

1. Введение

Реализовать метод секущих для нахождения корня функции. В методе Ньютона требуется вычислять производную функции, что не всегда удобно. Можно заменить производную первой разделённой разностью, найденной по двум последним итерациям, т. е. заменить касательную секущей. Тогда получим Для

$$x_{n+1} = x_n - \frac{(x_n - x_{n-1})f(x_n)}{f(x_n) - f(x_{n-1})}$$

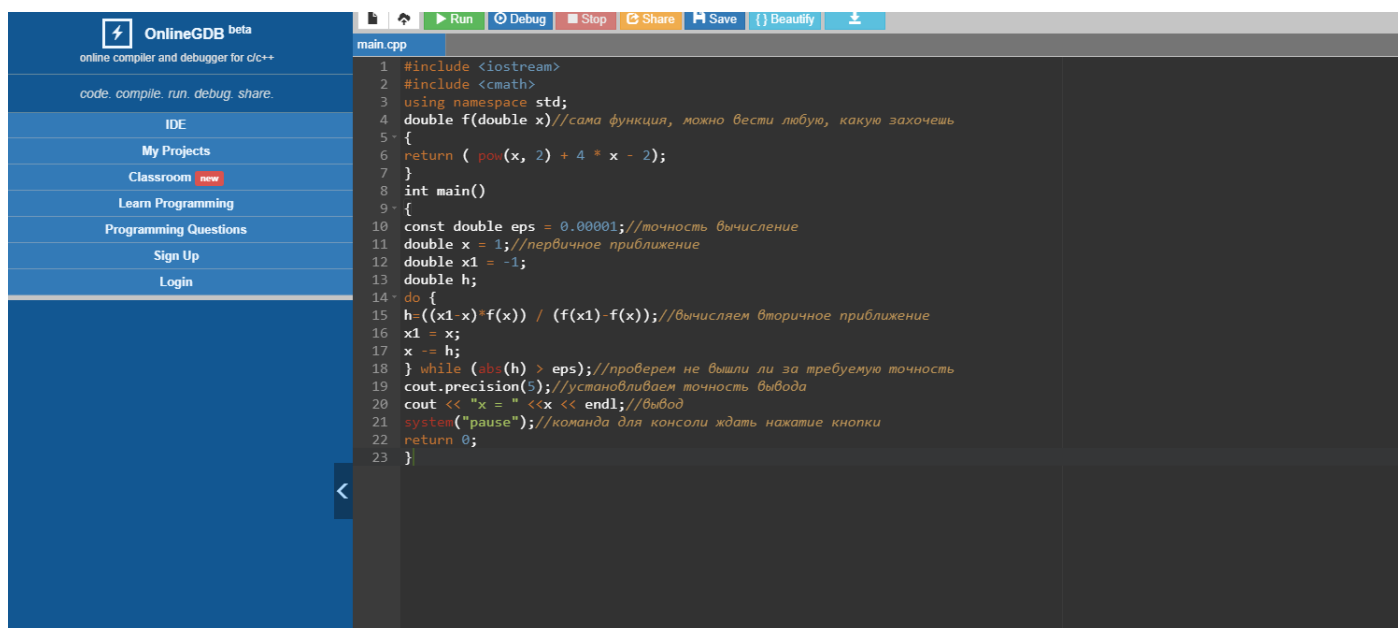
Рис. 1. формула

начала процесса надо задать x_0 и x_1 . Такие процессы, где для очередного приближения надо знать два предыдущих, называют двух шаговыми.

2. Код приложения

```
#include <iostream>
#include <cmath>
using namespace std;
double f(double x)//сама функция, можно вести любую, какую захочешь
{
return ( pow(x, 2) + 4 * x - 2);
}
int main()
{
const double eps = 0.00001;//точность вычисление
double x = 1;//первичное приближение
double x1 = -1;
double h;
do {
h=((x1-x)*f(x)) / (f(x1)-f(x));//вычисляем вторичное приближение
x1 = x;
x -= h;
} while (abs(h) > eps);//проверем не вышли ли за требуемую точность
cout.precision(5);//устанавливаем точность вывода
cout << "x = " <<x << endl;//вывод
system("pause");//команда для консоли ждать нажатие кнопки
return 0;
}
```

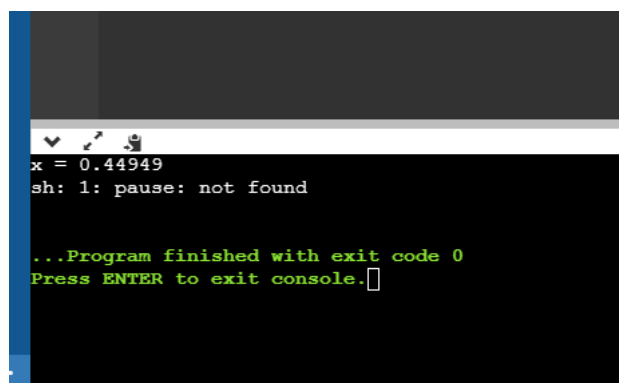
3. Скриньшот программы



The screenshot shows the OnlineGDB web interface. On the left is a sidebar with navigation links: IDE, My Projects, Classroom (marked 'new'), Learn Programming, Programming Questions, Sign Up, and Login. The main area displays a C++ file named 'main.cpp' with the following code:

```
1 #include <iostream>
2 #include <cmath>
3 using namespace std;
4 double f(double x)//сама функция, можно вести любую, какую захочешь
5 {
6     return ( pow(x, 2) + 4 * x - 2);
7 }
8 int main()
9 {
10     const double eps = 0.00001;//точность вычисление
11     double x = 1;//первичное приближение
12     double x1 = -1;
13     double h;
14     do {
15         h=((x1-x)*f(x)) / (f(x1)-f(x));//вычисляем вторичное приближение
16         x1 = x;
17         x -= h;
18     } while (abs(h) > eps);//проверим не вышли ли за требуемую точность
19     cout.precision(5);//устанавливаем точность вывода
20     cout << "x = " <<x << endl;//вывод
21     system("pause");//команда для консоли ждать нажатие кнопки
22     return 0;
23 }
```

Рис. 2. скринь



The screenshot shows the console output of the program. It displays the calculated value of x and a message indicating the program has finished.

```
x = 0.44949
sh: 1: pause: not found

...Program finished with exit code 0
Press ENTER to exit console.
```

Рис. 3. скринь

4. библиографические ссылки

Для изучения «внутренностей» \TeX необходимо изучить [1], а для использования \LaTeX лучше почитать [2, 3].

Список литературы

- [1] Кнут Д.Э. Всё про $\text{T}_\text{E}\text{X}$. — Москва: Изд. Вильямс, 2003 г. 550 с.
- [2] Львовский С.М. Набор и верстка в системе $\text{L}_\text{A}\text{T}_\text{E}\text{X}$. — 3-е издание, исправленное и дополненное, 2003 г.
- [3] Воронцов К.В. $\text{L}_\text{A}\text{T}_\text{E}\text{X}$ в примерах. 2005 г.