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 функций»**

**ПО ДИСЦИПЛИНЕ «основы алгоритмизации и программирования»**

Выполнил: студент учебной группы

ИСПк-203-52-00

Кузнецов Захар Николаевич

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2025

Цель работы – изучить базовую структуру организации программы и основные конструкции языка программирования Pascal

Вариант 8

Задание:

1. Написать программу, вычисляющую значение функций:
2. -x -sin(x), если x < -7;
3. ln(x) \* tg(x) – cos (2 \* x), если -7 <= x <= 0;
4. 100 \* -x + x(0.1\*x)/-x, если 0 <= x < 2;
5. x(0.1\*x)/8, если 2 <= x;
6. Вычислить значение функций на интервале [-9;4] с шагом 0.1

Алгоритм – это последовательность четко определенных действий, выполнение которых ведет к решению задачи

Виды алгоритмов:

* Линейный алгоритм
* Разветвляющийся алгоритм
* Циклический алгоритм

Условный алгоритм – часть программы, которая выполняется только при соблюдении определенного условия.

Виды условных алгоритмов:

* Простое условие
* Двусловный выбор
* Множественный выбор
* Выбор по значению
* Вложенные условия
* Условия с логическими операторами
* Тернарный оператор

Использованные условные алгоритмы:

Циклический алгоритм — алгоритм, в котором некоторая последовательность действий выполняется многократно, пока выполняется заданное условие или для каждого элемента набора данных.

Виды циклических алгоритмов:

* С предусловием
* С постусловием
* С параметром

Использованные циклические алгоритмы:

1. Задача: -х -sin(x), если х<-7

Код программы:

program zadanie1;

uses Math;

var

x, result: Real;

begin

Write('Введите значение x (x < -7): ');

ReadLn(x);

if x < -7 then

begin

result := -x - sin(x);

WriteLn('f(', x:0:2, ') = ', result:0:4);

end

else

WriteLn('Ошибка: x должен быть меньше -7');

Write('Нажмите Enter для выхода...');

ReadLn;

end.

program zadanie1;

uses Math;

var x, result: Real;

begin

Write('Введите значение x (x < -7): ');

ReadLn(x);

if x < -7 then

begin

result := -x - sin(x);

WriteLn('f(', x:0:2, ') = ', result:0:4);

end

else

begin

WriteLn('Ошибка: x должен быть меньше -7');

end;

ReadLn; // Ожидание нажатия Enter

end.

Задача: ln(x)\*tg(x)-cos(2\*x), если -7 <= x < 0

Код программы

import math

def calculate\_function(x):

if -7 <= x < 0:

n\_x = x\*\*2

ta\_x = math.sin(x)

return n\_x \* ta\_x - math.cos(2 \* x)

else:

return None

x = float(input("Введите x: "))

result = calculate\_function(x)

if result is not None:

print(f"f({x}) = {result}")

else:

print("x должен быть в диапазоне [-7, 0)")

program zadanie2;

uses Math;

var

x, result: Real;

begin

Write('Введите x: ');

ReadLn(x);

if (-7 <= x) and (x < 0) then

begin

result := (x \* x) \* sin(x) - cos(2 \* x);

WriteLn('f(', x:0:2, ') = ', result:0:4);

end

else

WriteLn('x должен быть в диапазоне [-7, 0)');

ReadLn;

end.

Задача: 100 \* -x + x(0.1\*x)/-x, если 0 <= x < 2

Код программы

def calculate\_function(x):

if 0 <= x < 2:

return 100 \* -x + (x \* (0.1 \* x)) / -x

else:

return None

x = float(input("Введите x: "))

result = calculate\_function(x)

if result is not None:

print(f"f({x}) = {result}")

else:

print("x должен быть в диапазоне [0, 2)")

program zadanie3;

var

x, result: Real;

begin

Write('Введите x: ');

ReadLn(x);

if (0 <= x) and (x < 2) then

begin

result := 100 \* (-x) + (x \* (0.1 \* x)) / (-x);

WriteLn('f(', x:0:2, ') = ', result:0:4);

end

else

WriteLn('x должен быть в диапазоне [0, 2)');

ReadLn;

end.

Задача: x(0.1\*x)/8, если 2 <= x

Код программы:

def calculate\_function(x):

if 2 <= x:

return x \* (0.1 \* x) / 8

else:

return None

x = float(input("Введите x: "))

result = calculate\_function(x)

if result is not None:

print(f"f({x}) = {result}")

else:

print("x должен быть >= 2")

program zadanie4;

var

x, result: Real;

begin

Write('Введите x: ');

ReadLn(x);

if x >= 2 then

begin

result := x \* (0.1 \* x) / 8;

WriteLn('f(', x:0:2, ') = ', result:0:4);

end

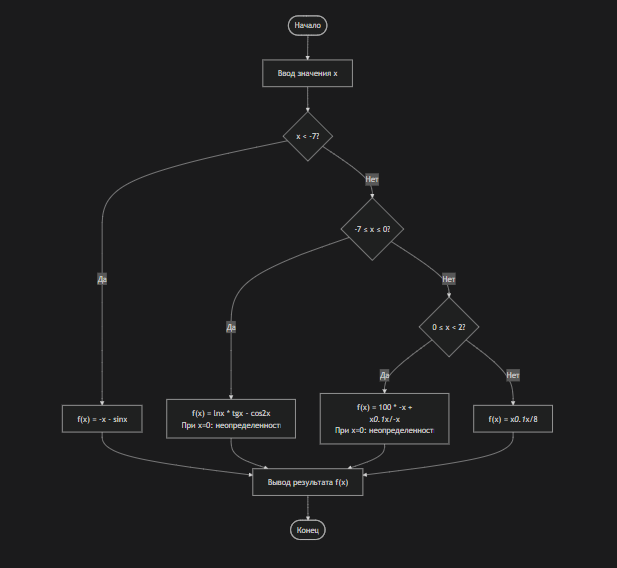
else

WriteLn('x должен быть >= 2');

ReadLn;

end.

Блок схема



2.Вычислить значение функций на интервале [-9; 4] c шагом 0.1;

Паскаль

Питон

program CalcFunc;

uses Math;

var x, r: real;

function f(x: real): real;

begin

if x < -7 then

f := -x - sin(x)

else if (x >= -7) and (x <= 0) then

begin

if x = 0 then

f := NaN

else

f := ln(abs(x)) \* tan(x) - cos(2\*x);

end

else if (x >= 0) and (x < 2) then

begin

if x = 0 then

f := NaN

else

f := -100\*x + (0.1\*x\*x)/(-x);

end

else

f := 0.0125 \* x \* x;

end;

begin

writeln('x':6, 'f(x)':15);

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

x := -9.0;

while x <= 4.0 do

begin

r := f(x);

if IsNan(r) then

writeln(x:6:1, 'неопределенность':20)

else

writeln(x:6:1, r:15:6);

x := x + 0.1;

end;

end.

import math

def f(x):

if x < -7:

return -x - math.sin(x)

elif -7 <= x <= 0:

if x >= 0:

return float('nan')

return math.log(abs(x)) \* math.tan(x) - math.cos(2\*x)

elif 0 <= x < 2:

if x == 0:

return float('nan')

return -100\*x + (0.1\*x\*x)/(-x)

else:

return 0.0125 \* x \* x

print("x\t\tf(x)")

print("-" \* 30)

x = -9.0

while x <= 4.0:

result = f(x)

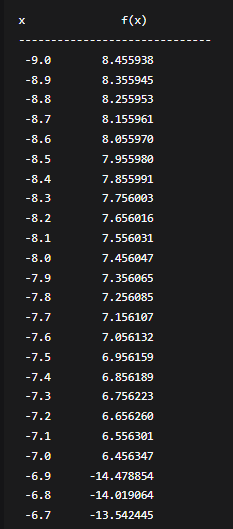
if math.isnan(result):

print(f"{x:6.1f}\tнеопределенность")

else:

print(f"{x:6.1f}\t{result:12.6f}")

x = round(x + 0.1, 1)

Вычисления

Вывод: В этой работе мы научились строить блок схемы с множеством алгоритмов и циклов. Это поможет нам писать коды и понимать программирование на шаг выше.