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»**

Выполнила: студентка учебной группы

ИСПк-202-52-00

Широнина Анна Андреевна

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2024

**Цель работы:** освоить принципы работы в графическом режиме; получить базовые навыки взаимодействия с графическими примитивами.

1. **Формулировка заданий.**

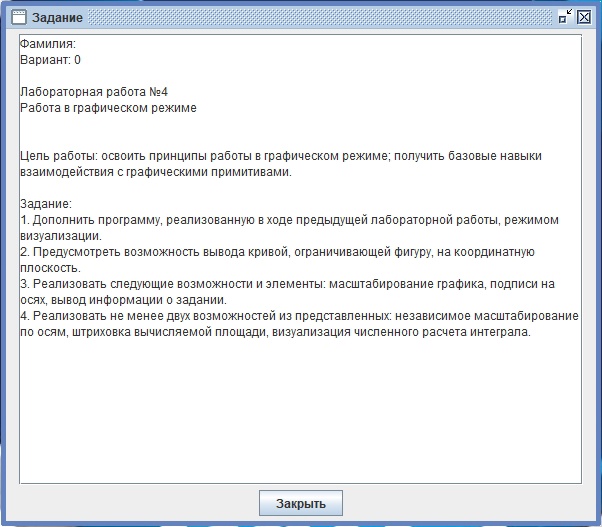
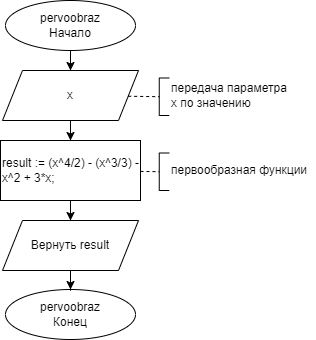
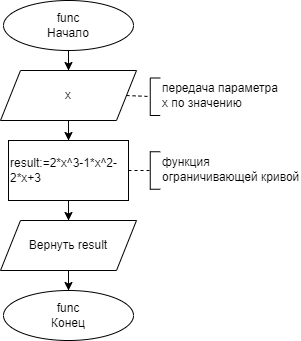


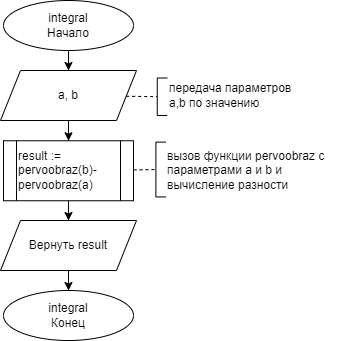
Рисунок 1 – Постановка задачки

1. **Описание алгоритма**

Для начала мы запрашиваем размеры графического окна. Установка размеров графического окна. Далее запрашиваем масштаб графика и пределы участка графика. Отрисовка графика. Запрашиваем данные о трапециях (нижний предел интегрирования, верхний предел интегрирования и количество трапеций для интегрирования). Визуализация численного расчета интеграла. Отрисовка трапеций и штриховки.

1. **Схема алгоритма**

****

****

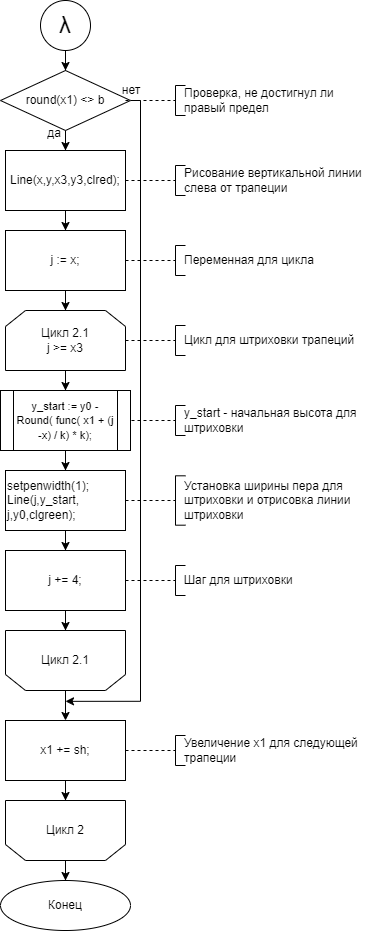
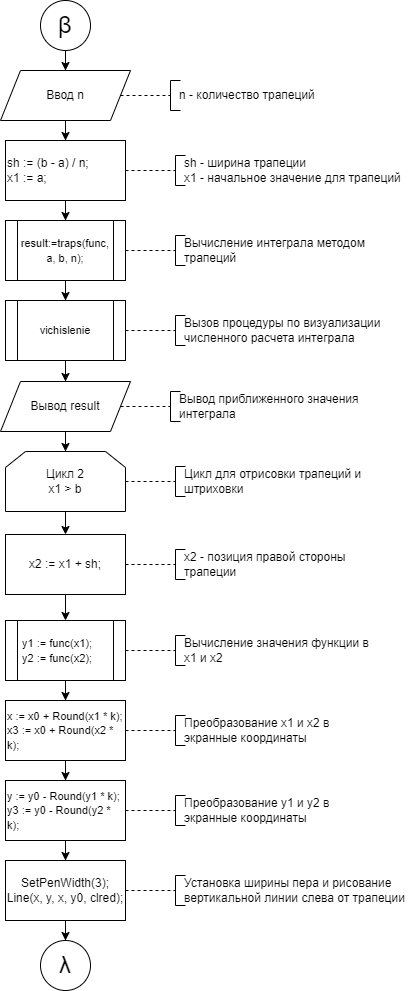
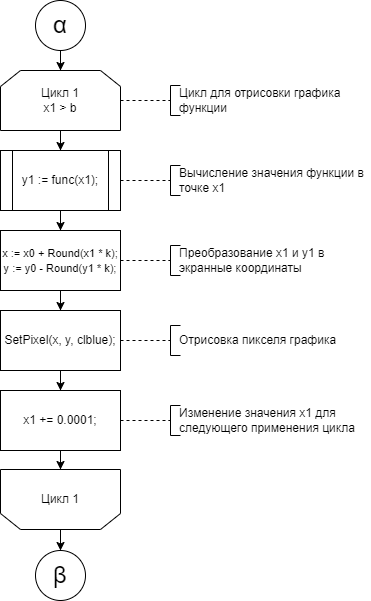
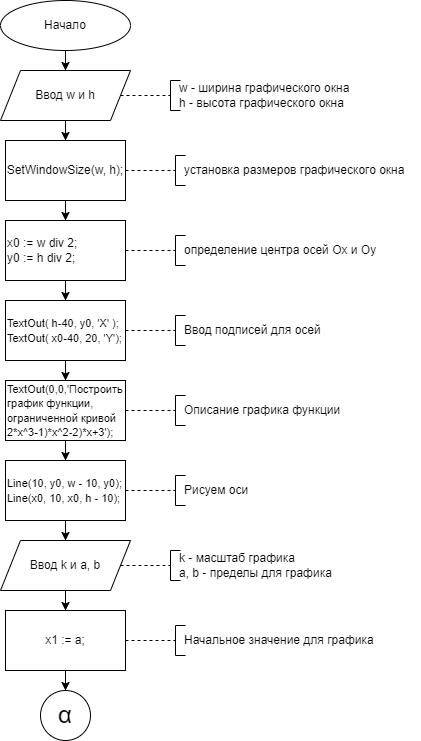
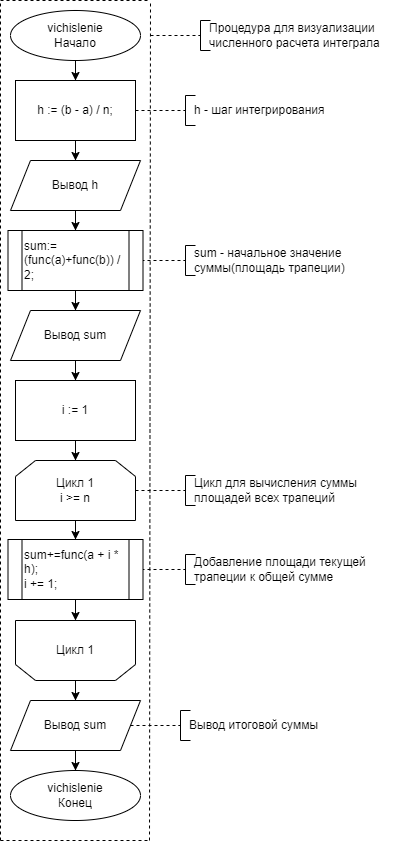
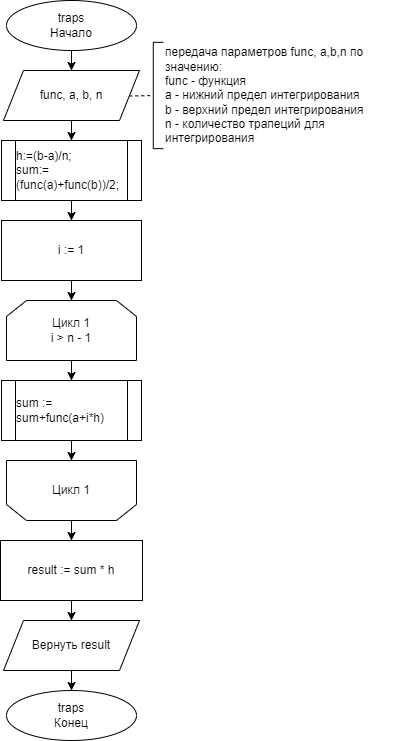
****

Рисунок 2 – Схема алгоритма задачи

1. **Код программы**

**uses** GraphABC;

**var**

a, b, x1, y1, k, sh, y2, x2, result, sum: real;

n, x, y, x0, y0, w, y3, x3, h: integer;

**type**

FuncType = **function**(x: Real): Real;

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

**begin**

result := 2 \* power(x, 3) + (-1) \* power(x, 2) + (-2) \* x + (3);

**end**;

**function** traps(func: FuncType;a, b: real; n: integer): real;

**var**

h, sum: real;

i: integer;

**begin**

h := (b - a) / n;

sum := (func(a) + func(b)) / 2;

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

sum := sum + func(a + i \* h);

result := sum \* h;

**end**;

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

**begin**

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

**end**;

**function** integral(a, b: real): real;

**begin**

result := pervoobraz(b) - pervoobraz(a);

**end**;

**procedure** vichislenie;

**var** i : integer;

h : real;

**begin**

writeln('Численный расчет интеграла: ');

h := (b - a) / n;

writeln('h = (b - a) / n = ', h);

sum := (func(a) + func(b)) / 2;

writeln('sum = (func(a) + func(b)) / 2 = ', sum);

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

sum := sum + func(a + i \* h);

writeln('Сложение площадей всех трапеций: sum = ', sum);

**end**;

**begin**

writeln('Введите ширину: ');

readln(w);

writeln('Введите высоту: ');

readln(h);

SetWindowSize(w, h);

x0 := w **div** 2;

y0 := h **div** 2;

TextOut(h + 40, y0, 'Х');

TextOut(x0 - 15, 20, 'Y');

TextOut(0,0,'Построить график функции, ограниченной кривой 2 \* x^3 - 1 \* x^2 - 2 \* x + 3');

Line(10, y0, w - 10, y0);

Line(x0, 10, x0, h - 10);

writeln('Введите масштаб графика: ');

readln(k);

writeln('Введите пределы участка для графика: ');

readln(a, b);

x1 := a;

// Рисуем график функции

**while** x1 <= b **do**

**begin**

y1 := func(x1);

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

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

SetPixel(x, y, clblue);

x1 += 0.0001;

**end**;

writeln('Введите количество трапеций: ');

readln(n);

sh := (b - a) / n;

x1 := a;

result := traps(func, a, b, n);

vichislenie;

writeln('Приближенное значение интеграла: ', result);

// Рисуем трапеции и штриховку

**while** x1 <= b **do**

**begin**

y1 := func(x1);

x2 := x1 + sh;

y2 := func(x2);

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

x3 := x0 + Round(x2 \* k);

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

y3 := y0 - Round(y2 \* k);

SetPenWidth(3);

Line(x, y, x, y0, clred);

**if** round(x1) <> b **then**

**begin**

Line(x, y, x3, y3, clred);

// Штриховка трапеций

**var** j: integer;

j:=x;

**while** j<x3 **do**

**begin**

**var** y\_start := y0 - Round(func(x1 + (j - x) / k) \* k);

setpenwidth(1);

Line(j, y\_start, j, y0, clgreen);

j+=4;

**end**;

**end**;

x1 += sh;

**end**;

**end**.

1. **Результат выполнения программы**

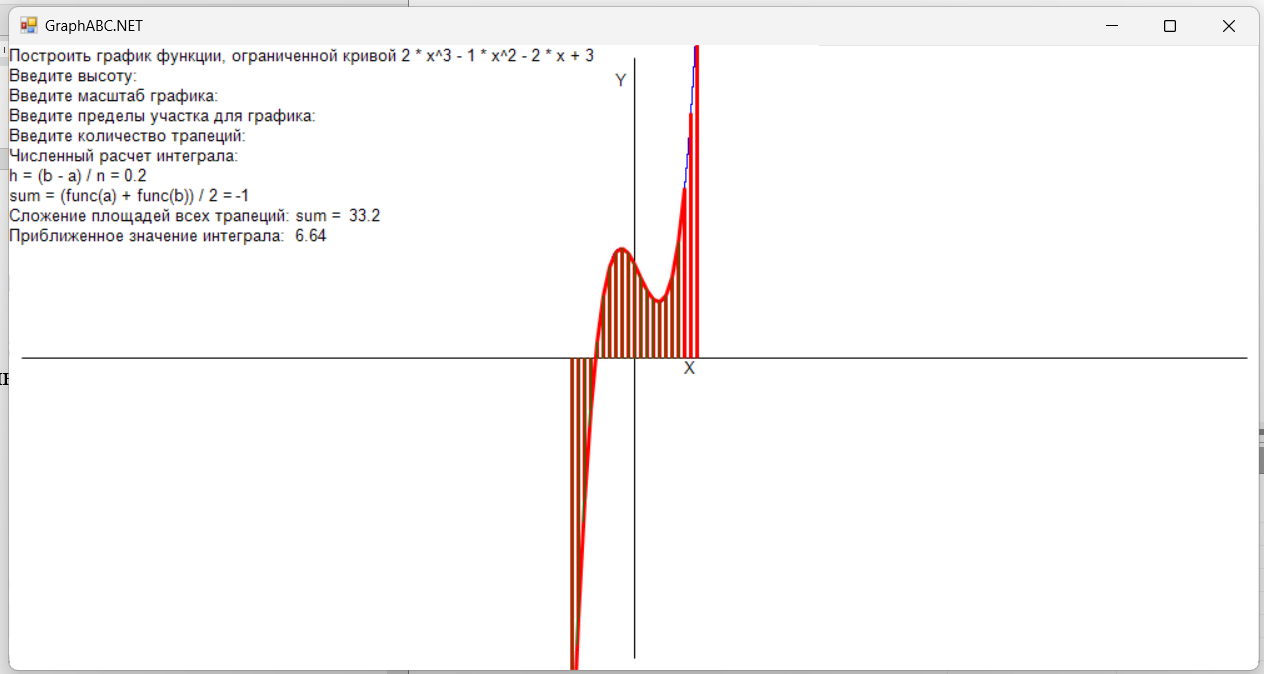
****

Рисунок 3 – Результат выполнения программы

1. **Вывод**

В ходе данной работы были успешно освоены принципы работы в графическом режиме, что позволило глубже понять основы компьютерной графики и взаимодействия с графическими примитивами. Участие в практических заданиях дало возможность наглядно ознакомиться с методами создания и манипуляции графическими объектами, такими как линии, фигуры и текст. Эти базовые навыки являются важным шагом на пути к более сложным задачам в области графического дизайна и программирования.

Для работы над схемами алгоритма необходимо четко сформулировать проблему, которую необходимо решить, собрать входные данные, необходимые для алгоритма, описать последовательность шагов, которые необходимо выполнить для решения задачи, использовать стандартные символы для построения схемы алгоритма (овалы – начало и конец, прямоугольники – действия или процессы, ромбы – условия или проверки и т.д.), проверить алгоритм на разных входных данных для выявления возможных ошибок, проанализировать алгоритм и внести изменения для повышения его эффективности, записать и оформить полученные результаты. Работа над схемами алгоритма помогает визуализировать процесс и обеспечить более легкое понимание и оптимизацию решения задачи.