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»**

Выполнил: студент учебной группы

ИСПк-204-52-00

Колобов Александр Алексеевич

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2023

Цель работы

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

Формулировка задания

Вариант 9.

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 расчёта интеграла.

Описание алгоритма

Данный алгоритм создаёт интерфейс на основе библиотеки GraphABC, в которой предоставляется возможность найти площадь криволинейной прямой функции с помощью двух методов: метод интеграла и метод трапеций. Также в главном меню можно назначить другие пределы, оценить погрешность полученного результата между двумя методами, изменить общий масштаб и изменить масштаб по осям X и Y. В новом окне появилась возможность просматривать функцию.

Схема алгоритма с комментариями

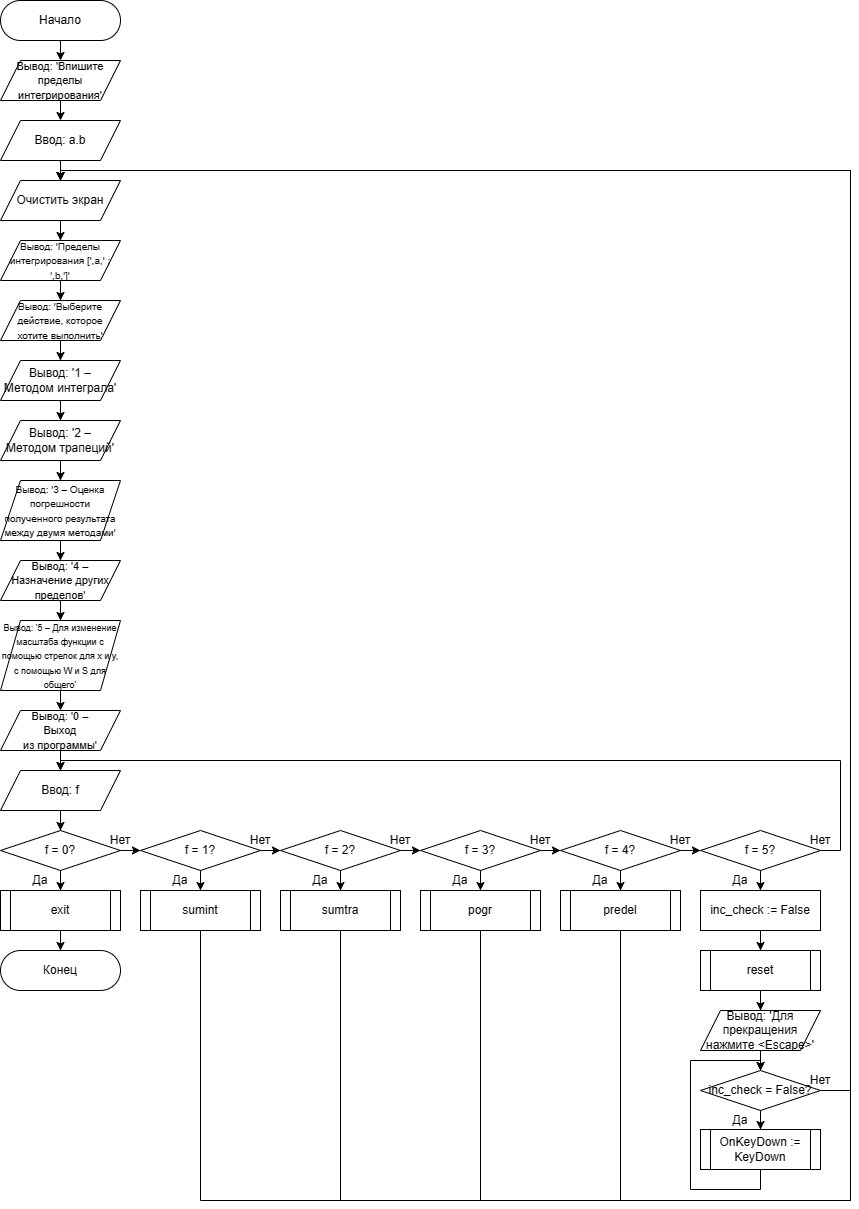


Рисунок 1 – Схема главного алгоритма

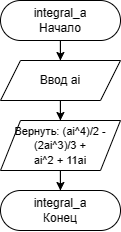


Рисунок 2 – Схема алгоритма процедуры integral

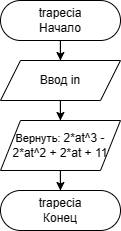


Рисунок 3 – Схема алгоритма функции trapecia

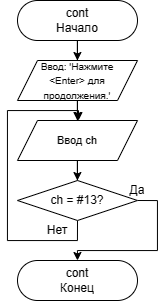


Рисунок 4 – Схема алгоритма процедуры cont

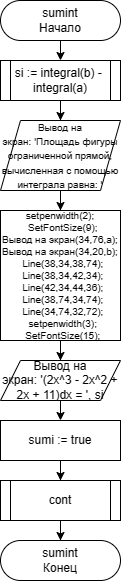


Рисунок 5 – Схема алгоритма процедуры sumint

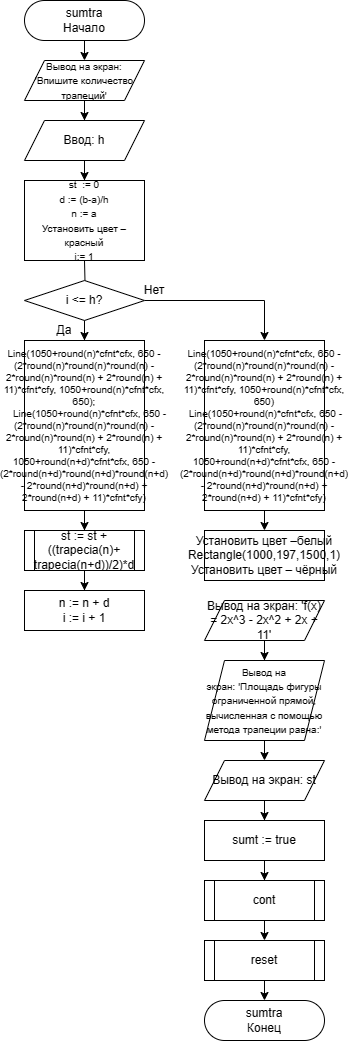


Рисунок 6 – Схема алгоритма процедуры sumtra

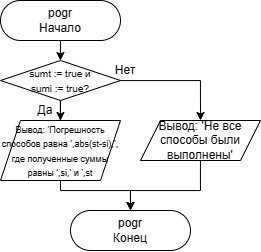


Рисунок 7 – Схема алгоритма процедуры pogr

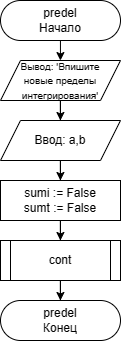


Рисунок 8 – Схема алгоритма процедуры predel

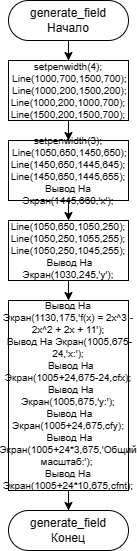
**

Рисунок 9 - Схема алгоритма процедуры generate\_field

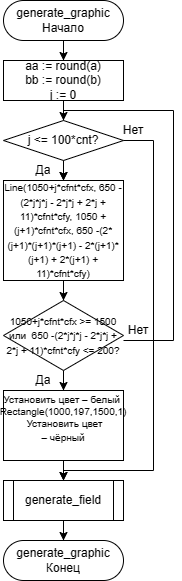
**

Рисунок 10 - Схема алгоритма процедуры generate\_graphic

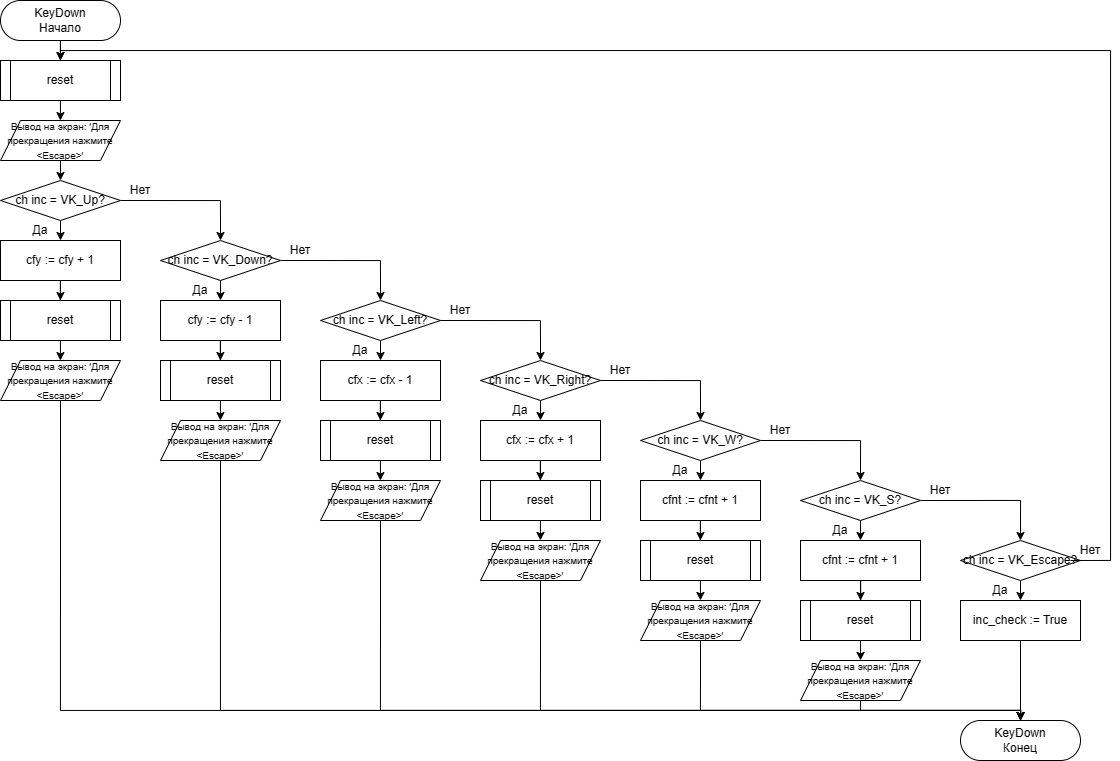


Рисунок 11 - Схема алгоритма процедуры KeyDown

Код программы

//uses Crt;

**uses** GraphABC;

**var** a,b,si,st,d,n : real; h,f,cfnt,ch\_inc,cfy,cfx : integer; sumt, sumi,inc\_check : boolean; ch : char;

**function** integral(ai: real) : real;

**begin**

Result := ((ai\*ai\*ai\*ai)/2 - (2\*ai\*ai\*ai)/3 + ai\*ai + 11\*ai);

**end**;

**function** main\_func(at: real) : real;

**begin**

Result := (2\*at\*at\*at - 2\*at\*at + 2\*at + 11);

**end**;

**procedure** generate\_field;

**begin**

setpenwidth(4);

Line(1000,700,1500,700);

Line(1000,200,1500,200);

Line(1000,200,1000,700);

Line(1500,200,1500,700);

setpenwidth(3);

Line(1050,650,1450,650);

Line(1450,650,1445,645);

Line(1450,650,1445,655);

TextOut(1445,660,'x');

Line(1050,650,1050,250);

Line(1050,250,1055,255);

Line(1050,250,1045,255);

TextOut(1030,245,'y');

TextOut(1130,175,'f(x) = 2x^3 - 2x^2 + 2x + 11');

TextOut(1005,675-24,'x:');

TextOut(1005+24,675-24,cfx);

TextOut(1005,675,'y:');

TextOut(1005+24,675,cfy);

TextOut(1005+24\*3,675,'Общий масштаб:');

TextOut(1005+24\*10,675,cfnt);

**end**;

**procedure** generate\_graphic;

**var** j,aa,bb : integer;

**begin**

aa := round(a);

bb := round(b);

**for** j := 0 **to** 100\*cfnt **do begin**

Line(1050+j\*cfnt\*cfx, 650 -(2\*j\*j\*j - 2\*j\*j + 2\*j + 11)\*cfnt\*cfy, 1050 + (j+1)\*cfnt\*cfx, 650 -(2\*(j+1)\*(j+1)\*(j+1) - 2\*(j+1)\*(j+1) + 2\*(j+1) + 11)\*cfnt\*cfy);

**if** (1050+j\*cfnt\*cfx >= 1500) **or** (650 -(2\*j\*j\*j - 2\*j\*j + 2\*j + 11)\*cfnt\*cfy <= 200) **then begin**

SetPenColor(clWhite);

Rectangle(1000,197,1500,1);

SetPenColor(clBlack);

**break**;

**end**;

//LineTo(1050+j\*2,650 -(2\*j\*j\*j - 2\*j\*j + 2\*j + 11)\*2);

//SetPixel(j,700 -(2\*j\*j\*j - 2\*j\*j + 2\*j + 11), clBlack);

**end**;

generate\_field;

**end**;

**procedure** reset;

**begin**

ClearWindow;

generate\_field;

generate\_graphic;

**end**;

**procedure** cont;

**begin**

TextOut(0,24\*20,'Нажмите <Enter> для продолжения.');

readln(ch);

**if** ch <> #13 **then** cont **else** TextOut(0,24\*20,' ');

**end**;

**procedure** sumint;

**begin**

reset;

si := integral(b) - integral(a);

TextOut(0,24\*0,'Площадь фигуры ограниченной прямой, вычисленная с помощью интеграла равна: ');

setpenwidth(2);

SetFontSize(9);

TextOut(34,76,a);

TextOut(34,20,b);

Line(38,34,38,74);

Line(38,34,42,34);

Line(42,34,44,36);

Line(38,74,34,74);

Line(34,74,32,72);

setpenwidth(3);

SetFontSize(15);

TextOut(48,40,'(2x^3 - 2x^2 + 2x + 11)dx = ');

TextOut(300,40,si);

sumi := true;

cont;

**end**;

**procedure** sumtra;

**var** i : integer; ff : real;

**begin**

reset;

st := 0;

TextOut(0,24\*0,'Впишите количество трапеций:');

readln(h);

d := (b-a)/h;

n := a;

SetPenColor(clRed);

**for** i := 1 **to** h **do begin**

Line(1050+round(n)\*cfnt\*cfx, 650 -(2\*round(n)\*round(n)\*round(n) - 2\*round(n)\*round(n) + 2\*round(n) + 11)\*cfnt\*cfy, 1050+round(n)\*cfnt\*cfx, 650);

Line(1050+round(n)\*cfnt\*cfx, 650 -(2\*round(n)\*round(n)\*round(n) - 2\*round(n)\*round(n) + 2\*round(n) + 11)\*cfnt\*cfy, 1050+round(n+d)\*cfnt\*cfx, 650 -(2\*round(n+d)\*round(n+d)\*round(n+d) - 2\*round(n+d)\*round(n+d) + 2\*round(n+d) + 11)\*cfnt\*cfy);

st := st + ((main\_func(n)+main\_func(n+d))/2)\*d;

n := n + d;

**end**;

Line(1050+round(n)\*cfnt\*cfx, 650 -(2\*round(n)\*round(n)\*round(n) - 2\*round(n)\*round(n) + 2\*round(n) + 11)\*cfnt\*cfy, 1050+round(n)\*cfnt\*cfx, 650);

Line(1050+round(n)\*cfnt\*cfx, 650 -(2\*round(n)\*round(n)\*round(n) - 2\*round(n)\*round(n) + 2\*round(n) + 11)\*cfnt\*cfy, 1050+round(n+d)\*cfnt\*cfx, 650 -(2\*round(n+d)\*round(n+d)\*round(n+d) - 2\*round(n+d)\*round(n+d) + 2\*round(n+d) + 11)\*cfnt\*cfy);

SetPenColor(clWhite);

Rectangle(1000,197,1500,1);

SetPenColor(clBlack);

TextOut(1130,175,'f(x) = 2x^3 - 2x^2 + 2x + 11');

TextOut(0,24\*1,'Площадь фигуры ограниченной прямой, вычисленная с помощью метода трапеции равна:');

TextOut(0,24\*2,st);

sumt := true;

cont;

reset;

**end**;

**procedure** KeyDown(ch\_inc : integer);

**begin**

reset;

TextOut(0,24\*0,'Для прекращения нажмите <Escape>');

//repeat

//cfnt\_check := cfnt;

**case** ch\_inc **of**

VK\_Up : **begin** cfy := cfy + 1; reset; TextOut(0,24\*0,'Для прекращения нажмите <Escape>'); **end**;

VK\_Down : **begin** cfy := cfy - 1; reset; TextOut(0,24\*0,'Для прекращения нажмите <Escape>'); **end**;

VK\_Left : **begin** cfx := cfx - 1; reset; TextOut(0,24\*0,'Для прекращения нажмите <Escape>'); **end**;

VK\_Right : **begin** cfx := cfx + 1; reset; TextOut(0,24\*0,'Для прекращения нажмите <Escape>'); **end**;

VK\_W : **begin** cfnt := cfnt + 1; reset; TextOut(0,24\*0,'Для прекращения нажмите <Escape>'); **end**;

VK\_S : **begin** cfnt := cfnt - 1; reset; TextOut(0,24\*0,'Для прекращения нажмите <Escape>'); **end**;

VK\_Escape : inc\_check := True;

**end**;

//if cfnt\_check <> cfnt then generate\_graphic;

//until ch\_inc = VK\_Escape;

**end**;

**procedure** pogr;

**begin**

reset;

**if** sumt **and** sumi **then begin**

TextOut(0,24\*0,'Погрешность способов равна: ');

TextOut(0,24\*1,abs(st-si));

TextOut(0,24\*3,'Где полученные суммы были равны: ');

TextOut(0,24\*4,si);

TextOut(0,24\*5,st); **end else** TextOut(0,24\*0,'Не все способы были выполнены');

cont;

**end**;

**procedure** predel;

**begin**

reset;

TextOut(0,24\*0,'Впишите пределы интегрирования:');

readln(a,b);

sumi := False;

sumt := False;

cont;

**end**;

**begin**

SetWindowWidth(1600);

SetWindowHeight(900);

SetFontSize(15);

cfnt := 1;

cfx := 1; {коэффициенты масштаба}

cfy := 1;

predel;

//TextOut(0,0,'Впишите коэффициент масштабирования:');

//readln(cfnt);

**repeat**

reset;

//ClrScr;

TextOut(0,24\*0,'Пределы интегрирования:');

TextOut(0,24\*1,'a = ');

TextOut(36,24\*1,a);

TextOut(0,24\*2,'b = ');

TextOut(36,24\*2,b);

TextOut(0,24\*3,'Выберите действие, которое хотите выполнить');

TextOut(0,24\*4,'1 – Методом интеграла');

TextOut(0,24\*5,'2 – Методом трапеций');

TextOut(0,24\*6,'3 – Оценка погрешности полученного результата между двумя методами');

TextOut(0,24\*7,'4 – Назначение других пределов');

TextOut(0,24\*8,'5 – Для изменение масштаба функции с помощью стрелок для x и y, с помощью W и S для общего');

TextOut(0,24\*9,'0 – Выход из программы');

readln(f);

**case** f **of**

0: CloseWindow;

1: sumint;

2: sumtra;

3: pogr;

4: predel;

5: **begin**; inc\_check := False; reset; TextOut(0,24\*0,'Для прекращения нажмите <Escape>'); **while** inc\_check = False **do begin** OnKeyDown := KeyDown; **end**; **end**;

**end**;

**until** f = 0;

**end**.

Результат выполнения программы

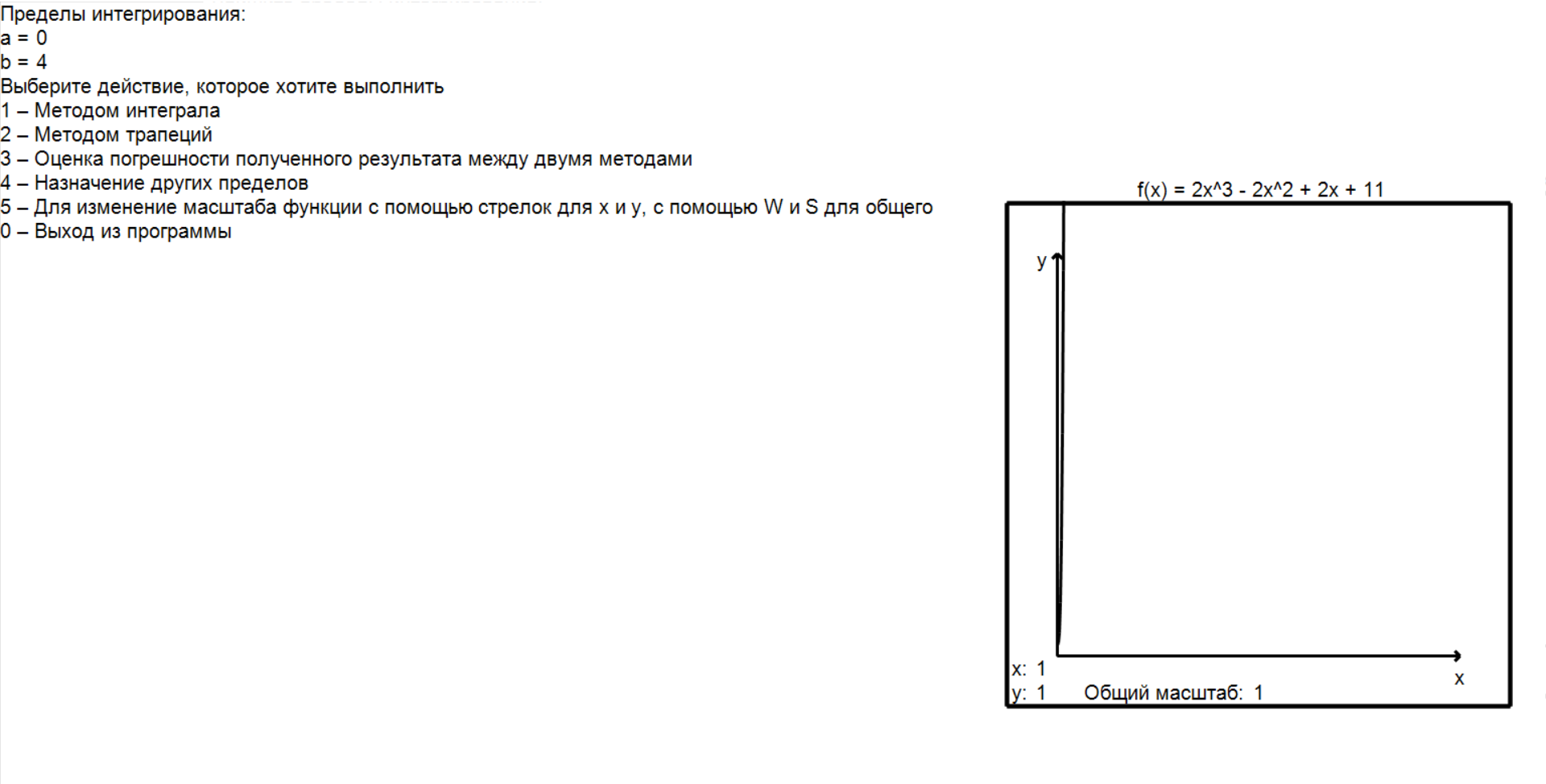


Рисунок 12 - Результат выполнения программы

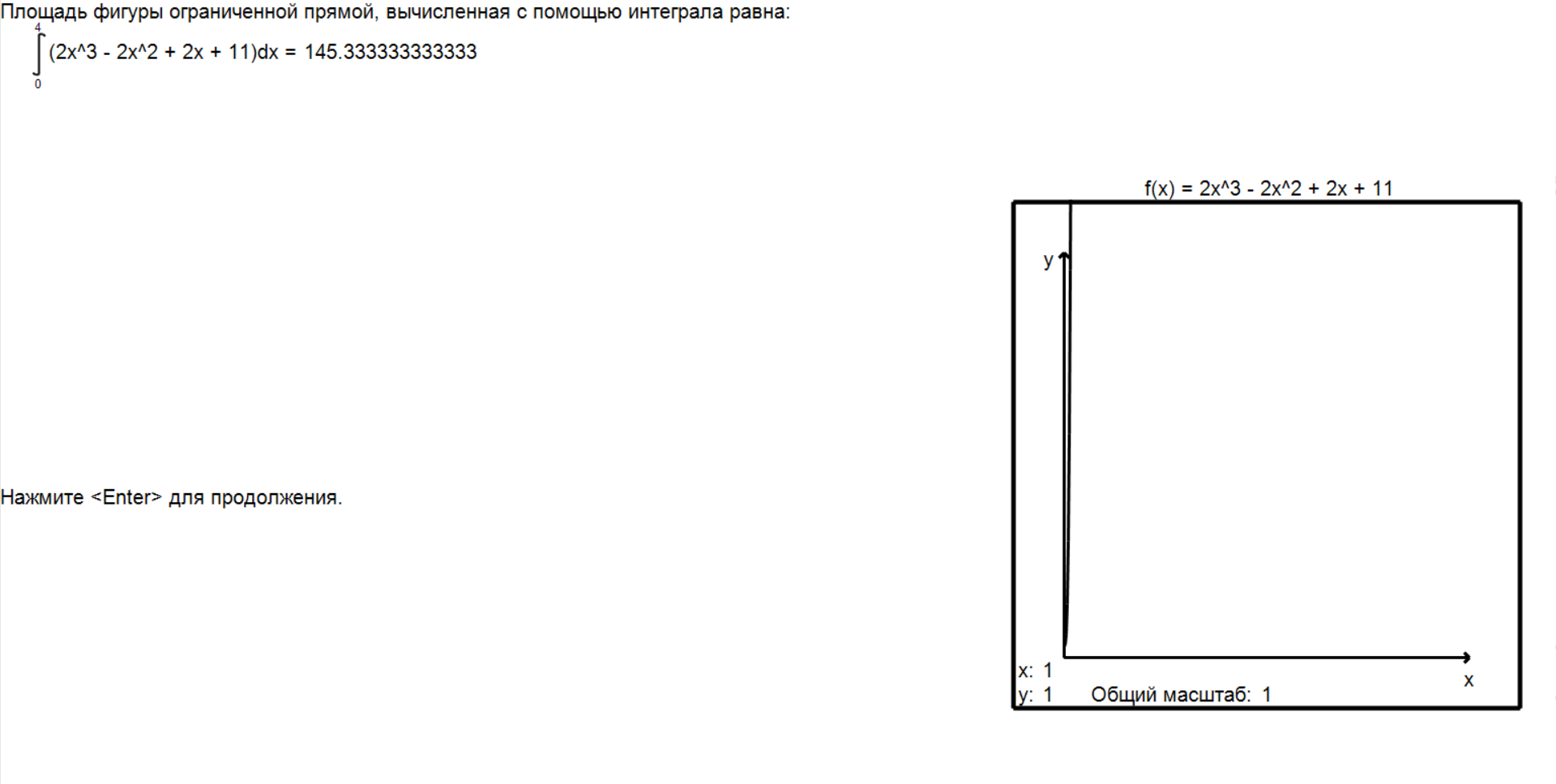


Рисунок 13 – Результат выполнения процедуры sumint

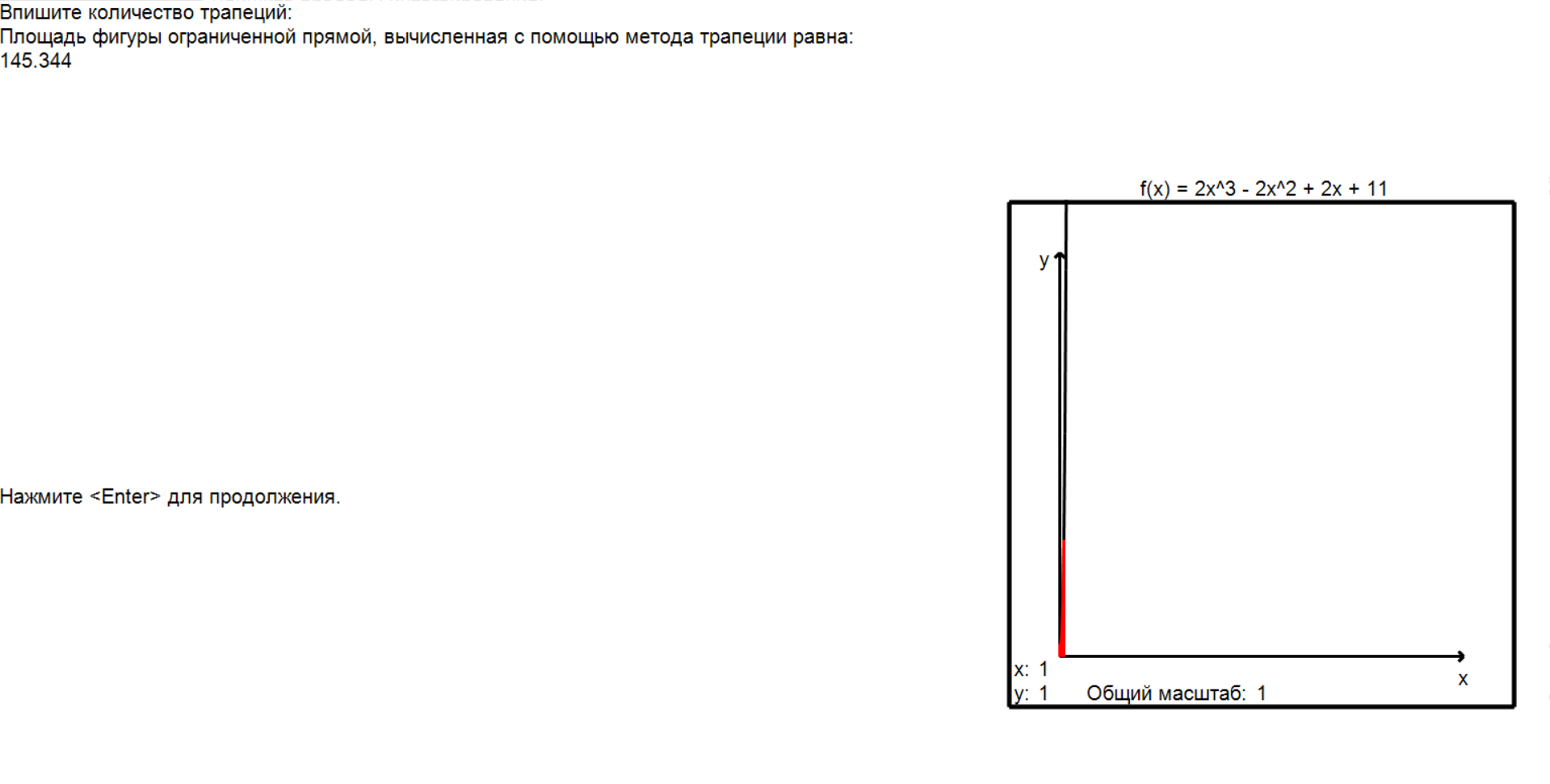


Рисунок 11 – Результат выполнения процедуры sumtra

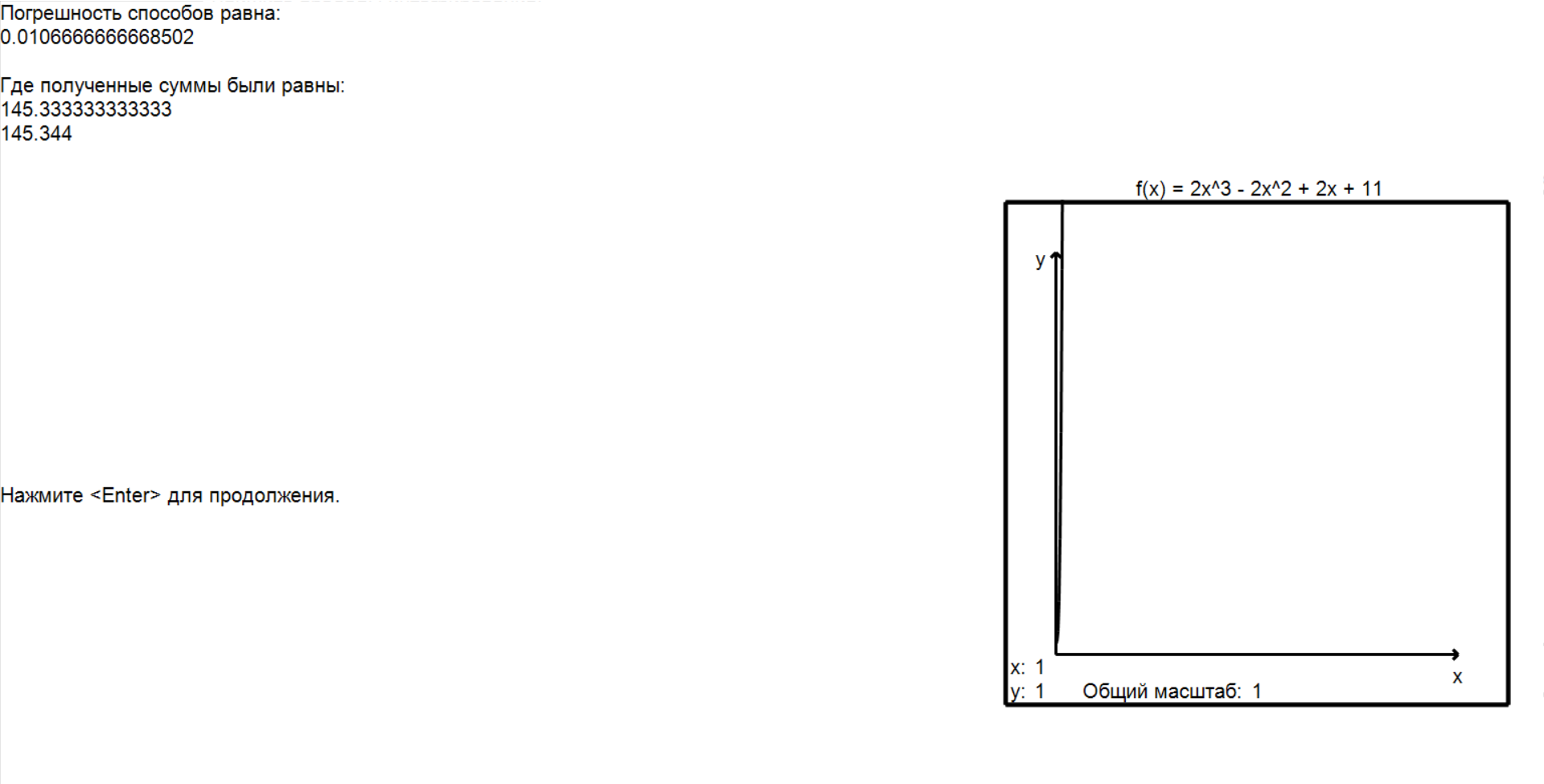


Рисунок 14 – Результат выполнения процедуры pogr

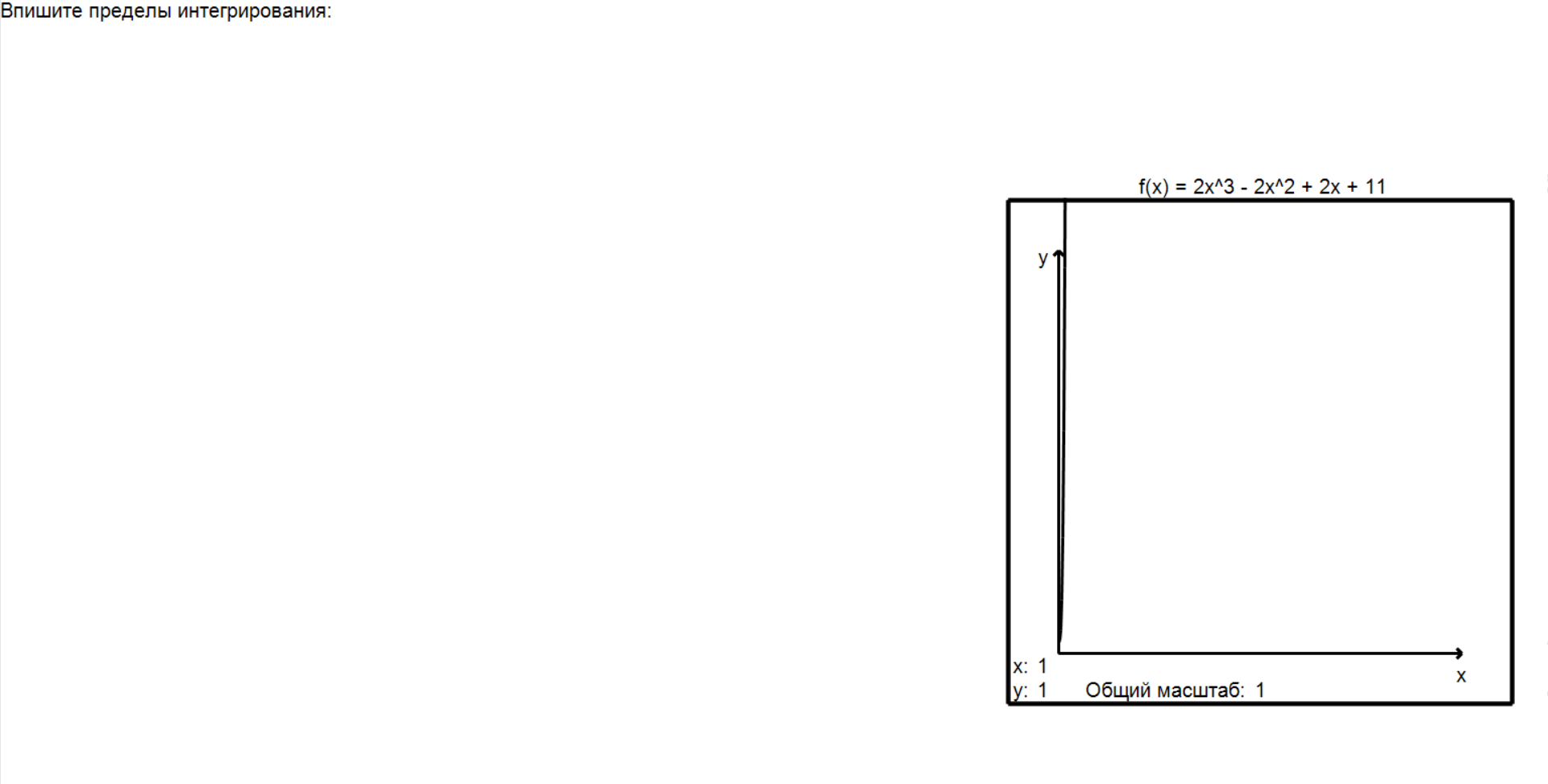


Рисунок 13 – Результат выполнения процедуры predel

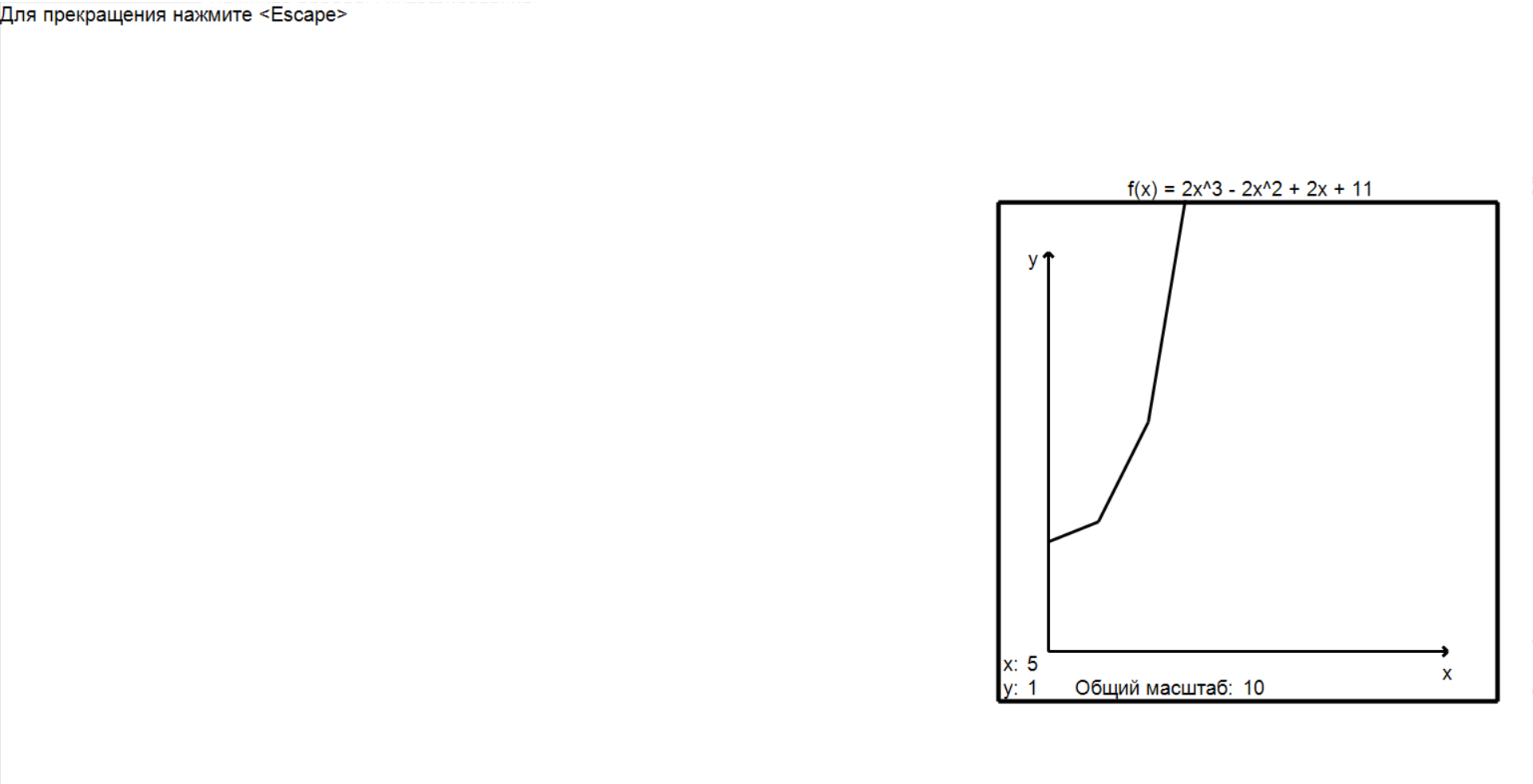


Рисунок 15 – Результат выполнения процедуры KeyDown

Вывод

В ходе выполнения данной работы мы смогли применить изученный материал лабораторных занятий на практике, использовав библиотеку GraphABC, команды SetFontSize, SetWindowWidth, SetWindowHeight, ClearWindow, SetPenColor, TextOut, Line, Rectangle, процедуры, функции, условный оператор, цикл for, создание переменных и операторов ввода и вывода, а также мы построили схему алгоритма с помощью бесплатного онлайн-сервиса app.diagrams.net.

В результате выполнения данной работы 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.

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.