



**«Московский государственный технический университет
имени Н.Э. Баумана»
(национальный исследовательский университет)
(МГТУ им. Н.Э. Баумана)**

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ
КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

О т ч е т

по лабораторной работе № 1

Название лабораторной работы: Создание консольных приложений
в среде Lazarus

Дисциплина: Основы программирования

Студент гр. ИУ6-13Б 05.09.2022 В. К. Залыгин
(Подпись, дата) (И.О. Фамилия)

Преподаватель _____
(Подпись, дата) (И.О. Фамилия)

Москва, 2022

Часть 1

Задание 1

Создать консольное приложение для вычисления корней квадратного уравнения.

Ход работы:

1. Создаю проект с шаблоном «Пустая программа».

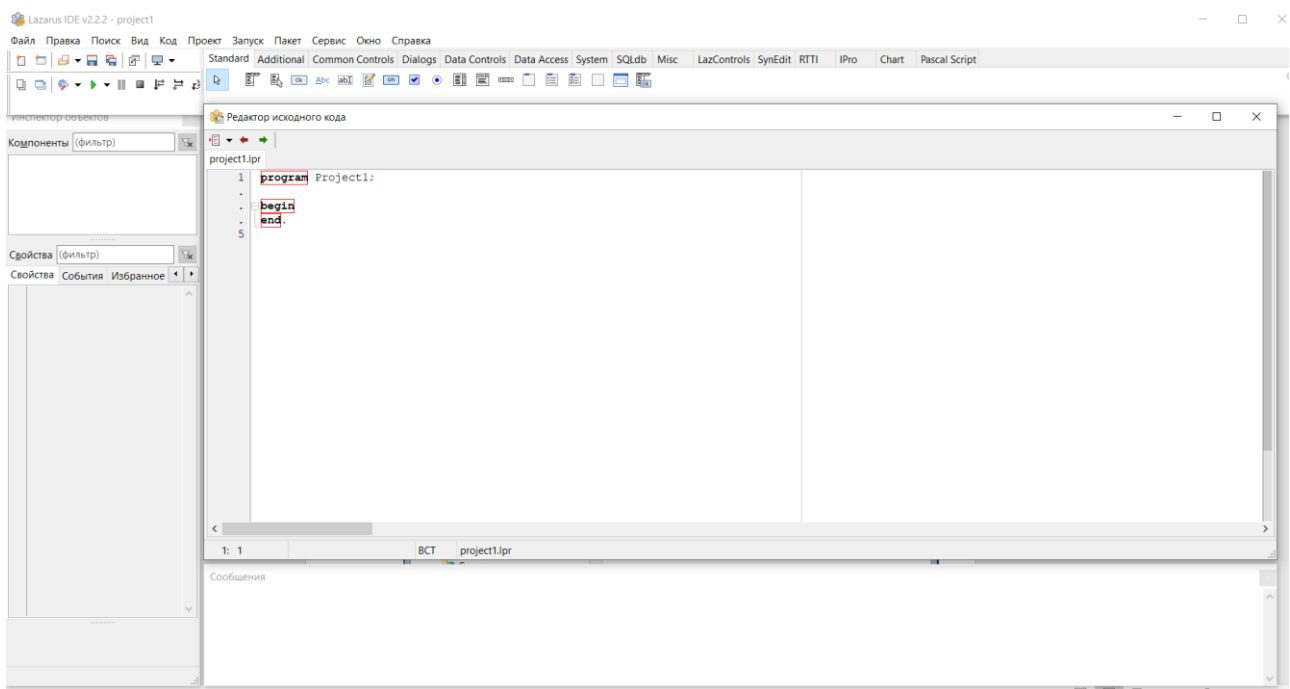


Рисунок 1 – Создание пустой программы

2. Сохраняю проект в папку.

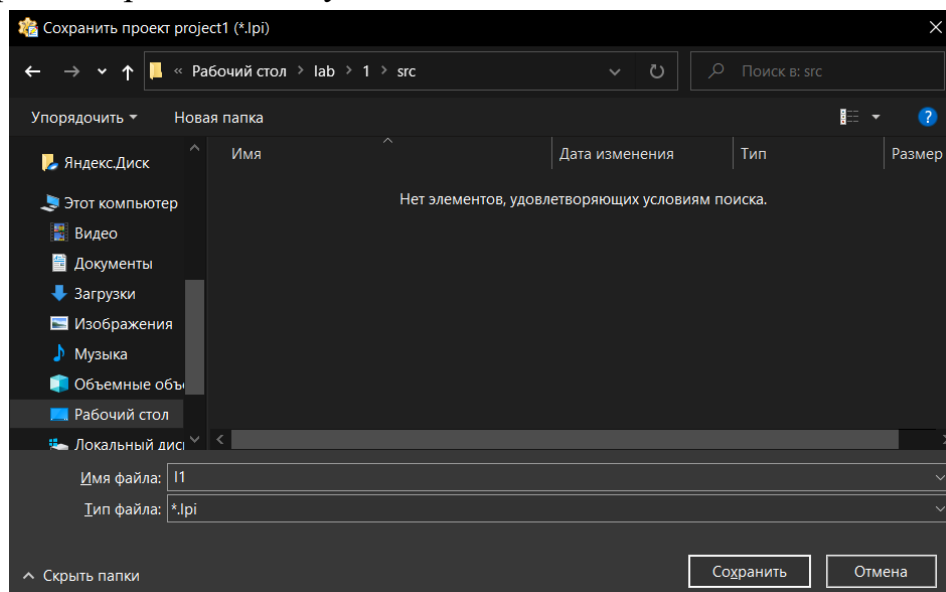


Рисунок 2 – Сохранение программы

3. Ввожу код.

```
1  program l1;  
  .  uses SysUtils;  
  .  var a, b, c, d, e, x1, x2: single;  
  .  begin  
  .      WriteLn('Input A, B, C:');  
  .      ReadLn(a, b, c);  
  .      d := b*b - 4*a*c;  
  .      if d >= 0 then  
  .          begin  
  .              e := 2*a;  
  .              x1 := (-b+sqrt(d))/e;  
  .              x2 := (-b-sqrt(d))/e;  
  .              WriteLn('x1 = ', x1:6:1, ' x2 = ', x2:6:1);  
  .          end  
  .      else WriteLn('No result');  
  .      ReadLn;  
  .  end.  
18
```

Рисунок 3 – Введённый код

4. При нажатии на кнопку F9 проект собирается и компилируется.

Сообщения

Компиляция проекта, цель: l1.exe: Успешно

Рисунок 4 – Результат сборки и компиляции

5. Программа запускается и корректно выводит ответ.

```
C:\Users\syava\OneDrive\Рабочий стол\lab\1\src\l1.exe  
Input A, B, C:  
1 3 2  
x1 =  -1.0 x2 =  -2.0  
_
```

Рисунок 5 – Исполнение программы

Задание 2

Таблица 1 – Типы ошибок и диагностические сообщения

	Ошибка	Физический смысл ошибки	Проявление ошибки	Расшифровка сообщения	Этап
1.	<i>Var AB, C, X1, X2, D, E;</i>	Вместо переменных А и В описана переменная АВ	Получено сообщение: Identifier not found «А» и др.	Не найдена переменная А	Компиляция
2.	<i>Readln(A,B,C)</i>	Нет “;”	Syntax error, “;” expected	Не поставлена точка с запятой	Компиляция
3.	<i>E=2*A;</i>	Использование неверного оператора	Illegal expression	Ожидается оператор присваивания, но он не найден	Компиляция
4.	<i>D:=spr(B)-4*A*C;</i>	Вызвана несуществующая функция	Identifier not found “spr”	Функции под именем “spr” не найдена	Компоновка
5.	Исходные данные: 0 1 3	Деление на 0	Invalid operation	Попытка выполнить недопустимую операцию	Исполнение
6.	Исходные данные: 1 1 3	Введены некорректные данные	No result	Нет возможности решить уравнение с данными коэффициентами	Исполнение
7.	<i>{ E:=2*A; }</i>	Пропущена строка вычислений	Variable “e” does not seem to be initialized	Переменной не задано значение	Исполнение
8.	<i>Read(A,C);</i>	Не определено (не введено) значение В	Variable “c” does not seem to be initialized	Значение для переменной не считано с консоли	Исполнение

Выводы: Получены знания о том, как создавать проект консольного приложения из шаблона, писать простые приложения, вносить и исправлять различного рода ошибки, возникающие на разных этапах работы программы.

Задание 3

1. Я внёс ошибку 7 в программу. В момент вычисления корней переменная E имеет значение 0. Так как в программе пропущена строка с инициализацией этой переменной, в ней записано значение по умолчанию, то есть 0.

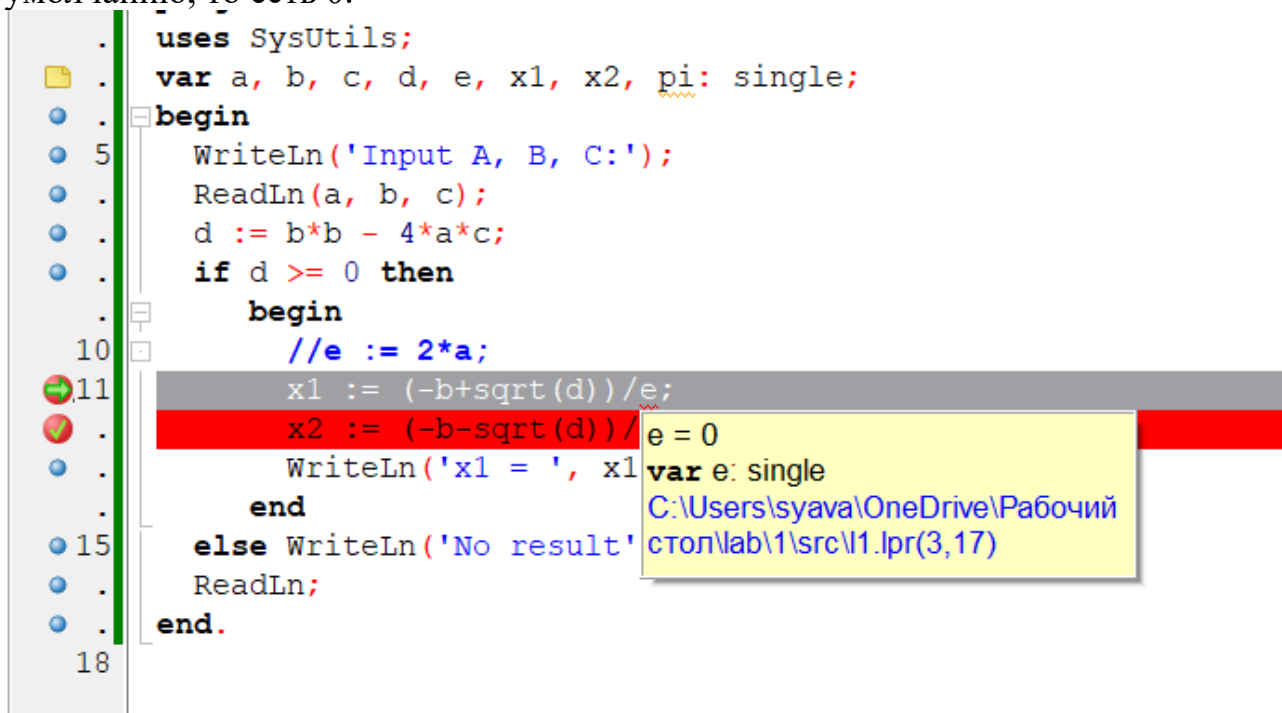


Рисунок 6 – Результат внесения в программу ошибки 7

2. Я внёс ошибку 6 в программу. В момент исполнения переменная D имеет значение меньше 0. Ошибка обнаруживается во время проверки условия в блоке if.

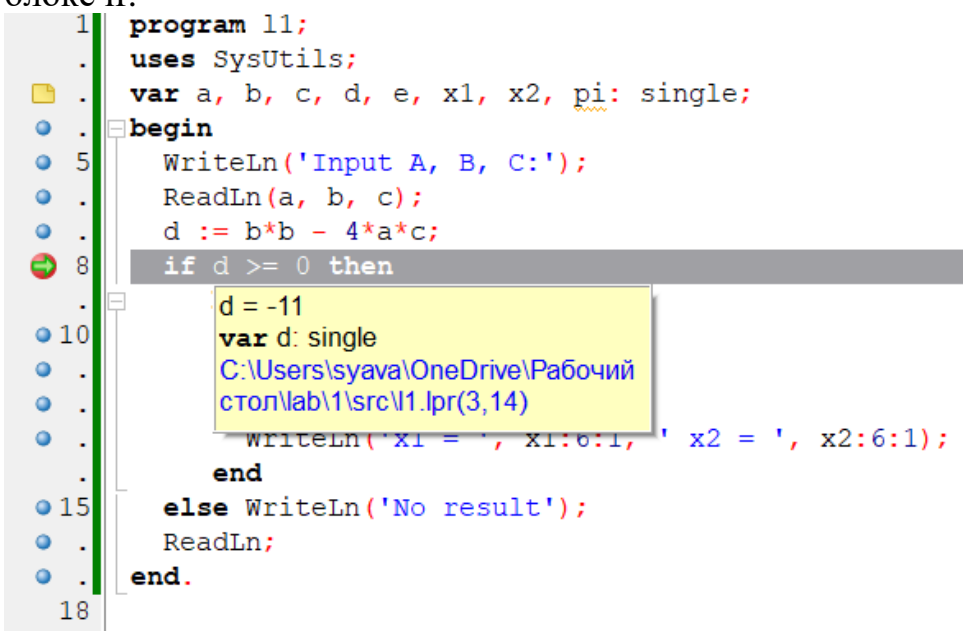


Рисунок 7 - Результат внесения в программу ошибки 6

3. При входных данных 1 1 3 в переменные записываются следующие значения:
 $A = 1, B = 1, C = 3.$

Вывод: изучены средства отладки в программе Lazarus, получены знания о точках останова и просмотре переменных в разные моменты исполнения программы.

Часть 2

Созданную в Miro схему можно увидеть на рисунке 8.

Созданную в Lucid схему можно увидеть на рисунке 9.

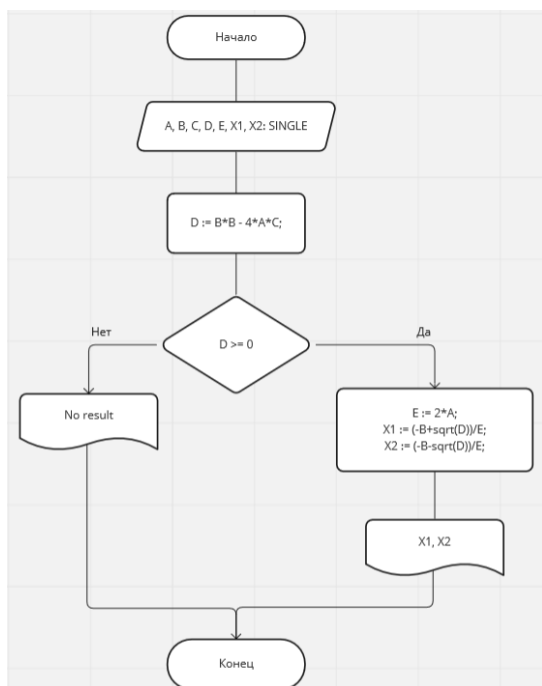


Рисунок 8 – диаграмма в Miro

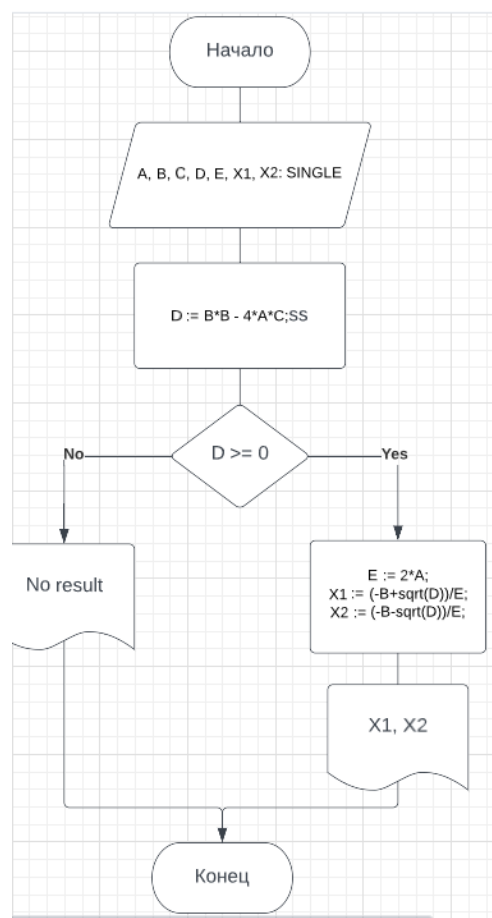


Рисунок 9 – диаграмма в Lucid

Вывод: было изучено несколько программ для построения схем. На взгляд автора, наибольшим удобством обладает сервис Lucid.