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»**

Выполнил: студент учебной группы

ИСПк-203-52-00

Криницын Олег Сергеевич

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

1. **Формулировка задания:**

Вариант 7.

1. Реализовать программу вычисления площади фигуры, ограниченной кривой 1\*x^3+(-1)\*x^2+(-4)\*x+(17) и осью ОХ (в положительной части оси OY).
2. Вычисление определенного интеграла должно выполняться численно, с применением методом трапеций.
3. Пределы интегрирования вводятся пользователям.
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.
7. **Схема алгоритма с комментариями:**

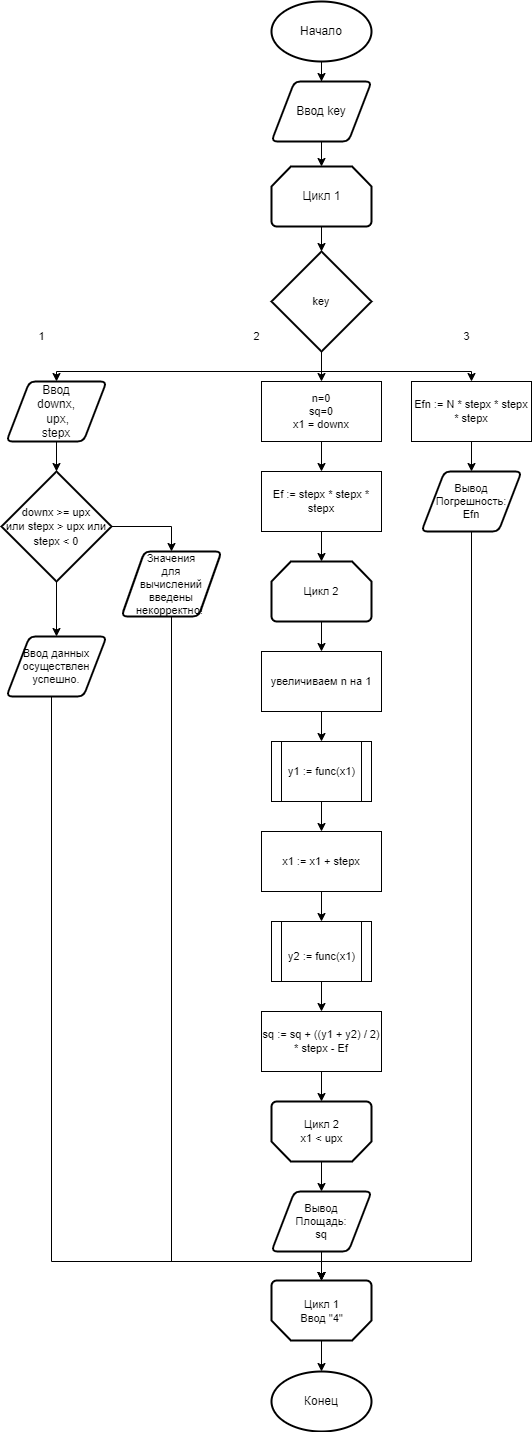
****

Рисунок 1 - Схема алгоритма программы

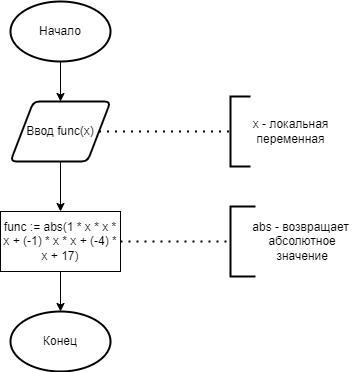


Рисунок 2 - Схема алгоритма функции

1. **Код программы**

**uses**

crt;

**procedure** menu;

**begin**

TextColor(Green);

writeln('Нажмите "1" чтобы осуществить ввод данных для вычислений.');

TextColor(Red);

writeln('Нажмите "2" чтобы осуществить вычисление площади фигуры.');

TextColor(Yellow);

writeln('Нажмите "3" чтобы вычислить погрешности.');

TextColor(Blue);

writeln('Нажмите "4" чтобы выйти из программы.');

**end**;

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

**begin**

func := abs(1 \* x \* x \* x + (-1) \* x \* x + (-4) \* x + 17);

**end**;

**var**

downx: real;

upx: real;

stepx: real;

y1, y2: real;

sq: real;

x1: real;

key: char;

Efn: real;

Ef: real;

N: integer;

**begin**

menu;

**repeat**

key := ReadKey;

**case** key **of**

'1':

**begin**

TextColor(Green);

Writeln('Введите нижнюю границу интегрирования:');

Readln(downx); ClrScr;

Writeln('Введите верхнюю границу интегрирования:');

Readln(upx); ClrScr;

Writeln('Введите шаг интегрирования:');

Readln(stepx); ClrScr;

ClrScr;

**if** (downx >= upx) **or** (stepx > upx) **or** (stepx < 0) **then**

**begin**

Writeln('Значения для вычислений введены некорректно! Выберете нужный пункт меню');

**end**

**else**

TextColor(Magenta);

Writeln('Ввод данных осуществлен успешно.');

menu;

**end**;

'2':

**begin**

N := 0;

sq := 0;

x1 := downx;

Ef := stepx \* stepx \* stepx;

**repeat**

inc(N);

y1 := func(x1);

x1 := x1 + stepx;

y2 := func(x1);

sq := sq + ((y1 + y2) / 2) \* stepx - Ef;

**until** x1 > upx;

TextColor(Cyan);

ClrScr;

Writeln('Площадь вычисляемой фигуры равна: ', sq:0:4);

menu;

**end**;

'3':

**begin**

Efn := N \* stepx \* stepx \* stepx;

TextColor(Brown);

ClrScr;

Writeln('Погрешность: ', Efn:0:4);

menu;

**end**;

**end**;

**until** key = '4'

**end**.

1. **Результат выполнения программы**

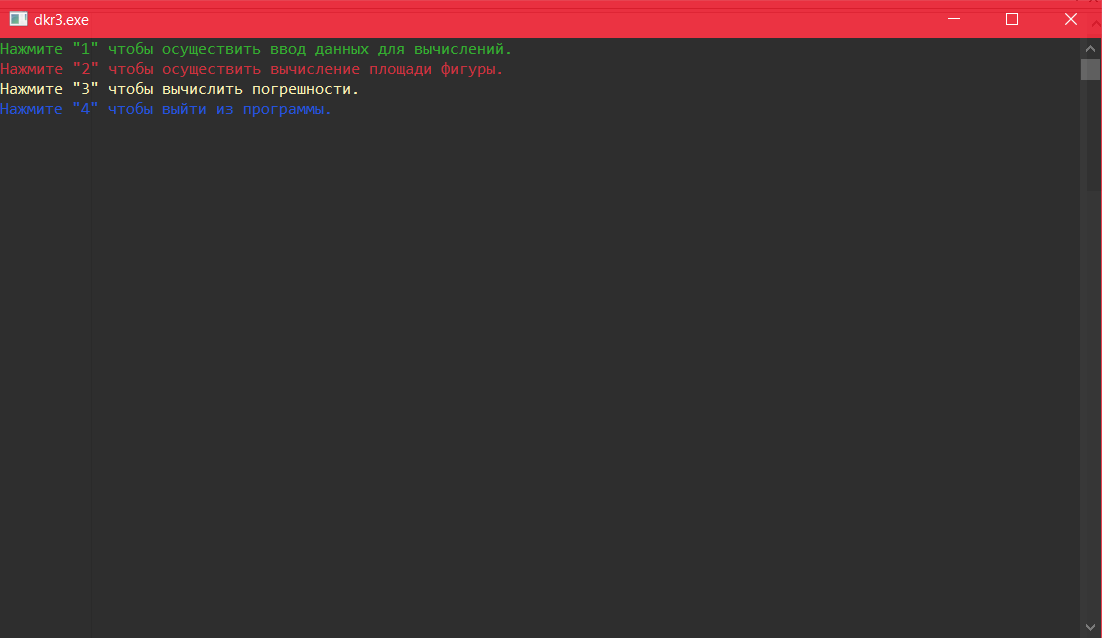
****

Рисунок 3 - Результат выполнения программы

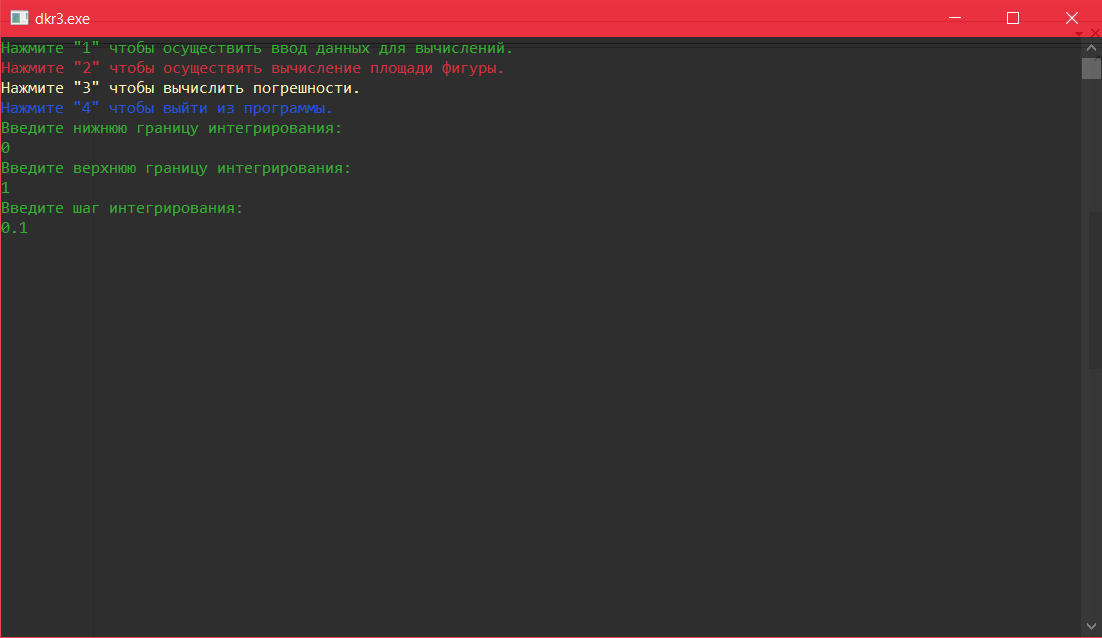


Рисунок 4 - Результат выполнения программы

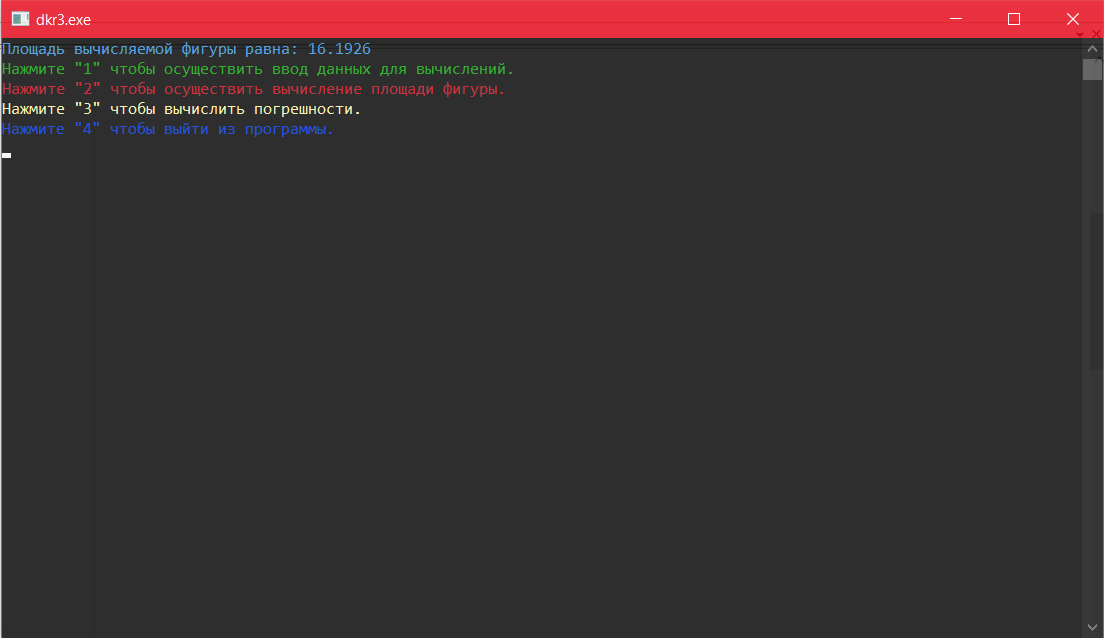


Рисунок 5 - Результат выполнения программы

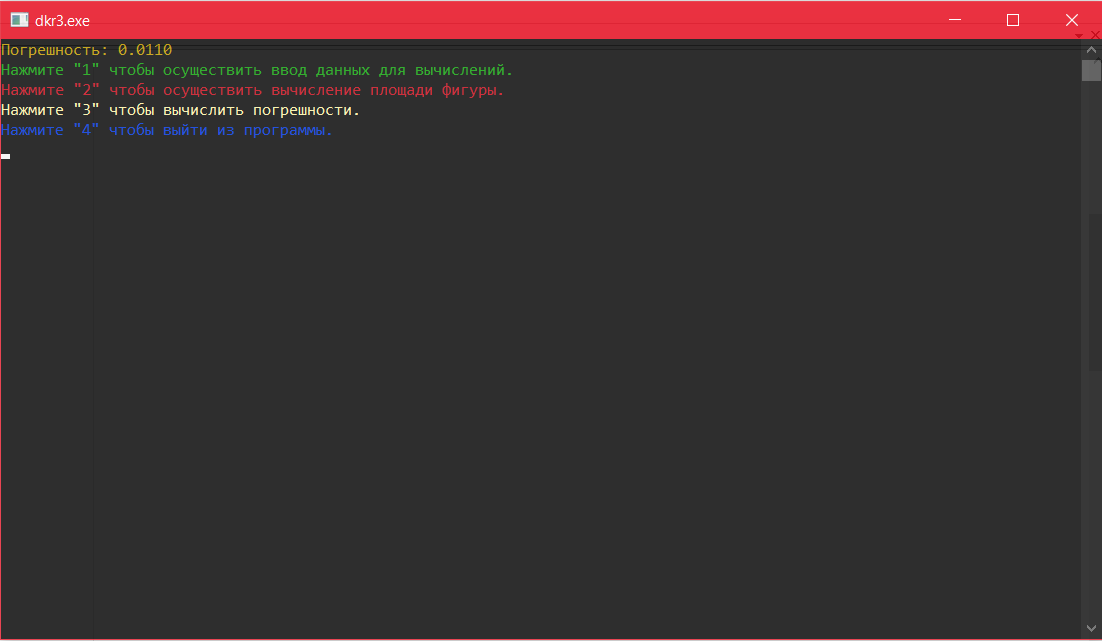


Рисунок 6 - Результат выполнения программы

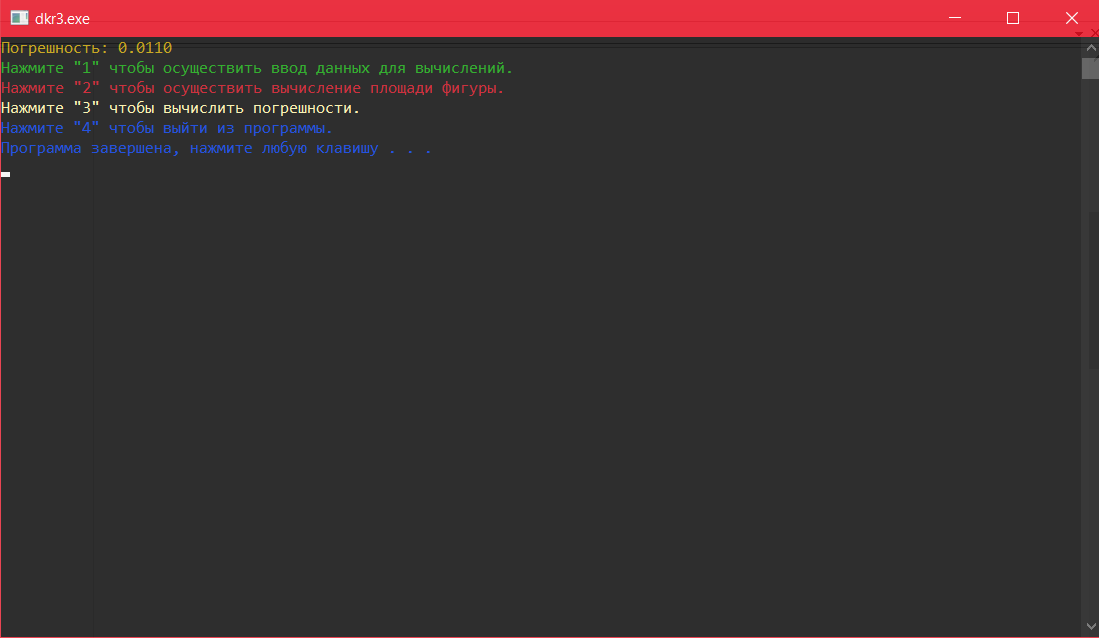


Рисунок 7 - Результат выполнения программы

1. **Вывод**

Нами была поставлена цель работы: освоить синтаксис построения процедур и функций, изучить способы передачи данных в подпрограммы, получить навыки организации минимального пользовательского интерфейса. Работа была разделена на несколько этапов: цель, формулировка задания, создание схемы алгоритма, написание кода программы, результат выполнения программы, вывод. Схема алгоритма была создана с помощью сайта  [(https://app.diagrams.net/?src=about)](https://products.aspose.app/diagram/ru/flowchart). В схеме задания используются два цикла и две функции(ReadKey и func(x1)). Function ReadKey - это функция, которая реагирует на нажатие клавиши и сообщает о том, какая именно клавиша была нажата. func(x1) - в этой функции мы присваиваем func (1 \* x^3 + (-1) \* x^2 + (-4) \* x + 17). Код программы был написан на PascalABC. Итак, делая вывод, хочу сказать, что в ходе работы все цели были достигнуты.