Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ОТЧЕТ**

**ПО ДОМАШНЕЙ КОНТРОЛЬНОЙ РАБОТЕ №3**

**«ИЗУЧЕНИЕ БАЗОВЫХ ПРИНЦИПОВ ОРГАНИЗАЦИИ ПРОЦЕДУР И ФУНКЦИЙ»**

**ПО ДИСЦИПЛИНЕ «ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»**

Выполнила: студентка учебной группы

ИСПк-202-52-00

Щемелева Мария Андреевна

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2022

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

2. Формулировка задания

Вариант 10

1. Реализовать программу вычисления площади фигуры, ограниченной кривой 1\*x^3+1\*x^2+17 и осью OX (в положительной части по оси OY).
2. Вычисление определенного интеграла должно выполняться численно, с применением метода трапеций.
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.

3. Схема алгоритма с комментариями

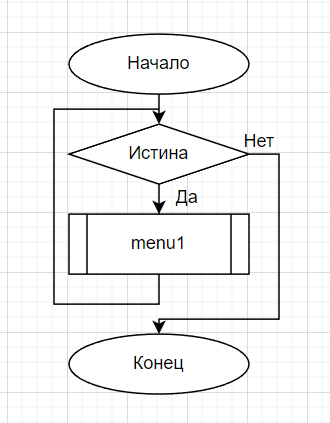


Рисунок 1 - Алгоритм решения программы

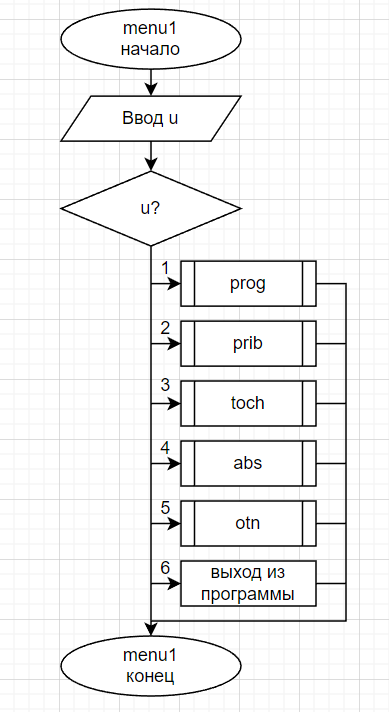


Рисунок 2 - Алгоритм решения подпрограммы menu1

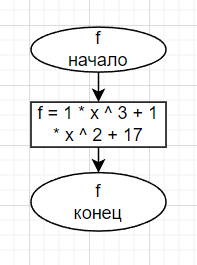


Рисунок 3 - Алгоритм решения функции f

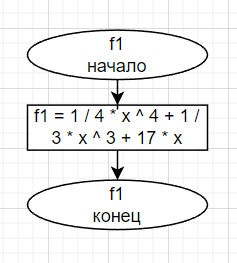
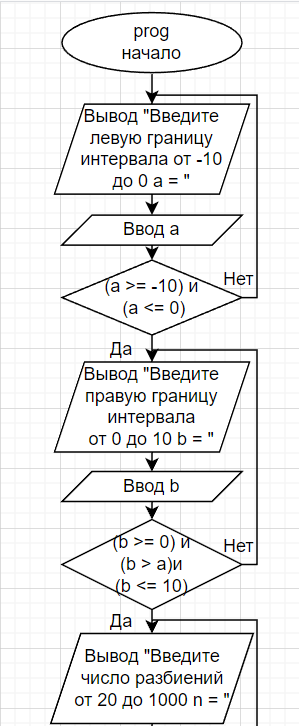


Рисунок 4 - Алгоритм решения функции f1



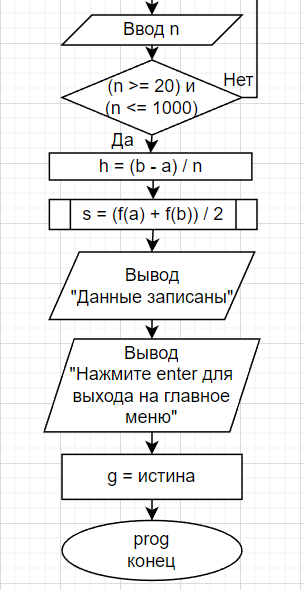
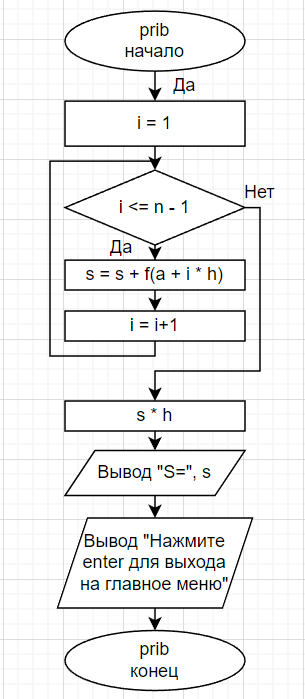


Рисунок 5 - Алгоритм решения подпрограммы prog

  
Рисунок 6 - Алгоритм решения подпрограммы prib добавить

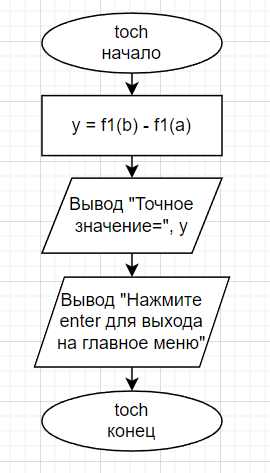


Рисунок 7 – Алгоритм решения подпрограммы toch

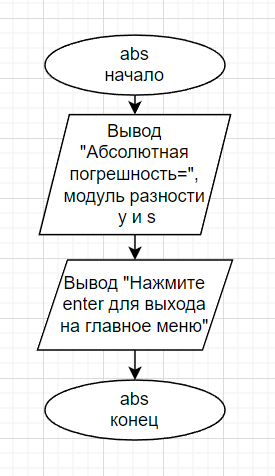


Рисунок 8 – Алгоритм решения подпрограммы abs

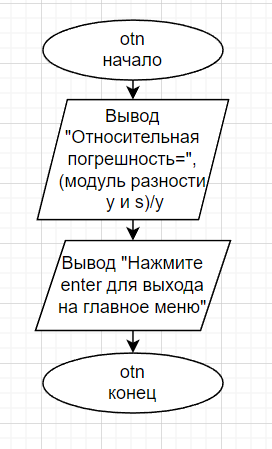


Рисунок 9 – Алгоритм решения подпрограммы otn

4. Код программы

**uses** crt;

**var**

a, b, h, s, y: real;

n: integer;

u: char;

g: boolean;

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

**begin**

f := 1 \* power(x, 3) + 1 \* power(x, 2) + 17;

**end**;

**function** f1(x: real): real;

**begin**

f1 := 1 / 4 \* power(x, 4) + 1 / 3 \* power(x, 3) + 17 \* x;

**end**;

**procedure** prog;

**begin**

clrscr;

**repeat**

write('Введите левую границу интервала от -10 до 0 a = ');

readln(a);

**until** (a >= -10) **and** (a <= 0);

**repeat**

write('Введите правую границу интервала от 0 до 10 b = ');

readln(b);

**until** (b >= 0) **and** (b > a) **and** (b <= 10);

**repeat**

write('Введите число разбиений от 20 до 1000 n = ');

readln(n);

**until** (n >= 20) **and** (n <= 1000);

h := (b - a) / n;

s := (f(a) + f(b)) / 2;

writeln ('Данные записаны)');

writeln ('Нажмите enter для выхода на главное меню');

g:= true;

readkey;

**end**;

**procedure** prib;

**begin**

clrscr;

**if** g **then begin**

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

s := s + f(a + i \* h);

s := s \* h;

writeln('S=', s:0:3);

**end else** writeln ('Введите сначала данные');;

writeln ('Нажмите enter для выхода на главное меню');

readkey;

**end**;

**procedure** toch;

**begin**

clrscr;

**if** g **then begin**

y := f1(b) - f1(a);

writeln('Точное значение=', y:0:3);

**end else** writeln ('Введите сначала данные');

writeln ('Нажмите enter для выхода на главное меню');

readkey;

**end**;

**procedure** abs;

**begin**

clrscr;

**if** g **then begin**

y := f1(b) - f1(a);

writeln('Абсолютная погрешность=', abs(y - s):0:3);

**end else** writeln ('Введите сначала данные');

writeln ('Нажмите enter для выхода на главное меню');

readkey;

**end**;

**procedure** otn;

**begin**

clrscr;

**if** g **then begin**

y := f1(b) - f1(a);

writeln('Относительная погрешность=', abs(y - s) / y:0:5);

**end else** writeln ('Введите сначала данные');

writeln ('Нажмите enter для выхода на главное меню');

readkey;

**end**;

**procedure** menu1;

**begin**

clrscr;

println('Чтобы ввести данные нажмите 1');

println('Чтобы выбрать приближенное значение введите 2');

println('Чтобы выбрать точное значение введите 3');

println('Чтобы выбрать абсолютную погрешность введите 4');

println('Чтобы выбрать относительную программу введите 5');

println ('Чобы ввести данные заново нажмите 6');

print('Выберите программу: ');

read(u);

**case** u **of**

'1': prog;

'2': prib;

'3': toch;

'4': abs;

'5': otn;

'6': **exit**;

**end**;

**end**;

**begin**

**while** true **do**

menu1();

**end**.

5. Результат выполнения программы

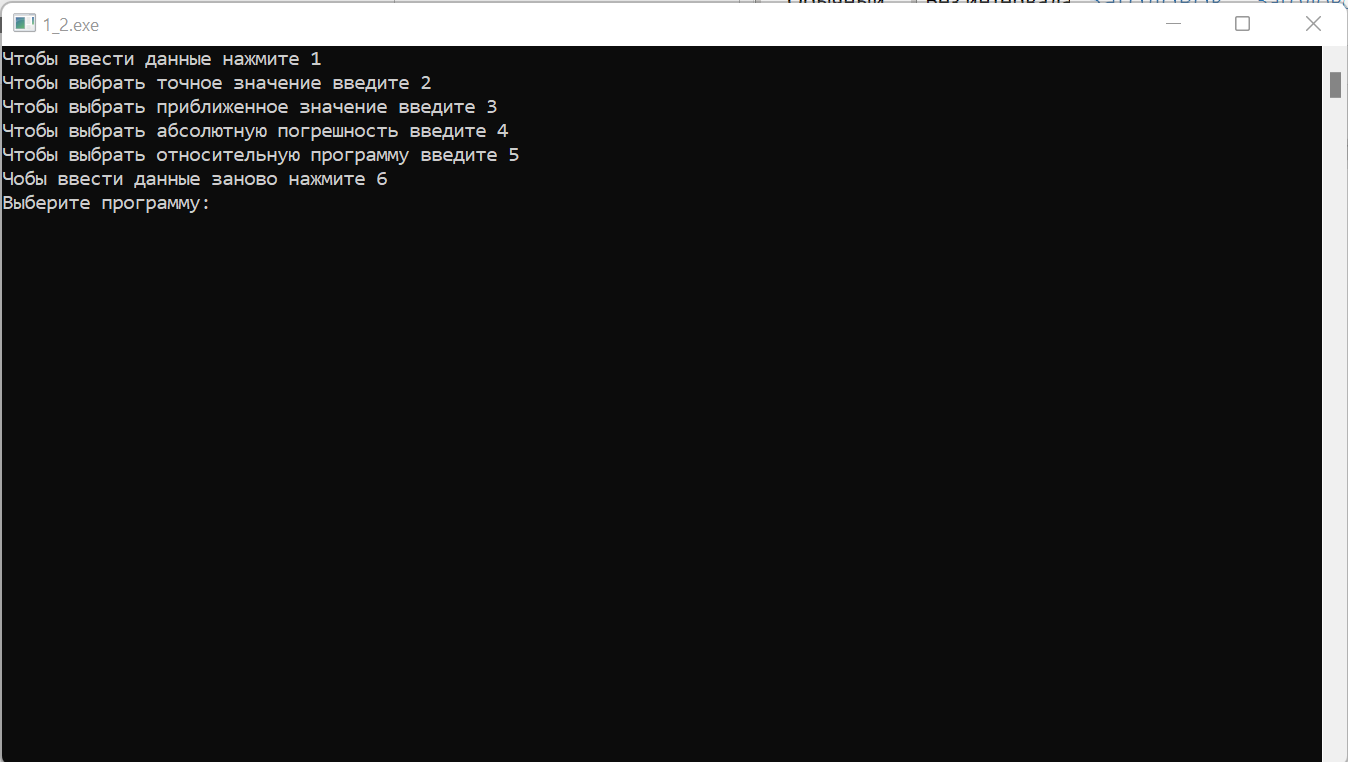


Рисунок 10 – Case-menu

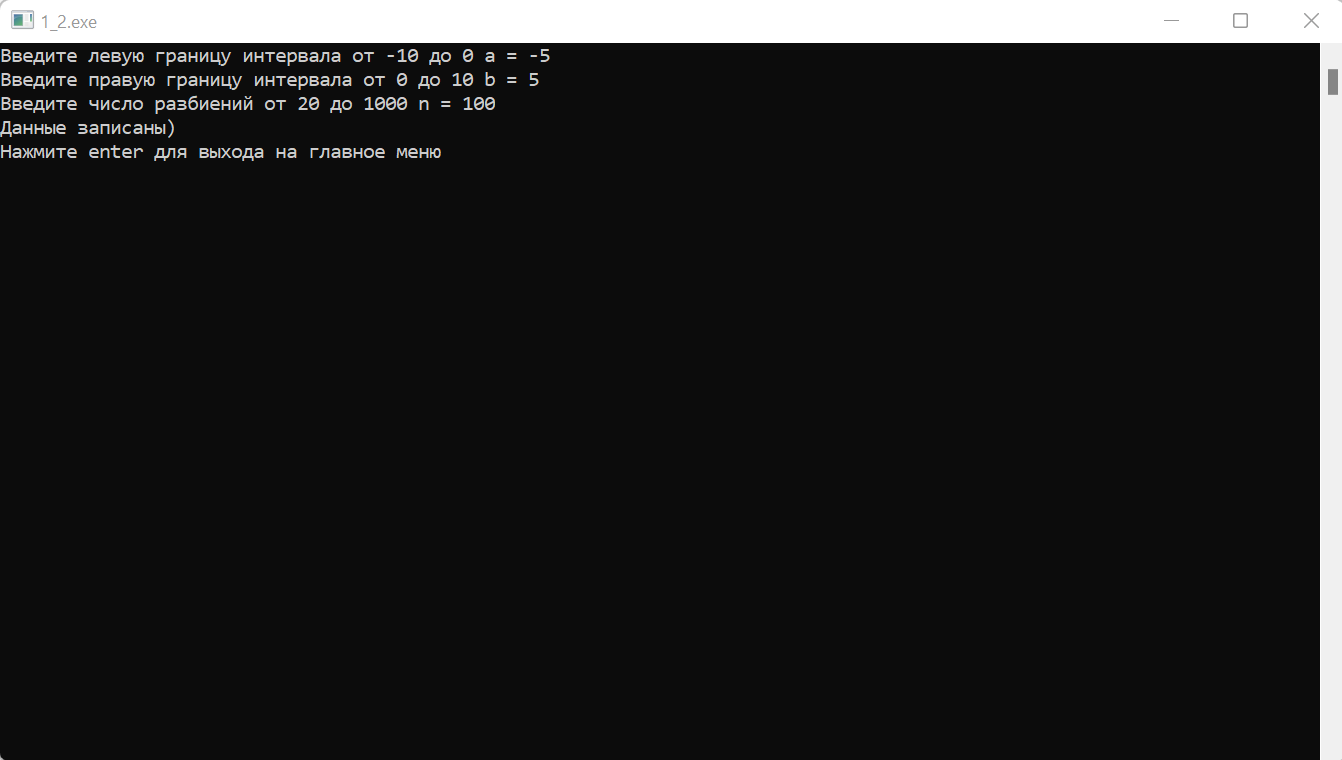


Рисунок 11 – Окно вывода программы

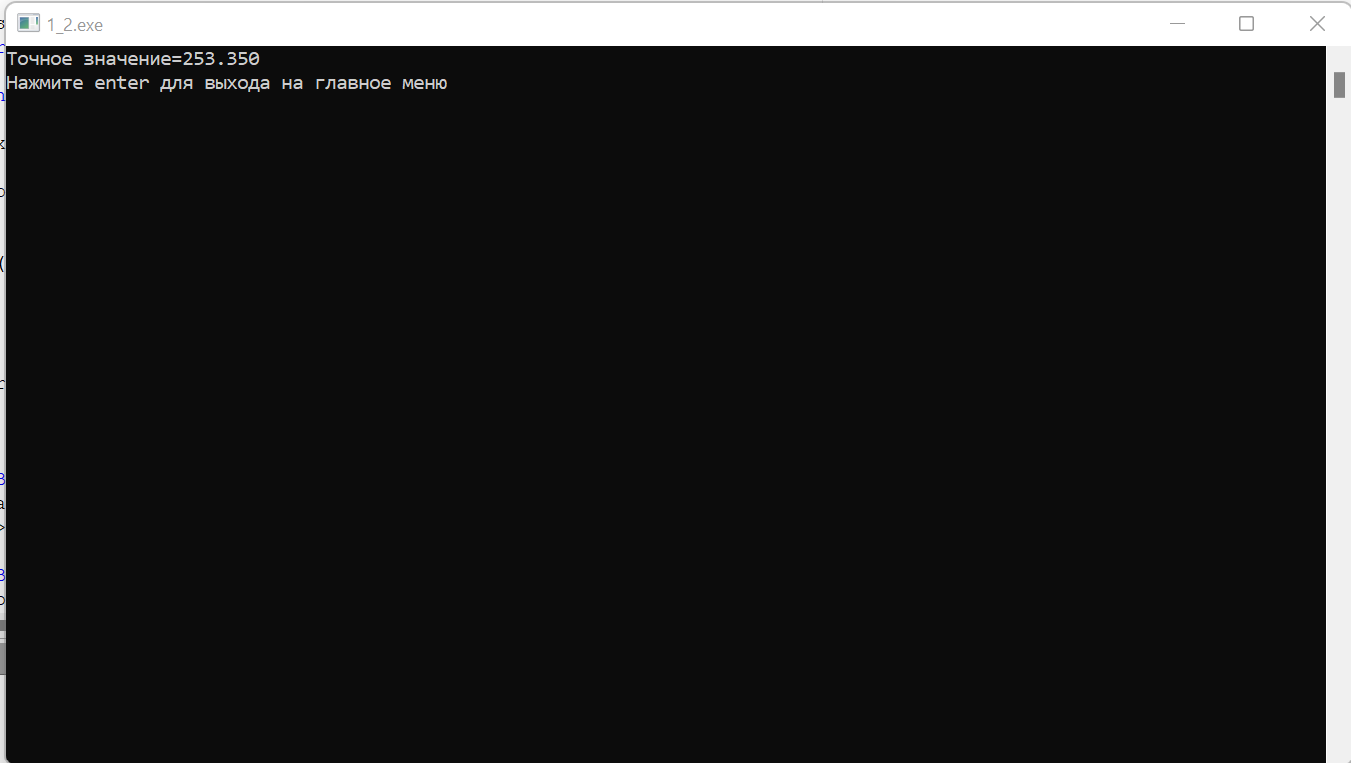


Рисунок 12 – Окно вывода программы

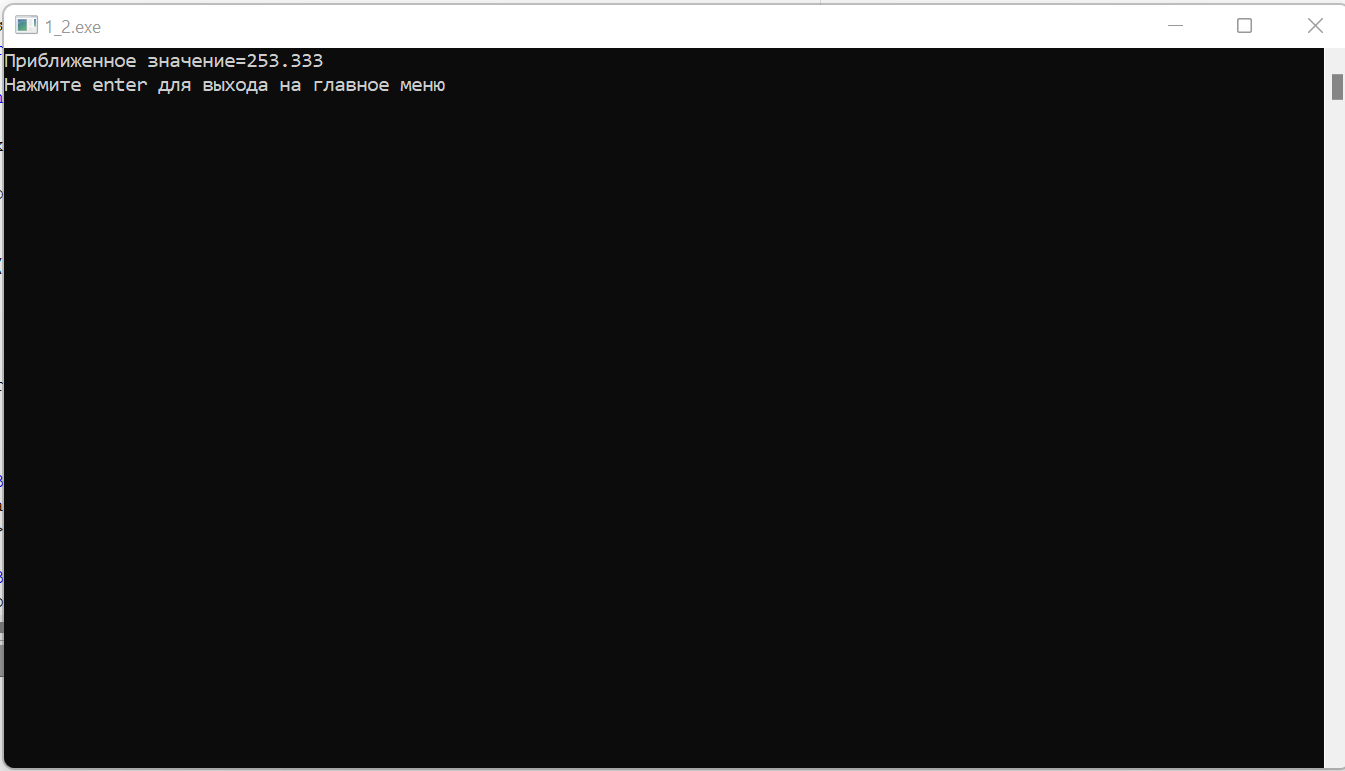


Рисунок 13 – Окно вывода программы

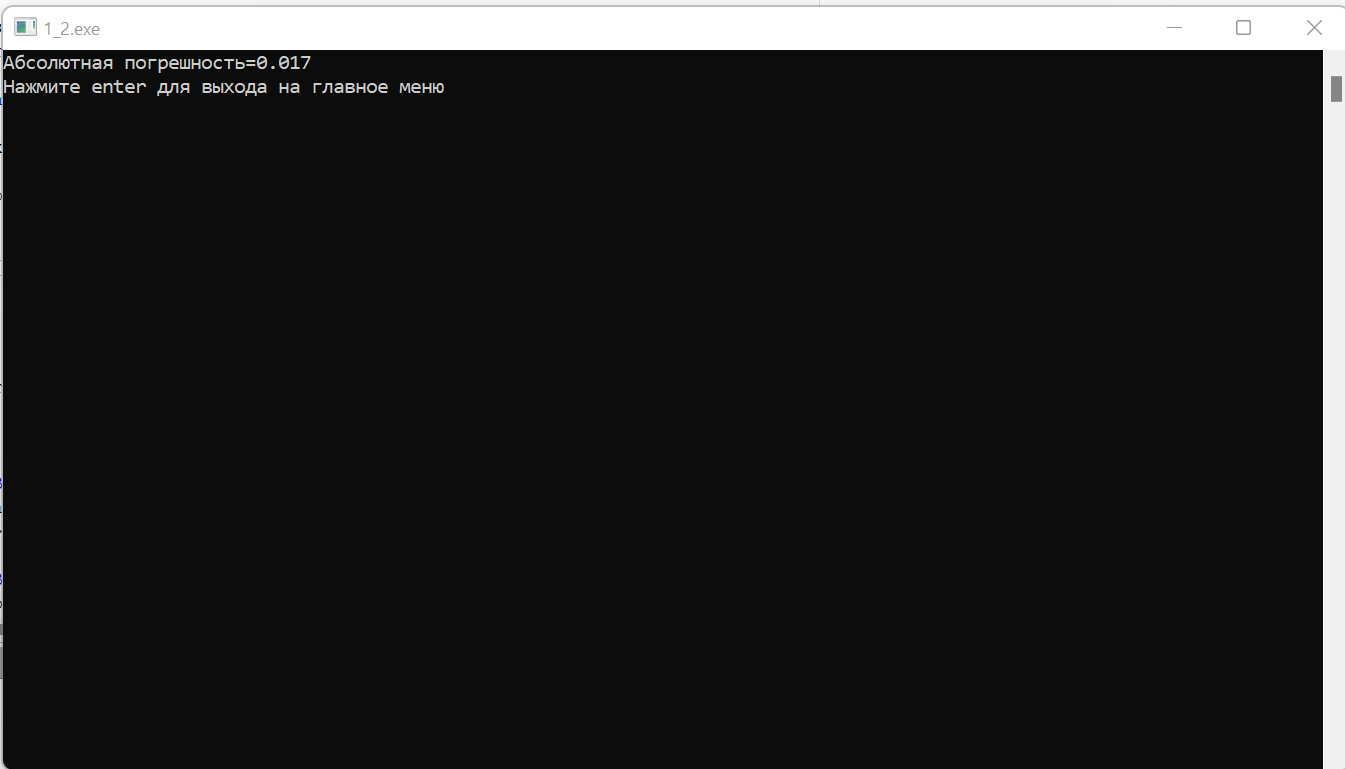


Рисунок 14 – Окно вывода программы

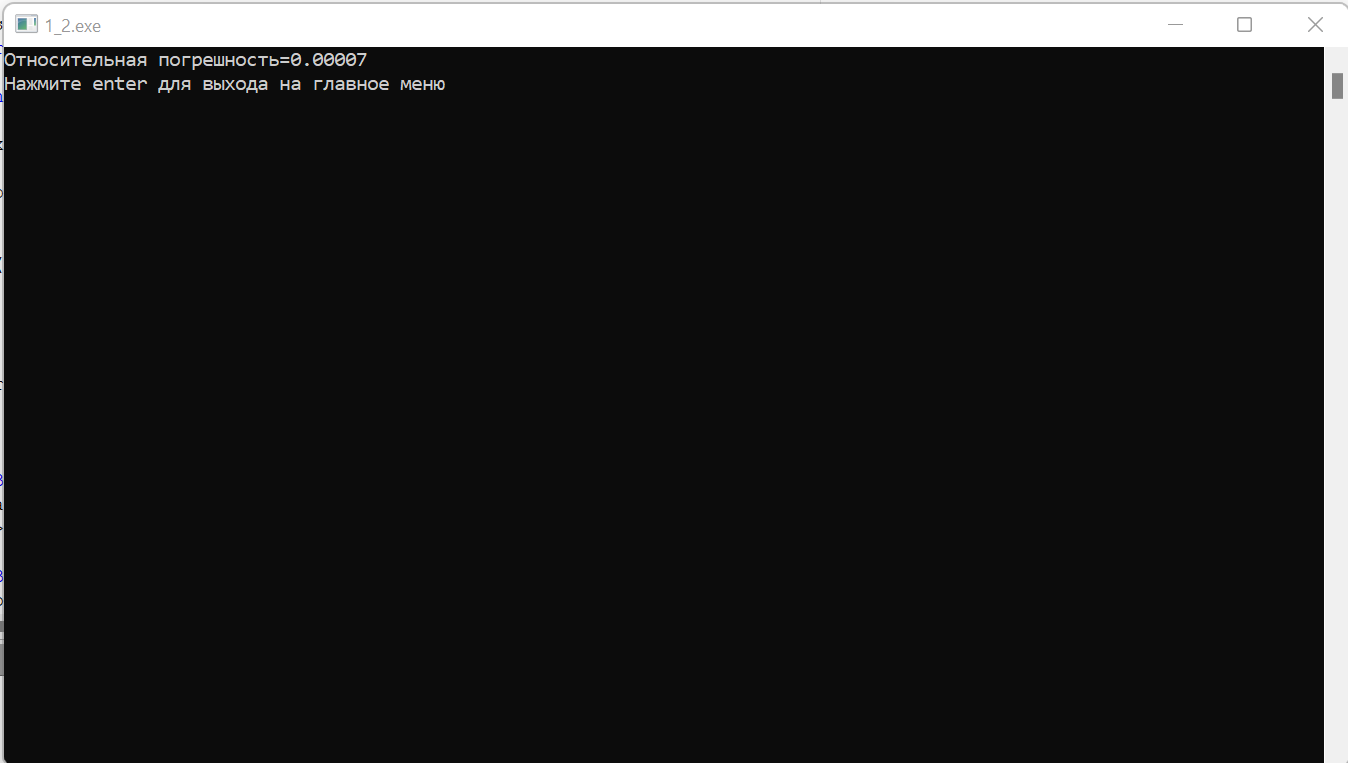


Рисунок 15 – Окно вывода программы

6. Вывод

В процессе решения задачи прошло ознакомление c CASE-меню, readkey, case of, procedure, function. Задание было выполнено с затруднениями, так как нужно было реализовать возможность оценки погрешности полученного результата. Вторая трудность заключалась в том, что было необходимо использовать процедуры и функции. Так же было затруднительно написать case-меню. Все задачи и цели были выполнены в выделенные рамки 2 недели.

На одном из этапов создании кода, в процессе выполнения программы возникла проблема: в создании case-меню и в применении procedure. Следующим действием было написание схемы алгоритма, пришлось закрепить знания по программе draw.io, Программа была изучена, поэтому не возникло проблем с draw.io.

Знания, умения, навыки, полученные за период практики, явились отличным стимулом для активной работы в освоении будущей специальности.

Учебная практика помогла лучше закрепить case of, procedure, CASE- меню в Pascal и была достигнута цель: применение полученных знаний и закрепление новых навыков. В ходе работы было все выполнено и сделано вовремя.