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-203-52-00

Куликов Иван Алексеевич

Преподаватель:

Сергеева Елизавета Григорьевна

Киров, 2022

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

**Задание (Вариант 12):**

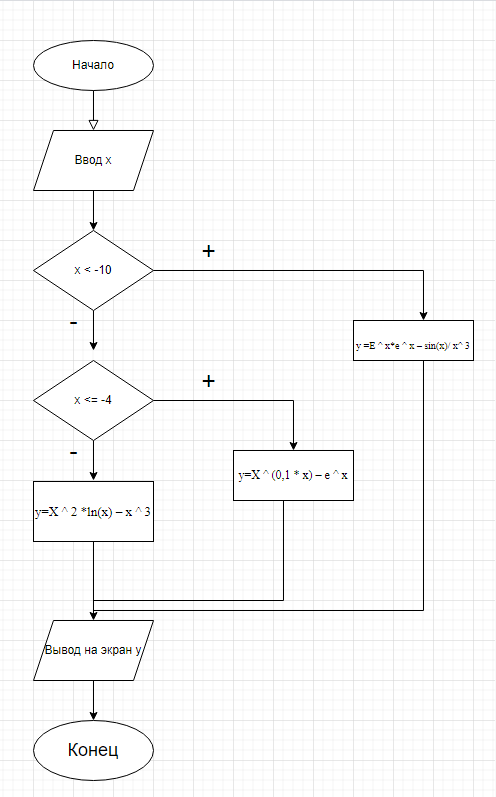
1. Написать программу, вычисляющую функции:

E ^ x \* e ^ x – sin(x) / x ^ 3, если x < -10;

X ^ (0,1 \* x) – e ^ x, если -10 <= x < -4;

X ^ 2 \* ln(x) – x ^ 3, если -4 < = x.

1. Вычислить значение функции на интервале [ -12; -2] с шагом 0,1.

**Схема алгоритма**

1)

Рисунок 1 – Задание №1. Схема алгоритма

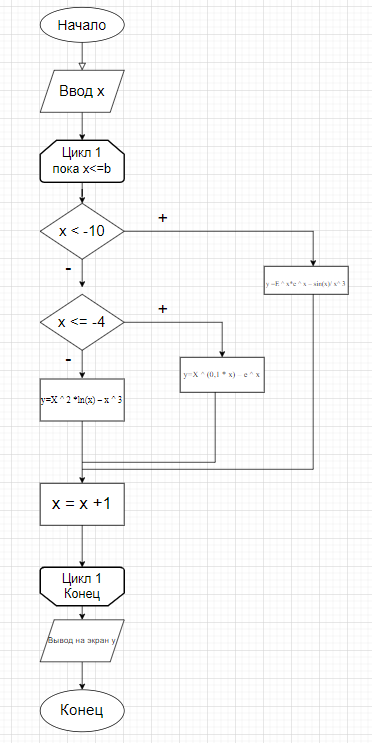
2)

Рисунок 2 – Задание №2. Схема алгоритма

**Код программы:**

1)

program z1\_lr1;

var x,func:real;

begin

read(x);

if (x<-10) then func:=power(e,x) \* power(e,x) + sin (x)/power(x,3);

if (x<=-4)and (x>=-10) then func:=power(x,0.1 \* x)- power(e,x);

if (x>-4) then func:=sqr(x)\*ln(x) - power(x,3);

writeln('y = ',func:4:2);

end.

2)

program z1\_lr1;

var a,b,h,x,func:real;

begin

read(a,b,h);

x:=a;

while x<=b do

begin

if (x<-10) then func:=power(e,x) \* power(e,x) + (sin (x)/power(x,3));

if (x<=-4) and (x>=-10) then func:=power(x,0.1 \*x)- power(e,x);

if x>-4 then func:=sqr(x)\*ln(x) -power(x,3);

writeln('x = ',x,' y = ',func:3:2);

x:=x+h;

end;

end.

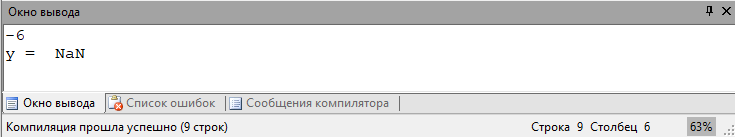
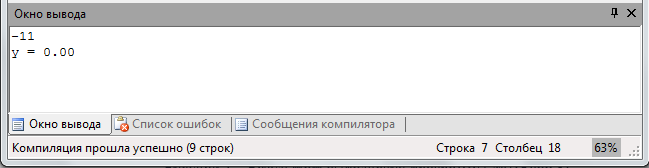
**Результат выполнения программы**

Рисунок 3 – Результат выполнения программы №1(Ввод 1)

Рисунок 4 – Результат выполнения программы №1(Ввод 2)[[1]](#footnote-1)

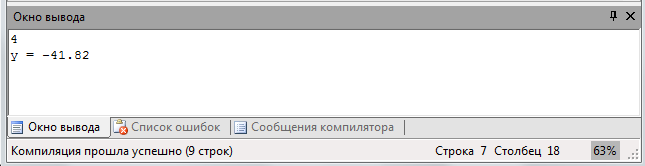


Рисунок 5 – Результат выполнения программы №1(Ввод 3)

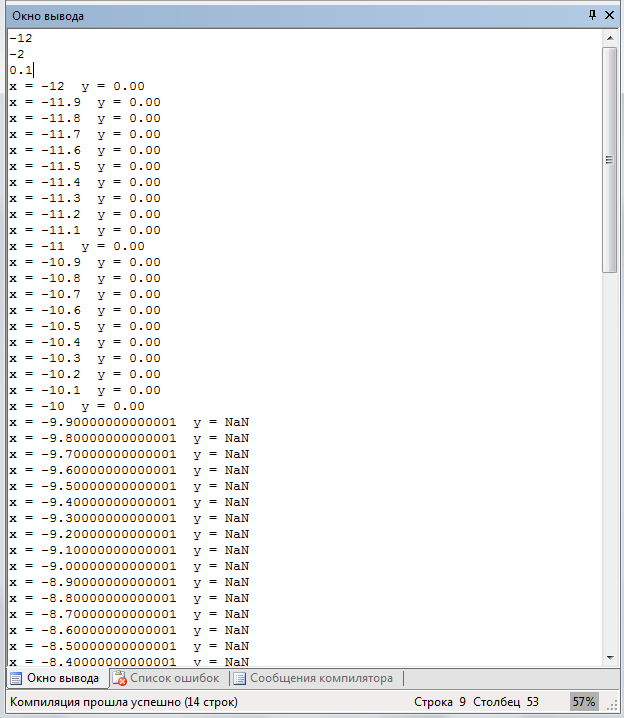


Рисунок 6 – Результат выполнения программы №2

**Вывод:**

Исходя из общего положения и пользуясь в качестве довода и логического обоснования совокупность ранее упомянутых эмпирических знаний можно сделать закономерный вывод, что в ходе работы мы использовали знания, полученные нами на столь замечательной и бесподобной дисциплине, как Основы алгоритмизации и программирования. После лабораторной работы мы можем смело сказать, что полученные на лекционных занятиях знания и умения в полной мере пригодились для грамотного и быстрого выполнения лабораторной работы №1.

Входе выполнения данной лабораторной работы мы столкнулись с неизвестным сокращением NaN, которое появилось при втором вводе данных в первую программу (см. рис. 4). После, мы узнали о том, что обозначает данное сокращение: NaN (не число) – числовой тип данных, значение которого является неопределенным или непредставимым (не соответствует ни одному из стандартных типов). Эти значения возникают при выполнении операций, которые имеют неопределенный числовой результат. Также стоит отметить тот факт, что во время написания кода на языке программирования Pascal мы использовали функцию power (возводит число в степень), о которой мы узнали во время работы над заданием. Таким образом, благодаря лабораторной работе №1 мы не только отработали уже полученные знания, но и узнали что-то новое.

С иными сложностями мы не сталкивались.

Подводя итоги, можно смело утверждать, что мы изучили базовую структуру организации программы и основные конструкции языка программирования Pascal.

1. NaN (Not a Number) – одно из особых состояний числа с плавающей запятой. Используется во многих математических библиотеках и математических сопроцессорах. Данное состояние может возникнуть в различных случаях, например, когда предыдущая математическая операция завершилась с неопределённым результатом или если в ячейку памяти попало не удовлетворяющее условиям число. [↑](#footnote-ref-1)