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»**

Выполнила: студентка учебной группы

ИСПк-202-52-00

Широнина Анна Андреевна

Преподаватель:

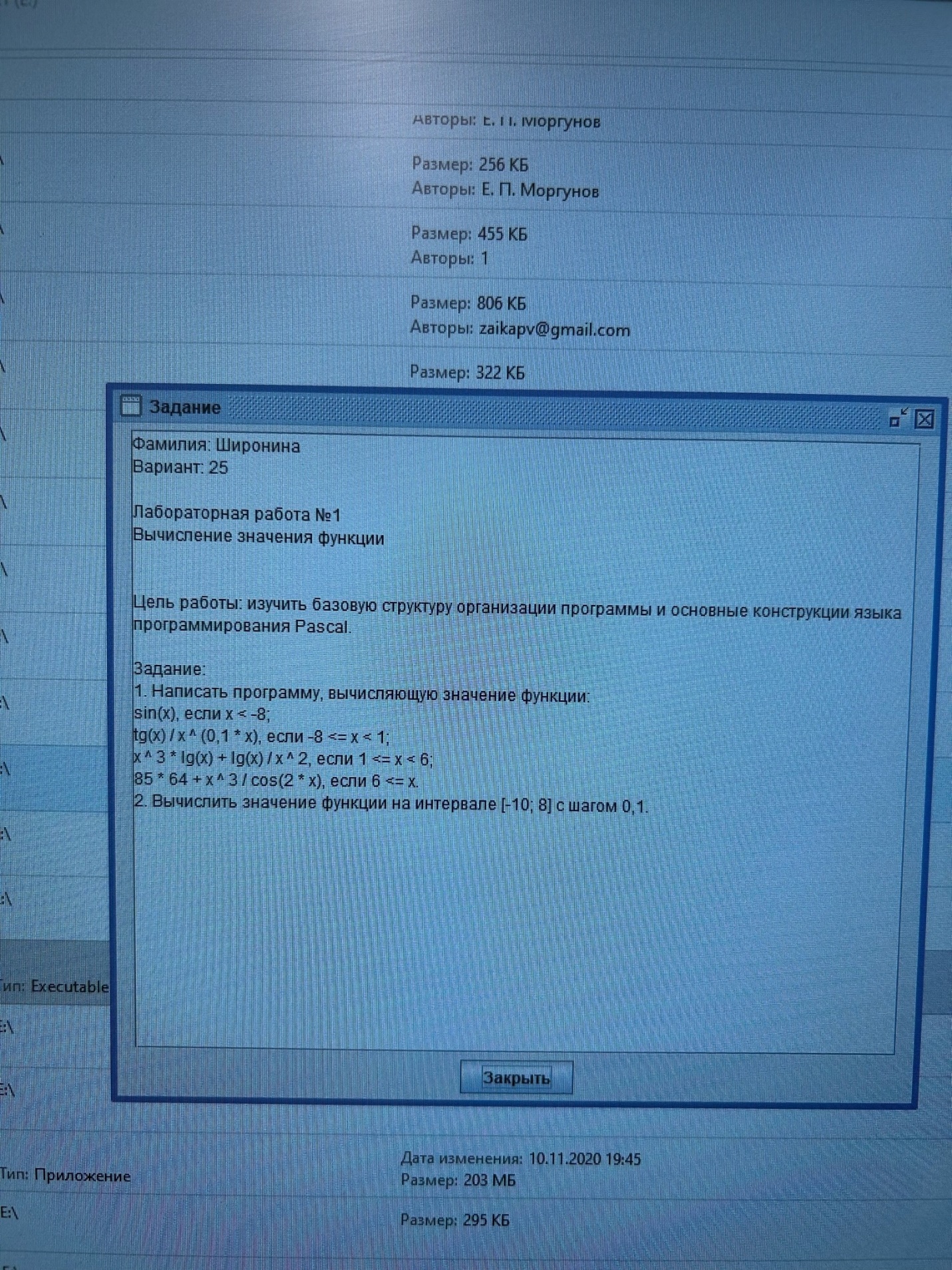
Сергеева Елизавета Григорьевна

Киров

2024

**Цель работы**: изучить базовую структуру организации программы и основным конструкция языка программирования Pascal.

1. **Формулировка заданий. Вариант 25**

****

**1.1 Задача 1**

**Постановка задачи**

Написать программу вычисляющую значение функции:

, если x < -8;

, если -8 <= x < 1;

, если 1 <= x < 6;

, если 6 <= x.

**1.2 Задача 2**

**Постановка задачи**

Вычислить значение функции на интервале [-10; 8] с шагом 0,1:

, если x < -8;

, если -8 <= x < 1;

, если 1 <= x < 6;

, если 6 <= x.

1. **Описание алгоритма и ответы на вопросы**

***1. Что такое алгоритм?***

Алгоритм – это набор инструкций, описывающих порядок действий исполнителя для достижения результата решения задачи за конечное число действий.

***2. Перечислите виды алгоритмов.***

* Линейный
* Разветвляющийся
* Циклический

***3. Дайте определение условному алгоритму.***

Условный алгоритм — это алгоритм, содержащий хотя бы одну проверку условия, в результате которой обеспечивается переход на один из возможных вариантов решения;

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

Существует две реализации структуры ветвления – полная и неполная:

Полная форма ветвления означает, что осуществляется выбор между двумя действиями. Если проверка условия даёт результат «да», то выбирается первое действие, иначе — второе действие.

Краткая форма ветвления предполагает, что если условие истинно, то выполняется первая команда, иначе — никакие действия не выполняются.

Для решения задачи использовался полный вид условного алгоритма, так как, если после условия ответ был ДА, то происходили математические вычисления и выводилось решение, если ответ НЕТ, то действия переходили к следующему условию и так далее.

***5. Дайте определение циклическому алгоритму.***

Циклический — алгоритм, предусматривающий многократное повторение одного и того же действия (одних и тех же операций) над новыми исходными данными. К циклическим алгоритмам сводится большинство методов вычислений и перебора вариантов.

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

Циклы с предусловием

Если в цикле с предусловием входящие в тело цикла команды могут не выполняться ни разу (если начальное значение параметра цикла удовлетворяет условию выхода из цикла).

Циклы с постусловием

В цикле с постусловием – выполняются как минимум один раз (даже если начальное значение параметра цикла удовлетворяет условию выхода из него).

Для решения данной использовался цикл с предусловием.

***7. Описание алгоритма***

**7.1 Задача 1**

Для начала работы программы мы вводим значение переменной X. Далее мы переходим к первому условию программы, которое имеет вид, что если x < -8, то используется функция . Если условие не выполняется, то мы переходим ко второму условию, которое имеет вид, что если -8 <= x < 1, то используется выражение . Далее, если второе условие нам также не подходит, то переходим к третьему условию, которое выполняется при 1 <= x < 6, и используется выражение . И ,если нам не подходят все три выше описанных условия, то мы переходим к четвертому условию, звучит он так: если 6 <= x, то выполняется выражение . Решение записывается в переменную Y, и обязательно выводится результат одного из четырех выражений.

**7.2 Задача 2**

Для начала работы программы мы вводим значение переменной X и задаем значения для интервала и шага. Далее мы переходим к первому условию программы, которое имеет вид, что если x < -8, то используется функция . Если условие не выполняется, то мы переходим ко второму условию, которое имеет вид, что если -8 <= x < 1, то используется выражение . Далее, если второе условие нам также не подходит, то переходим к третьему условию, которое выполняется при 1 <= x < 6, и используется выражение . И ,если нам не подходят все три выше описанных условия, то мы переходим к четвертому условию, звучит он так: если 6 <= x, то выполняется выражение . Решение по выбранной благодаря условиям функции каждого значения по интервалу записывается в переменную Y и по очереди выводится с каждым шагом.

1. **Схема алгоритма**
   1. **Задача 1**

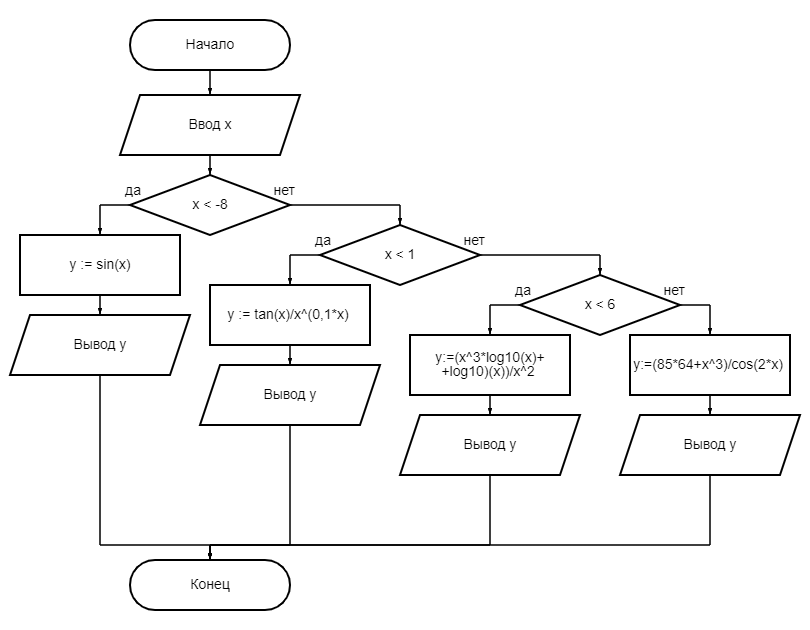
****

Рисунок 1 – Алгоритм решения задачи №1

* 1. **Задача 2**

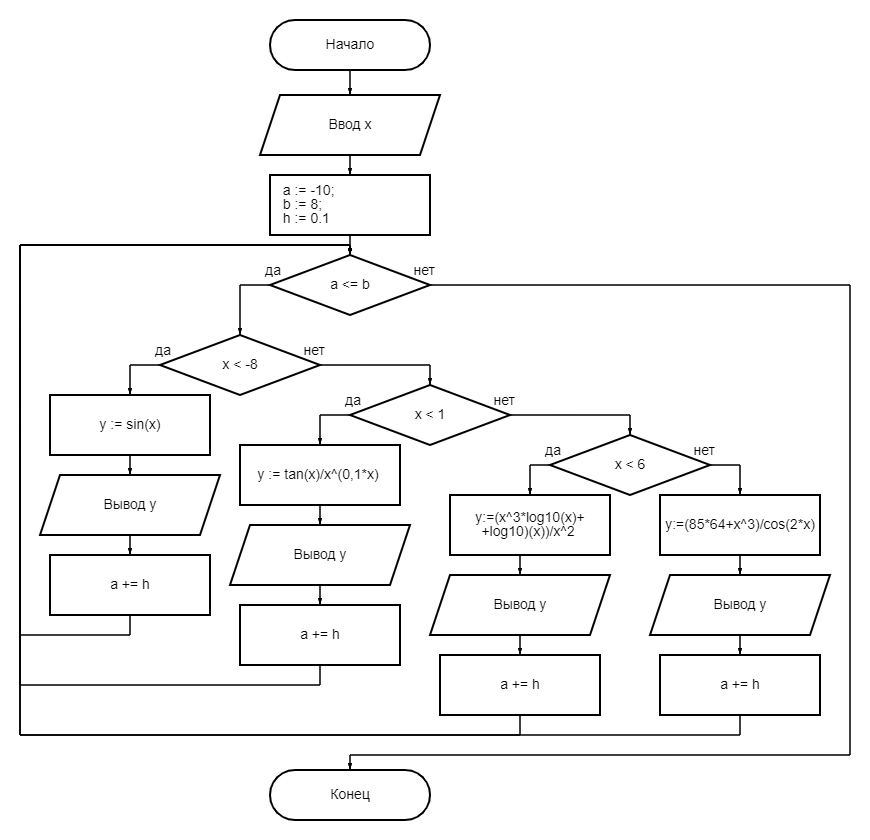
****

Рисунок 2 – Алгоритм решения задачи №2

1. **Код программы**

**4.1 Задача 1**

**var**

x, y: real;

**begin**

readln(x);

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

**begin**

y := sin(x);

writeln(y);

**end**

**else if** x < 1 **then**

**begin**

y := tan(x) / power(x, 0.1 \* x);

writeln(y)

**end**

**else if** x < 6 **then**

**begin**

y := (power(x, 3) \* log10(x) + log10(x)) / power(x, 2);

writeln(y)

**end**

**else**

**begin**

y := (85 \* 64 + power(x, 3)) / cos(2 \* x);

writeln(y)

**end**

**end**.

**4.2 Задача 2**

**var**

a, b, h, x, y: real;

**begin**

readln(x);

a := -10;

b := 8;

h := 0.1;

**begin**

**while** a <= b **do**

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

**begin**

y := sin(a);

writeln(y);

a += h;

**end**

**else if** x < 1 **then**

**begin**

y := tan(a) / power(a, 0.1 \* a);

writeln(y);

a += h;

**end**

**else if** x < 6 **then**

**begin**

y := (power(a, 3) \* log10(a) + log10(a)) / power(a, 2);

writeln(y);

a += h;

**end**

**else**

**begin**

y := (85 \* 64 + power(a, 3)) / cos(2 \* a);

writeln(y);

a += h;

**end**

**end**

**end**.

1. **Результат выполнения программы**

**5.1 Задача 1**

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
| -9 | -0.412118485241757 |
| -6 | NaN |
| 5 | 3.52280882185353 |
| 8 | -6215.152799396 |

\*NaN выводится, потому что отрицательное число может быть возведено только в целую степень

**5.2 Задача 2**

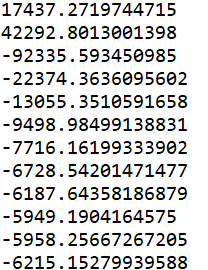
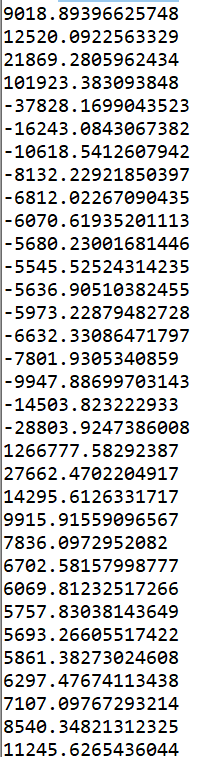
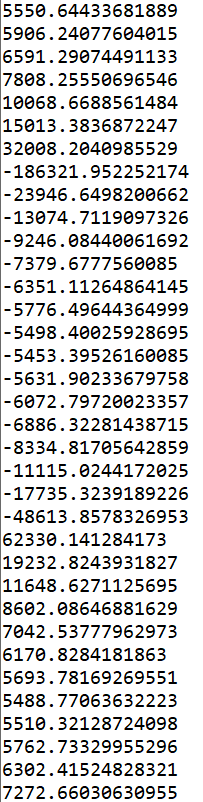
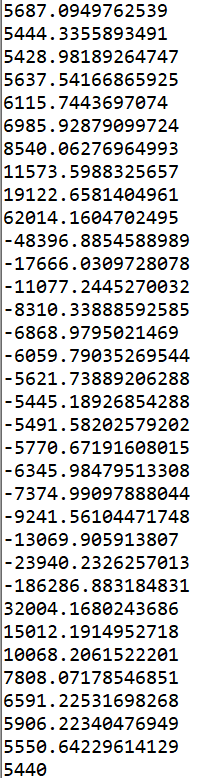
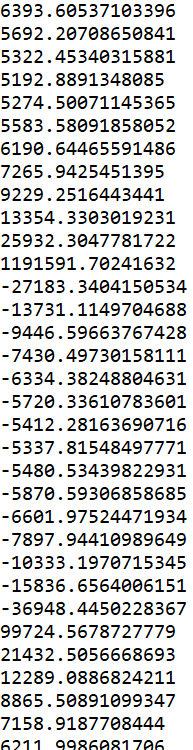
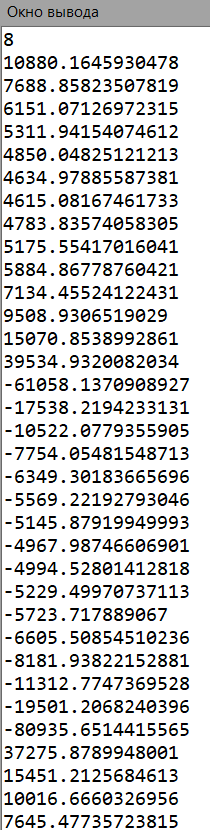
****

Рисунок 3 – Результат выполнения программы задачи №2

1. **Вывод**

В ходе работы мы изучили базовую структуру организации программы и основные конструкции языка программирования Pascal. Используя различные конструкты, такие как переменные, операторные конструкции (условные операторы и циклы) и т.д, мы получили представление о том, как создаются и функционируют программы на Pascal.

Таким образом, для решения задач, связанных с циклами и условиями в языке Pascal, необходимо выбрать подходящий оператор цикла или условия, описать блок операторов, который будет выполнен, и обеспечить выполнение условий, при которых этот блок будет выполнен. Важно учитывать возможные варианты развития ситуации, и обрабатывать возможные ошибки и крайние случаи.

Для работы над схемами алгоритма необходимо четко сформулировать проблему, которую необходимо решить, собрать входные данные, необходимые для алгоритма, описать последовательность шагов, которые необходимо выполнить для решения задачи, использовать стандартные символы для построения схемы алгоритма (овалы – начало и конец, прямоугольники – действия или процессы, ромбы – условия или проверки и т.д.), проверить алгоритм на разных входных данных для выявления возможных ошибок, проанализировать алгоритм и внести изменения для повышения его эффективности, записать и оформить полученные результаты. Работа над схемами алгоритма помогает визуализировать процесс и обеспечить более легкое понимание и оптимизацию решения задачи.