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-202-52-00

Ситников Иван Владимирович

Преподаватель:

Кузьминых Ангелина Владимировна

Киров

2023

**Цель работы:** изучить базовую структуру организации программы и основные конструкции языка программирования Pascal.

**Задание**

1. Написать программу, вычисляющую значение функции:

cos(x) + ln(x), если x < -9;

sin(x) - cos(x), если -9 <= x < 0;

lg(x) / ln(x) - x^(1/3) / 55, если 0 <= x < 4;

ln(x), если 4 <= x.

1. Вычислить значение функции на интервале [-11; 6] с шагом 0,3.

**Описание алгоритма**

Алгоритм рассчитывает и выводит значения функции res для заданных значений x в заданном диапазоне [-11; 6]. Он использует цикл while, который начинается с x = -11 и увеличивает x на 0,3 на каждой итерации до тех пор, пока x не превысит 6.

1. Инициализация переменных:

Устанавливаем начальное значение x = -11.

1. Использование цикла с предусловием(while):

Пока x не превышает 6, продолжаем выполнение.

1. Расчет значения функции res:

В зависимости от значения x, вычисляется res согласно заданным условиям:

Если x меньше -9: res = cos(x) + ln(-x).

Если x в диапазоне [-9, 0): res = sin(x) - cos(x).

Если x в диапазоне [0, 4): res = log10(x) / ln(x) - (power(x, 1/3) / 55).

Иначе (когда x больше или равно 4): res = ln(x).

1. Вывод результата:

Выводим сообщение в формате 'Для x = значение\_x Результат: значение\_res', где значение\_x и значение\_res заменяются соответствующими значениями x и res.

1. Увеличение x:

Увеличиваем x на 0,3 для следующей итерации цикла.

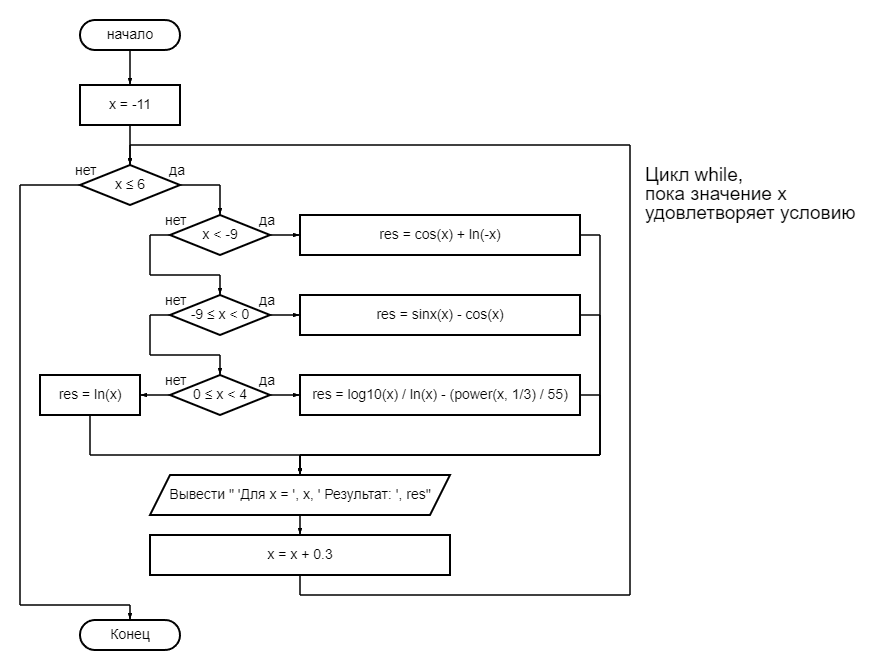
1. Повторение цикла:

Возвращаемся к шагу 2 и продолжаем выполнение цикла, пока x не превысит 6.

1. Завершение алгоритма:

По достижении x значения 6 завершаем цикл while, завершая выполнение программы.

**Схема алгоритма**



**Код программы**

**program** function1;

**var**

x, res: real;

**begin**

x := -11;

**while** x <= 6 **do**

**begin**

**if** x < -9 **then**

res := cos(x) + ln(-x)

**else if** (x >= -9) **and** (x < 0) **then**

res := sin(x) - cos(x)

**else if** (x >= 0) **and** (x < 4) **then**

res := log10(x) / ln(x) - (power(x, 1/3) / 55)

**else**

res := ln(x);

writeln('Для x = ', x, ' Результат: ', res);

x := x + 0.3;

**end**;

**end**.

**Результаты выполнения программы**

|  |  |
| --- | --- |
| Значение x | Значение функции для x |
| -11  -10.7  -10.4  -10.1  -9.8  -9.5  -9.2  -8.9  -8.6  -8.3  -8.0  -7.7  -7.4  -7.1  -6.8  -6.5  -6.2  -5.9  -5.6  -5.3  -5.0  -4.7  -4.4  -4.1  -3.8  -3.5  -3.2  -2.9  -2.6  -2.3  -2.0  -1.7  -1.4  -1.1  -0.8  -0.5  -0.2  0.1  0.4  0.7  1.0  1.3  1.6  1.9  2.2  2.5  2.8  3.1  3.4  3.7  4.0  4.3  4.6  4.9  5.2  5.5  5.8 | 2.40232097078642  2.07895445974652  1.7808215487201  1.53196724367803  1.35195611357177  1.25411964241012  1.24435986265083  0.36441435278322  -0.0556770505541084  -0.470794988785682  -0.843858212814775  -1.14154209591487  -1.33725542338602  -1.41351570656868  -1.36351084148843  -1.19170761381583  -0.913452694205715  -0.553601765913793  -0.144299240637922  0.277893106044747  0.675262089199918  1.012311921027  1.25893494386794  1.39310105759768  1.40282560285714  1.28723991498041  1.05666891922233  0.731708835935604  0.341387381547479  -0.0794291908968999  -0.493150590278543  -0.862820316156947  -1.1554168728887  -1.34480348148701  -1.41406280024669  -1.35700810049457  -1.1787359086363  0.425855229478501  0.420898003726378  0.418150772780659  0.416112663721434  0.414450974938504  0.413028898352613  0.411775166819067  0.410647364687772  0.409617958116025  0.408667941059379  0.407783577615493  0.40695458015407  0.406173020561942  1.38629436111989  1.45861502269952  1.52605630349505  1.58923520511658  1.64865862558738  1.70474809223843  1.75785791755237 |

**Вывод**

Программа и алгоритм, разработанные в ходе работы, представляют собой пример эффективного использования базовой структуры организации программы на языке Pascal и основных конструкций, таких как циклы и условные операторы. Они позволяют вычислить значения функции для различных интервалов в соответствии с заданными условиями.

Используя различные математические функции, такие как cos, sin, ln, lg и power, программа демонстрирует, как можно реализовать сложные вычисления в зависимости от значений переменной x. Это позволяет лучше понять принципы математических вычислений в контексте программирования.

Алгоритм также иллюстрирует важность эффективного управления циклами и шагами, что является ключевым аспектом при работе с большим количеством данных или при необходимости повторения операций для разных значений переменной.

В итоге, программа не только вычисляет значения функции на заданном интервале, но и служит примером организации, демонстрируя применение основных конструкций и методов программирования на языке Pascal.