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 программированния»**

Выполнил: студент учебной группы

ИСПк-204-52-00

Черных Иван Антонович

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2024

Цель работы

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

2.Формулировка задания (с вариантом)

Вариант 14

1. Реализовать программу вычисления площади фигуры, ограниченной кривой  
2 \* x ^ 3 + (1) \* x ^ 2 + (-2) \* x + (16) и осью OX (в положительной части по оси OY).  
2. Вычисление определенного интеграла должно выполняться численно, с применением метода левых 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.

3.Описание алгоритма

1. Ввод данных: Программа должна запрашивать у пользователя:

* Нижний предел интегрирования a;
* Верхний предел интегрирования b;
* Количество разбиений n, которое определяет точность вычислений.

2. Численное вычисление интеграла методом левых прямоугольников: Метод левых прямоугольников для интегрирования вычисляет площадь полосы по высоте функции в начале интервала.

3. Оценка погрешности:

Для оценки погрешности можно использовать оценку погрешности численного метода для интегралов, основанную на второй производной функции.

4. Структура программы:

Программа должна использовать меню, через которое пользователь может:

* Ввести пределы интегрирования и количество разбиений.
* Вычислить интеграл методом левых прямоугольников.
* Оценить погрешность полученного результата.
* Выйти из программы.

5. Структура программы:

Функция для вычисления интеграла методом левых прямоугольников:

* Эта функция принимает на вход пределы a, b и количество разбиений n, а также вычисляет численное значение интеграла.

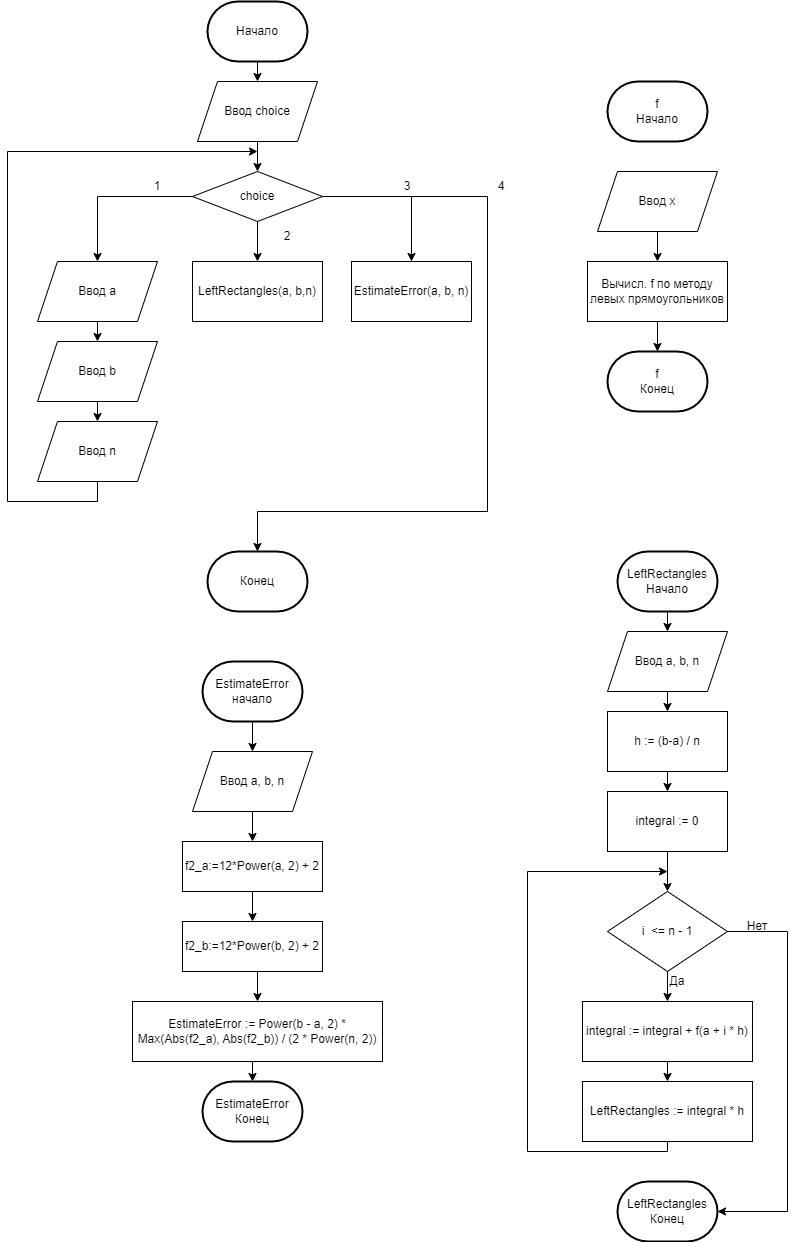
Функция для оценки погрешности:

* Эта функция оценивает погрешность по формуле, основанной на второй производной функции.

Основная программа:

* Основной цикл программы предлагает пользователю ввести данные, вычислить интеграл или погрешность, а затем выводит результаты.

Схема



Код для программы в Pascal.

**Задание 1**

**program** dkr3;

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

**begin**

f := 2 \* Power(x, 3) + Power(x, 2) - 2 \* x + 16;

**end**;

**function** LeftRectangles(a, b: real; n: integer): real;

**var** h, integral: real; i: integer;

**begin**

h := (b - a) / n;

integral := 0;

**for** i := 0 **to** n - 1 **do**

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

LeftRectangles := integral \* h;

**end**;

**function** EstimateError(a, b: real; n: integer): real;

**var** f2\_a, f2\_b: real;

**begin**

f2\_a := 12 \* Power(a, 2) + 2;

f2\_b := 12 \* Power(b, 2) + 2;

EstimateError := Power(b - a, 2) \* Max(Abs(f2\_a), Abs(f2\_b)) / (2 \* Power(n, 2));

**end**;

**var** a, b: real; n, choice: integer;

**begin**

**repeat**

writeln('1. Ввести пределы интегрирования');

writeln('2. Вычислить интеграл методом левых прямоугольников');

writeln('3. Оценить погрешность');

writeln('4. Выход');

write('Выбор: '); readln(choice);

**case** choice **of**

1: **begin**

write('Введите нижний предел (a): '); readln(a);

write('Введите верхний предел (b): '); readln(b);

write('Введите количество разбиений (n): '); readln(n);

**end**;

2: writeln('Интеграл: ', LeftRectangles(a, b, n):0:6);

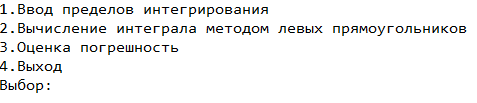
3: writeln('Погрешность: ', EstimateError(a, b, n):0:6);

**end**;

**until** choice = 4;

**end**.

6.Результат выполнения программы









Вывод:

Изучение языка программирования Pascal углубило понимание принципов разработки программ и эффективного использования языковых конструкций для решения различных задач. В процессе освоения были освоены не только синтаксические особенности, но и ключевые принципы программирования, что способствовало развитию логического мышления.

Также была изучена методология создания алгоритмических схем, что помогает упорядочить решения и лучше понять взаимодействие компонентов программы. Владение алгоритмическими методами стало основой для будущих разработок и более сложных решений.

Таким образом, цели были успешно достигнуты, и полученные навыки будут активно использоваться в следующих проектах. Базовое понимание Pascal укрепило уверенность и повысило мотивацию для дальнейшего изучения информатики и программирования.