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- 204-52-00

Черный Даниил Олегович  
Преподаватель:  
Сергеева Елизавета Григорьевна

г. Киров

2023

**Цель работы**

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

**Формулировка задания**

Вариант 19

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.

**Описание алгоритма**

Алгоритм визуализирует расчёт интеграла, между графиком функции

2\*x^3+(1)\*x^2+(-1)\*x+(14) 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 в координатной плоскости
2. Визуализация расчёта интеграла:
3. Масштабирование графика:

Масштабирование графика функции относительно оси X и оси Y.

**Схема алгоритма**

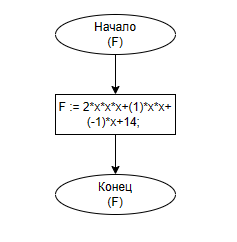


Рисунок 1 – Функция F

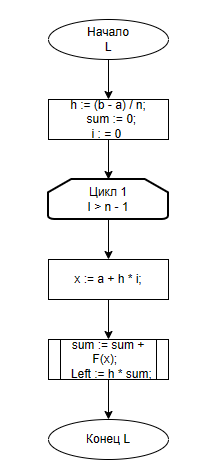


Рисунок 2 – Функция L

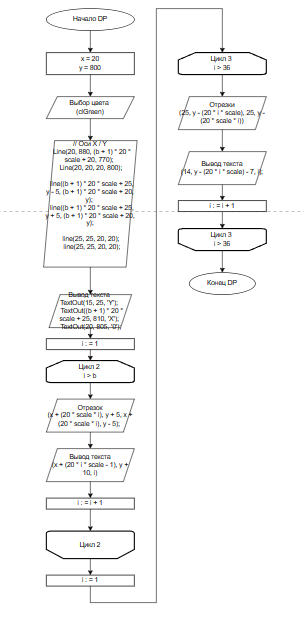


Рисунок 3 – процедура dp

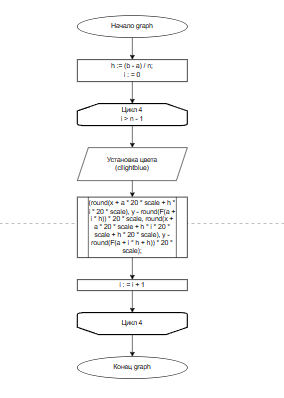


Рисунок 4 – Функция graph

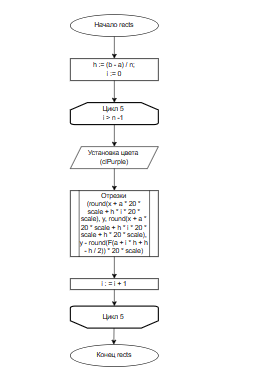


Рисунок 5 – Функция rects

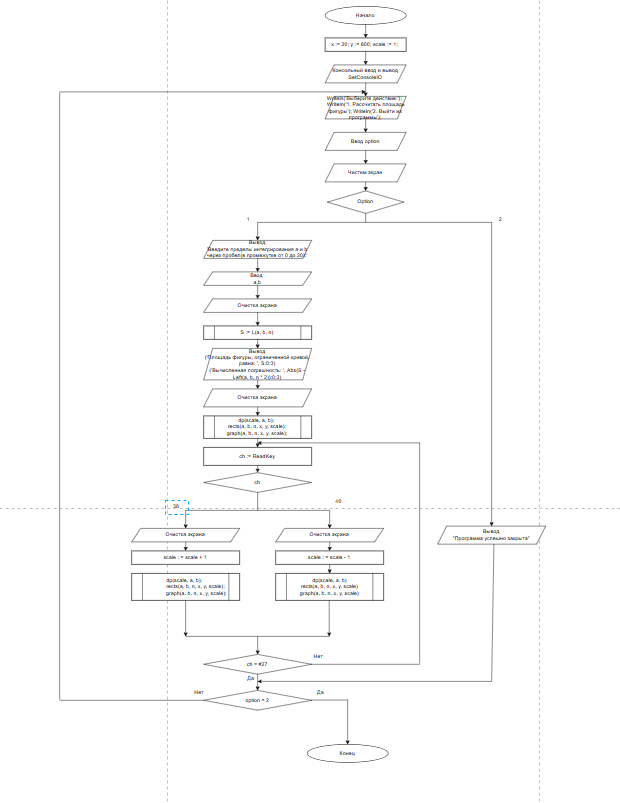


Рисунок 6 – Тело программы

**Код программы**

**uses crt,GraphABC;**

**var S : real;**

**a,b,x:integer;**

**ca:Byte;**

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

**begin**

**F := 2\*(x\*x\*x)+((-1)\*x)+14;**

**end;**

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

**var**

**h, x, sum,S: 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;**

**Left := h \* sum;**

**end;**

**var**

**n, y, option, scale: Integer;**

**ch: char;**

**procedure dp(scale, a, b: integer);**

**var**

**i, x, y: integer;**

**begin**

**x := 30;**

**y := 770;**

**SetPenColor(clgreen);**

**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(cllightblue);**

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

**rectangle(round(x + a \* 20 \* scale + h \* i \* 20 \* scale), y, round(x + a \* 20 \* scale + h \* i \* 20 \* scale + h \* 20 \* scale), y - round(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 := Left(a, b, n);**

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

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

**window.Clear;**

**dp(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;**

**dp(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;**

**dp(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('Пока');**

**end;**

**until option = 2;**

**end.**

**Результат выполнения программы**

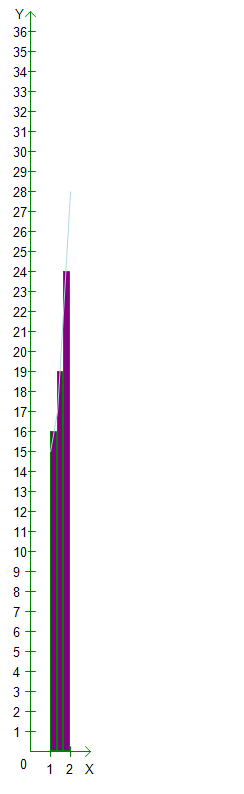


Рисунок 8 – Результат выполнения

**Вывод**

В результате домашней контрольной работы №4 мы достигли заданной нам цели.

Также мы освоили принципы работы в графическом режиме, получили базовые навыки взаимодействия с графическими примитивами, а также выполнили данное нам практическое задание, использовав изученный материал на практике.

Воспользовались интернет ресурсом draw.io для написания схемы алгоритма

Таким образом, выполнение домашней контрольной работы №4 принесло нам ценный опыт и знания, которые будут полезны в нашем дальнейшем образовании. Мы научились применять определённые базовые конструкции языка программирования, а также разрабатывать схему алгоритма для решения задачи. Этот опыт будет полезен нам будущем для решений новых задач и работой над реальными проектами.