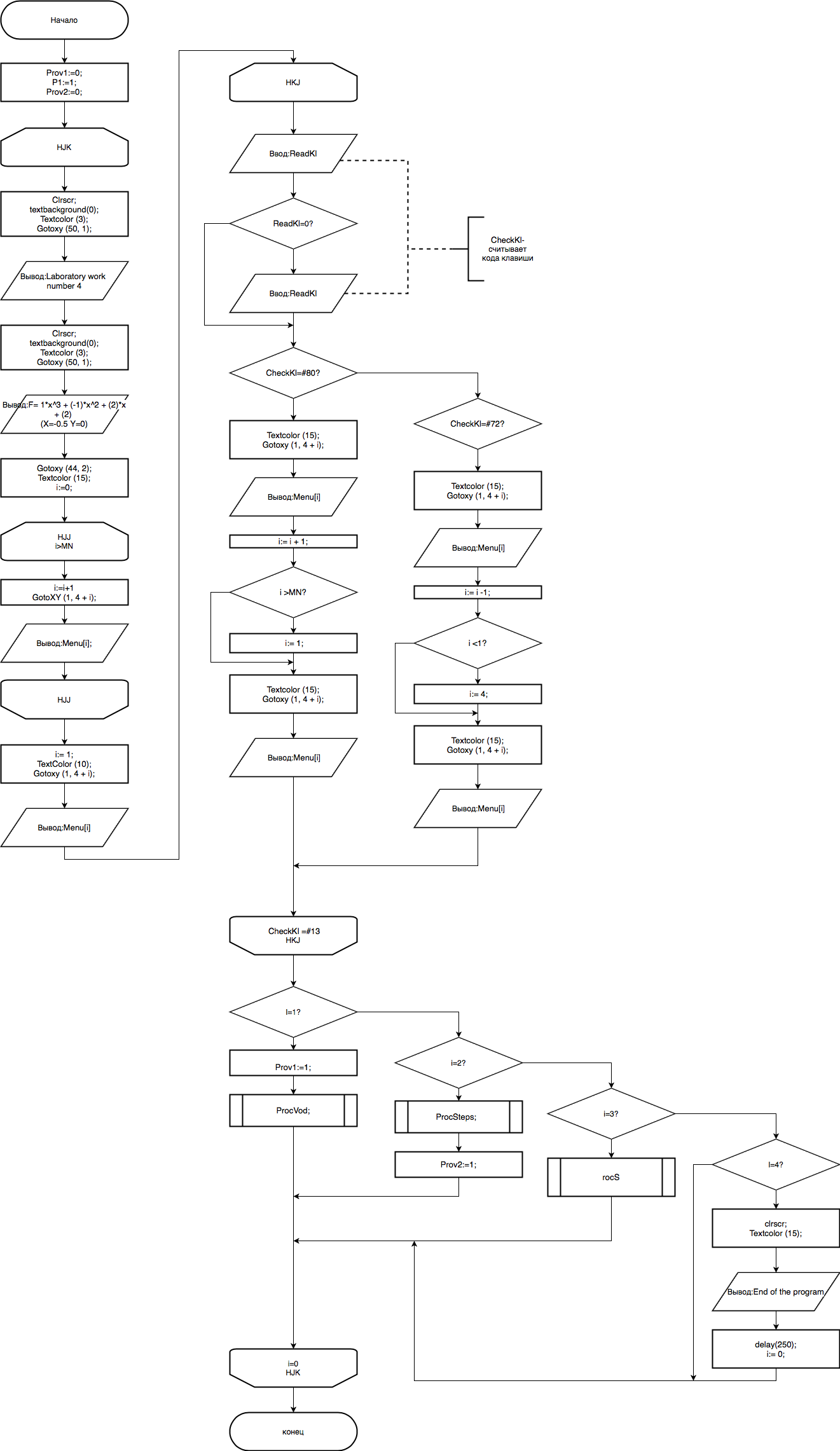
Вариант: 900

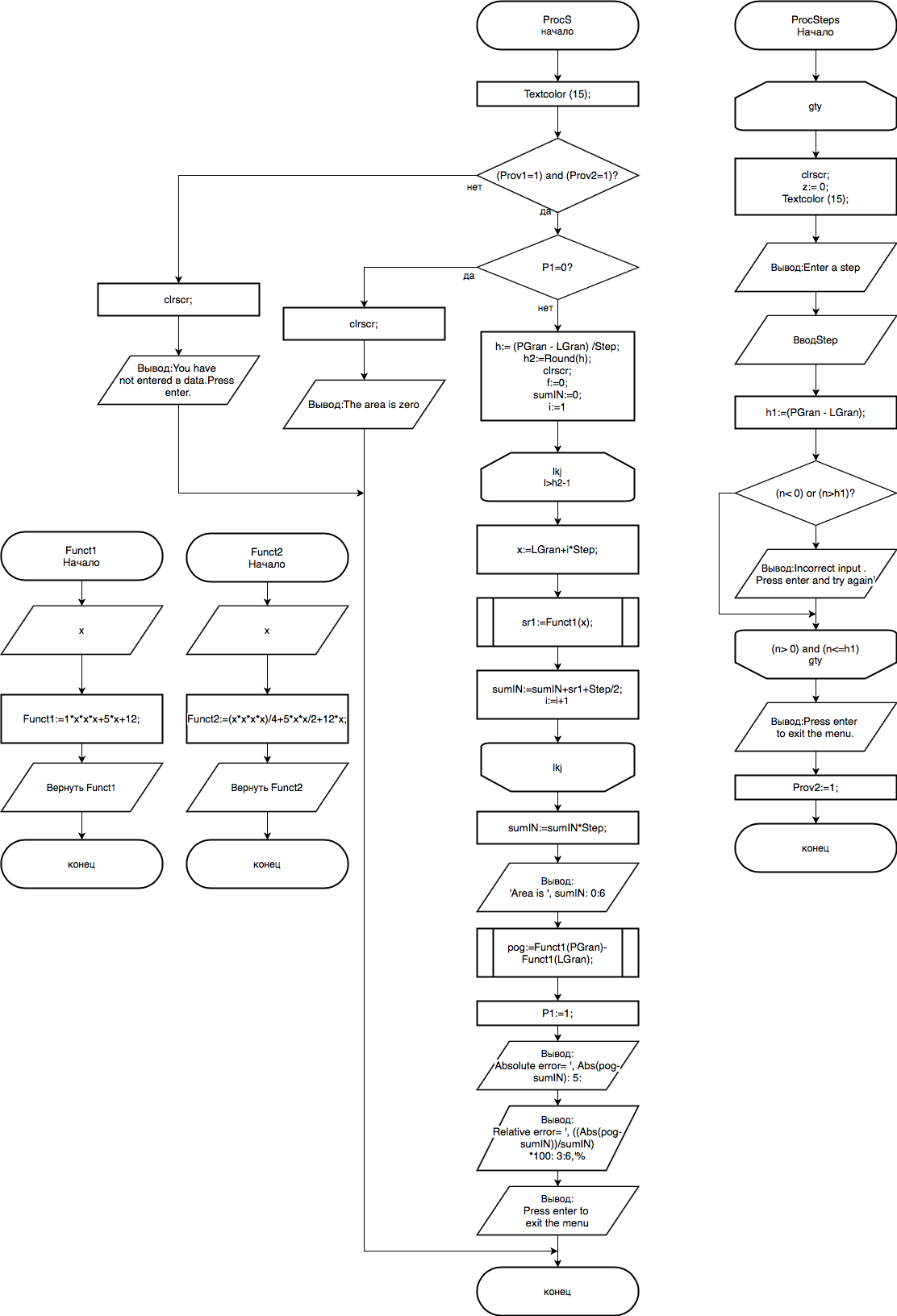
Цель работы: освоить синтаксис построения процедур и функций, изучить способы передачи данных в подпрограммы, получить навыки организации минимального пользовательского интерфейса.

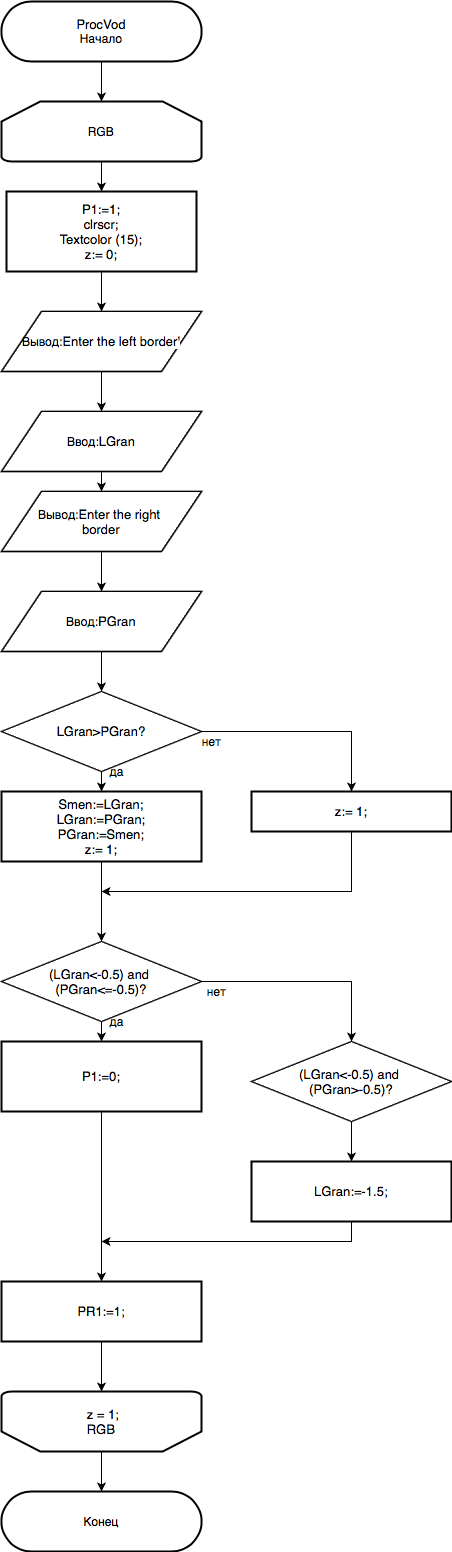
Задание:

1. Реализовать программу вычисления площади фигуры, ограниченной кривой и осью OX(в положительной части по оси OY).
2. Вычисление определенного интеграла должно выполняться численно, с применение метода средних прямоугольников.
3. Пределы интегрирования вводятся пользователем.
4. Взаимодействие с пользователем должно осуществляться посредством case-меню.
5. Требуется реализовать возможность оценки погрешности полученного результата.
6. Необходимо использовать процедуры и функции там, где это целесообразно.

Схема алгоритма:



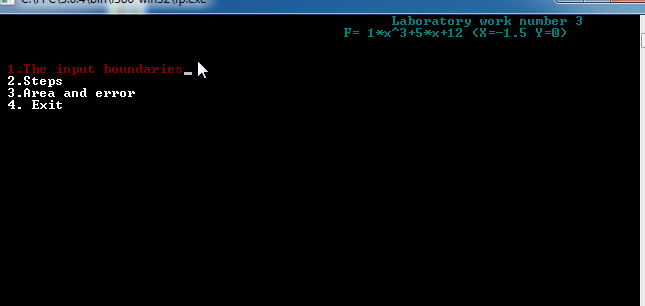


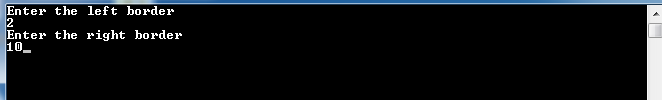


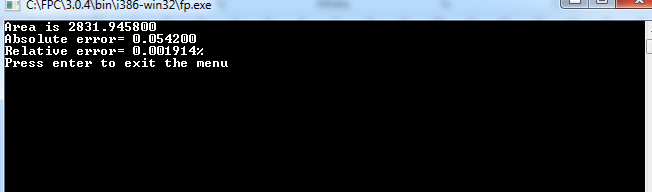
Исходный код

program laba3;   
{$mode delphi}   
uses wincrt,graph,crt;   
const   
MN= 4;   
Menu:array[1..MN]   
of string = (' 1.The input boundaries',   
' 2.Steps',   
' 3.Area and error',   
' 4. Exit');   
var LGran, PGran,P1, h, sumIN, pog,x,f,h1,Smen,Prov1:real;   
z, i,h2,Prov2:longint;   
n:real;   
ReadKl:char;   
Gd,Gm:smallint;   
function Funct1(x: real): real;   
begin   
Funct1:=x\*x\*x-2\*x\*x+2\*x+2;   
end;   
function Funct2(x: real): real;   
begin   
Funct2:=(x\*x\*x\*x)/4+(2\*x\*x\*x)/3+(x\*x)+2\*x;   
end;   
procedure ProcVod;   
begin   
repeat   
P1:=1;   
clrscr;   
Textcolor (15);   
z:= 0;   
writeln ('Enter the left border');   
readln (LGran);   
writeln ('Enter the right border');   
readln (PGran);   
if LGran>PGran then   
begin   
Smen:=LGran;   
LGran:=PGran;   
PGran:=Smen;   
z:= 1;   
end   
else   
z:= 1;   
if (LGran<-0.5) and (PGran<=-0.5) then   
begin   
P1:=0;   
end   
else   
if (LGran<-0.5) and (PGran>-0.5)then   
LGran:=-0.5;   
Prov1:=1;   
until z = 1;   
end;   
  
  
procedure ProcSteps;   
begin   
repeat   
clrscr;   
z:= 0;   
Textcolor (15);   
writeln ('Enter a step');   
readln (n);   
h1:=(PGran - LGran);   
if (n < 0) or (n>h1) then   
begin   
writeln ('Incorrect input . Press enter and try again');   
readln;   
end;   
until (n > 0) and (n<=h1);   
writeln ('Press enter to exit the menu.');   
Prov2:=1;   
readln;   
end;   
  
  
procedure ProcS;   
var ip:integer;   
sr1:real;   
begin   
Textcolor (15);   
if (Prov1=1) and (Prov2=1) then   
begin   
if P1=0 then   
begin   
clrscr;   
Writeln('The square error is equal to zero');   
end   
else   
begin   
h:= (PGran - LGran) / n;   
h2:=Round(h);   
Textcolor (15);   
clrscr;   
f:=0;   
sumIN:=0;   
for ip:= 1 to h2-1 do   
begin   
x:=LGran+ip\*n;   
sr1:=Funct1(x);   
sumIN:=sumIN+sr1+n/2;   
end;   
sumIN:=sumIN\*n;   
writeln ('Area is ', sumIN: 0:6);   
begin   
pog:=Funct1(PGran)-Funct1(LGran);   
P1:=1;   
end;   
Writeln ('Absolute error= ', Abs(pog-sumIN): 5:6);   
Writeln ('Relative error= ', ((Abs(pog-sumIN))/sumIN)\*100: 3:6,'%');   
writeln ('Press enter to exit the menu');   
end   
end   
else   
begin   
clrscr;   
Writeln('You have not entered [data.press](https://vk.com/away.php?to=http%3A%2F%2Fdata.press&cc_key=" \t "_blank) enter.');   
end;   
readln;   
end;   
  
  
begin   
Prov1:=0;   
P1:=1;   
Prov2:=0;   
repeat   
Clrscr;   
textbackground(0);   
Textcolor (15);   
Gotoxy (50, 1);   
Writeln ('Laboratory work number 3');   
Textcolor (15);   
Gotoxy (44, 2);   
Writeln ('F= 2\*x^3+2\*x (X=-0.75 Y=0)');   
Gotoxy (44, 2);   
Textcolor (15);   
for i:= 1 to MN do   
begin   
GotoXY (1, 4 + i);   
Write (Menu[i]);   
end;   
i:= 1;   
TextColor (10);   
Gotoxy (1, 4 + i);   
Write (Menu[i]);   
repeat   
ReadKl:=readkey;   
if ReadKl = #0 then   
ReadKl:=readkey;   
case ReadKl of   
#72:   
begin   
TextColor (15);   
Gotoxy (1, 4 + i);   
Write (Menu[i]);   
i:= i - 1;   
if i <1 then i:=4;   
Textcolor (10);   
Gotoxy (1, 4 + i);   
Write (Menu[i]);   
end;   
#80:   
begin   
Textcolor (15);   
Gotoxy (1, 4 + i);   
Write (Menu[i]);   
i:= i + 1;   
if i >MN then i:= 1;   
Textcolor (10);   
gotoxy (1, 4 + i);   
Write (Menu[i]);   
end;   
end;   
until ReadKl =#13;   
case i of   
1:   
begin   
Prov1:=1;   
ProcVod;   
end;   
2:   
begin   
ProcSteps;   
Prov2:=1;   
end;   
3: ProcS;   
4:   
begin   
clrscr;   
Textcolor (15);   
writeln ('End of the program...');   
i:= 0;   
end;   
end;   
until i = 0;   
end.

Фото отчет выполненой программы:







Вывод: В ходе выполнение лабораторной работы по программированию были получены:

* Навыки работы с вычисление площади с помощью метода левых прямоугольников
* Навыки работы с case-меню и посредством него взаимодействие с пользователем
* Навык проверки корректности данных вводимых пользователем.
* Навык оценки погрешности как абсолютной, так и относительной.

При выполнение лабораторной работы были выявлены сложности такие как:

* Выбор между процедурой и функцией. Выбор был сделан на функцию, так как она возвращает численное значение из подпрограммы в программу.
* Работа с структурой case-меню.
* Решение вывода площади в отрицательной части функции.