Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ОТЧЕТ**

**ПО ДОМАШНЕЙ КОНТРОЛЬНОЙ РАБОТЕ №4**

**«ИЗУЧЕНИЕ БАЗОВЫХ ПРИНЦИПОВ ОРГАНИЗАЦИИ ПРОЦЕДУР И ФУНКЦИЙ»**

**ПО ДИСЦИПЛИНЕ «ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»**

Выполнил: студент учебной группы ИСПк- 205-52-00

Огнев Григорий Сергеевич

Преподаватель:

Сергеева Елизавета Григорьевна

г. Киров

2023

1. **Цель работы**: освоить принципы работы в графическом режиме; получить базовые навыки взаимодействия с графическими примитивами.
2. **Задание**:
3. Дополнить программу, реализованную в ходе предыдущей лабораторной работы, режимом визуализации.
4. Предусмотреть возможность вывода кривой, ограничивающей фигуру, на координатную плоскость.
5. Реализовать следующие возможности и элементы: масштабирование графика, подписи на осях, вывод информации о задании.
6. Реализовать не менее двух возможностей из представленных: независимое масштабирование по осям, штриховка вычисляемой площади, визуализация численного расчета интеграла.

**3. Описание алгоритма**:

1) Визуализация графика функции:

Сначала строится первая четверть координатной плоскости

(положительные значения x и y).

2) Построение графика:

Построение графика функции в координатной плоскости.

3) Визуализация расчёта интеграла:

Под графиком функции строятся прямоугольники,

визуализирующие вычисление интеграла.

4) Масштабирование графика:

Масштабирование графика функции относительно оси X и оси Y.

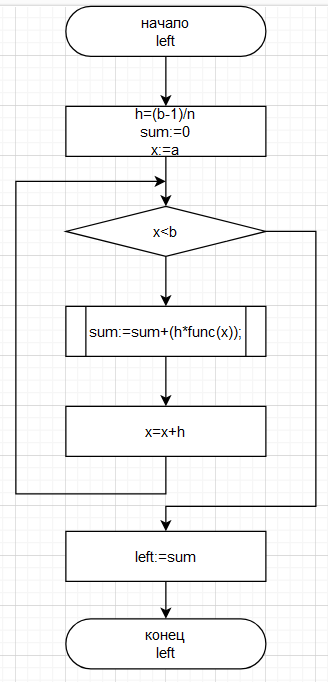
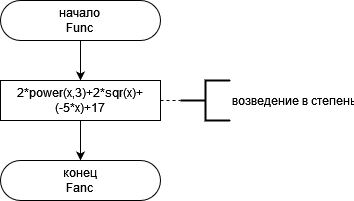
**4. Схема алгоритма**:

Рисунок 2 Left



Рисунок 1 Funs

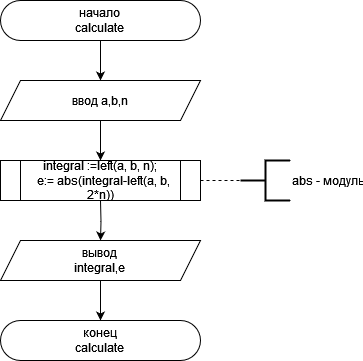


Рисунок 3 calculate

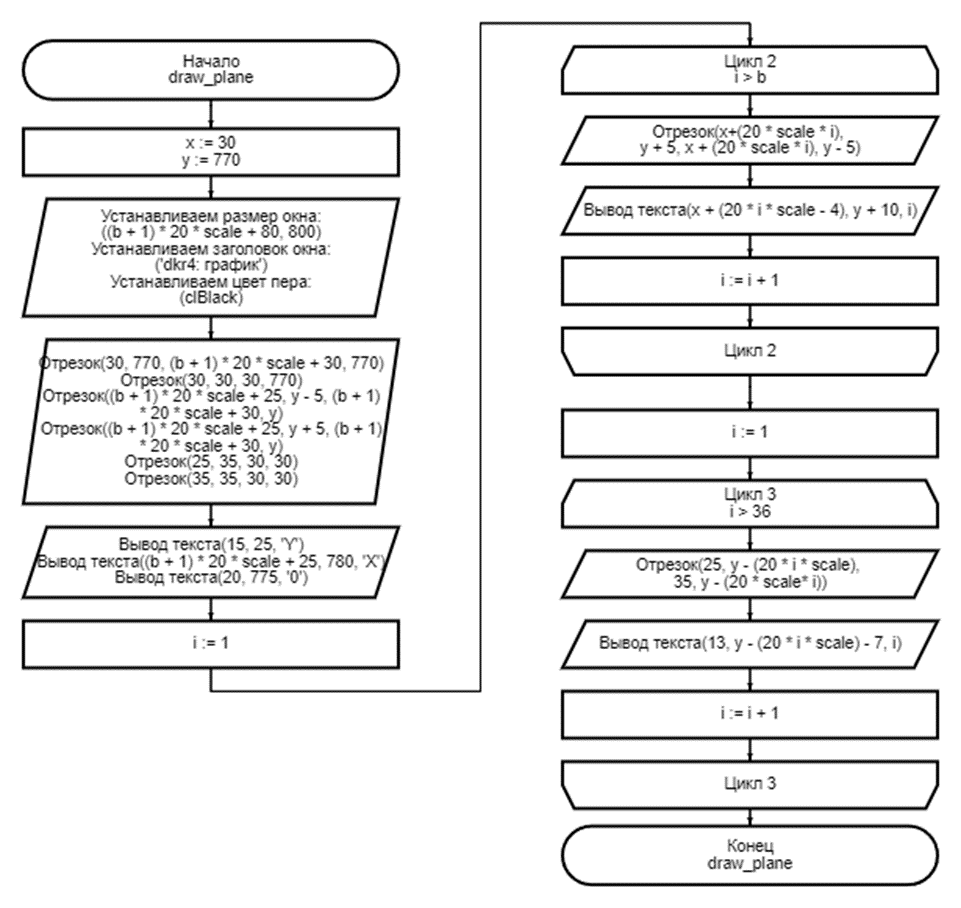


Рисунок 4 - draw plane

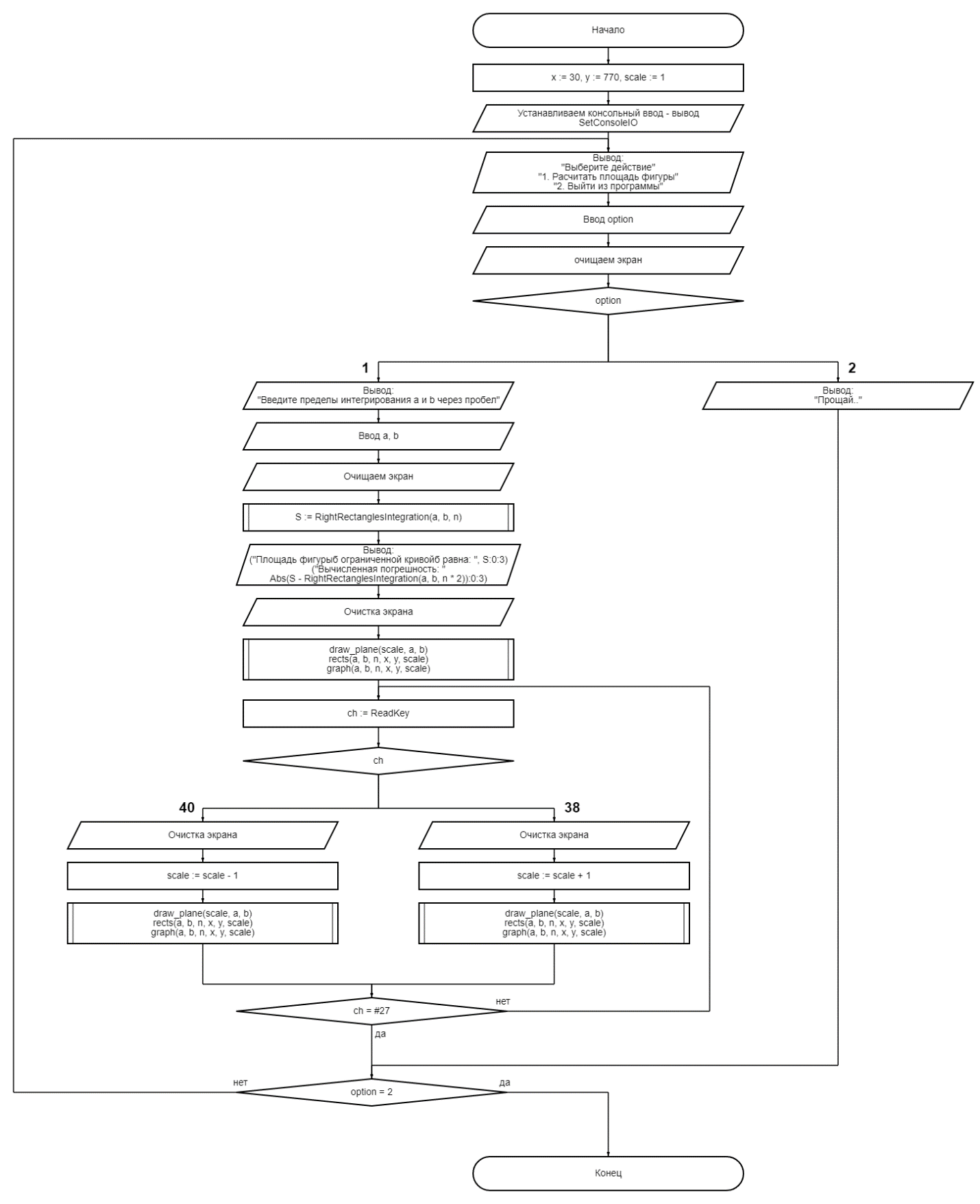
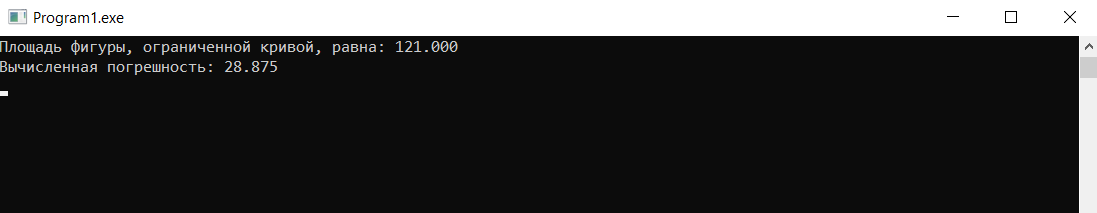
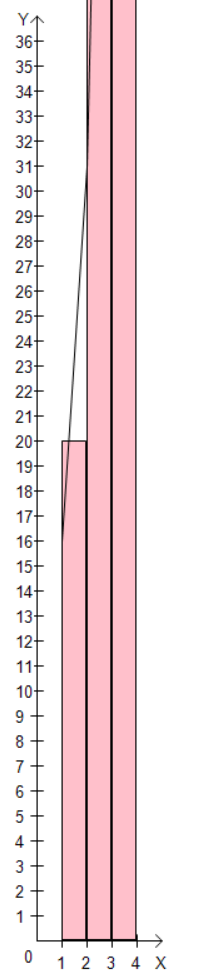




Рисунок 5 тело

**6.Резульат выполнения программы**



**6 Код программы:**

**uses** Crt, GraphABC;

**function** F(x: Real): Real;

**begin**

F :=2\*power(x,3)+2\*sqr(x)+(-5\*x)+17;

**end**;

**function** RightRectanglesIntegration(a, b: Real; n: Integer): Real;

**var**

h, x, sum: Real;

i: Integer;

**begin**

h := (b - a) / n;

sum := 0;

**for** i := 0 **to** n - 1 **do**

**begin**

x := a + h \* i;

sum := sum + F(x);

**end**;

RightRectanglesIntegration := h \* sum;

**end**;

**var**

S: Real;

a, b, n, x, y, option, scale: Integer;

ch: char;

**procedure** draw\_plane(scale, a, b: integer);

**var**

i, x, y: integer;

**begin**

x := 30;

y := 770;

SetWindowSize((b + 1) \* 20 \* scale + 80, 800);

SetWindowTitle('dkr4: график');

SetPenColor(clBlack);

Line(30, 770, (b + 1) \* 20 \* scale + 30, 770); // Ось X

Line(30, 30, 30, 770); // Ось Y

line((b + 1) \* 20 \* scale + 25, y - 5, (b + 1) \* 20 \* scale + 30, y);

line((b + 1) \* 20 \* scale + 25, y + 5, (b + 1) \* 20 \* scale + 30, y);

line(25, 35, 30, 30);

line(35, 35, 30, 30);

TextOut(15, 25, 'Y');

TextOut((b + 1) \* 20 \* scale + 25, 780, 'X');

TextOut(20, 775, '0');

**for** i := 1 **to** b **do**

**begin**

line(x + (20 \* scale \* i), y + 5, x + (20 \* scale \* i), y - 5);

TextOut(x + (20 \* i \* scale - 4), y + 10, i);

**end**;

**for** i := 1 **to** 36 **do**

**begin**

line(25, y - (20 \* i \* scale), 35, y - (20 \* scale \* i));

TextOut(13, y - (20 \* i \* scale) - 7, i);

**end**;

**end**;

**function** graph(a, b, n, x, y, scale: integer): boolean;

**var**

i: integer;

h: Real;

**begin**

h := (b - a) / n;

**for** i := 0 **to** n - 1 **do**

**begin**

SetPenColor(clBlack);

line(trunc(x + a \* 20 \* scale + h \* i \* 20 \* scale), y - trunc(F(a + i \* h)) \* 20 \* scale, trunc(x + a \* 20 \* scale + h \* i \* 20 \* scale + h \* 20 \* scale), y - trunc(F(a + i \* h + h)) \* 20 \* scale);

**end**;

**end**;

**function** rects(a, b, n, x, y, scale: integer): boolean;

**var**

i: integer;

h: Real;

**begin**

h := (b - a) / n;

**for** i := 0 **to** n - 1 **do**

**begin**

SetBrushColor(clPink);

rectangle(trunc(x + a \* 20 \* scale + h \* i \* 20 \* scale), y, trunc(x + a \* 20 \* scale + h \* i \* 20 \* scale + h \* 20 \* scale), y - trunc(F(a + i \* h + h - h / 2)) \* 20 \* scale);

**end**;

**end**;

**begin**

x := 30;

y := 770;

scale := 1;

SetConsoleIO;

**repeat**

Writeln('Выберите действие:');

Writeln('1. Рассчитать площадь фигуры');

Writeln('2. Выйти из программы');

Readln(option);

ClrScr;

**case** option **of**

1:

**begin**

Writeln('Введите пределы интегрирования a и b через пробел(в промежутке от 0 до 20):');

Readln(a, b);

ClrScr;

Writeln('Введите количество отрезков для деления:');

Readln(n);

ClrScr;

S := RightRectanglesIntegration(a, b, n);

Writeln('Площадь фигуры, ограниченной кривой, равна: ', S:0:3);

Writeln('Вычисленная погрешность: ', Abs(S - RightRectanglesIntegration(a, b, n \* 2)):0:3);

window.Clear;

draw\_plane(scale, a, b);

rects(a, b, n, x, y, scale);

graph(a, b, n, x, y, scale);

**repeat**

ch := ReadKey;

**begin**

ch := ReadKey;

**case** ch **of**

#40:

**begin**

window.Clear;

scale := scale - 1;

draw\_plane(scale, a, b);

rects(a, b, n, x, y, scale);

graph(a, b, n, x, y, scale);

**end**;

#38:

**begin**

window.Clear;

scale := scale + 1;

draw\_plane(scale, a, b);

rects(a, b, n, x, y, scale);

graph(a, b, n, x, y, scale);

**end**;

**end**;

**end**;

**until** ch = #27;

**end**;

2: Writeln('прощай..');

**end**;

**until** option = 2;

**end**.

**7.Вывод**:

Лабораторная работа посвящена использованию метода прямоугольников для расчета приближенной площади под кривой на заданном интервале и оценки погрешности данного метода. Алгоритм включает в себя определение функции Func(x) для вычисления значений кривой, использование функции Left(a, b) для вычисления площади под кривой методом прямоугольников и процедуры Eval(a, b, n) для оценки погрешности метода. Пользователю предлагается ввести пределы интегрирования (a, b) и количество прямоугольников (n), после чего производится вычисление площади и погрешности.

Таким образом, лабораторная работа направлена на применение метода прямоугольников для нахождения площади криволинейных фигур, а также на оценку погрешности данного метода, что предоставляет студенту понимание и навыки использования численных методов интегрирования.