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

Черепанов Илья Николаевич

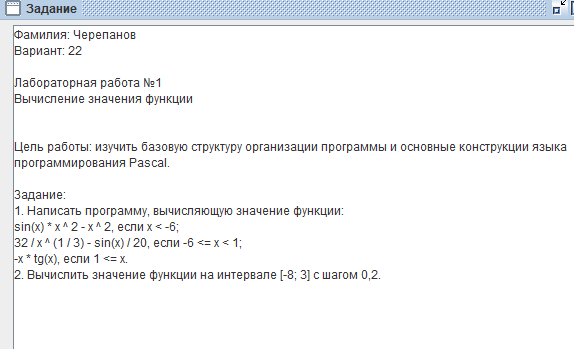
Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2025

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

****

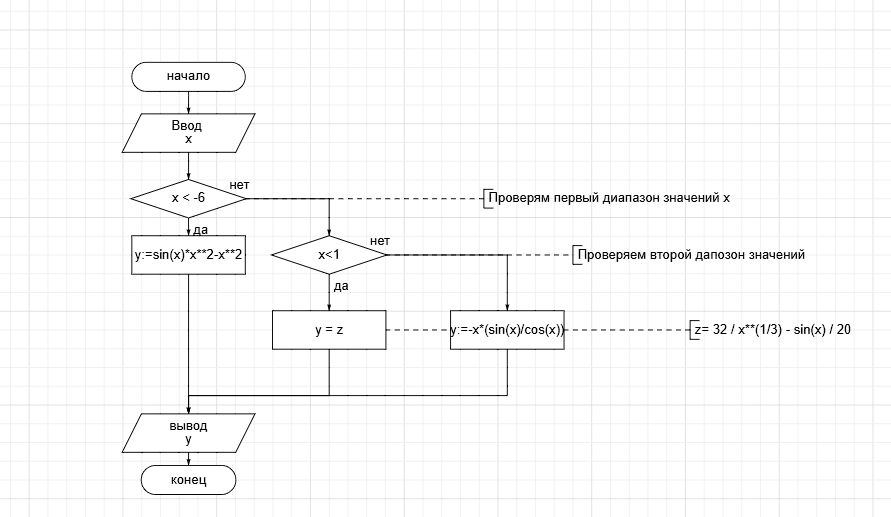
(Рисунок №1)

**Алгоритм кода**

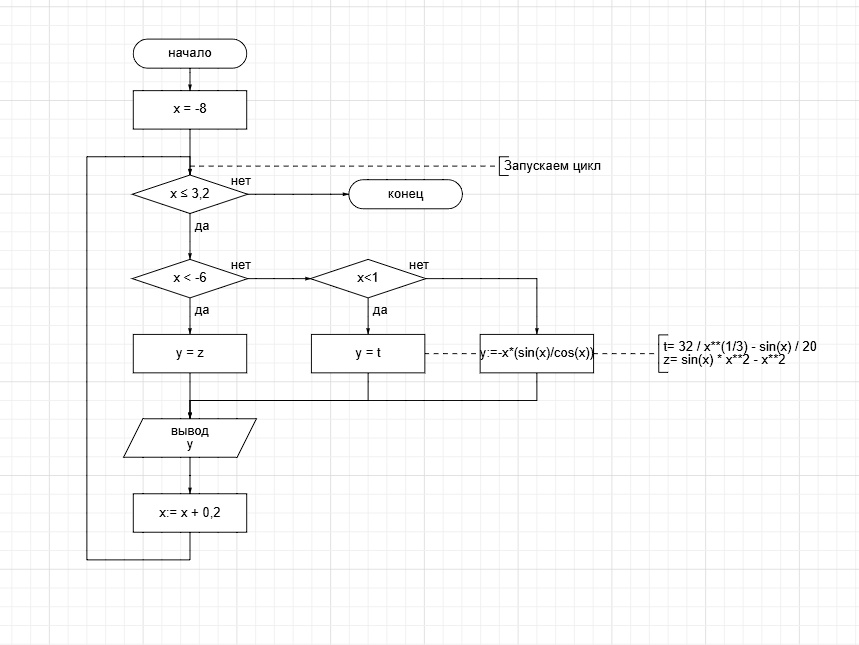
1. Задаем число
2. Проверяем в каком числовом оно диапазоне находится
3. В зависимости от диапазона находим значение функции по данной нам формуле
4. Выводим ответ

**Ответы на вопросы**

1. **Алгоритм** - это последовательность чётко определённых шагов или инструкций, которые описывают, как выполнить задачу или решить проблему.
2. Алгоритмы подразделяются на: линейный, разветвляющийся и циклический.
3. Условный алгоритм - это такая организация алгоритма, при которой в зависимости от выполнения или невыполнения некоторого условия выполняется та или другая последовательность команд.
4. Условные алгоритмы делятся на полные и неполные. Это алгоритмические конструкции, в которых выполнение порядка действий зависит от истинности или ложности условия. В задачах использовались полные алгоритмы, то есть алгоритмы, в которых для каждого условия есть определённый результат: если одно условие выполняется, выводится одно сообщение, если нет - другое.



(Рисунок №2: схема алгоритма 1)



(Рисунок №3: схема алгоритма 2)

**Код программы**

import math

x = float(input('Введите значение x: '))

if x < -6:

y = math.sin(x) \* x\*\*2 - x\*\*2

elif -6 <= x < 1:

y = 32 / (x\*\*(1/3)) - math.sin(x) / 20

if -6 <= x < 0:

y = "не определено"

else:

y = -x \* (math.sin(x) / math.cos(x))

print('Значение функции = ', y)

(Задача 1 Python)

import math

print('Вычисление функции на интервале [-8; 3] с шагом 0.2:')

x = -8.0

while x <= 3.2:

if x < -6:

y = math.sin(x) \* x \*\* 2 - x \*\* 2

elif -6 <= x < 1:

y = 32 / (x \*\* (1 / 3)) - math.sin(x) / 20

if -6 <= x < 0:

y = 'не определено'

else:

y = -x \* (math.sin(x) / math.cos(x))

print('x = ', round(x,2), 'y = ', y)

x += 0.2

(Задача 2 Python)

**var** x, y:real;

**begin**

writeln('Введите значение x');

readln(x);

**if** x<-6 **then**

y:= sin(x) \* x\*\*2 - x\*\*2

**else if** (-6 <= x) **and** (x < 1) **then**

y:= 32 / x\*\*(1/3) - sin(x) / 20

**else**

y:= -x\*(sin(x)/cos(x));

writeln('Значение функции = ',y:0:2)

**end**.

(Задача 1 Pascal)

var x, y: real;

i: integer;

begin

writeln('Вычисление функции на интервале [-8; 3] с шагом 0.2:');

x := -8.0;

while x <= 3.2 do

begin

if x < -6 then

y:= sin(x) \* x\*\*2 - x\*\*2

else if (-6 <= x) and (x < 1) then

y:= 32 / x\*\*(1/3) - sin(x) / 20

else

y:= -x\*(sin(x)/cos(x));

writeln('x = ', x:0:2, ' y = ', y:0:2);

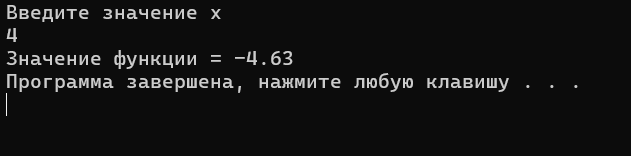
x := x + 0.2;

end;

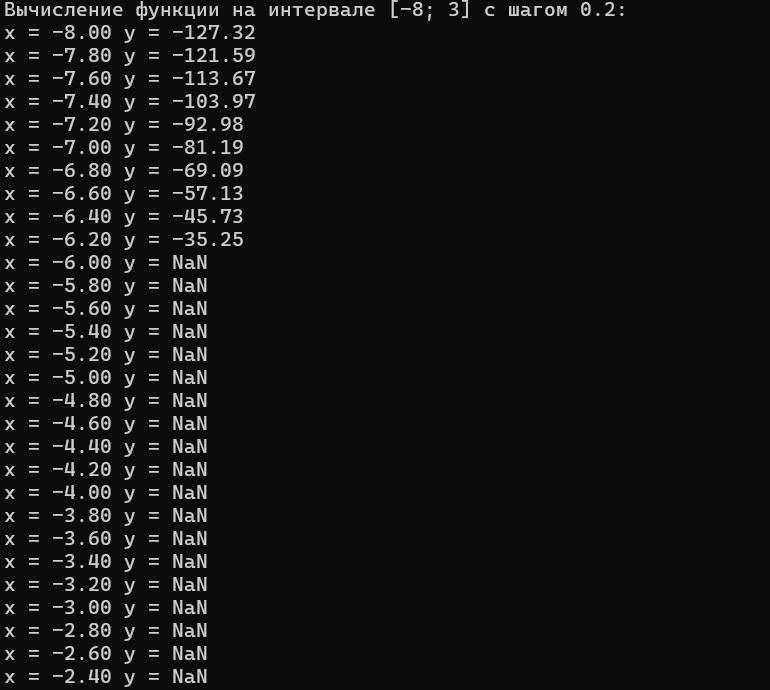
end.

(Задача 2 Pascal)

**Результат выполнения программы**



(Рисунок №4)



(Рисунок №5)

**Вывод**

В результате выполнения лабораторной работы №1 на тему «Вычисление значения функции» мы многое узнали, а также многое вспомнили. Были изучены основные элементы языков программирования Pascal и Python, необходимые для построения алгоритмов, таких как оператор присваивания, условные конструкции и циклические структуры. Особое внимание уделялось вопросам корректности и точности вычислений, а также обработке особых случаев и ошибок, возникающих при работе с функциями. Проведенное тестирование подтвердило работоспособность разработанной программы и её способность правильно вычислять значения заданной функции в данном диапазоне аргументов. Данный опыт является важным этапом подготовки будущих специалистов в области информатики и программирования, способствуя формированию профессиональных компетенций, необходимых для успешной работы в сфере информационных технологий.