**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

Алембаев Артём Сергеевич

Преподаватель:

Сергеева Елизавета Григорьевна

1. **Цель работы**

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

1. **Формулировка задания (с вариантом)**

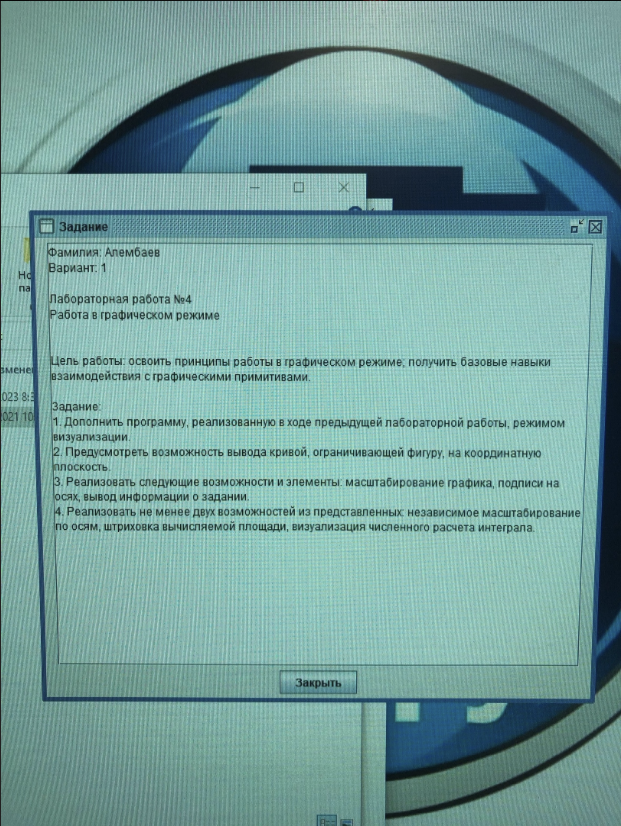
****

Рисунок 1 – Вариант

1. **Описание алгоритма**

Алгоритм состоит из нескольких функций:

Функция f(x), которая задает исходную функцию для интегрирования.

Функция f1(x, zoom), которая преобразует исходную функцию в соответствии с заданным масштабом графика.

Функция integral\_graph(a, b, mode, graph), которая вычисляет интеграл функции методом прямоугольников и, при необходимости, строит график функции.

Главная часть алгоритма выполняет следующую последовательность действий:

Устанавливает размеры окна графического интерфейса.

Запрашивает у пользователя масштаб графика и функции.

В цикле предлагает пользователю выбор программы: вычисление интеграла или выход из программы.

В зависимости от выбора пользователя выполняет соответствующие действия:

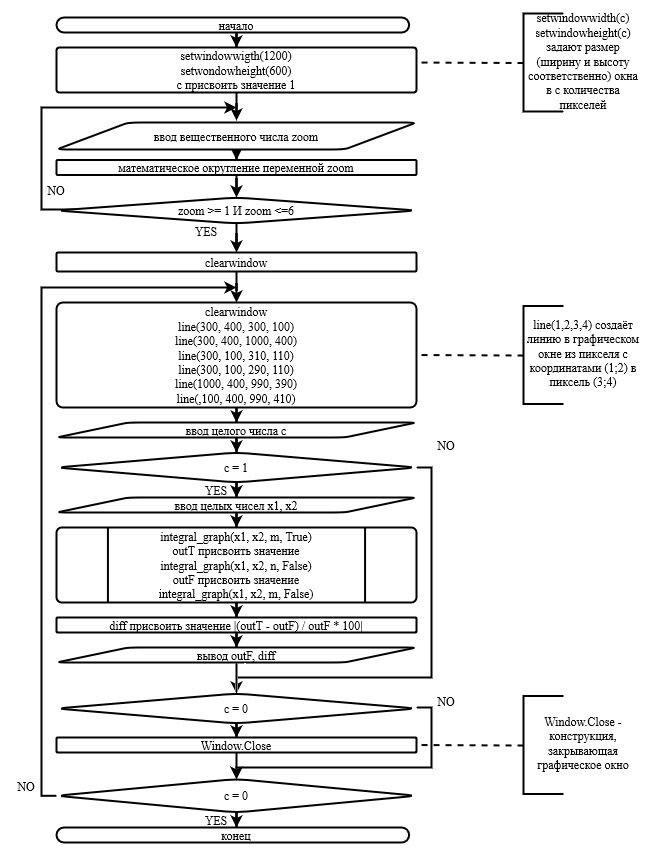
Если выбрано вычисление интеграла, запрашивает пределы интегрирования, строит график функции, вычисляет интеграл методом прямоугольников и выводит результаты на экран.

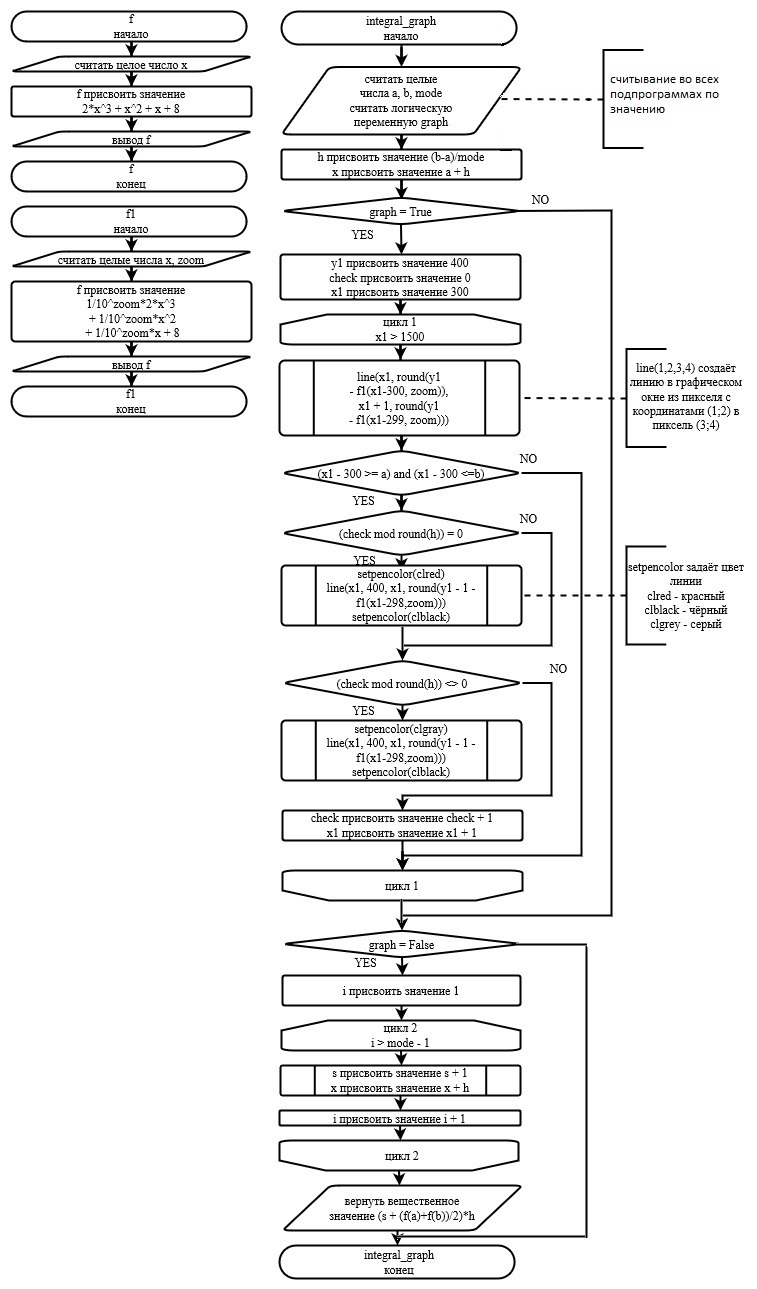
Если выбран выход, завершает работу программы.

Повторяет цикл, пока пользователь не выберет выход из программы.

В конце алгоритма происходит закрытие окна графического интерфейса при выборе пользователя выхода из программы.

1. **Схема алгоритма с комментариями**





1. **Код программы**

Uses GraphABC;

var

x1, x2 : integer;

outT, outF, diff, zoom : real;

c : byte;

const

n = 1000000000;

m = 10;

function f(x:real):real;begin f := x\*x\*x + 2\*x\*x + 3\*x + 18; end;

function f1(x, zoom:real):real;

begin

f1 := 1/power(10, zoom)\*x\*x\*x +

1/power(10, zoom)\*2\*x\*x +

1/power(10, zoom)\*3\*x +

18;

end;

function integral\_graph(a, b, mode: integer; graph : boolean): real;

var

h, s, x : real;

y1, check: integer;

begin

h := (b-a)/mode;

x:=a+h;

if graph = True then

begin

y1 := 400;

check := 0;

for x1:integer:= 300 to 1300 do

begin

line(x1, round(y1 - f1(x1-300, zoom)), x1 + 1, round(y1 - f1(x1-299, zoom)));

if (x1 - 300 >= a) and (x1 - 300 <= b) then

begin

if ((check mod round(h)) = 0) or (x1 = b) then

begin

setpencolor(clred);

line(x1, 400, x1, round(y1 - 1 - f1(x1-298, zoom)));

setpencolor(clBlack);

end;

if (check mod round(h)) <> 0 then

begin

setpencolor(clGray);

line(x1, 400, x1, round(y1 - 1 - f1(x1-298, zoom)));

setpencolor(clblack);

end;

check:=check+1;

end;

end;

end;

if graph = False then

begin

for i:integer := 1 to mode - 1 do

begin

s := s + f(x);

x:=x+h;

end;

integral\_graph := (s + (f(a) + f(b)) / 2) \* h;

end;

end;

begin

SetWindowWidth(1200);

SetWindowHeight(600);

c := 1;

while True do

begin

textout(5, 10, 'Введите масштаб графика и функции (рекомендуется максимальный) : ');

textout(5, 25, 'От 1 до 6');

read(zoom);

zoom:=round(zoom);

if (zoom >= 1) and (zoom <= 6) then break;

ClearWindow;

end;

repeat

ClearWindow;

// область графика

Line(300, 400, 300, 100);

Line(300, 400, 1000, 400);

Line(300, 100, 310, 110);

Line(300, 100, 290, 110);

Line(1000, 400, 990, 390);

Line(1000, 400, 990, 410);

textout(5, 10, 'Вычисление интеграла - 1');

textout(5, 25, 'Выход - 0');

textout(5, 40, 'Выберите программу: ');

textout(280, 100, 'y');

textout(990, 410, 'x');

textout(295, 405, '0');

readln(c);

Case c of

1:

begin

textout(5, 70, 'Введите пределы интегрирования: ');

readln(x1, x2);

textout(5, 90, 'Построение графика . . .');

integral\_graph(round(x1\*power(10, (zoom - 1)\*(1/zoom))),

round(x2\*power(10, (zoom - 1)\*(1/zoom))), m, True);

textout(5, 120, 'Идёт интегрирование . . . ');

outT := integral\_graph(x1, x2, n, False);

outF := integral\_graph(x1, x2, m, False);

diff := (outT - outF) / outF \* 100;

textout(5, 145, 'Результаты интегрирования: ');

textout(5, 160, outF+' ед^2');

textout(5, 180, 'Погрешность: ');

textout(5, 195, abs(diff)+'%');

textout(5, 230, 'Нажмите ENTER');

readln;

end;

0:

begin

textout(5, 70, 'Выключение программы . . .');

Sleep(300);

Window.Close;

end;

end;

until c = 0;

end.

1. **Результат выполнения программы**

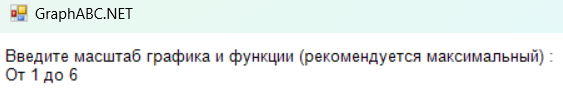
****

Рисунок 2 – Как выглядит изначально

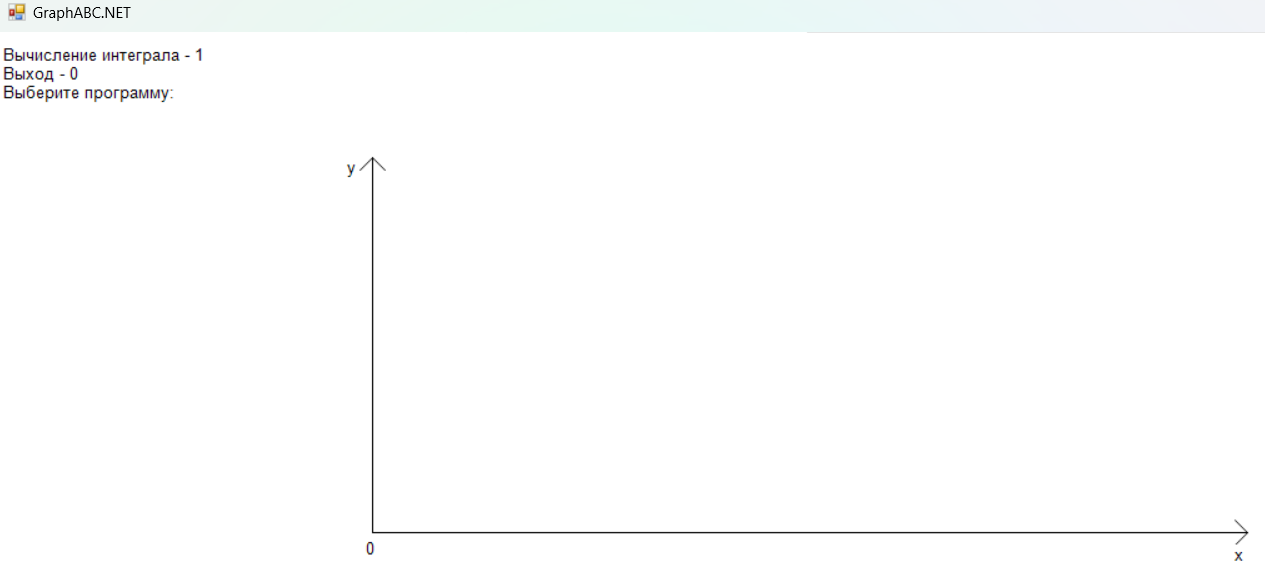


Рисунок 3 – Выбор программ

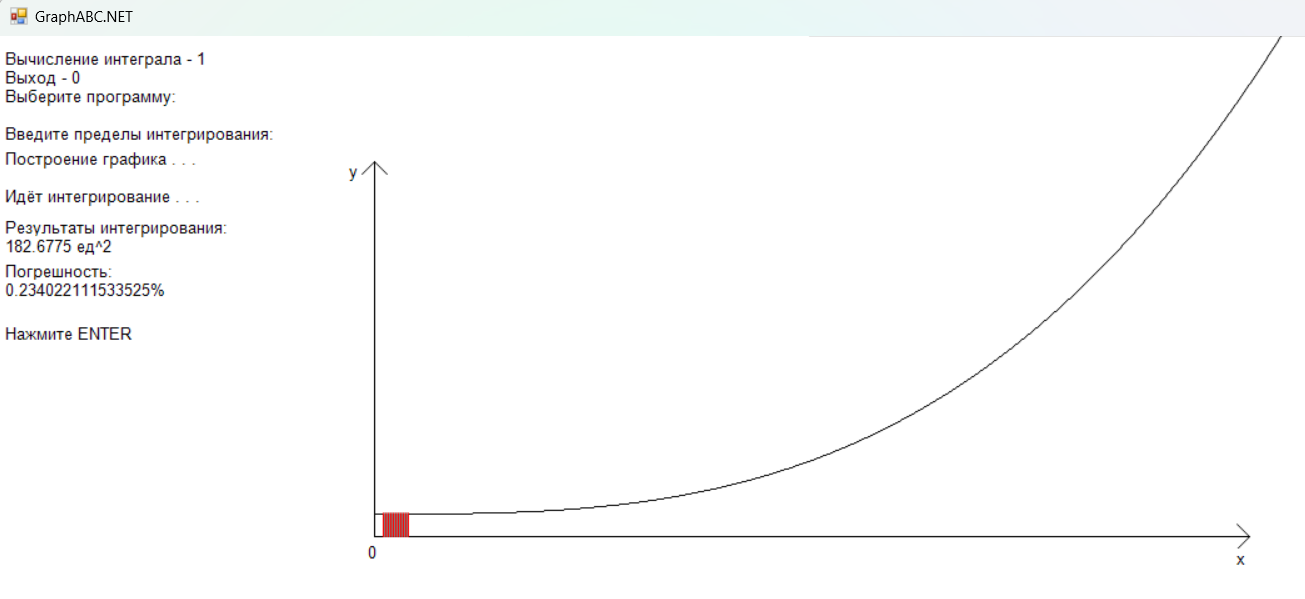


Рисунок 4 – Вывод рисунка

1. **Вывод**

Были изучены способ особого вывода данных при помощи встроенной коллекции, позволяющей вызывать case-меню. Изучены способы взаимодействия программ и подпрограмм в коде и в схемах алгоритмов, а также передача этих данных в графический вид, изучены понятия: линии, точки, квадраты, круги и другие объекты в графике. Изучены способ вычисления интеграла без дифференцирования, но при помощи разделения графиков, способами трапеции и прямоугольников, а также способ выражения этого через графику. Были изучены процедуры и функции коллекции GraphABC для отчистки данных с экрана пользователя и лучшего форматирования текста, изображения. Изучен метод нахождения относительной погрешности в вычислениях интеграла и других функций. Был изучен способ графического взаимодействия с пользователем. Все задачи были выполнены, а цель достигнута.