*Федеральное государственное бюджетное образовательное учреждение* *высшего образования*

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

ФАКУЛЬТЕТ \_\_\_\_ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ\_\_\_\_\_\_\_\_

КАФЕДРА \_\_\_\_\_\_Компьютерные Системы и сети (ИУ6)\_\_\_\_\_\_\_\_\_\_\_

**Отчет**

**по лабораторной работе № 1**

**Часть 1**

**Название лабораторной работы: Создание консольных приложений в среде Turbo Delphi 2006**

**Дисциплина: Основы программирования**

Студент гр. ИУ6-14Б **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

(Подпись, дата) (И.О. Фамилия)

Преподаватель **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

(Подпись, дата) (И.О. Фамилия)

Москва, 2018

**Задание 1**

1. Для создания заготовки консольного приложения мы выбрали пункт **File/New/ Other…** (пункт **File** подпункт **New** и подпункт **Other…**). На экране появляется окно выбора создаваемых проектов. Выберитем иконку **Console Application**. После этого на экране появилась заготовка консольного приложения со стандартным именем **Project1.**

2. В рабочей области на экране теперь два окна: панель структуры и вкладка многооконного редактора текстов программ.

В окне структуры отображается дерево объектов программы: переменных функций, типов, библиотек и т. п.

3. Сохраняем созданный проект. Чтобы избежать дублирования имен проектов, каждый отдельный проект сохраняем ***в*** ***отдельной папке***. Поэтому перед сохранением создаем новую папку для этого проекта, например, *проект1*. Затем используем пункт меню **File/Save**. На экране появляется диалог **Save Project1 as**. В дереве файлов находим созданную папку, задаем имя программы в окне *Имя файла*, например, *проект1* и нажимаем на кнопку **Сохранить.** После этого, изменится как имя файла проекта, так и имя на вкладке многооконного редактора программ.

4. На переименованной вкладке Example1 многооконного редактора программ высвечивается заготовка консольного приложения. В этом окне набираем текст программы.

5. Для выполнения программы ми используем кнопку быстрого вызова **Run** (зеленый треугольник на панели инструментов), пункт меню **Run/Run** иликлавишу **F9**.

6. На экране появилось окно Консоль, в которое будет выведен запрос на ввод чисел. Числа вводим через пробел или каждое число в своей строке, нажимая после ввода числа клавишу Enter. После ввода чисел в то же окно выводиться результат

При этом вид окна среды за окном консоли изменится, на нем появятся панели режима отладки программы: Call Stack – стек вызовов, Watch List – список отслеживаемых значений переменных, Local Variables – значения локальных переменных, Event Log – перечень событий. Особенности работы с этими окнами будут рассмотрены далее.

Код программы (рис. 1.1):

Стр. 2



рис. 1.1

*program Example1;*

*{$APPTYPE CONSOLE}*

*uses SysUtils;*

***Var A, B, C, D, E, X1, X2:Single;***

*Begin* ***WriteLn('Input A, B, C:');***

***ReadLn(A,B,C);***

***D:= Sqr(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.*

Результат выполнения программы (рис. 1.2):



рис. 1.2

Стр. 3

Блок схема данной программы (рис. 1.3):

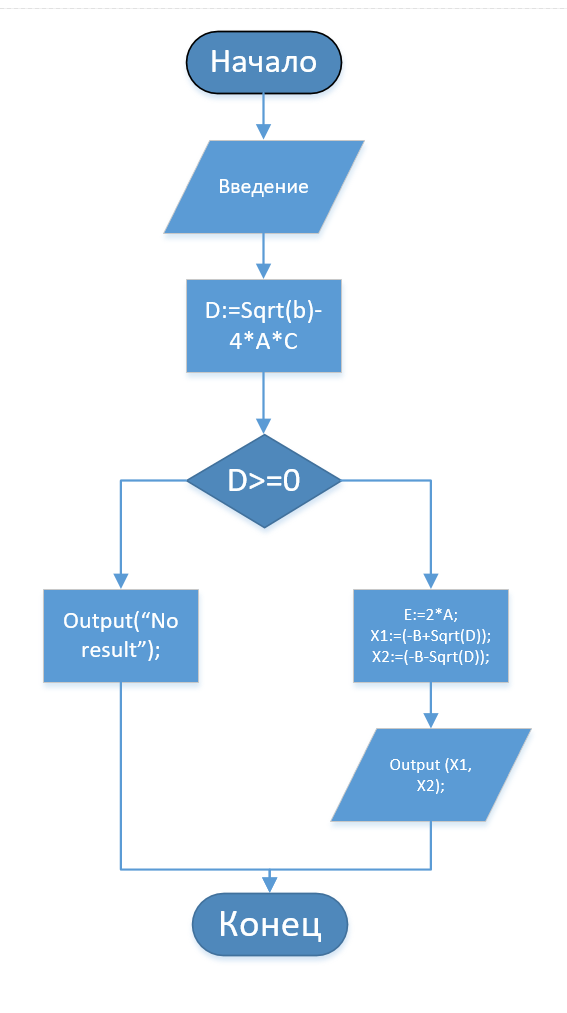


рис. 1.3

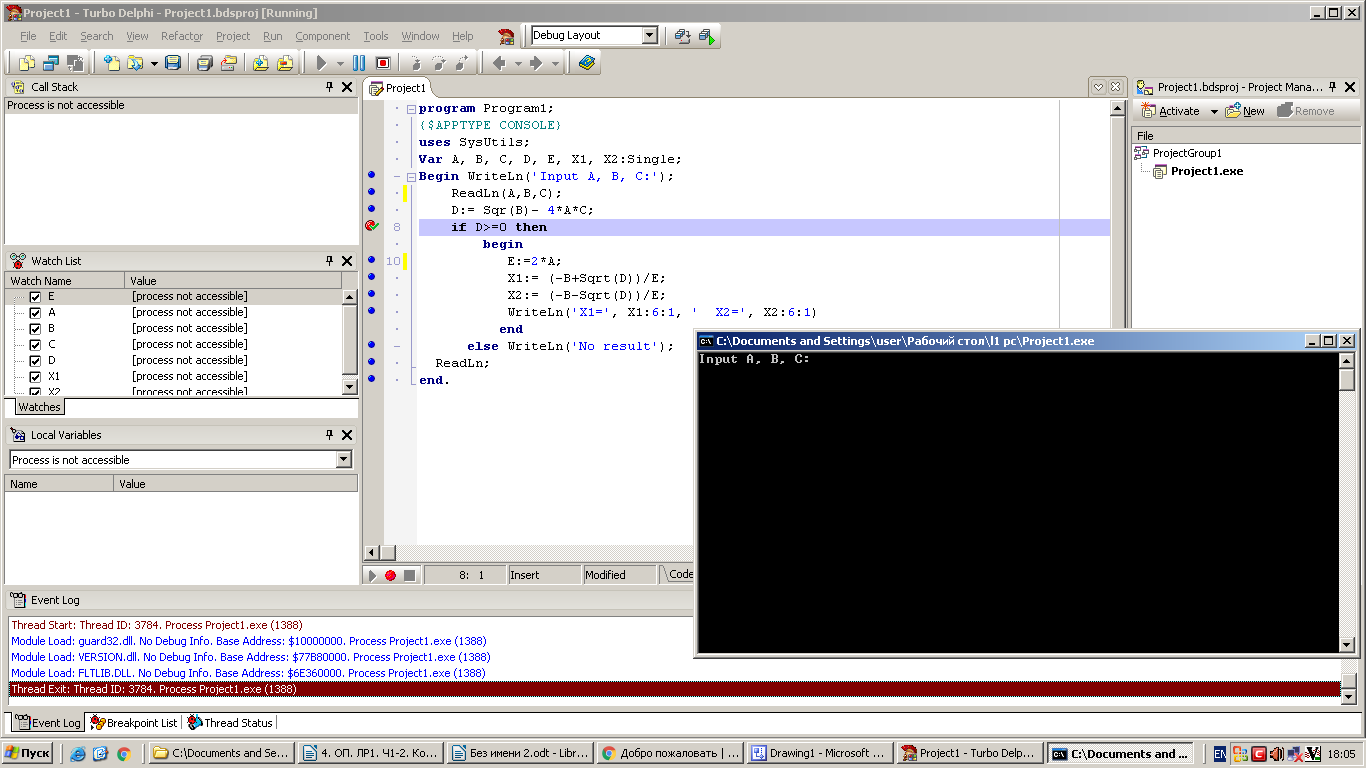
Код работающий программы с отладкой (рис. 1.4):

рис. 1.4

Стр. 4

**Задание2**

**Изучить диагностические сообщения Turbo Delphi.**

Поочередно внося ошибки в программу, мы фиксировали сообщения об ошибках в специальной таблице, представленной ниже. Классифицировали ошибки, расшифровывали сообщения системы и определяли этапы выполнения программы (компиляция, компоновка или выполнение), на котором была обнаружена данная ошибка.

**Таблица –** Диагностические сообщения

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Ошибка | Физический смысл ошибки | Проявление  ошибки | Расшифровка сообщения | Этап |
| 1. | *Var AB, C,*  *X1, X2, D, E;* | Вместо переменных A и B описана переменная AB | Получено сообщение: Undeclared identifier: 'A' | Не объявлена переменная A | Компиляция |
| 2. | Readln(A,B,C) | Нет (;) | Missing operator or semicolon | Отсутствует тоска с запятой | Компиляция |
| 3. | E=2\*A; | Отсутствует присвоение | ':=' expected but '=' found | Ожидается опирация присвоения (“:=”), но ее нет | Компиляция |
| 4. | *D:=spr(B)-4\*A\*C;* | Вызвана несуществующая функция | Undeclared identifier: 'spr' | Не существует оператора Spr | Компиляция |
| 5. | Исходные  данные: 0 1 3 | Переменная А не может являтся нолю | Floating point operation at 0x0040920f | Происходит деление на ноль | Выполнение |
| 6. | Исходные  данные: 1 1 3 | Дискриминант меньше 0 | No result | Нет результата | Выполнение |
| 7. | { *E:=2\*A;* } | Пропущена строка вычислений | Floating point divide by zero at 0x00409fc | Деление на переменную с плавующей запятой на число равное нулю | Компановка |
| 8. | *Read(A,C);* | Не определено (не введено) значение B | Выводил (1; -5; 4)  Выводит (2.2; -2.2) | Пропускает слагаемое и на место (“c”) ставит -5, из-за этого работает не правильно | Выполнение |

Стр. 5

**Задание3**

**Изучить средства отладки программ в среде Delphi.**

Задание:

1. Внесите в программу ошибку 7. Выполните пошаговую трассировку программы, отслеживая значения переменных A, B, C, D, E, x1, x2 в окне Watch. Какое значение имеет переменная E в момент вычисления корней уравнения? Почему?
2. Внесите в программу ошибку 6. Выполните пошаговую трассировку программы, отслеживая значение переменных A, B, C, D, E, x1, x2. Какое значение имеет переменная D? В какой момент обнаруживается ошибка?
3. Установите точку останова перед вычислением дискриминанта. Выполните программу до точки останова. Просмотрите значения переменных, подводя к ним курсор мыши.

3. Ответы на вопросы и выводы занести в отчет.

Ответ:

1. В момент ошибки переменная «E» равна 0. Программа выдает ошибку, так как происходит деление на ноль.

2. Переменная «D» равна -11. В момент, когда посчитался дискриминант и считаются переменные X1 и X2.

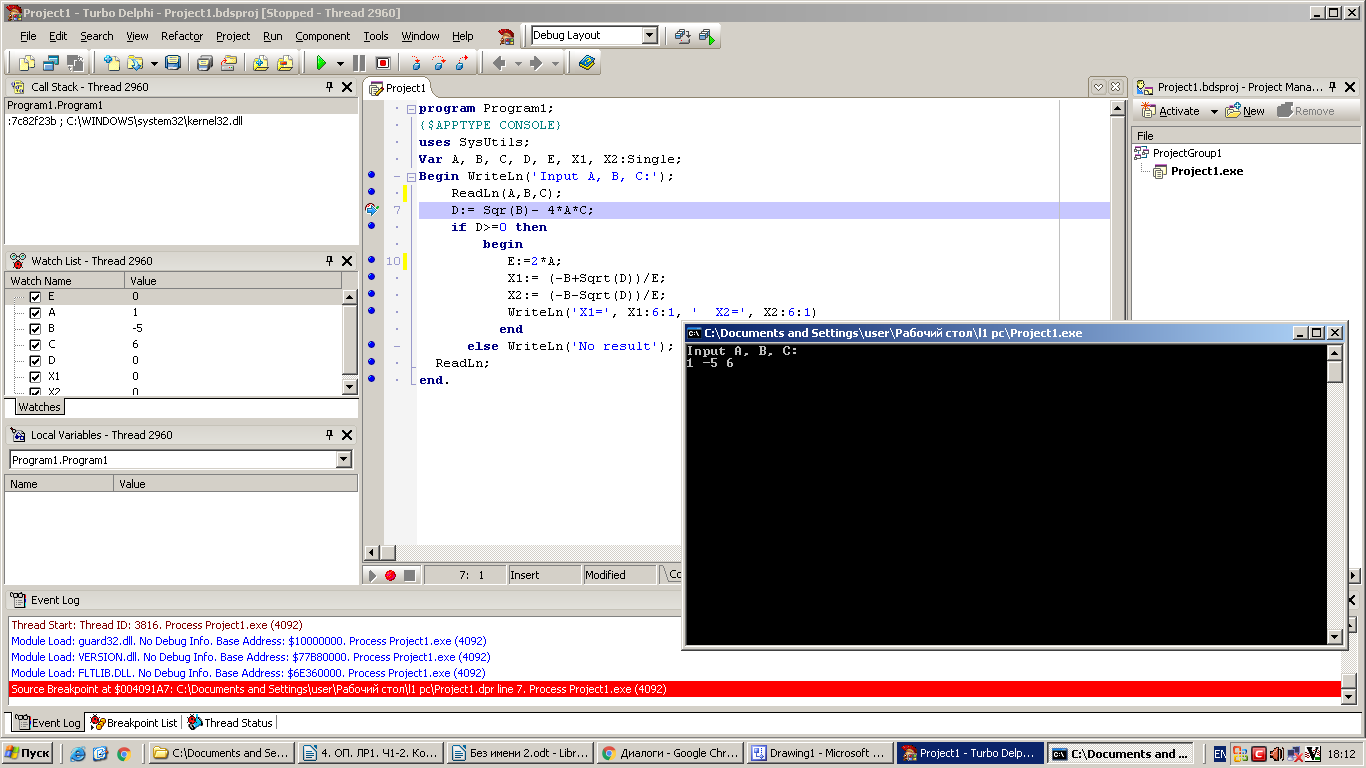
3. Значения переменных показаны на рисунке 1.5:

рис. 1.5

Стр. 6

**Вывод:** разработано консольное приложение в среде программирования Turbo Delphi с учётом работы отладчика.

**Часть 2**

**Создание схем алгоритмов средствами Microsoft Visio и OpenOffice Draw**

Схему созданную в Microsoft Vision можно увидеть на рисунке 2.1 слева.

Схему созданную в OpenOffice Draw можно увидеть на рисунке 2.1 справа.

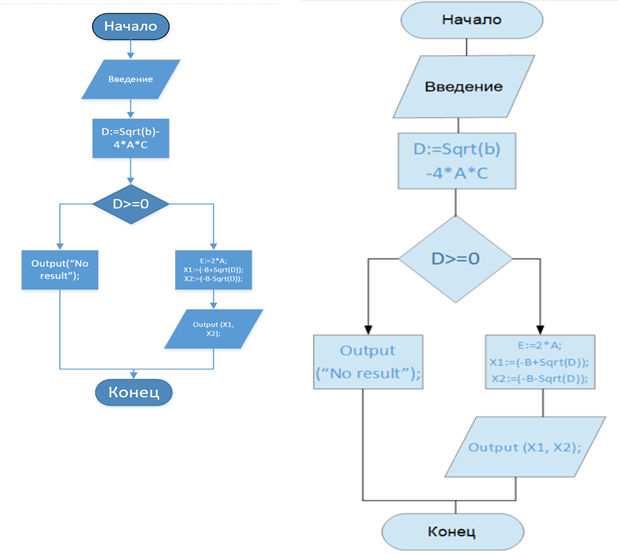


рис. 2.1

Вывод: для меня программа Microsoft Visio оказалась удобнее OpenOffice Draw

Стр. 7

Начало

Output(«No result»)

Конец

Output(X1,X2)

E:= 2\*A;

X1:=(-B + Sqrt(D))/E;

X2:=(-B-Sqrt(D))/E

Да

Нет

D >= 0

D := Sqr(B)- 4\*A\*C;

Input(a, b, c)