

Практическая работа №3

Тема «Организация циклов»

Цели работы:

1. Написать программы согласно заданному варианту – вычисление значения функции по графику, попадание выстрелов в мишень, ряды Тейлора.

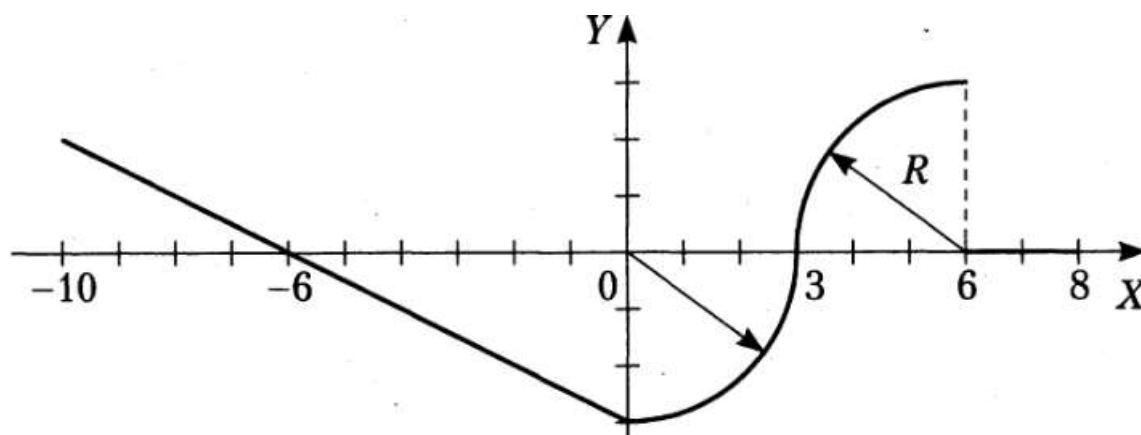
Оснащение:

Персональный компьютер под управлением ОС Windows 10.
Офисный пакет Microsoft Office 2016
Среда разработки Visual Studio 2019

Задание 1. Таблица значений функции

Вычислить и вывести на экран значение функции, заданной графически на интервале от x_n до x_k с шагом dx . Интервал и шаг задать таким образом, чтобы проверить все ветви программы. Таблицу снабдить шапкой и заголовком.

4



Программный код

```
#include <iostream>
#include <stdlib.h>
#include <windows.h>
char* Rus(const char* text);
using namespace std;
int main(int argc, char* argv[])
{
    double Xn, Xk, dX, a, b, c, x, F;
    cout << Rus("Введите Xn, Xk, dX, a, b, c\n");
    scanf_s("%lf%lf%lf%lf%lf%lf", &Xn, &Xk, &dX, &a, &b, &c);
    int a1, b1, c1, F1, i;
    a1 = (int)a;
```

					<i>AuCD.09.03.02.050000 ПР</i>		
Изм.	Лист	№ докум.	Подпись	Дат			
Разраб.	Ермошина В.А				Практическая работа №3 «Организация циклов»	Лит.	Лист
Провер.	Берёза А. Н.						2
Реценз						ИСОиП (филиал) ДГТУ в г.Шахты ИСТ-Тб11	
Н. Контр.							
Утверд.							

```

b1 = (int)b;
c1 = (int)c;
cout << "a=" << a << Rus(" целая часть a1=") << a1 << endl;
cout << "b=" << b << Rus(" целая часть b1=") << b1 << endl;
cout << "c=" << c << Rus(" целая часть c1=") << c1 << endl;
cout << "i| a | b | c | x | F |\n" << endl;
x = Xn;
i = 1;
while (x <= Xk)

{
    if ((x >= -10 and x < 0) || (x >= 0 and x < 3) || (x >= 3 and x < 6)) F = -a * x * x - b;
    else if (x > -10 && c == 0) F = (x - a) / x;
    {
        if (c == 0) cout << i << Rus("| F= Бесконечность x=") << x << " c=" << c << endl;

        else F = -x / c;
        F1 = (int)F;
        if (((a1 | b1) != 0) ^ ((a1 | c1) != 0))
            cout << i << " | " << a << " | " << b << " | " << c << " | " << x << " | " << F << endl;
        else cout << i << " | " << a << " | " << b << " | " << c << " | " << x << " | " << F1 << endl;
    }
    x += dX;
    i++;
}
system("PAUSE");
return 0;
}

char bufRus[256];
char* Rus(const char* text)
{
    CharToOemA(text, bufRus);
    return bufRus;
}

```

Скриншоты

```

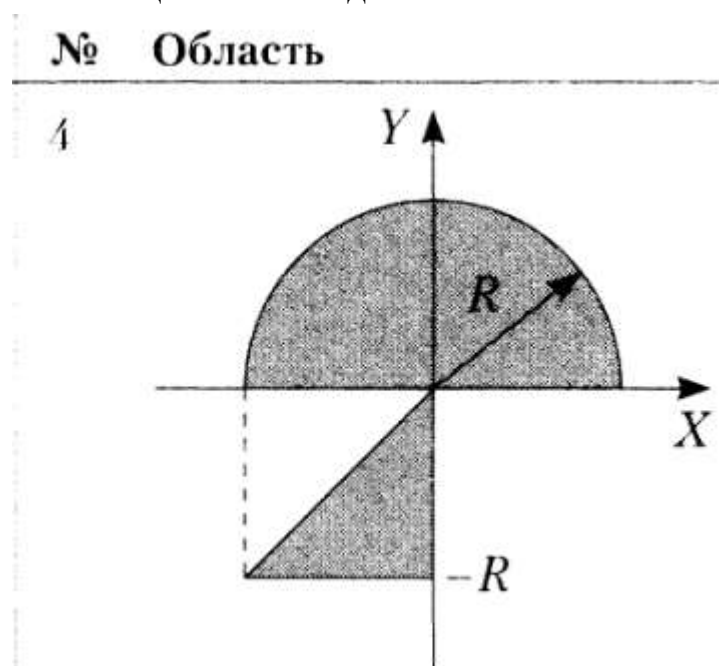
C:\Users\Asus\source\repos\3_1\Debug\3_1.exe
Введите Xn, Xk, dX, a, b, c
0 100 10 10 30 50
a=10 целая часть a1=10
b=30 целая часть b1=30
c=50 целая часть c1=50
i| a | b | c | x | F |
1| 10 | 30 | 50 | 0 | 0
2| 10 | 30 | 50 | 10 | -1
3| 10 | 30 | 50 | 20 | -1
4| 10 | 30 | 50 | 30 | -1
5| 10 | 30 | 50 | 40 | -1
6| 10 | 30 | 50 | 50 | -1
7| 10 | 30 | 50 | 60 | -1
8| 10 | 30 | 50 | 70 | -1
9| 10 | 30 | 50 | 80 | -1
10| 10 | 30 | 50 | 90 | -1
11| 10 | 30 | 50 | 100 | -2
Для продолжения нажмите любую клавишу . . .

```

Рисунок 1 - Результат работы программы

Задание 2. Серия выстрелов по мишени

Для десяти выстрелов, координаты которых задаются с клавиатуры, вывести текстовые сообщения о попадании в мишень.



Программный код

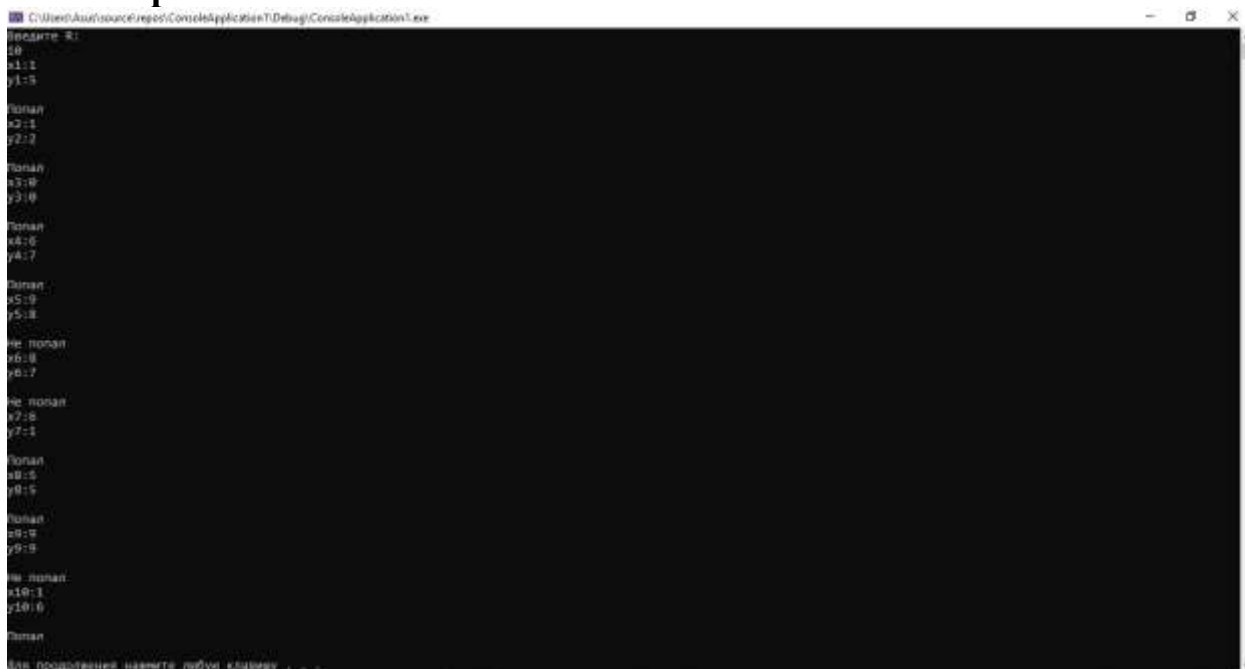
```
#include <iostream>
#include <math.h>
using namespace std;
int main()
{
    setlocale(LC_ALL, "");
    float R, x, y;
    cout << "Введите R:" << endl;
    cin >> R;
    for (int i = 0; i < 10; i++)
    {
        cout << "x" << i + 1 << ":";
        cin >> x;
        cout << "y" << i + 1 << ":";
        cin >> y;
        cout << endl;
        if ((x * x + y * y <= R * R) && ((y >= 0))) // условие для полуокружности
            cout << "Попал" << endl;
        else
        {
            if (x >= y && x <= 0 && y > -R) //условие для треугольника
                cout << "Попал" << endl;
            else
            {
                cout << "Не попал" << endl;
            }
        }
    }
}
```

```

cout << endl;
system("pause");
return 0;
}

```

Скриншоты



Задание 3. Ряды Тейлора

Вычислить и вывести на экран в виде таблицы значения функции, заданной с помощью ряда Тейлора. на интервале от x_1 до x_k с шагом dx с точностью ϵ . Таблицу снабдить шапкой и заголовком. Каждая строка таблицы должна содержать значение аргумента, значение функции и количество просуммированных членов ряда.

$$4. \ln(x+1) = \sum_{n=0}^{\infty} \frac{(-1)^n x^{n+1}}{n+1} = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} - \dots, \quad -1 < x < 1.$$

Программный код

```

#include <iostream>
#include <cmath>
using namespace std;
int main()
{
    setlocale(LC_CTYPE, "rus");
    double x_begin, x_end, x_step, n, sum = 0;
    cout << "Введите интервал x[-1 <= x < 1]:\n\t";
    cout << "Начальное значение:\n\t";
    cin >> x_begin;
    cout << "Конечное значение:\n\t";
    cin >> x_end;
    cout << "Введите шаг:\n\t ";
    cin >> x_step;

```

```

cout << "Введите точность расчета [0 < n < 10]: ";
cin >> n;
cout << "| x | ln(1 - x)| Taylor |\n";
for (double x = x_begin; x <= x_end; x += x_step)
{
    sum += x;
    for (int i = 1; i < n; i++)
    if (i % 2 == 0)
    sum += pow(x, double(i)) / i;
    cout << x << " | " << log(1 - x) << " | " << -sum << endl;
    sum = 0;
}
cin.get();
cin.get();
return 0;
}

```

Скриншоты

```

C:\Users\Asus\source\repos\ConsoleApplication1\Debug\ConsoleApplication1.exe
Введите интервал x[-1 <= x < 1]:
Начальное значение:
-1
Конечное значение:
0.9
Введите шаг:
0.1
Введите точность расчета [0 < n < 10]: 2
| x | ln(1 - x)| Taylor |
-1 | 0.693147 | 1
-0.9 | 0.641854 | 0.9
-0.8 | 0.587787 | 0.8
-0.7 | 0.530628 | 0.7
-0.6 | 0.470004 | 0.6
-0.5 | 0.405465 | 0.5
-0.4 | 0.336472 | 0.4
-0.3 | 0.262364 | 0.3
-0.2 | 0.182322 | 0.2
-0.1 | 0.0953102 | 0.1
-1.38778e-16 | 2.22045e-16 | 1.38778e-16
0.1 | -0.105361 | -0.1
0.2 | -0.223144 | -0.2
0.3 | -0.356675 | -0.3
0.4 | -0.510826 | -0.4
0.5 | -0.693147 | -0.5
0.6 | -0.916291 | -0.6
0.7 | -1.20397 | -0.7
0.8 | -1.60944 | -0.8
0.9 | -2.30259 | -0.9

```

Вывод: В ходе выполнения практической работы написала программы «Таблица значений функции», «Серия выстрелов по мишени», «Ряды Тейлора».