

Министерство сельского хозяйства Российской Федерации
Департамент научно-технологической политики и образования
Федеральное государственное бюджетное
образовательное учреждение высшего профессионального образования
«Волгоградский государственный аграрный университет»
Кафедра «Математическое моделирование и информатика»

К.Е. Токарев

**ПРОГРАММИРОВАНИЕ АЛГОРИТМОВ
ЛИНЕЙНОЙ И РАЗВЕТВЛЯЮЩЕЙСЯ СТРУКТУРЫ
НА ЯЗЫКЕ TURBO PASCAL**

Методические указания к выполнению контрольной работы
по дисциплине «Информатика»
для студентов очной, заочной и сокращенно-заочной форм обучения

ОСНОВНЫЕ ПОНЯТИЯ

Язык программирования **Pascal** был разработан Николасом Виртом в шестидесятые годы прошлого века как учебный язык для студентов. Язык **Pascal** был назван в честь французского математика и философа Блеза Паскаля (1623-1662 гг.) - изобретателя первой в мире механической вычислительной машины. Современный **Turbo Pascal** сохранил простоту и структуру языка, разработанного Виртом. Это достаточно мощное средство программирования, предназначенное для написания программ различной сложности. На **Turbo Pascal** можно выполнить простые расчеты, составить программы для реализации сложных инженерных задач, обучающие программы, программы-оболочки, тестирующие программы и драйверы.

Алфавит языка

Программа на языке **Pascal** может содержать следующие символы:

- Прописные и строчные буквы латинского алфавита и знак подчеркивания ():
A, B, C.....x,y,z.
- Цифры 0, 1, 2 ..., 9.
- Специальные символы:
+ _ * / = < > . , ; :

@ ' () [] { } \$ # ^

В качестве имен программ, констант, переменных, различных процедур, функций и других объектов языка используются имена идентификаторы, которые представляют собой совокупность букв, цифр и символа подчеркивания, начинающуюся с буквы или символа подчеркивания. Идентификатор не может содержать пробел. При написании могут быть использованы как прописные, так и строчные буквы. Каждое имя (идентификатор) должно быть уникальным. Длина имени не ограничена, но значимы только первые 63 символа. Большие и маленькие буквы равнозначны.

Данные в языке Turbo Pascal

Данные языка **Pascal** можно разделить на константы и переменные.

Константы не изменяют своего значения в процессе выполнения программы. Они описываются с помощью служебного слова *const*, за которым идет список имен констант каждому из которых с помощью символа << = >> присваивается значение. Одна константа от другой отделяется точкой с запятой, например:

```
const  
h=3;  
b=-1.5;  
c=`abc`;
```

Переменные могут изменять свое значение в процессе выполнения программы неограниченное число раз. Описание переменных начинается со служебного слова *var*, за которым следуют имена переменных и через двоеточие указывается их тип, например:

```
var  
a, b: real;
```

c, d: integer;

Типы данных в Turbo Pascal

Pascal является типизированным языком. Это означает, что в Pascal любые данные характеризуются своим типом. Тип данных определяется при их описании и не может быть изменен. Тип - это множество значений, которые могут принимать данные и множество операций, допустимых над этими значениями. Данные могут участвовать только в операциях, допустимых их типом.

Перечень типов данных в языке Turbo Pascal можно представить в виде следующей схемы:

1. Простые типы (скалярные):
 - Порядковые типы (целый, вещественный, символьный, логический, перечисляемый, интервальный).
 - Ссылочный тип.
2. Структурированные типы: строковый, регулярный, комбинированный, множественный, файловый.
3. Процедурные типы.

Целочисленные типы данных

Эта группа типов обозначает множества целых чисел в различных диапазонах. Целочисленные типы данных занимают в памяти от 1 до 4 байт. Все они представлены в таблице 1.

Таблица 1

Целочисленные типы данных

Тип	Диапазон	Размер в байтах
Byte	0..255	1
Word	0..65535	2
Shortint	-128.. 127	1
Integer	-32768..32767	2
Longint	-2147483648. .2 147483647	4

Значения целых типов могут изображаться в программе двумя способами: в десятичном виде и в шестнадцатеричном виде (в этом случае в начале числа ставится знак \$). Диапазон изменений шестнадцатеричных чисел от \$0000 до \$FFFF.

Вещественные типы данных.

Эта группа типов обозначает множества вещественных значений в различных диапазонах. Вещественные типы данных занимают в памяти от 4 до 10 байт. Все они представлены в таблице 2.

Тип	Диапазон	Мантисса	Размер в байтах
Real	2.9E-39..1.7E38	11-12	6
Single	1.5E-45..3.4E38	7-8	4
Double	5.0E-324..1.7E308	15-16	8
Extended	3.4E-4951..1.1E4932	19-20	10
Comp	-2E+63..2E+62	19-20	8

Вещественные данные могут быть как с *фиксированной*, так и с *плавающей* точкой. В числах с фиксированной точкой целая часть числа отделяется от дробной точкой. Например, 7.32, 45.123. -0.12.

Числа с плавающей точкой используются для представления очень больших или малых чисел. В этой форме число записывается в виде: $\pm mE\pm P$, где m - мантисса; E - символ, обозначающий основание десятичной системы счисления; P - порядок. Знак мантиссы определяет знак числа. Порядок представляет собой однозначное или двузначное число, указывающее, на сколько разрядов вправо (знак "+") или влево (знак "-") нужно переместить точку в мантиссе, чтобы получить фактическое значение представляемой величины. Например:

в Паскале

-3.2E-06

6.42E-02

в математике

$-3.2 \cdot 10^{-6}$

$6.42 \cdot 10^2$

Символьный тип данных.

Символьный тип - это множество символов кодовой таблицы компьютера ASCII. Каждый символ занимает 1 байт и может быть описан с помощью служебного слова *char*, например:

Var

a, b: char;

В тексте программы значения переменных и константы символьного типа должны быть заключены в апострофы: 'a', '+'.

Логический (булевский) тип данных.

Булевский тип обозначается идентификатором *boolean*. Данные этого типа могут принимать два значения: *true* (истина) или *false* (ложь). Для размещения в памяти переменной булевского типа требуется 1 байт. Например:

Var

a, b: boolean;

Перечисляемый тип.

Перечисляемый тип задается непосредственным перечислением значений, которые может принимать переменная данного типа, например:

Var

a, c: (red, blue, green);

b: (dog, cat);

Можно сначала ввести перечисляемый тип данных, а затем описать переменные этого типа. Для создания нового типа используется служебное слово *type*. Формат:

Type

<имя типа>=(<значение 1, значение 2, ..., значение N>);

Var

<идентификатор, ...> : < имя типа >;

Пример:

Type

Gaz=(*Ge*, *C*, *O,N*);

Metall=(*Na*, *K*, *Li*, *Cu*, *Zn*);

Var

G1, *G2*, *G3*: *Gaz*;

Met1, *Met2* : *Metall*;

Переменные *G1*, *G2*, *G3* и *Met1*, *Met2* могут принимать только одно из перечисленных значений. Попытка присвоить им любое другое значение вызовет программное прерывание.

Интервальный тип.

Интервальный тип данных позволяет задавать две константы, которые определяют границы изменения переменных данного типа. Значение первой константы должно быть меньше значения второй. Сами же они являются целочисленными или символьными, например:

Var

a, *c*: -7.. 4;

b: 'a'..'d';

Как и случае перечисляемого типа, можно предварительно ввести тип данных с помощью служебного слова *type*, а затем описывать переменные данного типа, например:

Type int = 0..9;

Var a,b : *int*;

Структура программы на языке Turbo Pascal

Структуру программы можно записать следующим образом:

```
Program ... ; { Заголовок программы }
Uses ... ; { Подключение библиотек (модулей) }
Label ... ; { Раздел объявления меток }
Const ... ; { Раздел объявления констант }
Type ... ; { Раздел объявления новых типов }
Var ... ; { Раздел объявления переменных }
Procedure ... ; { Описание своих процедур }
Function ... ; { Описание своих функций }
Begin { Начало основной программы }
...;
```

{Операторы }
...;
End.

Описание заголовка программы

Заголовок программы состоит из служебного слова *program*, имени программы, образованного по правилам использования идентификаторов **Pascal**, и точки с запятой.

Формат:

Program <имя_программы>;

Пример:

Program lab1;

Заголовок программы несет чисто смысловую нагрузку и может отсутствовать.

Описание используемых библиотек

Этот раздел состоит из зарезервированного слова *uses* и списка имен подключаемых стандартных и пользовательских библиотечных модулей.

Формат:

Uses <имя1>, <имя2>, ...;

Пример:

Uses Crt, Dos, MyLib;

Раздел описания меток

Перед любым оператором языка Pascal можно поставить метку, что позволяет выполнить прямой переход на этот оператор с помощью оператора перехода *goto* из любого места программы.

Метка состоит из имени и следующего за ним двоеточия. Именем может служить идентификатор или цифра (десятичное число от 1 до 9999). Перед употреблением метка должна быть описана.

Формат:

Label <имя1, имя2, ... >;

Пример:

Label M1,M2;

Begin

...

Goto M1;

...

M1:

...

End.

Если метка описана, но в разделе операторов не используется, то ошибки при этом не возникает, т.е. метки можно описывать и применять по мере расширения программы.

Раздел описания констант

В разделе описания констант производится присваивание идентификаторам констант постоянных значений. Раздел начинается зарезервированным словом *const*, за которым следует ряд выражений, присваивающих идентификаторам числовые или строковые значения. Выражения присваивания отделяются друг от друга точкой с запятой.

Формат:

Const <идентификатор>=<значение >;

Пример:

```
Const  
Max = 100;  
Name = 'Pit';  
Code = $124;
```

Удачное относительно мнемоники именование констант пользователя делает программу более читаемой и позволяет быстро вносить корректировку в программу при изменении алгоритма.

Раздел описания типов данных

Тип данных может быть описан либо в разделе описания переменных, либо в разделе описания типов. Выбор описания типа зависит только от программиста и специфики программы.

Формат:

Type <имя типа>=<значения типа>;

Пример:

```
Type  
Days = 1..31;  
Matr = array [1..10] of integer;
```

Каждое описание задает множество значений и связывает с этим множеством некоторое имя типа.

Например, в данном описании тип *Days* определяет множество целых чисел от 1 до 31, *Matr* - массив из 10 целых чисел.

Раздел описания переменных

Каждая встречающаяся в программе переменная должна быть описана. Описание обязательно предшествует использованию переменной.

Формат:

```
Var <идентификатор, ...>:<тип>;
```

Пример:

```
Var  
  A, B: integer;  
  Result: real;  
  Vxod: boolean;
```

Раздел описания процедур и функций

В этом разделе размещаются тела подпрограмм. Подпрограммой называется программная единица, имеющая имя, по которому она может быть вызвана из других частей программы. В языке **Pascal** роль подпрограмм выполняют процедуры и функции. В общем случае подпрограмма имеет ту же структуру, что и программа. Для описания подпрограмм используются зарезервированные слова *procedure* и *function*, которые записываются в начале подпрограммы.

Формат процедуры:

```
Procedure <имя процедуры> {<параметры>};  
  <разделы описаний>  
  <раздел операторов>  
end;
```

Формат функции:

```
Function < имя функции> {<параметры>}:<тип результата>;  
  <разделы описаний>  
  <раздел операторов>  
end;
```

Раздел операторов

Раздел операторов является основным, т.к. именно в нем с предварительно описанными переменными, константами, значениями функций выполняются действия, позволяющие получить результат.

Формат:

```
Begin  
  <оператор1>;  
  ...  
  <операторN>;  
End .
```

Операции и выражения в языке Pascal

Конструкция языка, задающая порядок выполнения действий над элементами данных, называется *выражением*. Выражение состоит из *операндов* (operand- элемент данных, участвующий в операции), - величин и выражений, над

которыми производится операция: круглых скобок и знаков операций. *Операции* определяют действия, которые надо выполнить над операндами.

Операции в языке **Pascal** подразделяются на арифметические, отношения, логические, строковые и др. Выражения соответственно называются арифметические, отношения, логические и т.д.

Основные *арифметические операции*, применяемые в **Turbo Pascal** приведены в таблице 3.

Арифметические операции

Таблица 3

Операция	Действие	Тип операндов	Тип результата
+	Сложение	Integer, real	Integer, real
-	Вычитание	Integer, real	Integer, real
*	Умножение	Integer, real	Integer, real
/	Деление	Integer, real	Integer, real
Div	Деление нацело	Integer	Integer
Mod	Остаток от деления	Integer	Integer

Пример: $9/4 = 2.25$; $9 \text{ div } 4 = 2$; $9 \text{ mod } 4 = 1$.

Операции отношения выполняют сравнение двух операндов и определяют истинно выражение или ложно. Их результат - логический. Операции отношения приведены в таблице 4.

Операции отношения

Таблица 4

Операция	Название	Выражение	Результат
=	Равно	$A=B$	True, если A равно B
<>	Не равно	$A \neq B$	True, если A не равно B
>	Больше	$A > B$	True, если A больше B
<	Меньше	$A < B$	True, если A меньше B
>=	Больше или равно	$A \geq B$	True, если A больше или равно B
<=	Меньше или равно	$A \leq B$	True, если A меньше или равно B
in	Принадлежность	$A \text{ in } M$	True, если A находится в списке B

Логические операции выполняются над логическими данными. Определены следующие логические операции (табл. 5).

Таблица 5

Логические операции

Операция	Действие	Выражение	A	B	Результат
Not	Логическое отрицание	Not A	True		False
And	Логическое И	A and B	False		True
			True	True	True
			True	False	False

Or	Логическое ИЛИ	A or B	False	True	False
			False	False	False
			True	True	True
			True	False	True
Xor	Исключающее ИЛИ	A xor B	False	True	True
			False	False	False
			True	True	False
			True	False	True
			False	True	True
			False	False	False

Вычислите выражений производится по следующим правилам:

- 1, Запись выражения производится в одну строку без каких-либо подстрочных или надстрочных знаков с использованием только круглых скобок.
2. Выполняются операции внутри скобок. Если имеется несколько пар скобок, вычисления начинаются с самых внутренних.
3. Порядок вычислений внутри скобок определяется приоритетом (старшинством) операции:
 - а) вычисление значений функций;
 - б) not;
 - в) возведение в степень;
 - г) операции типа умножения (*, /, div, mod, and);
 - д) операции сложения (+, -, or, xor);
 - е) операции отношения.

Операция, имеющая высший приоритет выполняется первой.

4. При наличии нескольких операций одного приоритета, вычисления выполняются слева направо.

В **Turbo Pascal** определены стандартные функции над арифметическим операндами (таблица б)

Примеры:

$$\text{tg}(x) = \sin(x) / \cos(x);$$

$$\text{round}(6.8) = 7;$$

$$\text{arccos}(x) = \arctan(\sqrt{1-\text{sqr}(x)} / x);$$

$$\text{trunc}(2.1) = 2;$$

$$\text{arcsin}(x) = \arctan(x / \sqrt{1-\text{sqr}(x)});$$

$$\text{trunc}(6.8) = 6;$$

$$\log_b a = \ln(a) / \ln(b);$$

$$\text{frac}(2.1) = 0.1$$

$$\text{int}(1.3) = 1;$$

$$\text{frac}(6.8) = 0.8;$$

$$\text{int}(-1.3) = -1;$$

$$\text{odd}(3) = \text{true};$$

$$\text{round}(2.1) = 2;$$

$$\text{odd}(2) = \text{false};$$

Стандартные арифметические функции

таблица 6

Действие	Обозначение	Тип аргумента	Тип результата
Модуль числа	Abs (x)	Integer, real	Integer, real
Синус	Sin (x)	Integer, real	real
Косинус	Cos (x)	Integer, real	real
Арктангенс	Arctan (x)	Integer, real	real
π	Pi (x)		real
e^x	Exp (x)	Integer, real	real
Натуральный логарифм	Ln (x)	Integer, real	real
Десятичный логарифм	0.4343*Ln (x)	Integer, real	real
x^2	Sqr (x)	Integer, real	Integer, real
\sqrt{x}	Sqrt (x)	Integer, real	real
Вычисление целой части числа	Int (x)	real	real
Дробная часть числа	Frac (x)	real	real
Округление числа x	Round(x)	real	integer
Отсекание дробной части x	Trunc(x)	real	integer
Возвращает значение равное true, если x нечетное, false, если x четное	Odd (x)	integer	boolean
Генерация случайного числа от 0 до 1	Random	–	real
Генерация случайного числа от 0 до n	Random (n)	integer	integer
$x^n, x > 0$	Exp(n*ln(x))	Integer, real	Integer, real

$x^n, x < 0$	$- \text{Exp}(n * \ln(\text{abs } x))$	Integer, real	Integer, real
--------------	--	---------------	---------------

ОПЕРАТОРЫ

оператором называется предложение языка программирования, задающее полное описание некоторого действия, которое необходимо выполнить. Все операторы языка **Pascal** можно разделить на две группы: простые и структурные.

Простые операторы языка Паскаль

Оператор присваивания

Формат:

$\langle \text{идентификатор} \rangle := \langle \text{выражение} \rangle;$

Предписывает выполнить выражение, заданное в его правой части, и присвоить результат переменной, идентификатор которой расположен в левой части. Переменная и выражение должны иметь один и тот же тип, исключение, когда переменная вещественная, а выражение целочисленное.

Пример: $\text{Summa} := a + b.$

Оператор ввода Read и Readln

Ввод информации с клавиатуры осуществляется с помощью оператора *read*. Он может иметь один из следующих форматов:

$\text{Read}(x1, x2, \dots, xn);$

или

$\text{Readln}(x1, x2, \dots, xn),$

где $x1, x2, \dots, xn$ - список вводимых переменных.

Когда в программе встречается оператор *read*, его действие приостанавливается до тех пор, пока не будут введены исходные данные. Вводимые переменные отделяются друг от друга пробелом или *Enter*. Значения, вводимые с клавиатуры, должны иметь тот же тип, что и соответствующие им переменные, указанные в операторе *read* или *readln*. Если соответствие нарушено, возникает ошибка ввода-вывода.

Операторы *read* и *readln* аналогичны, их единственное отличие в следующем: после набора данных для оператора *readln* и нажатия клавиши *Enter* курсор переходит к началу новой строки, а для оператора *read* курсор остается на этой же строке.

Значения переменных логического и перечисляемого типов вводить с клавиатуры **нельзя**.

Оператор *readln*, записанный без параметров, используется для организации паузы - выполнение программы продолжится, как правило, только после нажатия на клавиатуре клавиши *Enter*.

Оператор вывода Write и Writeln

Для вывода информации (чисел, строк и булевых значений) на экран дисплея служат операторы *write* и *writeln*. Формат:

$\text{Write}(y1, y2, \dots, yn);$

Writeln (y1,y2, ..., yn),

где *y1, y2.....yn* - список выводимых переменных, констант, выражений (*y1, y2, ..., yn* не могут быть перечисляемого типа).

Операторы *write* и *writeln* последовательно выводят все переменные на экран дисплея. Если используется оператор *writeln*, то после вывода информации курсор перемещается на новую строку.

Оператор *writeln*, записанный без параметров, вызывает перевод строки.

Форматированный вывод информации

В операторах *write* и *writeln* имеется возможность записи выражения, определяющего ширину поля вывода.

Формат для данных *целого* и *строкового* типов:

Write (A: m),

где *m* - ширина поля вывода всего числа.

Пример:

Значение A	Выражение	Результат
134	<i>write (A: 6)</i>	<u> </u> 134

Величина *A* выводится в крайние правые позиции поля шириной *6*.

При выводе вещественных значений, кроме ширины поля вывода, через двоеточие надо указывать количество позиций, необходимых для дробной части числа. При форматированном выводе вещественных чисел эти числа выводятся в формате с фиксированной точкой.

Формат для данных *вещественного* типа:

Write (A: m: n),

где *m* - ширина поля вывода всего числа; *n* - количество знаков в дробной части числа ($0 \leq n \leq 24$). Если $n > 24$, то при выводе используется формат с плавающей точкой.

Пример:

Значение A	Выражение	Результат
51.04	<i>write (A: 8:3)</i>	<u> </u> 51.040
-46.78	<i>write (A: 7:2)</i>	<u> </u> -46.78

Если число не помещается в *m* позиций, то поле вывода расширяется до минимально необходимого.

Структурные операторы языка Паскаль

Структурные операторы представляют собой конструкции, построенные из других операторов по строго определенным правилам. Все структурные операторы можно разделить на три группы: *составные, условные, повтора*.

Составной оператор begin...end

Составной оператор представляет собой группу из произвольного числа операторов, отделенных друг от друга точкой с запятой, и ограниченную операторными скобками *begin* и *end*.

Формат:

```

begin
<оператор 1>;
<оператор 2>;
...
<оператор N>;
end ;

```

Составной оператор воспринимается как единое целое и может находиться в любом месте программы, где синтаксис языка допускает наличие оператора.

Условные операторы

Условные операторы предназначены для выбора к исполнению одного из возможных действий (операторов) в зависимости от некоторого условия (при этом одно из действий может отсутствовать). В качестве условий выбора используется значение логического выражения. В **Turbo Pascal** имеются два условных оператора: *if* и *case*.

Условный оператор *if...then...else*

Условный оператор *if* служит для организации процесса вычислений в зависимости от какого-либо логического условия.

Оператор *if* может принимать две формы записи (полную и короткую):

```

if <условие> then <оператор1>
else <оператор2>;
if <условие> then <оператор>;

```

В блок-схемах разветвленные алгоритмы изображаются так, как показано на рис. 1.

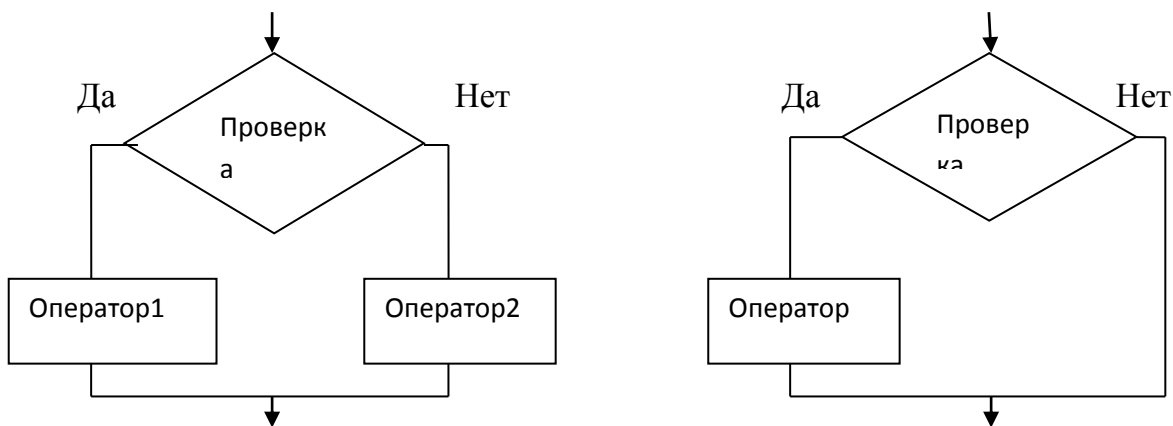


Рис.1. Фрагменты алгоритма разветвленной структуры

Оператор условия *if* выполняется следующим образом. Сначала вычисляется выражение, записанное в условии. В результате его вычисления получается значение булевого типа. В первом случае, если значение выражения есть *True* (истина), выполняется <оператор1>, указанный после слова *then*. Если результат вычисления выражения в условии есть *False* (ложь), то выполняется <оператор2>. Во втором случае — если результат выражения *True*, выполняется <оператор>, если *False* — выполняется оператор, следующий сразу за оператором *if*. По правилам каждая из ветвей может содержать либо один выполняемый оператор,

либо несколько, объединенных в составной оператор. Точка с запятой перед *else* не ставится.

Операторы *if* могут быть вложенными.

Формат записи:

```
if <условие1> then
    if <условие2> then <оператор1>
        else <оператор2>;
```

При вложениях всегда действует правило: служебное слово *else* всегда связывается с ближайшим по тексту служебным словом *if*, которое еще не связано со служебным словом *else*.

Оператор выбора case...of...end

Если один оператор *if* может обеспечить выбор из двух альтернатив, то оператор выбора *case* позволяет сделать выбор из произвольного числа имеющихся вариантов. Он состоит из выражения, называемого селектором (*selection* — выбор альтернативы), и списка параметров, каждому из которых предшествует список констант выбора (список может состоять и из одной константы).

Формат:

```
case <выражение-селектор> of
<список1>: <оператор 1>;
<список2>: <оператор2>;
...
<списокN>: <операторN>;
[else <оператор0>;]
end;
```

Здесь <выражение-селектор> - это выражение порядкового типа, в зависимости от значения которого принимается решение;

<список1>.. <списокN > - значения, с которыми сравнивается значение <селектора>;

<оператор1>, ..., < операторN > - операторы (возможно составные), из которых выполняется тот, с константой которого происходит первое совпадение значения <селектора>, <оператора0> выполнится, если значение ключа не совпадает ни с одной из констант <список1>.. <списокN> .

Ветвь *else* не обязательна и, в отличие от оператора *if*, перед ней можно ставить точку с запятой. Если для нескольких значений <селектора> действия совпадают, то эти константы можно перечислить через запятую перед двоеточием или даже задать диапазон значений (нижняя граница .. верхняя граница).

ЗАДАНИЯ К КОНТРОЛЬНОЙ РАБОТЕ

Контрольная работа выполняется каждым студентом в соответствии со своим индивидуальным вариантом. Номер варианта выбирается по двум последним цифрам шифра зачетной книжки. Если это число больше 30, то номер варианта равен сумме двух последних цифр.

Задание1. Вычислить и вывести на экран значения двух переменных.

Задание2. Вычислить и вывести на экран значение заданной функции, используя условный оператор *if*.

Задание 3. Решить задачу с использованием оператора *case*.

ПРИМЕР ВЫПОЛНЕНИЯ КОНТРОЛЬНОЙ РАБОТЫ

Задание 1. Вычислить и вывести на экран значения переменных y и p , используя расчетные формулы:

$$y = \operatorname{atg}^3(x^2) + \frac{z^2}{a^2+x^2},$$

$$p = \ln(a+x^2) + \sin^2\left(\frac{z}{a}\right)$$

где $a = 0.59$, $z = -4.8$, $x = 2.1$.

Блок-схема алгоритма решения представлена на рис.2. В блоке 3 вводятся исходные данные a , x , z . В блоке 4 вычисляется значение тангенса. В блоках 5 и 6 вычисляются значения функций y и p , а в блоке 7 осуществляется вывод вычисленных значений.

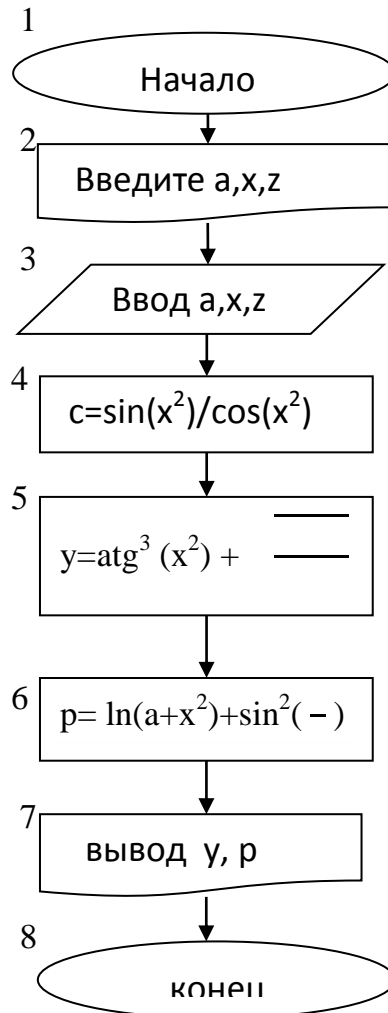


Рис.2. Блок-схема алгоритма решения

Программа, реализующая приведенный алгоритм имеет вид :

```

Program zadanie_1;
Var
  
```



```

a, x, z, c, y, p : real;
Begin
  Writeln ( 'Введите a, x, z ');
  Read (a, x, z);
  c := sin(x*x)/cos(x*x);
  y := a*exp(3*ln(c))*sqr(x)+ sqrt(z*z/(a*a+x*x));
  p := ln(a+x*x)+sqr(sin(z/a));
  Writeln ('y = ', y:8:3, ' p = ', p:8:3);
End.

```

Задание 2. Вычислить и вывести на экран значение заданной функции ax , при $x \leq 0$;

$f x = b \ln x$, при $0 < x \leq 5$;

0, в остальных случаях.

где $a = 15$; $b = 3$.

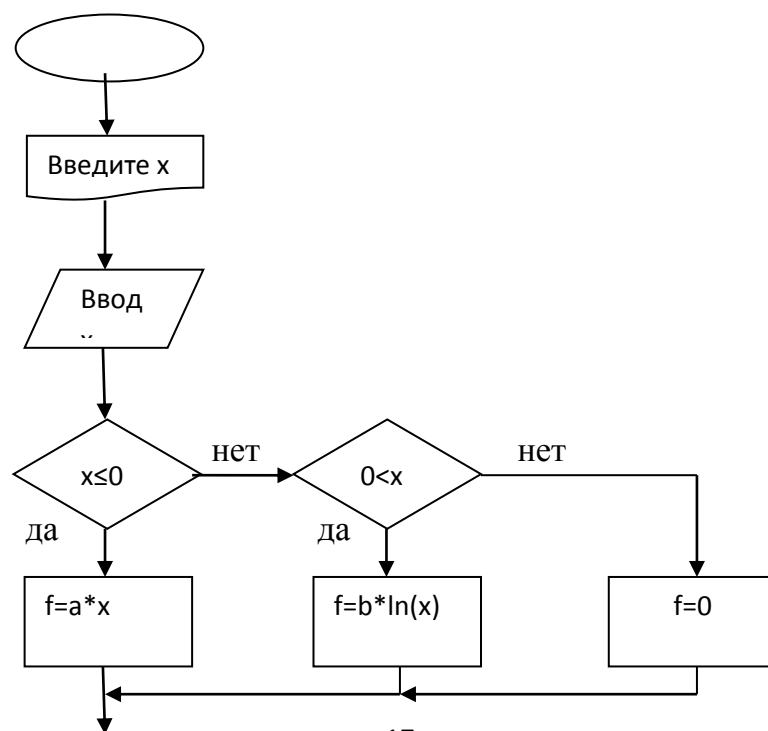
Блок-схема алгоритма представлена на рис.3.

Текст программы:

```

Program zadanie_2;
Const a = 15; b = 3;
Var f, x : real;
Begin
  Writeln ( 'Введите x ');
  Read (x);
  If x <= 0 then f := a*x
    else if(x>0) and (x<=5) then f := b*ln(x)
      else f := 0;
  Writeln ( ' f = ', f:8:3);
End.

```



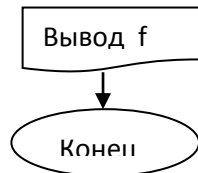


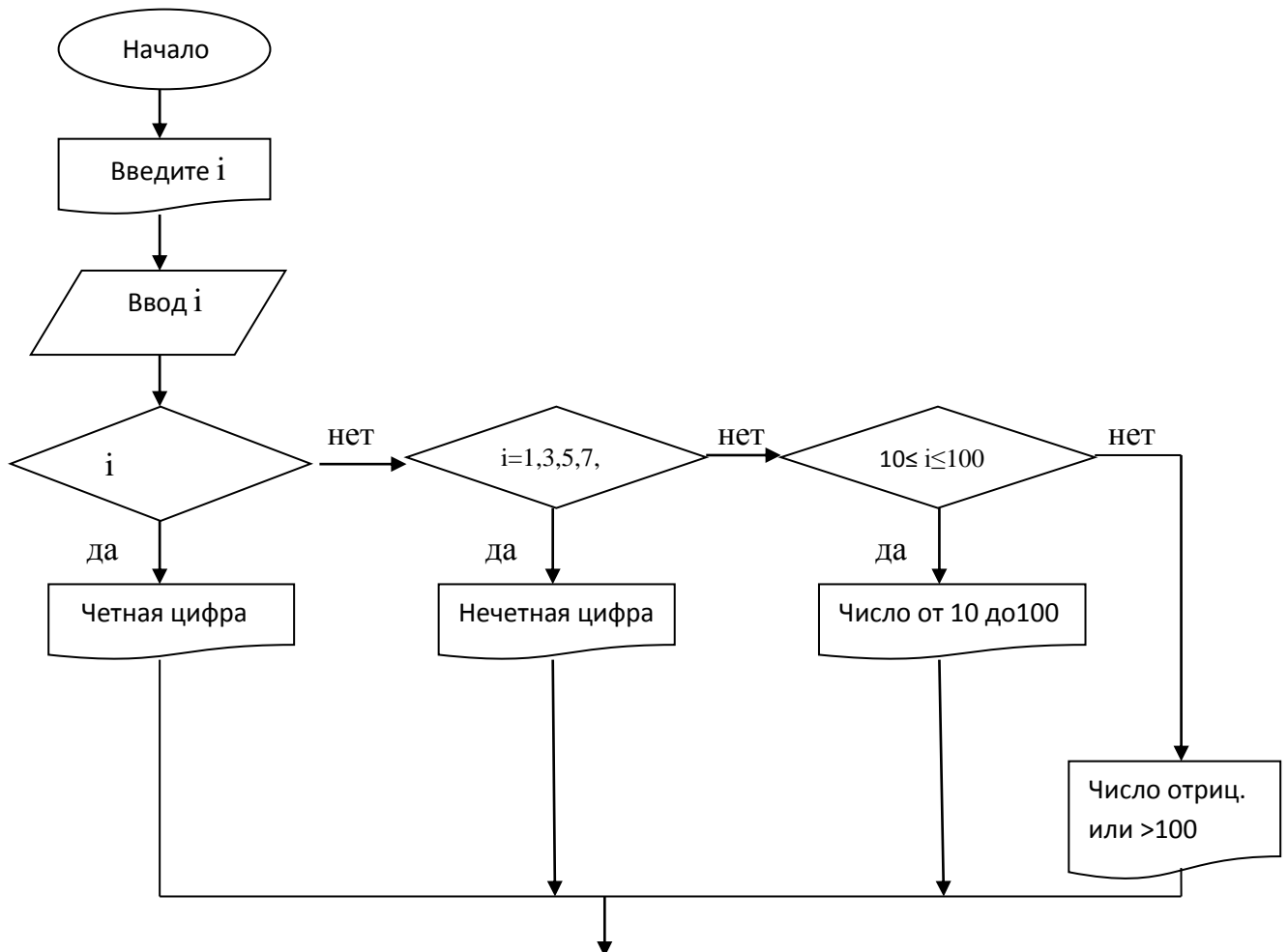
Рис.3. Блок-схема алгоритма

Задание 3. Вводится целое число. Если оно от 0 до 9, то определить четное оно или не четное, а если введенное число попадает в диапазон от 10 до 100 выдать соответствующее сообщение и если не попадает тоже вывести сообщение. Блок-схема алгоритма представлена на рис.4.

Текст программы:

```

Program zadanie_3;
Var i: integer;
Begin
  Write ('Введите целое число i');
  Readln(i);
  Case i of
    0,2,4,6,8 : writeln('Четная цифра');
    1,3,5,7,9 : writeln('Четная цифра');
    10..100 : writeln('Число от 10 до 100');
    else writeln('Число либо отрицательное, либо > 100');
  End;
End.
  
```



Конец

Рис.4. Блок-схема алгоритма

ЗАДАНИЕ 1

Вариант задания	Расчетные формулы	Значения исходных данных
1	$a = \frac{2 \cos x - \pi}{1 + \sin^2 y}$ $b = 1 + \frac{z}{3 + z^2}$	$x=1.426$ $y=-1.22$ $z=3.5$
2	$b = x^y x - \overline{y x}$ $g = y - x \frac{y - z}{1 + y - x^2}$	$x=1.825$ $y=18.225$ $z=-3.298$
3	$S = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!}$ $g = x \sin x^3 + \cos^2 y$	$x=0.335$ $y=0.025$
4	$y = e^{-bt} \sin at + b - \overline{bt + a}$ $S = b \sin at^2 \cos 2t - 1$	$a=-0.5$ $b=1.7$ $t=0.44$
5	$w = \frac{b^2 \sin^3 x + a}{x^2 + b} - \frac{x}{a^2 + b^2}$ $y = \cos^2 x^3 - \frac{x}{a^2 + b^2}$	$a=0.1$ $b=0.2$ $x=1$
6	$S = x^3 t g^2 x + b^2 + \frac{a}{x+b}$ $Q = \frac{bx^2 - a}{e^{ax} - 1}$	$a=16.5$ $b=3.4$ $x=0.61$
7	$R = \frac{x^2 x + 1}{b} - \sin^2 x + a$ $S = \frac{xb}{a} + \cos^2 x + b^3$	$a=0.7$ $b=0.05$ $x=0.5$
8	$y = \sin^3 x^2 + a^2 - \frac{x}{b}$ $z = \frac{x^2}{a} + \cos x + b^3$	$a=1.1$ $b=0.004$ $x=0.2$
9	$f = \overline{mtg t + c \sin t}$ $z = m \cos bt \sin t + c$	$m=2$ $c=-1$ $t=1.2$ $b=0.7$
10	$y = btg^2 x - \frac{a}{\sin^2 x a}$ $d = ae^{-ax} \cos bx a$	$a=3.2$ $b=17.5$ $x=-4.8$
11	$f = \ln a + x^2 + \frac{\sin^2 x}{x + a} b$ $z = e^{-cx} \frac{x + \overline{x + a}}{x - x - a}$	$a=10.2$ $b=9.2$ $x=2.2$ $c=0.5$
12	$y = \frac{a^{2x} + xb^{-x} \cos a + b}{x + 1}$ $R = \frac{b^2 \sin^3 x + a}{x^2 + b} - \frac{x}{a^2 + b^2}$	$a=0.3$ $b=0.9$ $x=0.61$
13	$z = \frac{ax \sin 2x + e^{-2x} x + b}{x}$ $w = \cos^2 x^3 - \frac{x}{a^2 + b^2}$	$a=0.5$ $b=3.1$ $x=1.4$
14	$U = \frac{a^2 x + e^{-x} \cos bx}{bx - e^{-x} \sin bx + 1}$	$a=0.5$ $b=2.9$

	$f = e^{2x} \ln a + x - b^{3x} \ln b - x$	x=0.3
15	$z = \frac{\sin x}{m^2 \sin^2 x} - cm \ln mx$ $S = e^{-ax} \frac{x+1}{x+1} + e^{-bx} \frac{x+1}{x+1.5}$	a=0.5 c=2.1 b=1.8 x=1.7 m=0.7
16	$m = \frac{n+k}{l} + tg^3 n - k$ $H = \sin k^2 - e^{-nk} \frac{k^2+l^2}{k^2+l^2}$	n=0.5 k=0.3 l=1
17	$c = 1 - x + \frac{2x^2}{2!} + \frac{3x^3}{3!} + \frac{4x^4}{4!}$ $R = e^{yz} \frac{\sin^2 x^3 + \cos^3 x^2}{x^3 + \cos^3 x^2}$	x=1.18 y=0.75 z=-1.5
18	$R = a \frac{btg^3c + a^4 - lg 2c}{\sin 2c}$ $H = \frac{\sin 2c}{b^2 - a c} + e^{2c}$	a=0.5 b=0.3 c=1.23
19	$Q = \frac{a^3c + \cos c - \pi 4}{1 8 + \sin^3 d}$ $R = e^{-da} \frac{a + c - d}{a - c + d}$	a=0.72 c=1.82 d=1.23
20	$U = \frac{\cos f}{4 + g^2tg f^2} - ab \ln fg$ $H = e^{2f} \frac{1-f}{1-f} + e^{2g} \frac{b+a}{b+a}$	f=0.33 g=2.33 a=-0.3 b=4.75
21	$S = \sin \pi - 2x + \frac{\pi}{6} - \frac{1}{a^3}$ $y = \frac{x^5 - 0.3 x - 1}{\pi - 2x^2}$	a=0.25 x=0.1
22	$a = 2^x \cos xt + 1 - \frac{1}{xt+1}$ $b = xtg t^2 \cos 2t + 2t$	t=0.3 x=1.22
23	$y = \frac{a^2 \sin^2 x}{x^3 e^{a^2}} - 2 \cos^2 x$ $z = tg 2a + e^{-3x} \frac{x-2a^3}{x-2a^3}$	a=0.2 x=0.5
24	$f = 2^x y - \frac{1}{x y}$ $g = \cos x y + \sin^2 e^x$	x=0.4 y=0.15
25	$S = \frac{\cos^3 x}{m \cos^2 2x} + \ln mx$ $U = \frac{mx}{x^2} + \cos mx + 1^2$	x=1.2 m=0.7
26	$f = e^{-ax} \ln ax + 1 - tg^2 ax + 1$ $z = \frac{ax+1}{ax+1} - ax^3 \sin ax + 1^2$	a=0.12 x=1.5
27	$S = \frac{wtg^3 w + t \sin w}{t}$ $m = \cos^2 t^3 - \frac{t}{w^2 + t^2}$	w=0.72 t=-1.5
28	$Q = \sin^2 x^3 - x^3 \frac{y^2 + x^2}{y^2 + x^2}$ $S = e^y \frac{\sin^3 x^3}{2 + y^2tg y^2}$	x=1 y=0.6
29	$U = x^2lg x^3 + a + \cos^2 x^3 + a$ $w = \frac{x^2}{a} + e^{ax}tg^3 x^3 + a$	a=0.1 x=1.6
30	$y = b^2 \sin bt \cos t + b^2 - 1$ $z = \frac{bttg b^2 - 1}{b^2 - 1} + 2b^3$	b=0.5 t=0.3

ЗАДАНИЕ 2

№ вариант	функция	условие	исх. данные
1	$y = \frac{at^2 \ln t}{1 + e^{at} \cos bt}$	$1 \leq t \leq 2$ $t < 1$ $t > 2$	$a = -0.5$ $b = 2$
2	$y = \frac{\pi x^2 - \frac{7}{x^2}}{ax^3 + 7 \frac{x}{x}} \cdot \frac{1}{\lg x + 7 \frac{x}{x}}$	$x < 1.3$ $x = 1.3$ $x > 1.3$	$a = 1.5$
3	$\omega = \frac{ax^2 + bx + c}{\frac{a}{x} + x^2 + 1} \cdot \frac{a + bx}{x^2 + 1}$	$x < 1.2$ $x = 1.2$ $x > 1.2$	$a = 2.8$ $b = -0.3$ $c = 4$
4	$Q = \frac{\pi x^2 - \frac{7}{x^2}}{ax^2 + 7 \frac{x}{x}} \cdot \frac{1}{\ln x + 7 \frac{x}{x + a}}$	$x < 1.4$ $x = 1.4$ $x > 1.4$	$a = 1.65$
5	$y = \frac{1.5 \cos^2 x + 1.8ax}{x - 2^2 + 6} \cdot \frac{1}{3 \operatorname{tg} x}$	$x < 1$ $x = 1$ $1 < x < 2$ $x > 2$	$a = 2.3$
6	$\omega = \frac{x^3 \sqrt{x - a}}{x \sin ax} \cdot \frac{1}{e^{-ax} \cos ax}$	$x < a$ $x = a$ $x > a$	$a = 2.5$
7	$Q = \frac{bx - \lg bx}{1 + bx + \lg bx}$	$bx < 1$ $bx = 1$ $bx > 1$	$b = 1.5$
8	$y = \frac{\sin x \lg x}{\cos^2 x}$	$x > 3.5$ $x \leq 3.5$	$---$
9	$f = \frac{\lg x + 1}{\sin^2 \frac{ax}{x}}$	$x > 1$ $x \leq 1$	$a = 20.3$
10	$z = \frac{\frac{\ln x^3 + x^2}{x + t}}{x + t + 1} \cdot \frac{1}{\cos x + t \sin^2 x}$	$x < 0.5$ $x = 0.5$ $x > 0.5$	$t = 2.2$
11	$y = \frac{\frac{a + b}{e^x + \cos x}}{\frac{a + b}{x + 1}} \cdot \frac{1}{e^x + \sin x}$	$x < 2.8$ $2.8 \leq x < 6$ $x \geq 6$	$a = 2.6$ $b = -0.39$
12	$y = \frac{\operatorname{alg} x + \sqrt[3]{x}}{2a \cos x + 3x^2}$	$x > 1$ $x \leq 1$	$a = 0.9$

13	$\omega = \frac{\frac{a}{i} + bi^2 + c}{i + ai + bi^3}$	$i < 4$ $4 \leq i \leq 6$ $i > 6$	$a=2.1$ $b=1.8$ $c=-20.5$
14	$z = \frac{a \sin \frac{i^2+1}{n}}{\cos i + \frac{1}{n}}$	$\sin \frac{i^2+1}{n} \geq 0$ $\sin \frac{i^2+1}{n} < 0$	$a=0.3$ $n=10$
15	$z = \frac{\sin x}{\cos x \operatorname{tg} x}$	$x \leq 2$ $2 < x < 5$ $x \geq 5$	---
16	$z = \frac{\sin x + 1}{-x}$	$x > 0$ $x < 0$ $x = 0$	---
17	$Z = \frac{x^3}{3}$	$y \neq 0$ $y = 0$	$y = x - 0.5$
18	$Z = \frac{0}{1 - x}$	$x < -1$ $-1 \leq x \leq 1$ $x > 1$	---
19	$w = \frac{at^2 + b \sin t + 1}{at + b}$	$t < 0.1$ $t = 0.1$ $t > 0.1$	$a=2.5$ $b=0.4$
20	$z = \frac{c \cos \frac{i^2+1}{n}}{\sin \frac{i^2+1}{n}}$	$\cos \frac{i^2+1}{n} \geq 0$ $\cos \frac{i^2+1}{n} < 0$	$c=0.5$ $n=10$
21	$y = \frac{x^2 + e^x}{2 + \pi \sin^2 x + \ln x}$	$x < 4$ $x = 4$ $x > 4$	---
22	$y = \frac{b2^x + x}{x + 1^2}$	$x < 0$ $x \geq 0$	$b = 0.25$
23	$y = \frac{c \ln x^2}{3^{x+1} x + 1^3}$	$x < 0$ $x \geq 0$	$c = 0.3$
24	$y = \frac{x^2 + 2^x}{x + \pi \sin^2 x + \operatorname{tg} x}$	$x < 2$ $x = 2$ $x > 2$	---
25	$y = \frac{x + 1^3}{3x + x + 1 \cos^2 x + \sin x^2}$	$x < 1$ $x = 1$ $x > 1$	---
26	$y = \frac{0}{x^3 - x}$	$x \leq 0$ $0 \leq x \leq 1$ $x > 1$	---
27	$y = \frac{2x^3 - 7x^2 + 5x + 3}{x + \ln \cos 2x - 1}$	$x < 3$ $x \geq 3$	---
28	$y = \frac{\cos bx - 1}{5x^2 - b}$	$x \leq 0$ $0 < x \leq 1$ $x > 1$	$b = 2$
29	$y = \frac{\ln \cos 5x - 2x + 1}{0.35 \operatorname{tg} 0.1x - 2e^{-0.1x}}$	$x \leq -\pi$ $x > -\pi$	---

30	$y = \begin{cases} 2x^3 + 5x^2 \\ x^2 \\ 7e^{-x} \end{cases}$	$\begin{cases} x \leq -1 \\ -1 < x \leq 1 \\ x > 1 \end{cases}$	---
----	---	---	-----

ЗАДАНИЕ 3

- В зависимости от того, введена открытая скобка или закрытая, напечатать "открытая круглая скобка" или "закрытая фигурная скобка" (учитывать круглые, квадратные, фигурные скобки).
- В зависимости от введенного символа L, S, V программа должна вычислять длину окружности; площадь круга; объем цилиндра. Радиус окружности и высоту цилиндра ввести с клавиатуры.
- Определите, является ли введенная буква русского алфавита гласной.
- Напишите программу, которая по введенному числу из промежутка $0..24$, определяет время суток.
- Пусть имеются данные об автомобилях четырех моделей. Как входное данное ввести номер модели и получить характеристики: год выпуска и цену.
- Ввести первую букву названия страны. Вывести, название ее столицы.
- Ввести телефонный код города. Вывести название города.
- Дано целое число в диапазоне $1-7$. Вывести строку - название дня недели, соответствующее данному числу (1 - «понедельник», 2 - «вторник» и т. д.).
- Вычислить, значение функции по одной из формул ($x \geq 0$):
$$y = \begin{cases} x + a, & x < 10; \\ x - a, & 10 \leq x \leq 20; \\ x^2, & 20 < x < 30; \\ a^2 + x, & 30 \leq x < 60; \\ x^2 + a^2, & x \geq 60. \end{cases}$$
- Ввести число от 0 до 5 . Вывести его название на английском (или другом) языке.
- Пусть время суток обозначено номерами: 1 - утро, 2 - день, 3 - вечер. Билет в кино на утренний сеанс стоит 50 р., на дневной сеанс - 70 р., а на вечерний - 100 р. Какова выручка кинотеатра в одно из время суток, если в кино пришло n посетителей.
- Ввести номер телефона одного из четырех друзей. Вывести его имя.
- Элементы окружности пронумерованы следующим образом: 1 - радиус R , 2 - диаметр $D = 2R$, 3 - длина $L = 2\pi R$, 4 - площадь круга $S = \pi R^2$. Дан номер одного из этих элементов и его значение. Вывести значения остальных элементов данной окружности.
- Дано целое число K . Вывести строку-описание оценки, соответствующей числу K (1 - «плохо», 2 - «неудовлетворительно», 3 - «удовлетворительно», 4 - «хорошо», 5 - «отлично»). Если K не лежит в диапазоне $1-5$, не лежит в диапазоне $1-5$, то вывести строку «ошибка».
- Дан номер месяца - целое число в диапазоне $1-12$ (1 - январь, 2 - февраль и т. д.). Вывести название соответствующего времени года («зима», «весна», «лето», «осень»).
- Пусть время суток обозначено номерами: 1 - утро, 2 - день, 3 - вечер. Билет в кино на утренний сеанс стоит (детский - 25р., взрослый - 70р.), на дневной сеанс (детский - 50р., взрослый - 100р.), а на вечерний (детский - 70 р., взрослый - 100р.).

Какова выручка кинотеатра в одно из время суток, если в кино пришло n детей и m взрослых.

17. Робот может перемещаться в четырех направлениях («С»- север, «З»- запад, «Ю» — юг, «В»- восток) и принимать три цифровые команды: 0 — продолжать движение, 1 -поворот налево, -1 - поворот направо. Дан символ С — исходное направление робота и целое число N - посланная ему команда. Вывести направление робота после выполнения полученной команды.

18. Дан номер месяца— целое число в диапазоне 1-12 (1 — январь, 2— февраль и т. д.). Определить количество дней в этом месяце для не високосного года.

19. Арифметические действия нал числами пронумерованы следующим образом: 1— сложение, 2 - вычитание, 3— умножение, 4 —деление. Дан номер действия N (целое число в диапазоне 1-4) и вещественные числа A и B (B не равно 0), Выполнить над числами указанное действие и вывести результат.

20. Элементы равнобедренного прямоугольного треугольника пронумерованы следующим образом: 1 - катет a , 2 — гипотенуза $c = a \cdot (2)^{1/2}$, 3— высота h , опущенная на гипотенузу ($h = c/2$), 4 — площадь $S = c \cdot h/2$. Дан номер одного из этих элементов и его значение. Вывести значения остальных элементов данного треугольника.

21. Единицы длины пронумерованы следующим образом: 1- дециметр, 2 - километр, 3 - метр, 4 - миллиметр, 5 - сантиметр. Дан номер единицы длины (целое число в диапазоне 1- 5) и длина отрезка в этих единицах (вещественное число). Найти длину отрезка в метрах.

22. Элементы равностороннего треугольника пронумерованы следующим образом: 1 -сторона a , 2 -радиус R_1 , вписанной окружности ($R_1 = a \cdot (3)^{1/2} / 6$), 3 - радиус R_2 описанной окружности ($R_2 = 2 \cdot R_1$), 4 -площадь $S = a^2 \cdot (3)^{1/2} / 4$. Дан номер одного из этих элементов и его значение. Вывести значения остальных элементов данного треугольника,

23. Единицы массы пронумерованы следующим образом: 1- килограмм, 2- миллиграмм, 3 - грамм, 4- тонна, 5- центнер. Дан номер единицы массы (целое число в диапазоне 1 -5) и масса тела в этих единицах (вещественное число). Найти массу тела в килограммах.

24. Пусть населенные пункты обозначены номерами от 1 до 8.Стоимость одного билета к пункту k определяется так:

$$\text{Цена} = \begin{matrix} 22, & k = 1, \\ 25, & k = 2, 3, 4 \\ 30, & k = 5, 6, \\ 35, & k = 7, 8. \end{matrix}$$

Сколько стоит m билетов к населенному пункту, номер которого пользователь должен ввести с клавиатуры?

25. Вводится целое положительное число. Определить четное оно или нет.

26. Для целого числа N от 20 до 30 напечатать фразу "Мне N лет", учитывая при этом, что при некоторых значениях N слово "лет" надо заменить на слово "год" или "года".

27. Напечатать слово **СТУДЕНТ** в заданном падеже (**И** - именительный, **Р** - родительный, **Д** - дательный, **В** - винительный, **Т** - творительный, **П** - предложный) единственного числа.

28 . Для целого числа N от 1 до 8 напечатать фразу "Мы нашли N грибов", согласовав окончание слова "гриб".

29. Вычислить значение функции по одной из формул ($x \geq 0$):

$$y = \begin{cases} x/c, & x \leq 5; \\ x + c, & 5 < x \leq 15 \\ \bar{x} + c, & 15 < x < 25; \\ \bar{x}/c, & x > 25. \end{cases}$$

30. Мастям игральных карт присвоены порядковые номера: 1 — пики, 2 — трефы, 3 — бубны, 4 — червы. Достоинству карт, старших десятки, присвоены номера: 11 — валет, 12 — дама, 13 — король, 14 — туз. Даны два целых числа: N — достоинство ($6 \leq N \leq 14$) и M — масть карты ($1 \leq M \leq 4$). Вывести название соответствующей карты вида «шестерка бубен», «дама червей», «туз треф» и т. п.

СОДЕРЖАНИЕ И ОФОРМЛЕНИЕ КОНТРОЛЬНОЙ РАБОТЫ

Пояснительная записка к контрольной работе оформляется в текстовом редакторе **MS Word** в формате **A4** и должна содержать:

1. Цель работы.
2. Задание.
3. Графические схемы алгоритмов с учетом требований ГОСТ 19.701-90.
4. Программы на алгоритмическом языке **Turbo Pascal**.
5. Результаты выполнения работы.