Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ОТЧЕТ**

**ПО КОНТРОЛЬНОЙ РАБОТЕ №4**

**«****РАБОТА В ГРАФИЧЕСКОМ РЕЖИМЕ»**

Выполнил: студент учебной группы

ИСПк-202-52-00

Сысолятин Никита Дмитриевич

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2022

1. **Цель контрольной работы.**

Освоить принципы работы в графическом режиме; получить базовые навыки взаимодействия с графическими примитивами.

1. **Формулировка задания.**

Вариант: 1

Задание:

1) Дополнить программу, реализованную в ходе предыдущей лабораторной работы, режимом визуализации.

2) Предусмотреть возможность вывода кривой, ограничивающей фигуру, на координатную плоскость.

3) Реализовать следующие возможности и элементы: масштабирование графика, подписи на осях, вывод информации о задании.

4) Реализовать не менее двух возможностей из представленных: независимое масштабирование по осям, штриховка вычисляемой площади, визуализация численного расчета интеграла

1. **Код программы.**

**Uses** GraphABC;

**var**

a, b: real;

**var**

n: integer;

**function** f(x: real): real;

**begin**

f := 2 \* power(x, 3) - 4 \* x + 3; // Функция

**end**;

**function** f1(x: real): real;

**begin**

f1 := 1 / 2 \* power(x, 4) - 2 \* power(x, 2) + 3 \* x; // Первообразная

**end**;

// интервал

**procedure** granizi;

**begin**

**repeat**

a := readlnreal('Введите левую границу интервала от -10 до 0 a = ');

**until** (a >= -10) **and** (a <= 0);

**repeat**

b := readlnreal('Введите правую границу интервала от 0 до 10 b = ');

**until** (b >= 0) **and** (b > a) **and** (b <= 10);

**end**;

//Процедура разбиений

**procedure** razbienie(**var** n: integer);

**begin**

**repeat**

n := readinteger('Введите число разбиений от 1 до 10000 n = ');

**until** (n >= 1) **and** (n <= 10000);

**end**;

//Процедура площади

**procedure** plos(**var** a, b: real; **var** n: integer; **var** h, s, y: real);

**begin**

**if** b = 0.0 **then if** a = 0.0 **then begin** writeln('Параметры интервалов не заданы'); **exit**; **end**;

h := (b - a) / n;

s := f1(b) - f1(a);

**for var** i := 1 **to** n **do**

s := s + f(a + i \* h);

s := s \* h;

y := f1(b) - f1(a);

readln;

**end**;

//график

**procedure** grafik(**var** gx, gy, x0, y0, k, x1, y1, i, e, r, x2: integer; x, y, h: real);

**begin**

gx := 900;

gy := 700;

k := 20;

setwindowsize(gx, gy);

x0 := gx **div** 2;

y0 := gy **div** 2;

//ось X

line(0, y0, gx, y0);

line(gx, y0, gx - 10, y0 + 10);

line(gx, y0, gx - 10, y0 - 10);

textout(gx - 10, y0 + 10, 'X');

//ocm Y

line(x0, 0, x0, gy);

line(x0, 0, x0 - 10, 10);

line(x0, 0, x0 + 10, 10);

textout(x0 + 10, 10, 'Y');

//циклс для цены деления оси X

i := round(a);

**while** i <= b **do**

**begin**

line(x0 + i \* k, y0 + 10, x0 + i \* k, y0 - 10);

textout(x0 + i \* k, y0 + 10, inttostr(i));

i += 1;

**end**;

//цикл для цены деления оси Y

i := -10;

**while** i < 11 **do**

**begin**

**if** i <> 0 **then**

**begin**

line(x0 + 10, y0 + i \* k, x0 - 10, y0 + i \* k);

textout(x0 + 10, y0 - i \* k, inttostr(i));

**end**;

i += 1;

**end**;

x := a;

x1 := x0 + round(x \* k);

y1 := y0 - round(y \* k);

MoveTo(x1, y1);

**while** x < b **do**

**begin**

y := 2 \* power(x, 3) - 4 \* x + 3;

x1 := x0 + round(x \* k);

y1 := y0 - round(y \* k);

lineTo(x1, y1);

x += 1 / n;

**end**;

// Построение прямоугольников

h := (b - a) / n;

x := a;

**while** x <= b **do**

**begin**

setpencolor(clred);

y := 2 \* power(x, 3) - 4 \* x + 3;

x1 := x0 + round(x \* k);

y1 := y0 - round(y \* k);

x2 := x0 + round((x + h) \* k);

rectangle(x1, y1, x2, y0);

x += h;

**end**;

readln;

**end**;

// Процедура меню

**procedure** menu(**var** a: integer);

**begin**

clearwindow;

textout(10, 10, 'Выберите действие: ');

textout(10, 30, '1 - Ввод интервала ');

textout(10, 50, '2 - Ввод кол-ва разбиений ');

textout(10, 70, '3 - Вывод площади ');

textout(10, 90, '4 - Вывод площади по формуле Ньютона-Лейбница ');

textout(10, 110, '5 - Вывод абсолютной погрешности ');

textout(10, 130, '6 - Относительная погрешность ');

textout(10, 150, '7 - График');

textout(10, 170, '8 - Конец');

read(a);

clearwindow;

**end**;

//Начало программы

**var**

h, s, y, x: real;

z, gx, gy, x0, y0, k, x1, y1, i, e, r, x2: integer;

**begin**

menu(z);

**while** true **do**

**begin**

**case** z **of**

1: **begin** granizi; menu(z); **end**;

2: **begin** razbienie(n); plos(a, b, n, h, s, y); menu(z); **end**;

3: **begin** writeln('S=', s:0:5); **if** s = 0.00 **then** writeln('Параметры для высчитывания площади не заданы'); menu(z); readln; **end**;

4: **begin** writeln('Аналитическое значение=', y:0:5); menu(z); readln; **end**;

5: **begin** writeln('Абсолютная погрешность=', abs(y - s):0:5); menu(z); readln; **end**;

6: **begin** writeln('Относительная погрешность=', abs(y - s) / y); menu(z); readln; **end**;

7: **begin** grafik(gx, gy, x0, y0, k, x1, y1, i, e, r, x2, x, y, h); readln; menu(z); **end**;

8: **begin** writeln('Спасибо за внимание'); writeln('Нажмите Enter чтобы выйти из приложения...'); **exit**; **end**;

**end**;

**end**;

**end**.

1. **Результат выполнения программы.**

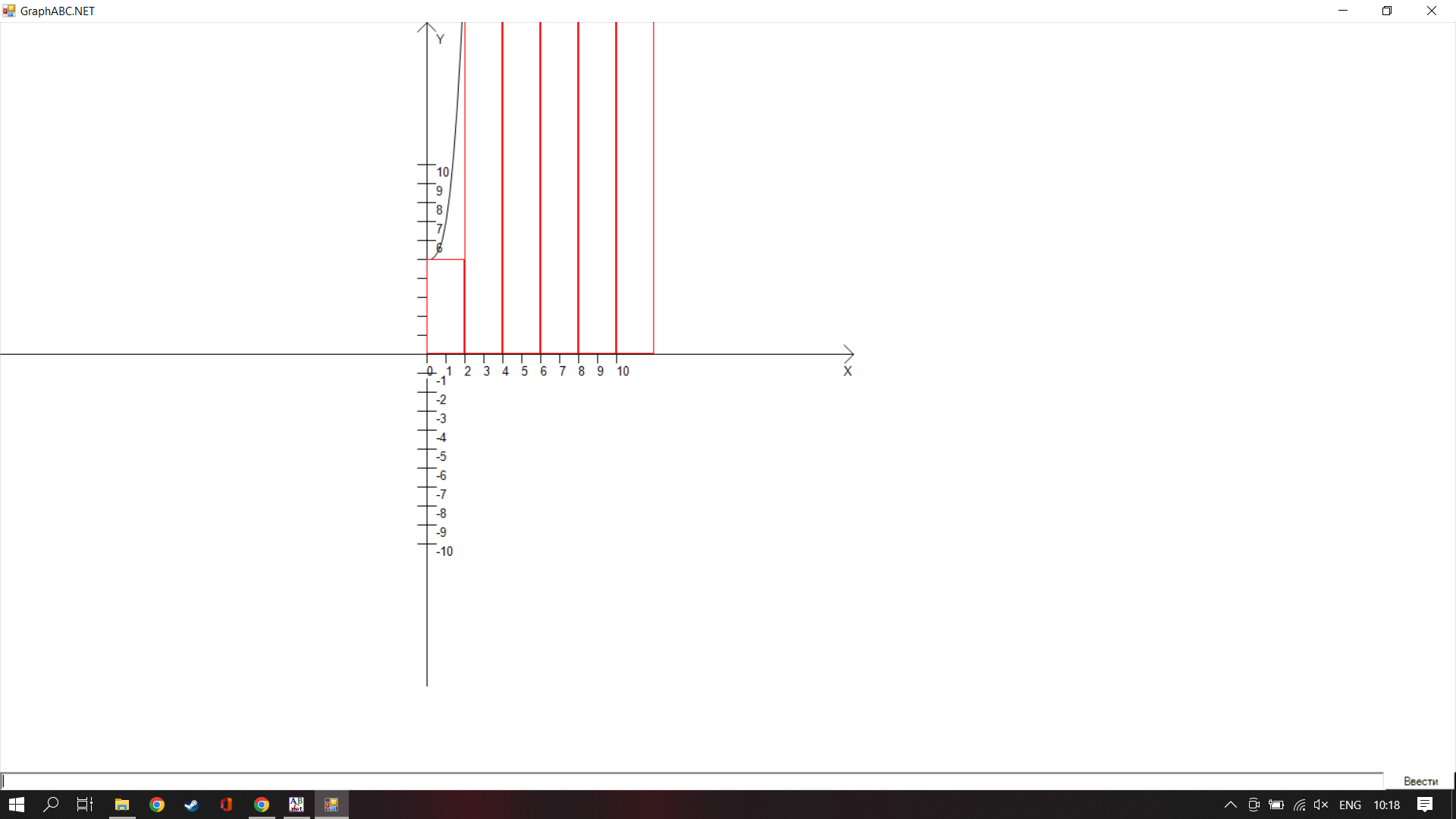


Рисунок 1 - Результат выполнения программы

1. **Вывод.**

В ходе выполнения контрольной работы №4 «Работа в графическом режиме», освоили принципы работы в графическом режиме, получили базовые навыки взаимодействия с графическими примитивами.

Модуль GraphABC представляет собой простую графическую библиотеку и предназначен для создания несобытийных графических и анимационных программ в процедурном и частично в объектном стиле. Рисование осуществляется в специальном графическом окне, возможность рисования в нескольких окнах отсутствует.

Научились создавать и использовать графический режим. Во время работы возникали небольшие трудности, которые удалось решить с помощью преподавателя.