Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ОТЧЕТ**

**ПО ДОМАШНЕЙ КОНТРОЛЬНОЙ РАБОТЕ №1**

**«Вычисление значения функции»**

**ПО ДИСЦИПЛИНЕ «Основы Алгоритмизации и Программирования»**

Выполнил: студент учебной группы

ИСПк-205-52-00

Черных Иван Антонович

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2024

**Цель работы**

Изучить базовую структуру организации программы и основные конструкции языка программирования Pascal

**Формулировка задания**

Вариант 17

Задание:

1. Написать программу вычисляющую значение функции

63/x ^ (0,1 \* x) - е ^ x/x ^ 3, если х < -8;

(x ^ 3/cos(x)) \* (100/e ^ x), если -8 <= x < -2;

-x \* е ^ x, если -2 <= x < 5;

x ^ (0,1 \* x) + cos(2 \* x), если 5 <= x.

2. Вычислить значение функции на интервале [-10; 7] с шагом 0,3.

**Описание алгоритма и ответы на вопросы**

1. Инициализация переменных:

* Задается начальное значение x = -10 (левая граница интервала) и шаг step = 0.3.
* Переменная y объявлена для хранения вычисленных значений функции.

1. Вывод заголовка таблицы:

* Выводится структура таблицы

1. Цикл вычислений:

* Цикл while x <= 7 запускается, пока значение x не превысит 7. Этот интервал соответствует области определения задачи.
* Если x < -8, используется формула y := 63 / power(x, 0.1 \* x) - exp(x) / power(x, 3). Для значений x <= 0, программа присваивает y = 0, чтобы избежать ошибок вычислений.
* Если -8 =< x < -2, применяется формула y := (power(x, 3) / cos(x)) \* (100 / exp(x)).
* Если -2 <= x < 5, применяется формула y := -x \* exp(x).
* Если x >= 5, применяется формула y := power(x, 0.1 \* x) + cos(2 \* x)
* Вычисляется значение y в соответствии с выбранной формулой

1. Увеличение значения x

* Значение x увеличивается на шаг (0.3).

1. Повтор итерации

* Цикл продолжается до тех пор, пока x <= 7.

1. Завершение программы

* Когда значение x превышает 7, цикл завершается, и программа завершает выполнение

**Ответы на вопросы**

1. Что такое алгоритм?  
Алгоритм — это точная, пошаговая последовательность действий, предназначенная для решения конкретной задачи или достижения определённого результата.

2. Перечислите виды алгоритмов.  
Основные виды алгоритмов:

* Линейные алгоритмы
* Разветвляющиеся алгоритмы
* Циклические алгоритмы

3. Дайте определение условному алгоритму.  
Условный алгоритм — это алгоритм, выполнение которого зависит от выполнения определённых условий. На каждом шаге такого алгоритма производится проверка условия, и в зависимости от его выполнения выбирается один из нескольких путей действий.

4. Виды условных алгоритмов. Какие использовали для решения задачи?  
Виды условных алгоритмов:

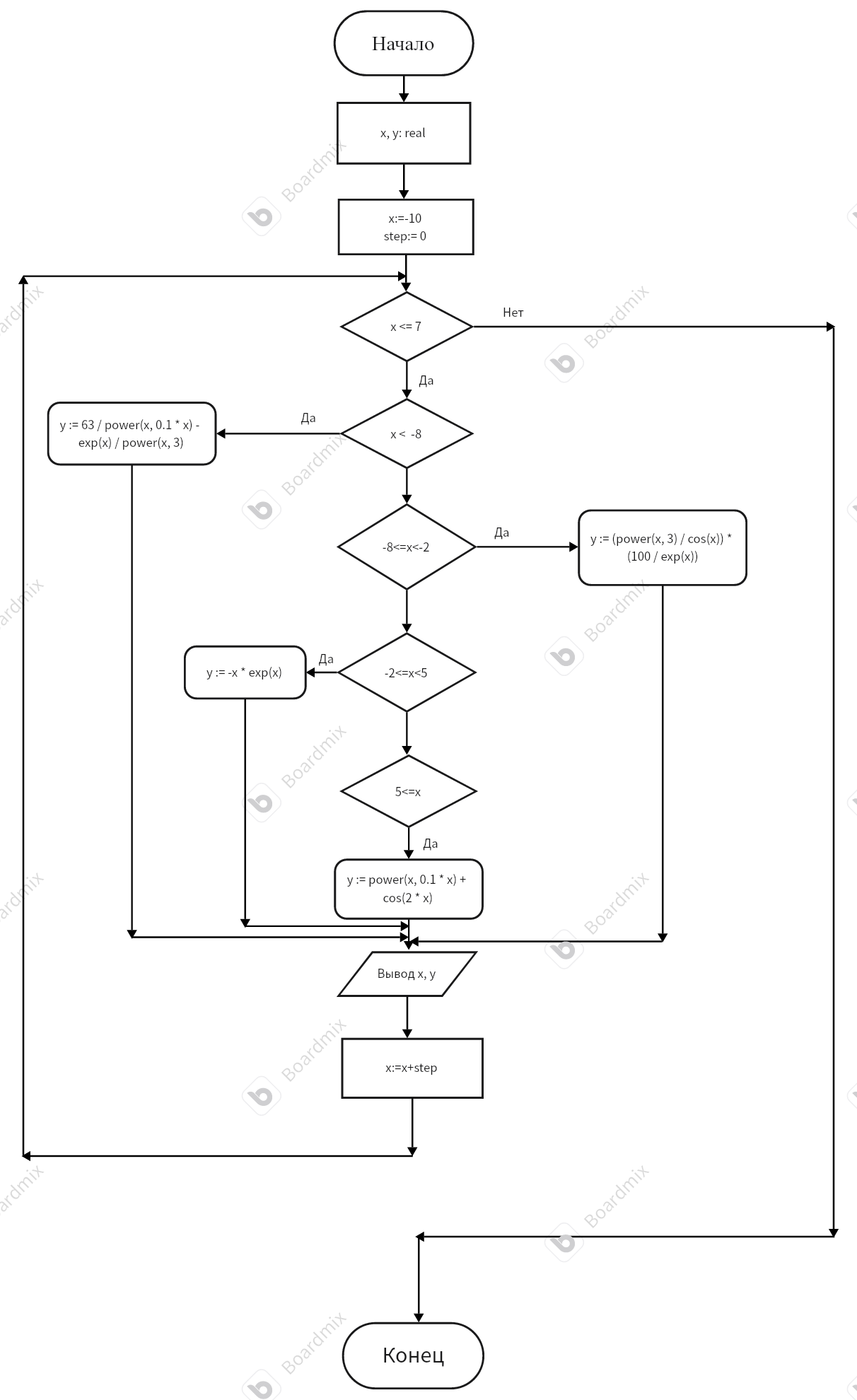
* Одноуровневые (Содержат только одно условие)
* Многоуровневые (Содержат несколько условий)
* Вложенные (Условия находятся внутри других условий)

5. Дайте определение циклическому алгоритму.  
Циклический алгоритм — это алгоритм, в котором определённые действия повторяются несколько раз, пока выполняется заданное условие или пока не достигнута конечная цель.

6. Виды циклических алгоритмов. Какие использовали для решения задачи?  
Виды циклических алгоритмов:

* С фиксированным числом повторений (например, *for*)
* С предусловием (например, *while*)
* С постусловием (например, *repeat...until*)

**Схема алгоритма**



**Код программы**

**program** dkr1;

**var** x, y: real; step: real;

**begin**

x := -10;

step := 0.3;

writeln('------TABLE------');

writeln(' x | y');

writeln('------------------');

**while** x <= 7 **do**

**begin**

**if** x < -8 **then**

y := 63 / power(x, 0.1 \* x) - exp(x) / power(x, 3)

**else if** (x >= -8) **and** (x < -2) **then**

y := (power(x, 3) / cos(x)) \* (100 / exp(x))

**else if** (x >= -2) **and** (x < 5) **then**

y := -x \* exp(x)

**else**

y := power(x, 0.1 \* x) + cos(2 \* x);

writeln('x = ', x:6:2, ' | y = ', y:10:3);

x := x + step;

**end**;

**end**.

**Результат выполнения программы**

------TABLE------

x | y

------------------

x = -10.00 | y = -630.000

x = -9.70 | y = 0

x = -9.40 | y = 0

x = -9.10 | y = 0

x = -8.80 | y = 0

x = -8.50 | y = 0

x = -8.20 | y = 0

x = -7.90 | y = 2890878113.476

x = -7.60 | y = -349104748.389

x = -7.30 | y = -109463304.913

x = -7.00 | y = -49893095.713

x = -6.70 | y = -26722016.221

x = -6.40 | y = -15885265.939

x = -6.10 | y = -10292331.021

x = -5.80 | y = -7277694.788

x = -5.50 | y = -5744653.110

x = -5.20 | y = -5440217.808

x = -4.90 | y = -8470783.112

x = -4.60 | y = 8634139.249

x = -4.30 | y = 1461991.418

x = -4.00 | y = 534585.130

x = -3.70 | y = 241572.601

x = -3.40 | y = 121815.390

x = -3.10 | y = 66187.159

x = -2.80 | y = 38312.919

x = -2.50 | y = 23759.968

x = -2.20 | y = 16329.339

x = -1.90 | y = 0.284

x = -1.60 | y = 0.323

x = -1.30 | y = 0.354

x = -1.00 | y = 0.368

x = -0.70 | y = 0.348

x = -0.40 | y = 0.268

x = -0.10 | y = 0.090

x = 0.20 | y = -0.244

x = 0.50 | y = -0.824

x = 0.80 | y = -1.780

x = 1.10 | y = -3.305

x = 1.40 | y = -5.677

x = 1.70 | y = -9.306

x = 2.00 | y = -14.778

x = 2.30 | y = -22.941

x = 2.60 | y = -35.006

x = 2.90 | y = -52.705

x = 3.20 | y = -78.504

x = 3.50 | y = -115.904

x = 3.80 | y = -169.865

x = 4.10 | y = -247.395

x = 4.40 | y = -358.384

x = 4.70 | y = -516.752

x = 5.00 | y = -742.066

x = 5.30 | y = 2.035

x = 5.60 | y = 2.827

x = 5.90 | y = 3.570

x = 6.20 | y = 4.086

x = 6.50 | y = 4.283

x = 6.80 | y = 4.194

**Вывод**В ходе выполнения задачи была изучена базовая структура программ на языке Pascal и основные конструкции, такие как условные операторы и циклы. Были использованы различные типы ветвлений для выполнения различных действий в зависимости от значений переменных, а также цикл с условием, что позволило организовать повторяющееся выполнение вычислений.