

Создание приложения “EnginCalc” в программной среде Lazarus
Автор: Шакиров Артур, 10 класс, 15 лет, МБОУ КСОШ имени героя
РФ В.И.Шарпатов, г. Новый Уренгой

Руководитель: Шарапов Тимур Фархатович, учитель информатики,
МБОУ КСОШ имени героя РФ В.И.Шарпатов, г. Новый Уренгой

Введение

В нашем обществе компьютер играет очень важную роль, уже трудно представить труды ученых, математиков, бухгалтеров, экономистов без использования вычислительной техники. Но компьютер сам не способен даже на простые операции, поэтому для того чтобы люди могли его использовать, необходимо программное обеспечение.

Программа управляет ресурсами компьютера и предоставляет их в распоряжение пользователя. Существует большое количество разнообразных по своим функциям и возможностям программ, но среди них выделяют два основных типа: системное и прикладное программное обеспечение. К системному программному обеспечению относят операционную систему и все программы, предназначенные для создания рабочей среды для прикладных программ, которые в свою очередь выполняют ту самую, необходимую пользователю, работу.

Актуальность этой темы обусловлена популярностью математических расчетов в повседневной работе, а т.к. в большинстве случаев работать происходит за компьютером, то, следовательно, калькулятор должен быть виртуальным.

Задачи:

1. Изучить основы языка Lazarus
2. Ознакомиться с технологией разработки программного обеспечения на всех этапах: от постановки задачи, выбора алгоритма, его реализации до сопровождения программы

3. Разработать программный продукт.

Программная среда: Lazarus

Предмет: Приложение EnginCalc

Методы: Поисковый, аналитический, ИКТ

1.Выбор программных средств и среды разработки реализации калькулятора

Для разработки и реализации приложения была выбрана следующая программа:

Free Pascal Compiler это свободно распространяемый компилятор языка Паскаль с открытыми исходными кодами, распространяется на условиях GNU General Public License (GNU GPL). Он совместим с Borland Pascal 7 и Object Pascal – Delphi, но при этом обладает рядом дополнительных возможностей, например, поддерживает перегрузку операторов.

Lazarus это бесплатный инструмент разработки с открытым кодом. IDE Lazarus представляет собой среду с графическим интерфейсом для быстрой разработки программ, аналогичную Delphi. В принципе такого набора достаточно для создания программ с графическим интерфейсом и приложений, работающих с базами данных и Интернетом.

Lazarus это стабильная богатая возможностями среда разработки для создания самостоятельных графических и консольных приложений. В настоящее время она работает на Linux, FreeBSD и Windows и предоставляет настраиваемый редактор кода и визуальную среду создания форм вместе с менеджером пакетов, отладчиком и графическим интерфейсом, полностью интегрированным с компилятором FreePascal

1.1 Описание IDLE Lazarus

При открытии приложения (Приложение 1) мы видим несколько компонентов, для начала разберем их.

Сверху располагается “ Главное окно ”, с помощью него можно управлять процессом разработки приложения. В нем предусмотрены команды управления файлами, компиляцией, редактированием, окнами и т.д

Слева находится ”Инспектор объектов”. В верхней части окна показывается множество объектов, снизу, расположены три вкладки: "Свойства", "События", "Избранное". Во вкладке "Свойства" перечисляются все свойства выбранного объекта. Во вкладке "События" перечисляются все события для объекта. Во вкладке "Избранное" избранные свойства и методы.

А справа расположен “Редактор исходного кода”(Приложение 2). Именно в этом окне я буду набирать код программы. Многие функции и возможности этого редактора совпадают с возможностями обычных текстовых редакторов, например Блокнота. Текст в редакторе можно выделять, копировать, вырезать, вставлять. Кроме того, в редакторе можно осуществлять поиск заданного фрагмента текста, выполнять вставку и замену. Но, конечно, этот редактор исходных текстов Lazarus обладает еще рядом дополнительных возможностей для комфортной работы применительно к разработке программ. Основное преимущество редактора заключается в том, что он обладает возможностями подсветки синтаксиса, причём не только Pascal, но и других языков, а также рядом других удобств.

2. Практическая часть

Начнем создание математического приложения с главной страницы. Главная страница – это страница, на которую попадает пользователь, как только открывает приложение (Приложение 3).

На этой странице я расположил 3 кнопки:

- Калькулятор (2.1 и приложение 4). На этой вкладке располагается инженерный калькулятор.
- Уравнения (2.2, 2.3, 2.4 и приложение 5). На этой вкладке расположено приложение для решения квадратных уравнений.

- Графики функций (2.5 и приложение 6). На этой вкладке расположено приложение для построения графика функции по множеству формул.

2.1 Калькулятор

Мы пользуемся простыми калькуляторами для математических вычислений в школе и для подсчета денег в магазине. Ученые, инженеры и статистики пользуются другими калькуляторами, способными выполнять сложные операции. Современные калькуляторы - это электронные приборы с маленькими силиконовыми микросхемами, производящими любые вычисления. Разновидность калькулятора - кассовый аппарат. Он суммирует цены и делает распечатку чека. Большинство кассовых аппаратов автоматически считывают цены по бар коду, указанному на каждом товаре. "Мозг" калькулятора может только складывать и вычитать. Умножение и деление он выполняет путем многократного сложения или вычитания.

Калькулятор - программа предназначенная для выполнения несложных вычислительных операций: сложения, вычитания, умножения и деления и т.д. В калькуляторе также предусмотрены возможности выполнения более сложных операций для инженерных и статистических вычислений (Приложение 4)

Перед созданием кода, я расположил однострочное поле Edit, которое отвечает за ввод и вывод чисел и кнопки Button. После этого я написал код программы в редакторе исходного кода (Приложение 2).

2.2 Решение квадратного уравнения.

Найти корни квадратного уравнения $AX^2+BX+C=0$, коэффициенты А, В, С заданы и вводятся с клавиатуры.

Из элементарной математики известна формула для нахождения корней этого уравнения:

$$X_{1,2} = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}$$

Однако эта формула применима только для случая действительных корней. Но мы считаем, что коэффициенты А, В, С могут быть произвольными, поэтому необходимо произвести анализ задачи и определить возможные варианты вычислений. Анализ задачи и определение возможных ситуаций, возникающих в ходе вычислений, является одной из важнейших функций программиста. Попытка запрограммировать только формулу может привести к неопределенной ситуации, если $A=0$, или $B^2 - 4AC < 0$. Именно программист должен предусмотреть возможность возникновения таких ситуаций и явным образом указать порядок вычислений в каждом конкретном случае.

Если $A=0$, это означает, что исходное уравнение выродилось в линейное $BX+C=0$. В этом случае решением его будет

$$X = -\frac{B}{C}$$

Так же, как в калькуляторе я расположил однострочные поля Edit, которые отвечают за ввод коэффициента и вывод ответа, кнопка Button, отвечающая за вывод ответа и, используя алгоритм решения, я написал код для решения уравнений 2 степени.

Если квадратное уравнение не имеет корней, он выводит только дискриминант.

2.3 Решение кубического уравнения.

В общем случае, корни кубического уравнения находятся по формуле Кардано. Для кубического уравнения

$$A_0x^3 + A_1x^2 + A_2x + A_3 = 0$$

находятся значения:

$$B_1 = \frac{A_1}{A_0}, B_2 = \frac{A_2}{A_0}, B_3 = \frac{A_3}{A_0}.$$

Далее находим

$$p = -\frac{B_1^3}{3} + B_2,$$

$$q = \frac{2B_1^3}{27} - \frac{B_1B_2}{3} + B_3.$$

Подставляем полученные p и q в формуле Кардано:

$$y = \sqrt[3]{-\frac{q}{2} + \sqrt{\frac{q^2}{4} + \frac{p^3}{27}}} + \sqrt[3]{-\frac{q}{2} - \sqrt{\frac{q^2}{4} + \frac{p^3}{27}}}$$

В итоге, находим корни исходного уравнения по формуле $x = y - \frac{B_1}{3}$.

Во вкладке Кубическое уравнение, я расположил однострочные поля Edit, которые отвечают за ввод коэффициентов, кнопку Button, отвечающая за вывод ответа в Мемо, где Мемо это простой текстовый многострочный редактор позволяющий вводить и выводить текст с клавиатуры, редактировать и сохранять в файл текстового формата. После, используя алгоритм решения я написал код для решения уравнений 3 степени

2.4 Решение уравнений 4 степени

В общем случае, приведенное уравнение четвертой степени вида

$$x^4 + Ax^3 + Bx^2 + Cx + D = 0 \text{ можно решить методом Феррари.}$$

Находится y_0 - любой из корней кубического уравнения

$y^3 - By^2 + (AC - 4D)y - A^2D + 4BD - C^2 = 0$. Затем решаются два квадратных уравнения

$$x^2 + \frac{A}{2}x + \frac{y_0}{2} \pm \sqrt{\left(\frac{A^2}{4} - B + y_0\right)x^2 + \left(\frac{A}{2}y_0 - C\right)x + \frac{y_0^2}{4} - D} = 0$$

в которых подкоренное выражение является полным квадратом.

Корни этих уравнений являются корнями исходного уравнения четвертой степени.

В этой вкладке, я расположил однострочные поля Edit, которые отвечают за ввод коэффициентов, кнопку Button, отвечающая за вывод ответа в Мемо. (Приложение 5) После, используя алгоритм решения, я написал код для решения уравнений 4 степени.

2.5 График функции

Это приложение строит графики элементарных функций по заданным формулам.

В форме “График функции” (Приложение 6) я расположил три вкладки, в каждую добавил разные функции. В первой вкладке находятся логарифмические и показательные функции, во второй – тригонометрические, а в третьей – степенные. В каждой вкладке есть однострочные поля Edit, которые отвечают за ввод $\min$, $\max$ – они отвечают за промежуток, коэффициентов a , b , c и кнопки Button, отвечающие за вывод ответа в chart определенной функции. На каждой кнопке написано, к какой функции он принадлежит.

Заключение

Мы считаем, что на современном этапе информатизации общества программирование на разных языках остается очень актуальным.

Цель работы достигнута в полном объеме. Результатом нашей работы стало готовое к работе приложение, которое выполняет множество различных простых и сложных вычислений.

В ходе проведенной работы нами были изучены основы языка Lazarus, мы познакомились с технологией разработки программного обеспечения на всех этапах: от постановки задачи, выбора алгоритма, его реализации до сопровождения программы и разработали приложение. Оценивая проделанную работу, можно сделать следующие выводы:

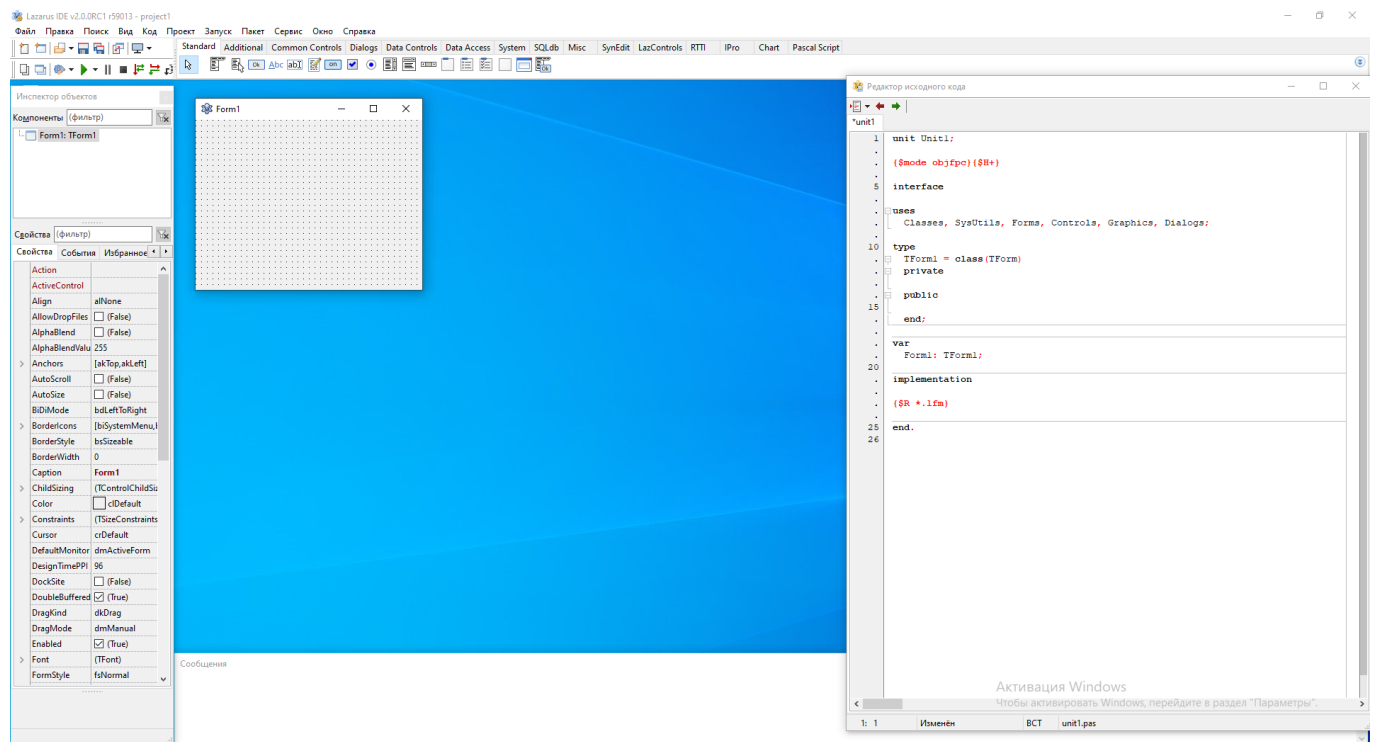
- Приложение предоставляет пользователям готовую продукцию, которая делает точные сложные вычисления, калькулятор прост и удобен в использовании, к тому же он с современным интерфейсом.

В дальнейшем разработанное приложение можно будет использовать в реальной жизни. Так же возможно обновление и добавление новых функций.

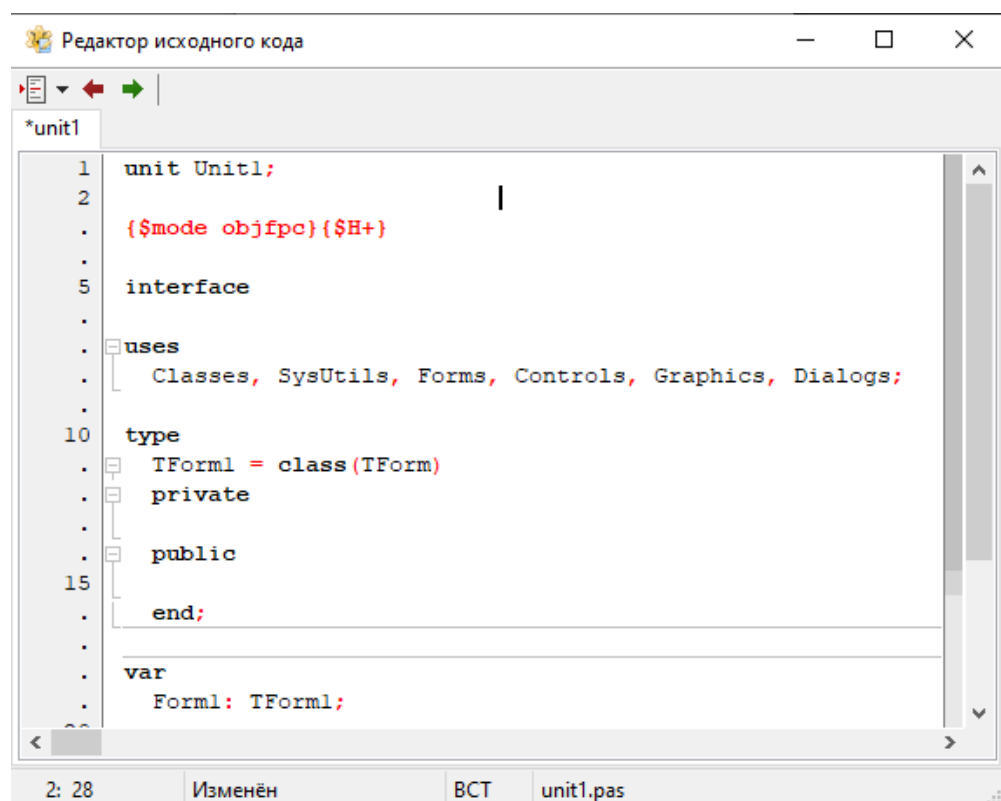
Данный продукт стабилен в работе и не требует высоких знаний для работы с ним.

Список использованных источников

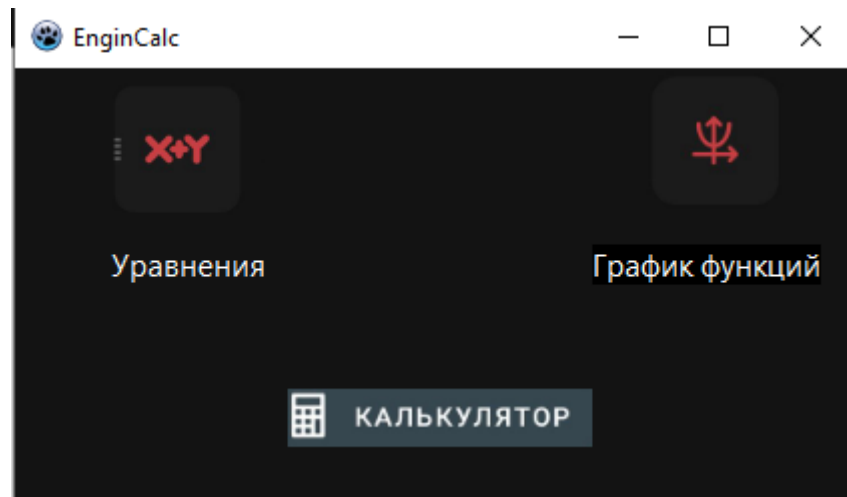
1. Учебник “Основы программирования в среде Lazarus” – Мансуров К.Т. – М.: Бином – 2010г. – 127 с. Дата обращения: 20.09.2021
2. Книга для чтения “Экспресс курс программирования в Lazarus” – Е.А. Попов – 2020г. – 97 с. Дата обращения: 20.09.2021
3. Ершов А. П. “Введение в теоретическое программирование: Беседы о методе” – М.: Книга по Требованию – 2012г. – 288 с. Дата обращения: 16.09.2021
4. Михайлов Т.Ю. “Характеристика среды Lazarus” – М.: Наука – 2009г. – 211 с. Дата обращения: 5.10.2021
5. Селиванова Д.М. ”Основы программирования в алгоритмической” – М.: Бином – 2010г. – 139 с. Дата обращения: 26.09.2021
6. Сайт Интернет-урок URL: <http://www.cleverstudents.ru/> Дата обращения: 8.10.2021
7. Учебные материалы URL: <https://studwood.ru/> Дата обращения: 17.10.2021



Приложение 1 “ Вид IDE Lazarus в Windows ”



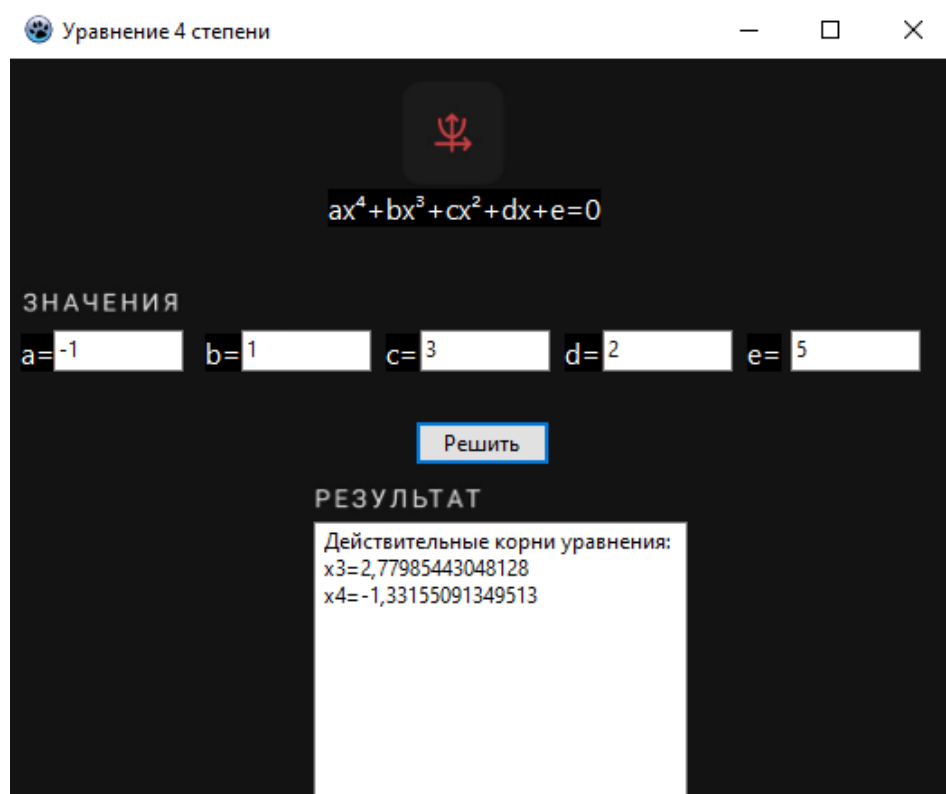
Приложение 2 “Редактор исходного кода”



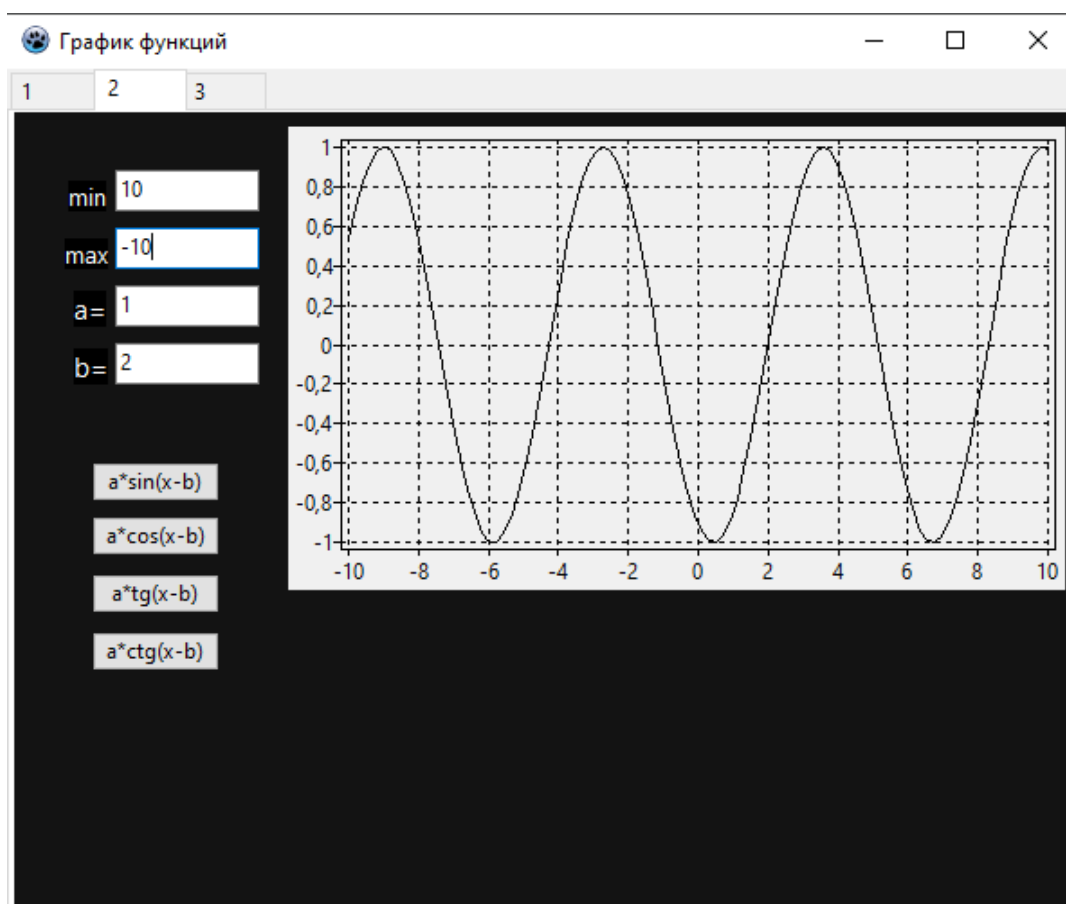
Приложение 3 “Главная страница”



Приложение 4 “Калькулятор”



Приложение 5 “Уравнение 4 степени”



Приложение 6 “Графики функций(на изображении график “ $a \cdot \sin(x-b)$ ”)”