**Лабораторная работа №9**

**1) Тема:**

Разветвляющиеся вычислительные процессы. Оператор выбора.

**2) Цель:**

Научиться работать с разветвляющимися вычислительными процессами средствами языка Pascal и компилятора Pascal ABC.

**3) Используемое оборудование:**

Компьютер, компилятор Pascal ABC.

**Задача 1**

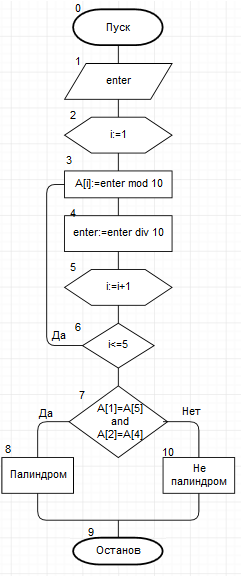
**4) Постановка задачи:**

Программа считывает с клавиатуры пятизначное число и определяет, является ли оно палиндромом. (Палиндром – это слово или число, которое одинаково читается как слева направо, так и справа налево, например: 12321, 57975).

**5) Математическая модель:**

Остатком от деления на 10, выделить цифры из числа и сравнить между собой первую и пятую, вторую и четвертую, если они совпадают, то число является палиндромом.

**6) Блок-схема:**



**7) Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Имя** | **Смысл** | **Тип** |
| I | Параметр цикла | Integer |
| Enter | Вводимое число | Integer |
| A | Массив цифр числа | integer |

**8) Код программы:**

**program** lr9;

**var**

A:**array**[1..5] **of** integer;

i,enter:integer;

**begin**

writeln('Введите пятизначное число: ');

readln(enter);

**for** i:=1 **to** 5 **do**

**begin**

A[i]:=enter **mod** 10;

enter:=enter **div** 10;

**end**;

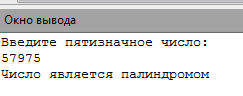
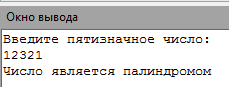
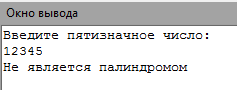
**if** (A[1]=A[5]) **and** (A[2]=A[4]) **then**

writeln('Число является палиндромом')

**else** writeln('Не является палиндромом')

**end**.

**9) Результат работы программы:**

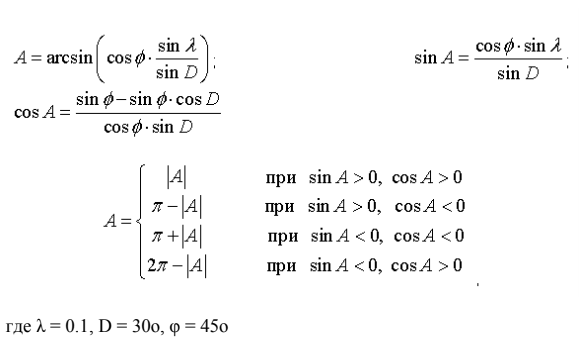


**Задача 2**

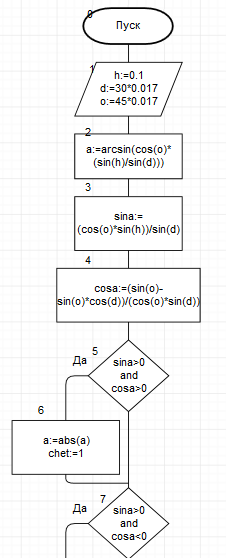
**4) Постановка задачи:**

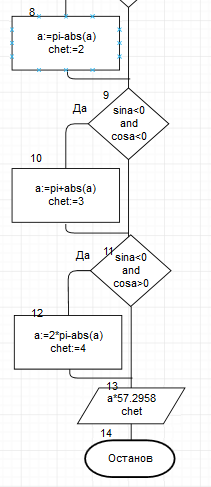
Определить четверть угла азимута А судна.

**5) Математическая модель:**



**6) Блок-схема:**





**7) Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Имя** | **Смысл** | **Тип** |
| O,d,h | Входные данные | Real |
| A | Угол азимута | Real |
| Cosa,sina | Промежуточные данные для рассчета | Real |
| Chet | Четверть, в которой находится азимут | integer |

**8) Код программы:**

**program** lr9;

**var** a, o, h, d,sina,cosa,chet: real;

**begin**

h:=0.1;

d:=30\*0.017;

o:=45\*0.017;

a:=arcsin(cos(o)\*(sin(h)/sin(d)));

sina:=(cos(o)\*sin(h))/sin(d);

cosa:=(sin(o)-sin(o)\*cos(d))/(cos(o)\*sin(d));

**if** (sina>0) **and** (cosa>0) **then begin** a:=abs(a); chet:=1; **end**;

**if** (sina>0) **and** (cosa<0) **then begin** a:=pi-abs(a); chet:=2; **end**;

**if** (sina<0) **and** (cosa<0) **then begin** a:=pi+abs(a); chet:=3; **end**;

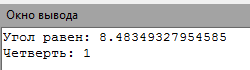
**if** (sina<0) **and** (cosa>0) **then begin** a:=2\*pi-abs(a); chet:=4; **end**;

writeln('Угол равен: ',a\*57.2958);

writeln('Четверть: ',chet);

**end**.

**9) Результат работы программы:**



**Задача 3**

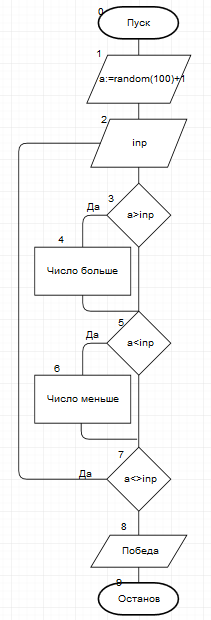
**4) Постановка задачи:**

«Угадай число». Составить программу, которая бы случайным образом загадывала число от 1 до 100, и предлагала Вам его угадать. При неправильном ответе, программа должна выводить сообщение о том, больше загаданное число или меньше. В случае победы выводится поздравление. Программа дает возможность вводить число до тех пор, пока пользователь не угадает.

**5) Математическая модель:**

-

**6) Блок-схема:**



**7) Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Имя** | **Смысл** | **Тип** |
| A | Загадываемое число | Integer |
| Inp | Число, вводимое пользователем | integer |

**8) Код программы**

**program** lr9;

**var**

a,inp: integer;

**begin**

a:=random(100)+1;

writeln('Число загадано');

**while** inp<>a **do**

**begin**

readln(inp);

**if** a>inp **then** writeln('Больше чем ', inp);

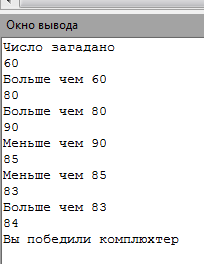
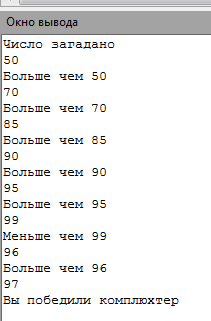
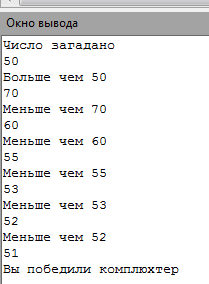
**if** a<inp **then** writeln('Меньше чем ', inp);

**end**;

writeln('Вы победили комплюхтер')

**end**.

**9) Результат работы программы:**



**Задача 4**

**4) Постановка задачи:**

Сформировать вывод слова «ворона» в зависимости от любого числительного, которое вводится с клавиатуры. Например: 1 – ворона, 3 – вороны, 5 – ворон.

**5) Математическая модель:**

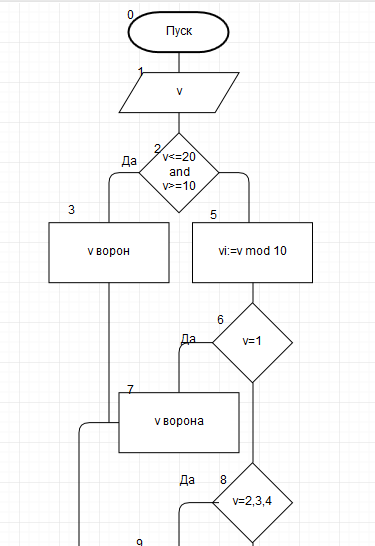
Чтобы выводить правильный падеж для любых числительных нужно выделить их в несколько групп:

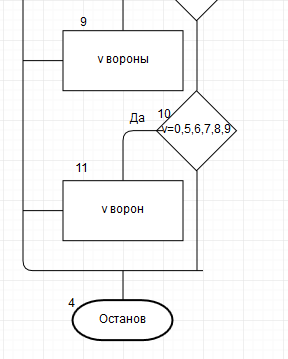
«Ворон»-числа оканчивающиеся на 0, 5, 6, 7, 8, 9, а также 10-20

«Ворона»-числа оканчивающиеся на 1

«Вороны»-числа оканчивающиеся на 2, 3, 4

**6) Блок-схема:**





**7) Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| **Имя** | **Смысл** | **Тип** |
| V | Кол-во ворон | Integer |
| Vi | Последняя цифра кол-ва ворон | integer |

**8) Код программы:**

**program** lr9;

**var**

v,vi: integer;

**begin**

writeln('Введите число ворон: ');

readln(v);

**if** (v<=20) **and** (v>=10) **then**

**begin**

writeln(v,' ворон');

**end**

**else**

**begin**

vi:=v **mod** 10;

**case** vi **of**

1: writeln(v,' ворона');

2,3,4: writeln(v,' вороны');

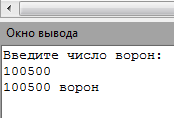
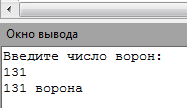
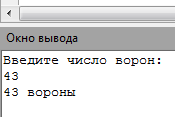
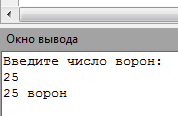
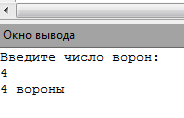
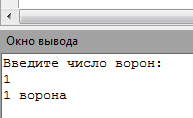
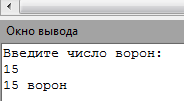
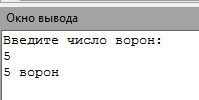
0,5,6,7,8,9: writeln(v,' ворон');

**end**;

**end**;

**end**.

**9) Результат работы программы:**



**10) Вывод:**

В ходе лабораторной работы я научился работать с разветвляющимися вычислительными процессами средствами языка Pascal и компилятора Pascal ABC.