**Министерство науки и высшего образования Российской Федерации**

федеральное государственное автономное образовательное учреждение

высшего образования

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Инженерная школа природных ресурсов

Направление подготовки Химическая технология

Отделение химической инженерии

**Составление программ с использованием   
процедур и функций**

**Лабораторная работа по дисциплине «Углубленный курс информатики»**

Выполнил студент гр. А.В. Исаева

(Подпись)

\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ 2020 г.

Отчет принят:

Преподаватель

доцент ОХИ ИШПР, к.т.н. В.А. Чузлов

(Подпись)

\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ 2020 г.

Томск 2020 г.

**Цель работы:** составление программ с использованием процедур и функций.

**Теоретическая часть**

При решении различных задач часто возникает необходимость проводить вычисления по одним и тем же алгоритмам, но с различными данными в одной программе. Такие вычисления целесообразно оформить в виде отдельных подпрограмм. Подпрограммы являются основой модульного программирования.

В языке Паскаль имеется две разновидности подпрограмм: ***процедуры*** и ***функции***.

Структура любой подпрограммы аналогична структуре всей программы. Текст подпрограммы должен быть помещен в тексте программы непосредственно перед основным блоком (после объявления констант, типов и переменных).

Все параметры, которые используются в подпрограмме, можно разделить на две категории: глобальные и локальные.

***Глобальные*** – это параметры, объявленные в разделе описаний основной программы, они действуют как в основной программе, так и в любой ее подпрограмме.

***Локальные*** – это параметры, объявленные внутри подпрограммы и доступны только ей самой. Они недоступны для операторов основной программы и других подпрограмм.

Использование подпрограмм связанно с двумя этапами:

-       описание подпрограмм;

-       обращение к подпрограмме.

 Процедуры

Процедуры используются в том случае, если подпрограмма имеет несколько результатов вычислений или результат является многомерной величиной, или не имеет результата.

Описание процедуры имеет вид:

**Procedure** <имя> (формальные параметры);

 <раздел описаний>

**label** <список меток>;

**const** <список констант>;

**type** <список типов>;

**var**<список переменных>

**Begin**

  <тело процедуры>

**End**;

и помещается в основной программе в разделе описаний.

**Практическая часть**

**Задание 1**

**Исходные данные**:



x=10

**Задание**

Составить программу для вычисления данного выражения. Возведение в степень оформить в виде подпрограммы-функции, вычисление y в виде подпрограммы-процедуры:

**Программная реализация**

**program** lb7;

**var**

x, y: real;

**function** pow(x, n: real): real;

**begin**

result := exp(ln(x) \* n)

**end**;

**procedure** f(x: real; **var** y: real);

**begin**

y := ((pow(x, 6) \* pow(-5 + x, 3)) / pow(1 + 2 \* x, 5));

**end**;

**begin**

x := 10;

f(x, y);

writeln(y);

**end**.

**Ответ**

30.6064908776742

**Задание 2**

**Исходные данные**: Описать вычисление f(x) по формуле:



**Задание**

1. используя подпрограмму-функцию;
2. используя подпрограмму-процедуру;

***x*** изменяется в интервале от -3 до 3 с шагом 1.

**Программная реализация**

1) **program** lb7;

**var**

y, x: real;

**function** f(x: real): real;

**begin**

**if (**x>=-2) **and (**x<2) **then**

**begin**

result := x \* x;

**end**

**else**

**if** x >= 2 **then**

**begin**

result := x \* x + 4 \* x + 5;

**end**

**else**

**begin**

result := 1;

**end**;

**end**;

**begin**

x := -3;

writeln('x',' ','y');

**repeat**

y := f(x);

writeln(x,' ', y);

x := x + 1;

**until** x > 3;

**end**.

2) **program** lb7;

**var**

y, x: real;

**procedure** pr(x: real; **var** y: real);

**begin**

**if (**x < 2) **and (**x >= -2) **then**

**begin**

y := x \* x;

**end**

**else**

**if** x >= 2.0 **then**

**begin**

y := x \* x + 4 \* x + 5;

**end**

**else**

**begin**

y := 1;

**end**;

**end**;

**begin**

x := -3;

writeln('x', ' ', 'y');

**repeat**

pr(x, y);

writeln(x, ' ', y);

x := x + 1;

**until** x > 3;

**end**.

**Ответ**

**1)** x y

-3 1

-2 4

-1 1

0 0

1 1

2 17

3 26

**2)** x y

-3 1

-2 4

-1 1

0 0

1 1

2 17

3 26

**Задание 3**

**Исходные данные**:



**Задание**

Составить программу для вычисления данного выражения. Вычисление нестандартных функций оформить в виде подпрограммы-функции, вычисление ***y*** оформить в виде процедуры (*z и a нужно объявить как локальные переменные процедуры*).

**Программная реализация**

**program** lb7;

**var**

x, y: real;

**function** pow(x, n: real): real;

**begin**

result := exp(ln(x) \* n)

**end**;

**function** ctg(x: real): real;

**begin**

result := cos(x) / sin(x)

**end**;

**procedure** f(x: real; **var** y: real);

**var**

z, a: real;

**begin**

a := 3 \* pow(x, 7 / 9) + ctg(x + 3);

z := 2 \* pow(x, 3 / 2) + ctg(x + 2);

y := ((ctg(x) - ctg(z)) / (ctg(a) + ctg(x))) - ctg(z);

**end**;

**begin**

x := 2;

f(x, y);

writeln(y);

**end**.

**Ответ**

3.59902982359085

**Задание 4**

**Исходные данные**:



**Задание**

Составить программу для вычисления ***Z***. Вычисление ***f(a)*** и возведение в степень оформить в виде подпрограмм-функций, вычисление ***Z*** оформить в виде процедуры.

**Программная реализация**

**program** lb7;

**var**

a, i, z: real;

**function** pow(x, n: real): real;

**begin**

result := exp(ln(x) \* n)

**end**;

**function** f(a: real): real;

**begin**

**if** a < 5 **then**

result := 1 / a;

**if** (a > 5) **and** (a <= 35) **then**

result := 1 + 2 \* pow(a, -3);

**if** a >= 35 **then**

result := pow(25, 1 / sqr(a));

**end**;

**procedure** p(a: real; **var** z: real);

**begin**

z := (f(a) \* f(a) / sin(a)) + 1 / a;

**end**;

**begin**

i := 1;

**repeat**

a := i \* i + Cos(-i + pow(3 + 2 \* i, 1 / 3));

p(a, z);

writeln(z:1:4);

i := i + 1;

**until** i > 10;

**end**.

**Ответ**

0.8979

0.1584

-5.4728

-11.2943

-1.1370

-2.0002

-0.9825

1.0331

3.2164

8.9944

**Выводы**

В ходе работы были составленные программы с использованием процедур и функций.