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: студент учебной группы

ИСПк-202-52-00

Долгополов Ярослав Максимович

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2024

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

1. **Формулировка задания. Вариант 8**

**Постановка задачи**

1. Реализовать программу вычисления площади фигуры, ограниченной кривой 2x3+ (-1)\*x2 + (-4)\*x + 5 и осью OX (в положительной части по оси OY).
2. Вычисление определенного интеграла должно выполняться численно, с применением 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.
7. **Описание алгоритма** 
   1. Вводим цикл 1. Тело цикла:
      1. Вводим число от 0 до 3.
      2. Если была введена 1, то осуществляем ввод данных (верхней границы интегрирования, нижней границы интегрирования и количество равных частей интервала интегрирования).
      3. Если была введена 2 и до этого были введены вводные данные (см. п. 2), то выводим результат вычисления определённого интеграла методом средних треугольников. Если до этого вводные данные не были введены, то выводим «Вводных данных нет».
      4. Если была введена 3 и до этого были введены вводные данные (см. п. 2), то выводим относительную погрешность вычисления определённого интеграла методом средних треугольников. Если до этого вводные данные не были введены, то выводим «Вводных данных нет».
      5. Ввод чисел осуществляется до тех пор, пока не будет введена 0.
8. **Схема алгоритма с комментариями**

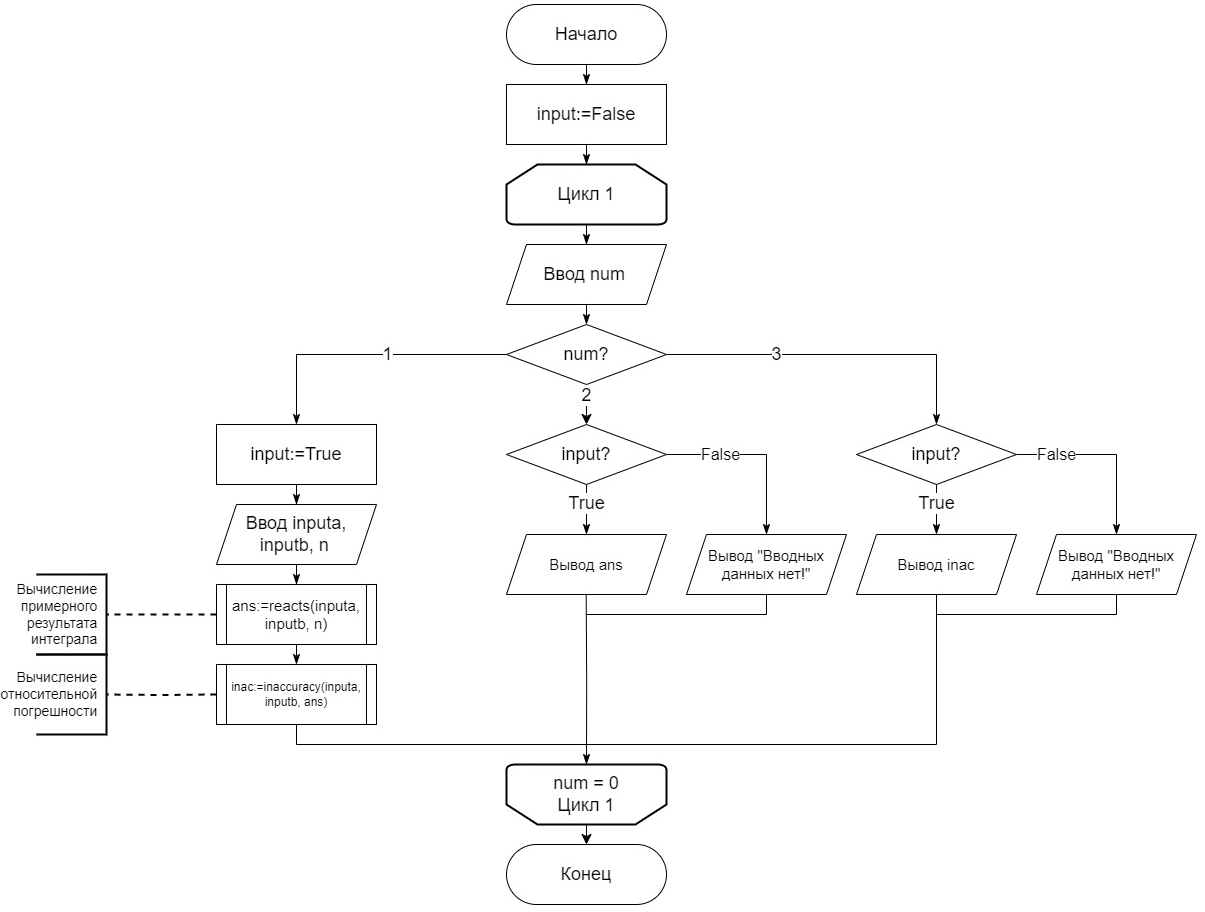
****

Рисунок 1 - Алгоритм решения задачи

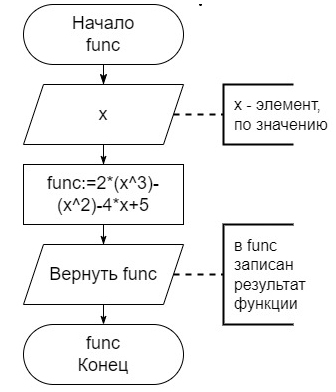
****

Рисунок 2 - Алгоритм выполнения функции func

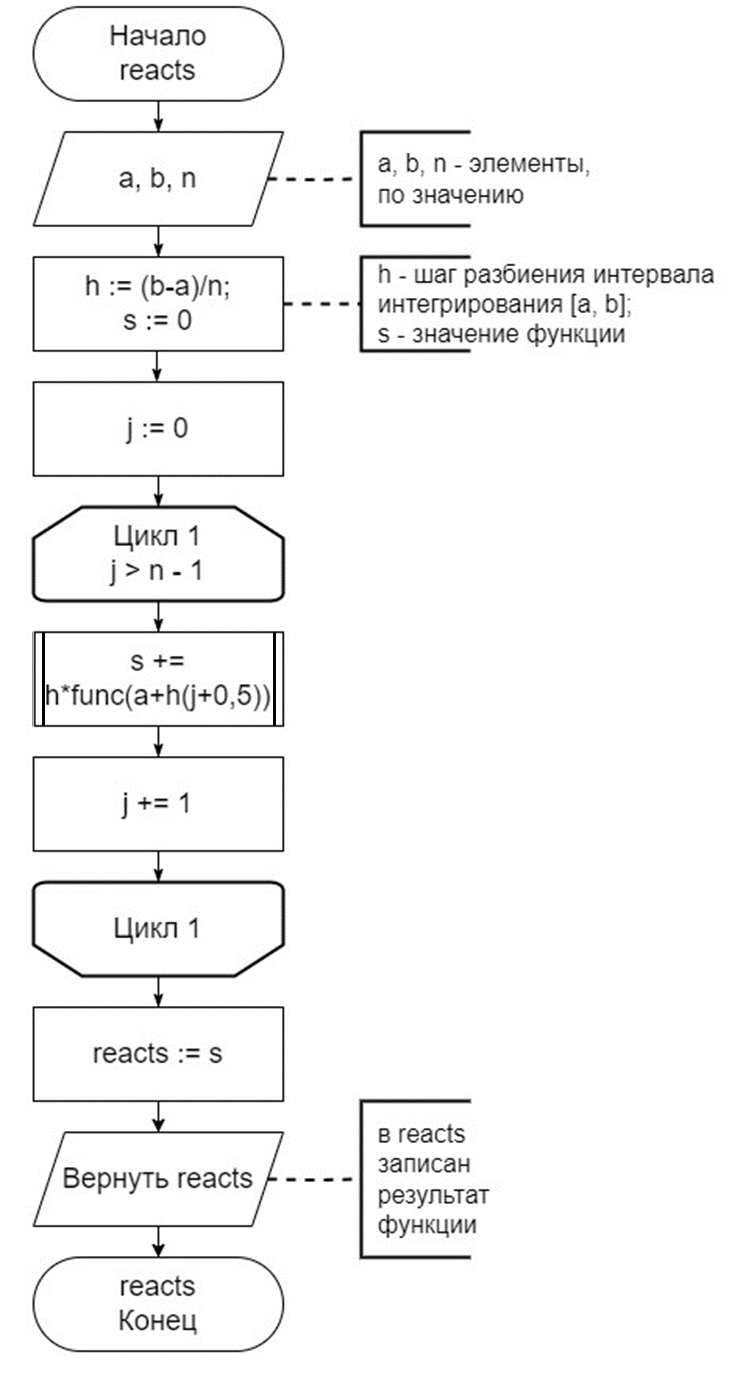


Рисунок 3 - Алгоритм выполнения функции reacts

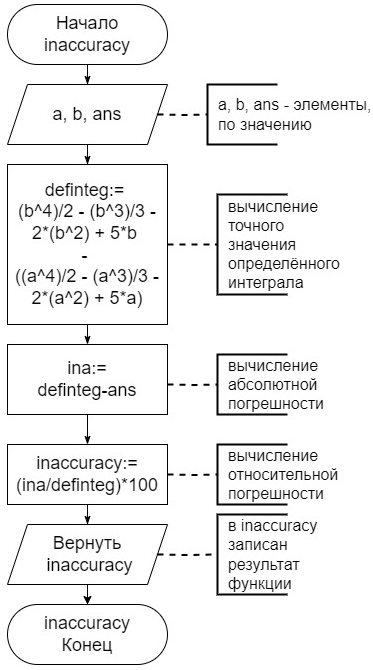
****

Рисунок 4 - Алгоритм выполнения функции inaccuracy

1. **Код программы**

**Uses** Crt;

**var**

inputa, inputb, ans, inac: real;

n, num: integer;

input: boolean;

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

**begin**

func:=2\*x\*sqr(x)-sqr(x)-4\*x+5;

**end**;

**Function** reacts(a, b: real; n: integer): real;

**var**

h, s, j: real;

**begin**

h := (b - a)/n;

j:=0.0;

**while** j <= (n-1) **do**

**begin**

s+=h\*func(a+h\*(j+0.5));

j+=1

**end**;

reacts:=s

**end**;

**Function** inaccuracy(a, b, ans: real): real;

**var**

definteg, ina: real;

**begin**

definteg:=(sqr(sqr(b))/2) - ((b\*sqr(b))/3) - 2\*sqr(b) + 5\*b - ((sqr(sqr(a))/2) - ((a\*sqr(a))/3) - 2\*sqr(a) + 5\*a);

ina:=definteg-ans;

inaccuracy:=(ina/definteg)\*100

**end**;

**begin**

writeln('Вычисление площади фигуры ограниченной кривой y = 2x^3 - x^2 - 4x + 5 и осью Ox.');

input:=False;

**repeat**

**begin**

writeln();

writeln('Возможные действия:');

writeln('1 - ввод');

writeln('2 - ответ');

writeln('3 - погрешность');

writeln('0 - выход');

write('Выберите действие: ');

readln(num);

**case** num **of**

1:

**begin**

input:=true;

write('Введите верхнюю границу интегрирования: ');

readln(inputb);

write('Введите нижнюю границу интегрирования: ');

readln(inputa);

write('Количество равных частей: ');

readln(n);

ans:=reacts(inputa, inputb, n);

inac:=inaccuracy(inputa, inputb, ans)

**end**;

2: **if** input **then** writeln('Ответ: ', ans) **else** writeln('Вводных данных нет!');

3: **if** input **then** writeln('Погрешность: ', inac) **else** writeln('Вводных данных нет!');

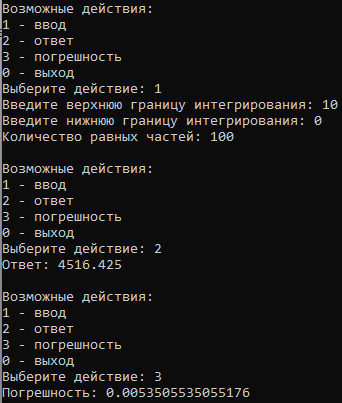
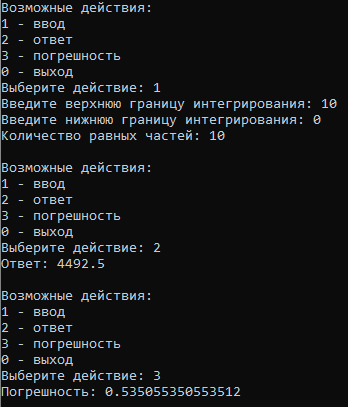
**end**

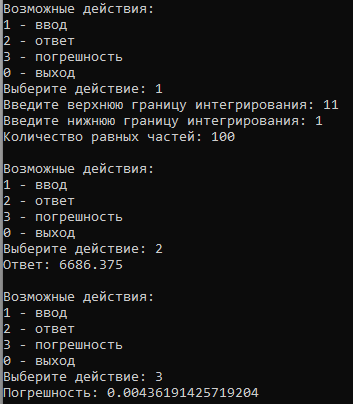
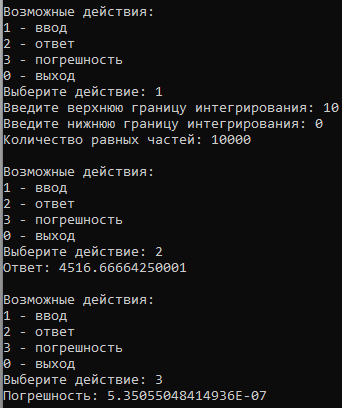
**end**

**until** num=0

**end**.

1. **Результат выполнения программы**

**** ****

**** ****

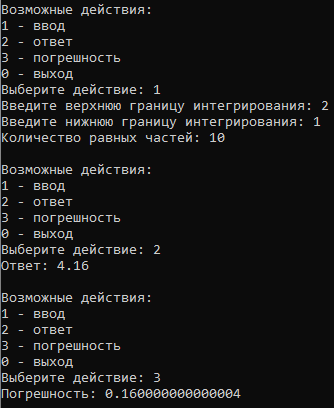
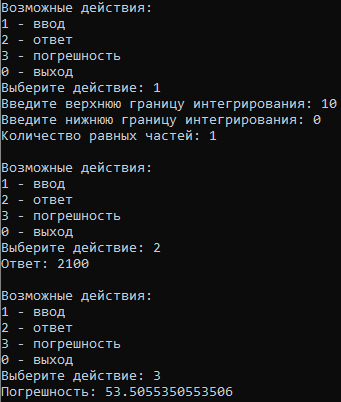
**** ****

Рисунок 5 - Результаты выполнения программы**Вывод**

В процессе работы над домашней контрольной мы изучили принципы построения процедур и функций и получили навыки организации минимального пользовательского интерфейса.

Для создания алгоритмических схем был использован draw.io — бесплатный онлайн-инструмент для создания и совместного редактирования диаграмм и схем.

В ходе работы мы столкнулись с некоторыми трудностями. Например, были проблемы с нахождением способа оценки погрешности. Было принято решение найти абсолютную погрешность вычитанием из точного результата, вычисленным по формуле Ньютона-Лейбница, приблизительного результата, полученного в ходе выполнения программы. Затем полученную абсолютную погрешность разделить на точный результат, чтобы получить относительную погрешность.

В итоге, благодаря полученным знаниям и устранённым ошибкам, мы смогли успешно решить задачу, применив код с заданием функции и введением пользовательского интерфейса, и расширили наши знания о языке программирования Pascal.