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**

**«РАБОТА В ГРАФИЧЕСКОМ РЕЖИМЕ»**

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

Выполнил: студент учебной группы ИСПк- 204-52-00

Никулин Арсений Игоревич

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.
7. **Описание алгоритма**:

Алгоритм визуализирует расчёт интеграла, между графиком функции

1 \* x ^ 3 + (-2) \* x ^ 2 + (2) \* x + (17) и осью OX (в положительной части оси OY).

1. Визуализация графика функции:

Сначала строится первая четверть координатной плоскости

(положительные значения x и y).

1. Построение графика:

Построение графика функции в координатной плоскости.

1. Визуализация расчёта интеграла:

Под графиком функции строятся прямоугольники,

визуализирующие вычисление интеграла.

1. Масштабирование графика:

Масштабирование графика функции относительно оси X и оси Y.

1. **Схема алгоритма**:

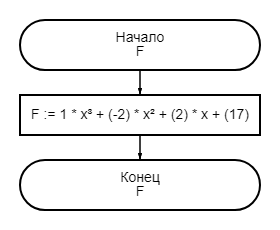
****

Рисунок 1.1 – Функция F

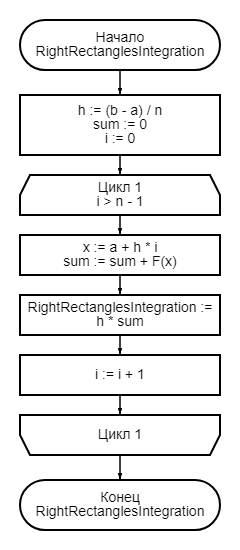


Рисунок 1.2 – Функция RightRectanglesIntegration

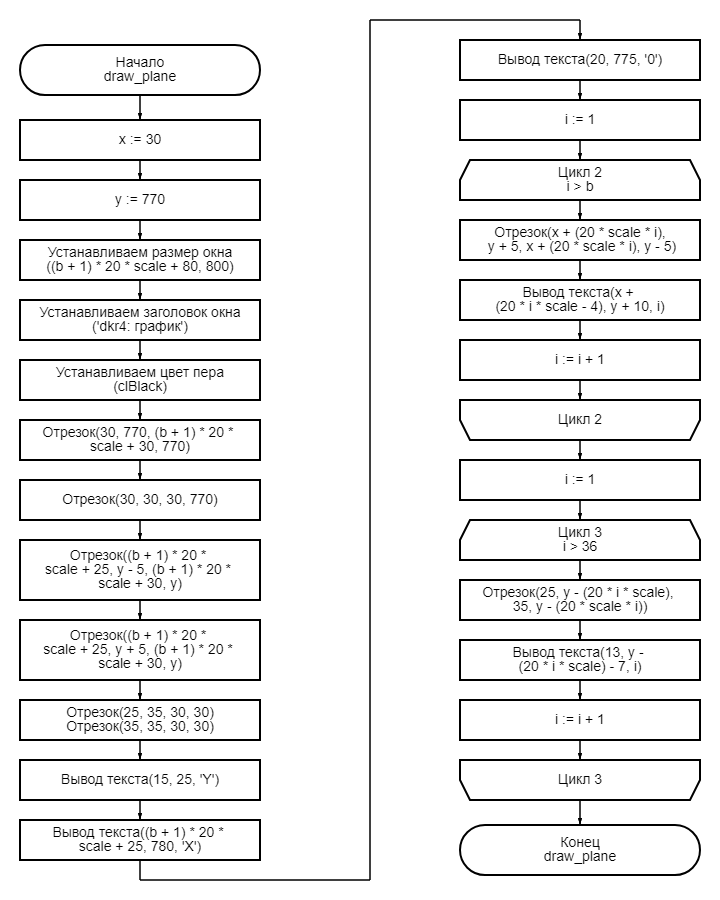


Рисунок 1.3 – Процедура draw\_plane

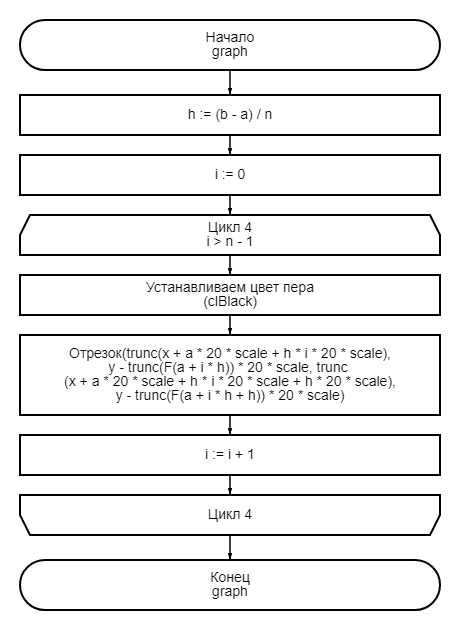


Рисунок 1.4 – Функция graph

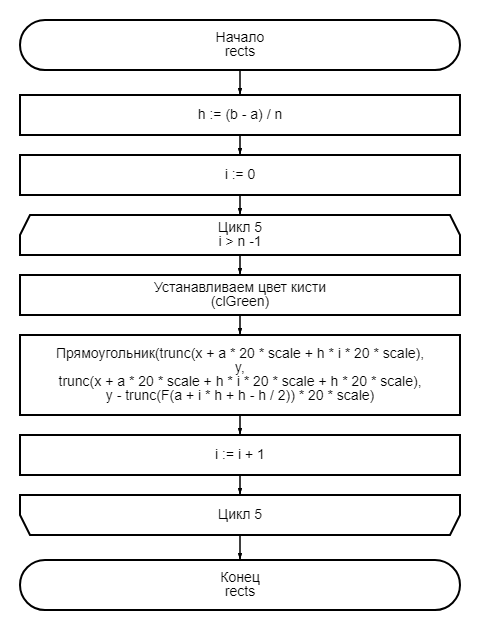


Рисунок 1.5 – Функция rects

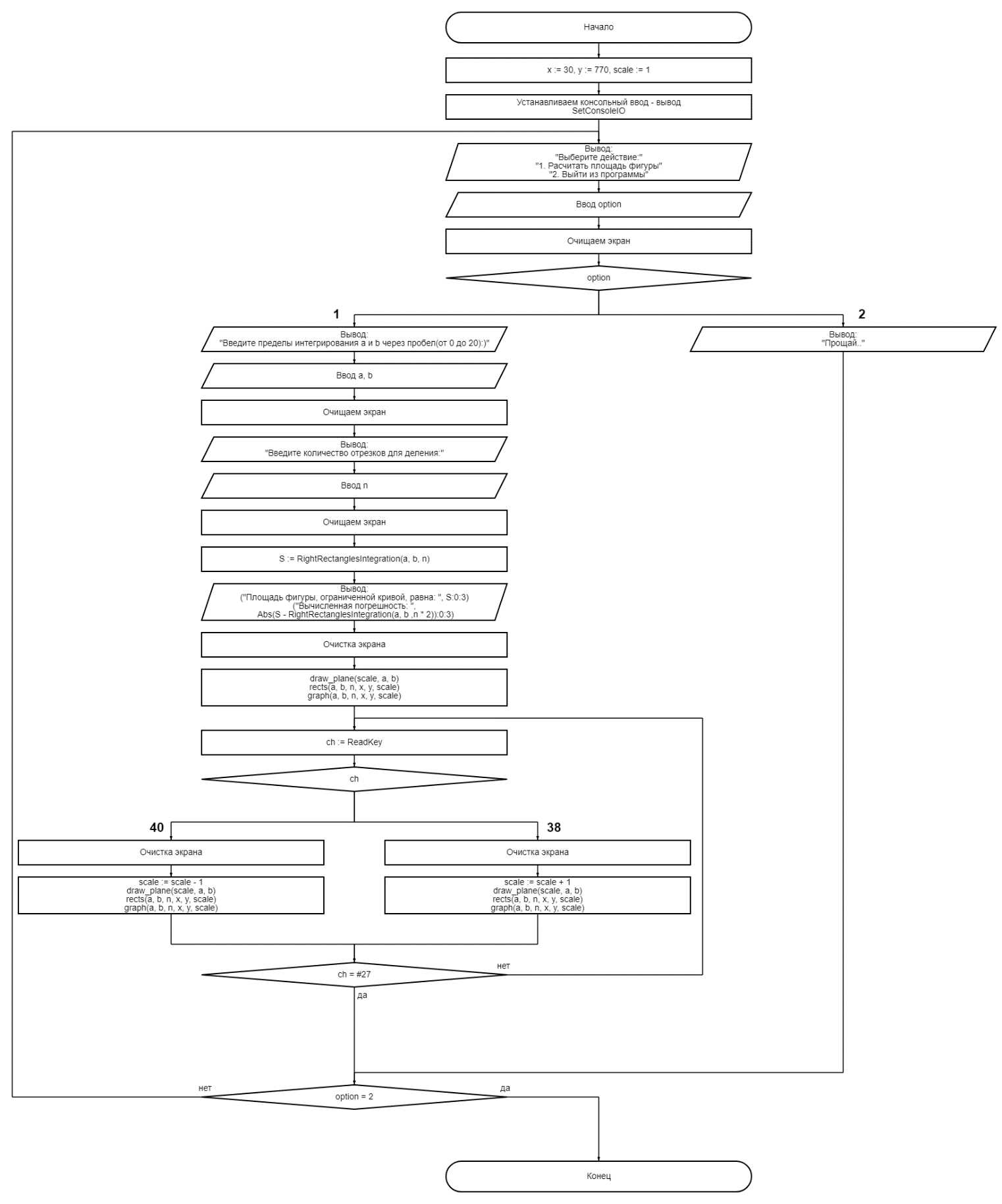


Рисунок 1.6 – Тело программы

1. **Код программы**:

**program** dkr4;

**uses** Crt, GraphABC;

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

**begin**

F := 1 \* x \* x \* x + (-2) \* x \* x + (2) \* 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(clGreen);

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**;

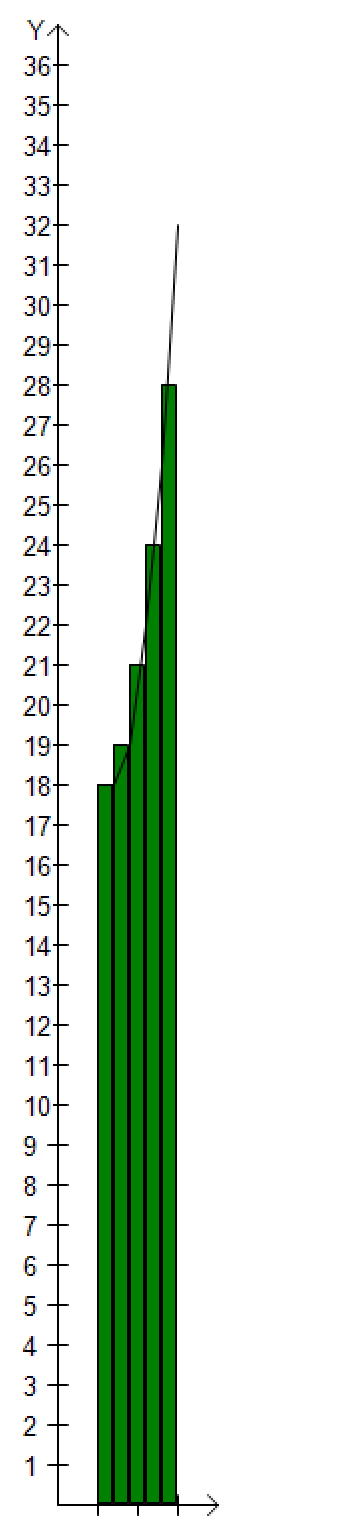
2: Writeln('прощай..');

**end**;

**until** option = 2;

**end**.

**6.Результат выполнения программы**:



**7.Вывод**:

Эта программа использует метод правых прямоугольников для вычисления площади фигуры, ограниченной кривой, по функции из прошлой лабораторной работы. Она также позволяет масштабировать график и отображает результат в виде прямоугольников. Кроме того, она предоставляет возможность пользователю управлять масштабом графика. В процессе выполнения задания возникали трудности, такие как: “Неверное количество передаваемых аргументов в функцию“, “описание переменной цикла for должно производиться в заголовке цикла”, также были ошибки в построении графика функции и неправильные подписи осей координат, реализация масштабизации графика также привела к большому количеству ошибок и неточностей построения, однако все ошибки и неисправности были устранены.

Вот такие пироги..