

Визуал Студия C++ 17.02.2019г.

```
// 1-я работающая программа решение уравнения методом Фиббоначи
```

```
#include <iostream>
```

```
using namespace std;
```

```
#define eps 1e-3
```

```
int F(int n)
```

```
{
```

```
    int f, f1(1), f2(1), m(0);
```

```
    while (m < n - 1)
```

```
    {
```

```
        f = f1 + f2;
```

```
        f1 = f2;
```

```
        f2 = f;
```

```
        ++m;
```

```
    }
```

```
    return f1;
```

```
}
```

```
double Fun(double x)
```

```
{
```

```
    return (x * x * x * x - 4 * x * x + 8); //здесь
```

```
    //может быть ваше уравнение
```

```
}
```

```
void Fib(double a, double b)
```

```
{
```

```
    double x1, x2, _x, xf1, xf2;
```

```
    int k(0);
```

```
    int N(0);
```

```
double fn1(1), fn2(1), fn, f = (b - a) / eps;
```

```
while (fn1 < f)
```

```
{
```

```
    fn = fn1 + fn2;
```

```
    fn1 = fn2;
```

```
    fn2 = fn;
```

```
    ++N;
```

```
}
```

```
x1 = a + (double)F(N - 2) / F(N) * (b - a) - (N & 1 ? -1 : 1) * eps / F(N);
```

```
x2 = a + (double)F(N - 1) / F(N) * (b - a) + (N & 1 ? -1 : 1) * eps / F(N);
```

```
xf1 = Fun(x1);
```

```
xf2 = Fun(x2);
```

P:

```
++k;
```

```
if (xf1 >= xf2)
```

```
{
```

```
    a = x1;
```

```
    x1 = x2;
```

```
    xf1 = xf2;
```

```
    x2 = a + (double)F(N - k - 1) / F(N - k) * (b - a) + ((N - k) & 1 ? -1 : 1) * eps / F(N - k);
```

```
    xf2 = Fun(x2);
```

```
}
```

```
else
```

```
{
```

```
    b = x2;
```

```
    x2 = x1;
```

```
    xf2 = xf1;
```

```
    x1 = a + (double)F(N - k - 2) / F(N - k) * (b - a) - ((N - k) & 1 ? -1 : 1) * eps / F(N - k);
```

```
    xf1 = Fun(x1);
```

```
}
```

```
if (fabs(b - a) <= eps)
```

```

    {
        _x = (a + b) / 2;
        cout << "\nРезультат:\nx = " << _x << "\t\tF(x) = " << Fun(_x) <<
            "\nКоличество итераций: " << k << endl;
    }
    else
        goto P;
}

//void
int main()
{
    setlocale(0, "");
    double a, b;
    cout << "Программа решает уравнение методом Фибоначчи.\nВведите границы:\na=";
    cin >> a;
    cout << "b=";
    cin >> b;
    Fib(a, b);
    system("pause");
}

```

// 2 работающая пограмма

//Программа решения уравнений методом Ньютона

```
#include <conio.h>
```

```
#include <math.h>
```

```
#include <iostream>
```

```
using namespace std;// определить область переменных
```

```
using std::cin; // определить конкретную область
```

```
using std::cout; //определить конкретную область
```

```
#define pi 3.14
```

```
double f(double x) {  
  
    return x * x - (cos(pi*x));  
}
```

```
double f1(double x) {  
  
    return 2 * x + (1 / x);  
}
```

```
double f2(double x) {  
  
    return 2 + (-1 / (x*x));  
}
```

```
int main() {  
    int n = 0;  
    double a, b, c, eps;  
    cout << "a="; cin >> a;  
    cout << "b="; cin >> b;  
    cout << "eps="; cin >> eps;  
    if (f(a)*f2(a) > 0) c = a;  
    else c = b;  
    do {  
        c = c - f(c) / f1(c);  
        n += 1;  
    } while (fabs(f(c)) >= eps);  
    cout << "c=" << c << "\n";  
    cout << "n=" << n << "\n";  
    _getch(); // без подчеркивания устарев.форма
```

```
return 0;
```

```
}
```