

# Диаграмма классов



## Описание классов

### NonlinearEquation:

Поля:

```
double a; // коэффициент перед X
double intervalStart; // начало интервала
double intervalEnd; // конец интервала
double epsilon; // точность
```

Конструктор и Деструктор:

```
NonlinearEquation(double coefficient, double start, double end, double eps) //
конструктор с параметрами
: a(coefficient), intervalStart(start), intervalEnd(end), epsilon(eps) {}

~NonlinearEquation() // деструктор без параметров {}
```

Методы:

```
double getA() const { // вызываем и получаем коэффициент}

double getIntervalStart() const { // вызываем и получаем начало отрезка}

double getIntervalEnd() const { //вызываем и получаем конец отрезка}

double getEpsilon() const { // вызываем и получаем точность}

double calculateEquationValue(double x) { // вызываем и получаем функцию}

double calculateEquationDerivative(double x) { // вызываем и получаем первую производную}

double solveByBisectionMethod() { // метод деления отрезка пополам}
```

## NewtonMethodSolver:

### Конструктор и Деструктор:

```
NewtonMethodSolver(double coefficient, double start, double end, double eps) //  
конструктор с параметрами  
    : NonlinearEquation(coefficient, start, end, eps) {}  
  
~NewtonMethodSolver() // деструктор без параметров{}
```

### Методы:

```
double solveByNewtonMethod(){};
```

## Описание пользовательского интерфейса:

### Функции пользователя:

1. Выбор метода решения

При входе в программу пользователь видит меню

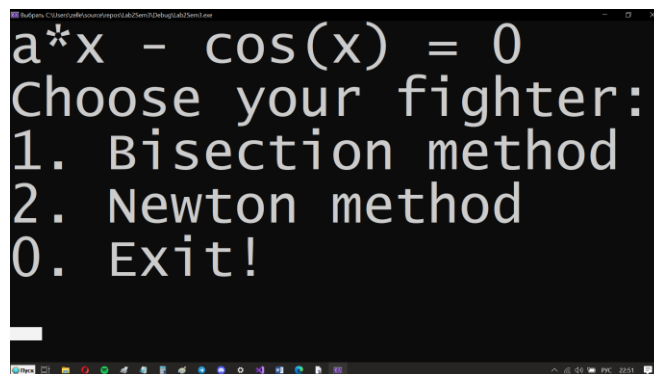


Рисунок 1 Выбор метода

С помощью ввода с клавиатуры пользователь выбирает метод решения  
Если сделать выбор, которого нет в программе, программа выдаст следующее окно:

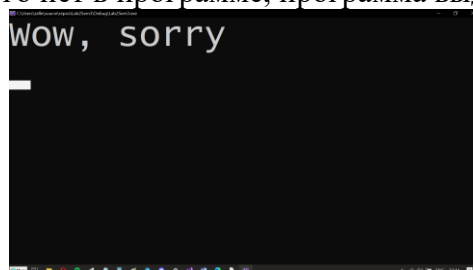


Рисунок 2 Ошибка ввода выбора метода

Нажав Enter, пользователь вернется на стартовый экран

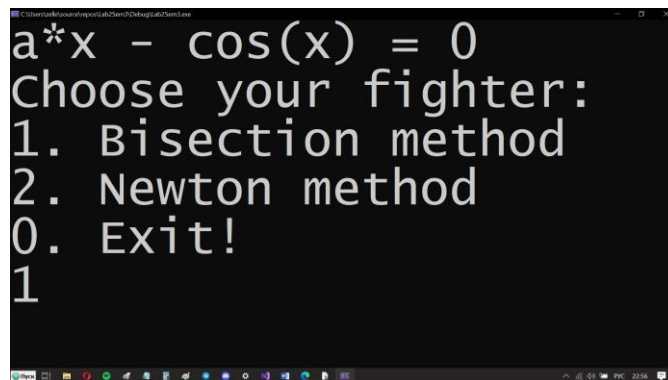


Рисунок 3 Ввод данных для выбора метода

Нажимая Enter, пользователь переходит на следующее окно

## 2. Ввод параметров

При вводе корректных значений выплывает следующее меню:

Ввод параметров происходит через ручной ввод (клавиатура). Пользователь вводит 4 параметра через пробел (коэффициент, начало отрезка, конец отрезка, точность)

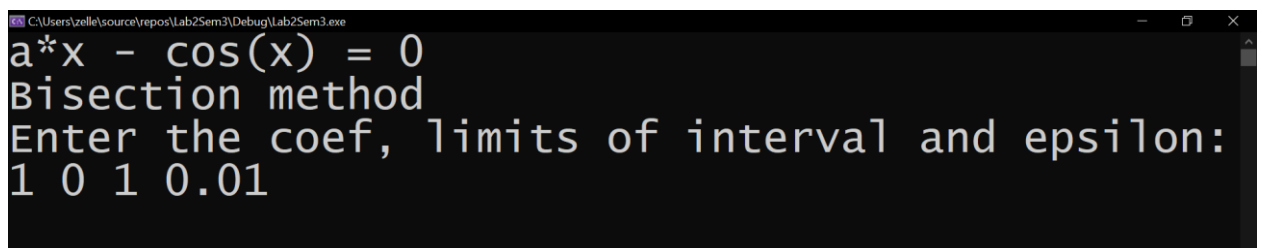


Рисунок 4 Ввод параметров

Нажимая Enter, пользователь получает результат

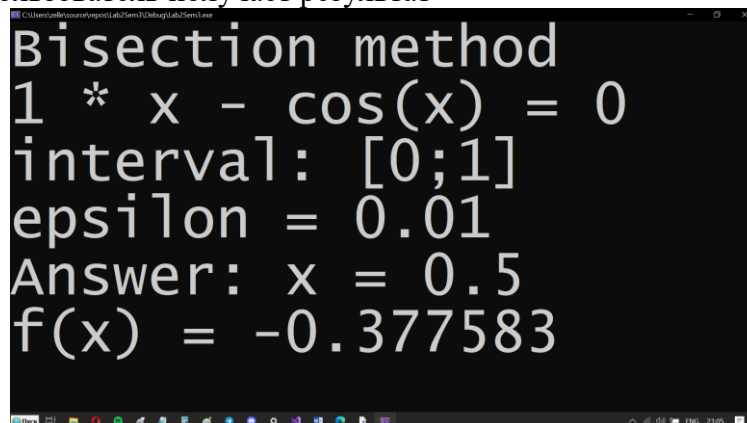


Рисунок 5 Результат

После нажатия Enter, пользователь переходит обратно на главный экран

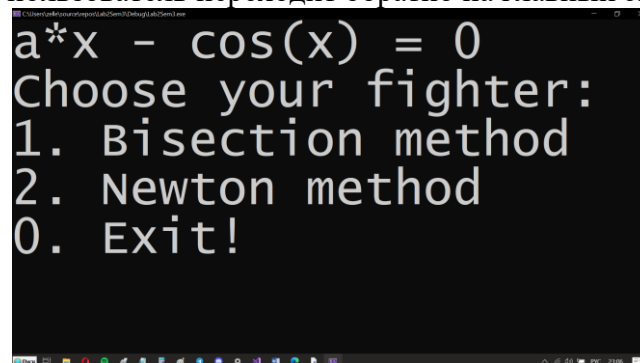


Рисунок 6 Переход на главный экран

И пользователь может заново выбрать метод, ввести данные и получить ответ или выйти из программы.

### 3. Выход из программы

При нажатии пользователем 0 и Enter – программа завершается

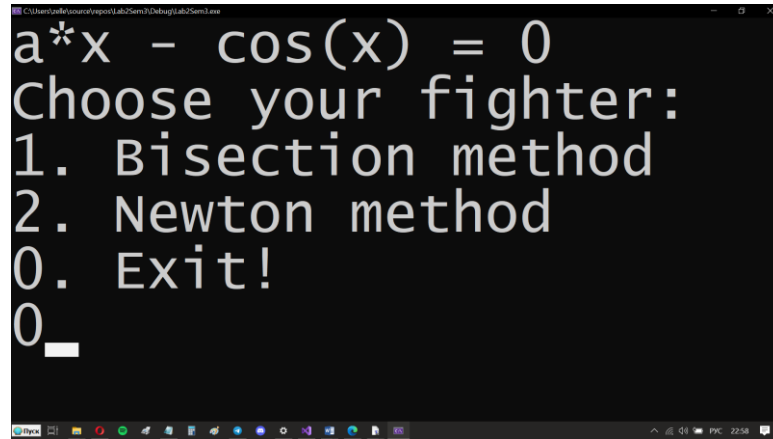


Рисунок 7 Ввод выхода из программы

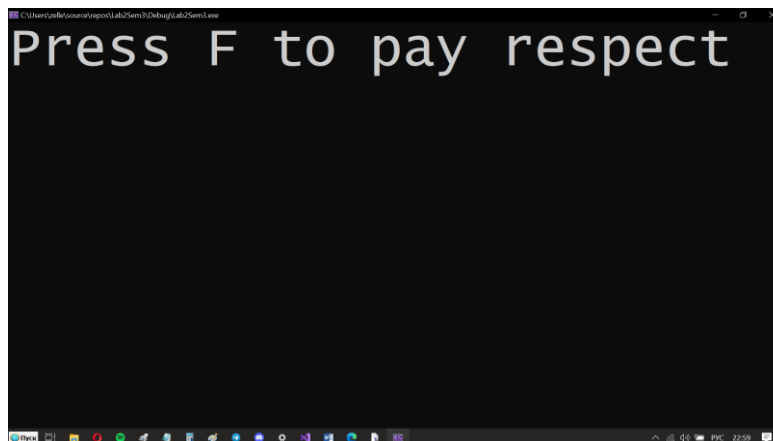


Рисунок 8 Выход из программы

При еще одном нажатии программа полностью завершается

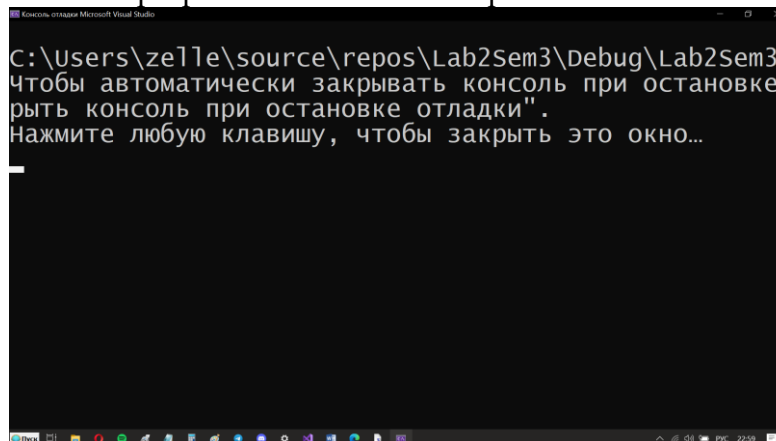


Рисунок 9 Полное завершение программы

# Тесты

```
Bisection method
10 * x - cos(x) = 0
interval: [0;5]
epsilon = 0.1
Answer: x = 0.078125
f(x) = -0.2157
```

Рисунок 11 Тест 1

```
Newton method
1 * x - cos(x) = 0
interval: [0.7;0.74]
epsilon = 1e-07
Answer: x = 0.739085
f(x) = 2.48759e-09
```

Рисунок 10 Тест 4

```
a*x - cos(x) = 0
Bisection method
Enter the coef, limits
45 1 5 0.2
Data isn't correct
```

Рисунок 13 Тест 2 некорректный ввод данных

```
a*x - cos(x) = 0
Newton method
Enter the coef, limits
-21 2 5 0.00001
Data isn't correct
```

Рисунок 12 Тест 5 некорректные данные

```
a*x - cos(x) = 0
Bisection method
Enter the coef, limits
-5 6 9 0.2
f(a)*f(b) > 0
```

Рисунок 15 Тест 3 знак функций на концах отрезка

```
a*x - cos(x) = 0
Newton method
Enter the coef, limits
2 5 9 0.1
f(a)*f(b) > 0
```

Рисунок 14 Тест 6 Знак функций на концах отрезка