1.Enabled:=false;

timer1.Enabled:=false;

end;

end;

procedure TForm12.Button2Click(Sender: TObject);

var randomNumber: Integer;

begin

// Генерируем случайное число от 1 до 100

randomNumber := Random(5) + 1;

// Используйте полученное случайное число по вашему усмотрению

// Например, выведите его на метку (Label)

Button2.Caption := IntToStr(randomNumber);

if randomNumber=1 then

AssignFile( f ,'1.txt');//связывается с разными файлами

if randomNumber=2 then

AssignFile( f ,'2.txt');

if randomNumber=3 then

AssignFile( f ,'3.txt');

if randomNumber=4 then

AssignFile( f ,'4.txt');

if randomNumber=5 then

AssignFile( f ,'5.txt');

Button2.Caption := IntToStr(randomNumber);

timer1.Enabled :=true;

progressbar1.Visible:=True;

RadioGroup1.Visible:=True;

button1.Visible:=true;

button2.Visible:=false;

reset(f); //Открываем файл для чтения

readln(f,s); //Считываем первую строку из файла

ball:=0; //изначально количество баллов 0

repeat

if (s[1]='-') then begin //Если первый символ строки ‘-‘ значит это вопрос

delete(s,1,1);

RadioGroup1.Caption:=s;

end

else if s[1]='\*' then begin //Если перв символ ‘\*’ значит это номер верного ответа

delete(s,1,1);

Nvern:=StrToInt(s);

end

else RadioGroup1.Items.Add(s); //Иначе это вариант ответа

readln(f,s); //Считываем следующую строку из файла

until (s[1]='-') or (Eof(f)); //Считывание и отправление вариантов ответов в RadiGroup до тех пор

// пока не достигнут следующий вопрос или конец файла

end;

procedure TForm12.Timer1Timer(Sender: TObject);

begin

progressbar1.Position :=progressbar1.Position +1 ;

if progressbar1.position = 420 then

begin

timer1.Enabled :=false;

Showmessage('Время вышло');

close;

end;

end; end.