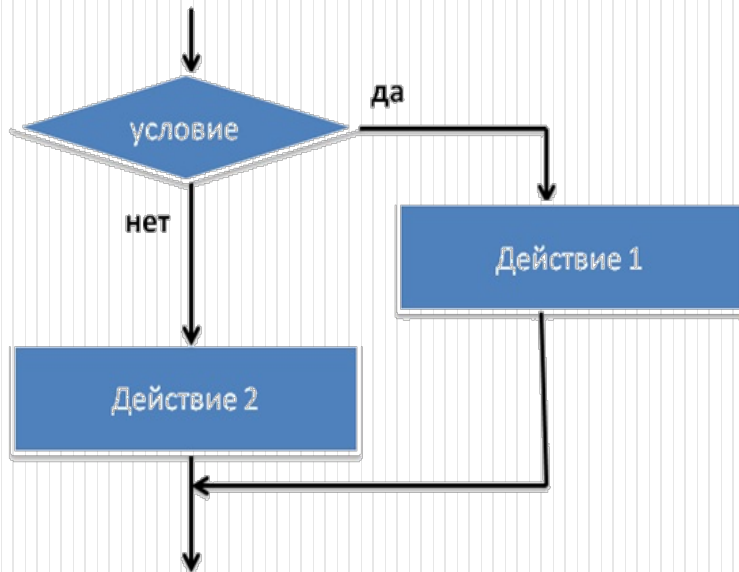




ПРОГРАММИРОВАНИЕ АЛГОРИТМОВ ВЕТВЛЕНИЯ



Автор: преподаватель
Романова Юлия Евгеньевна
ГОУ ВО МО «ГГТУ»

Промышленно-экономический колледж
Московская область, г. Орехово-Зуево

План урока:

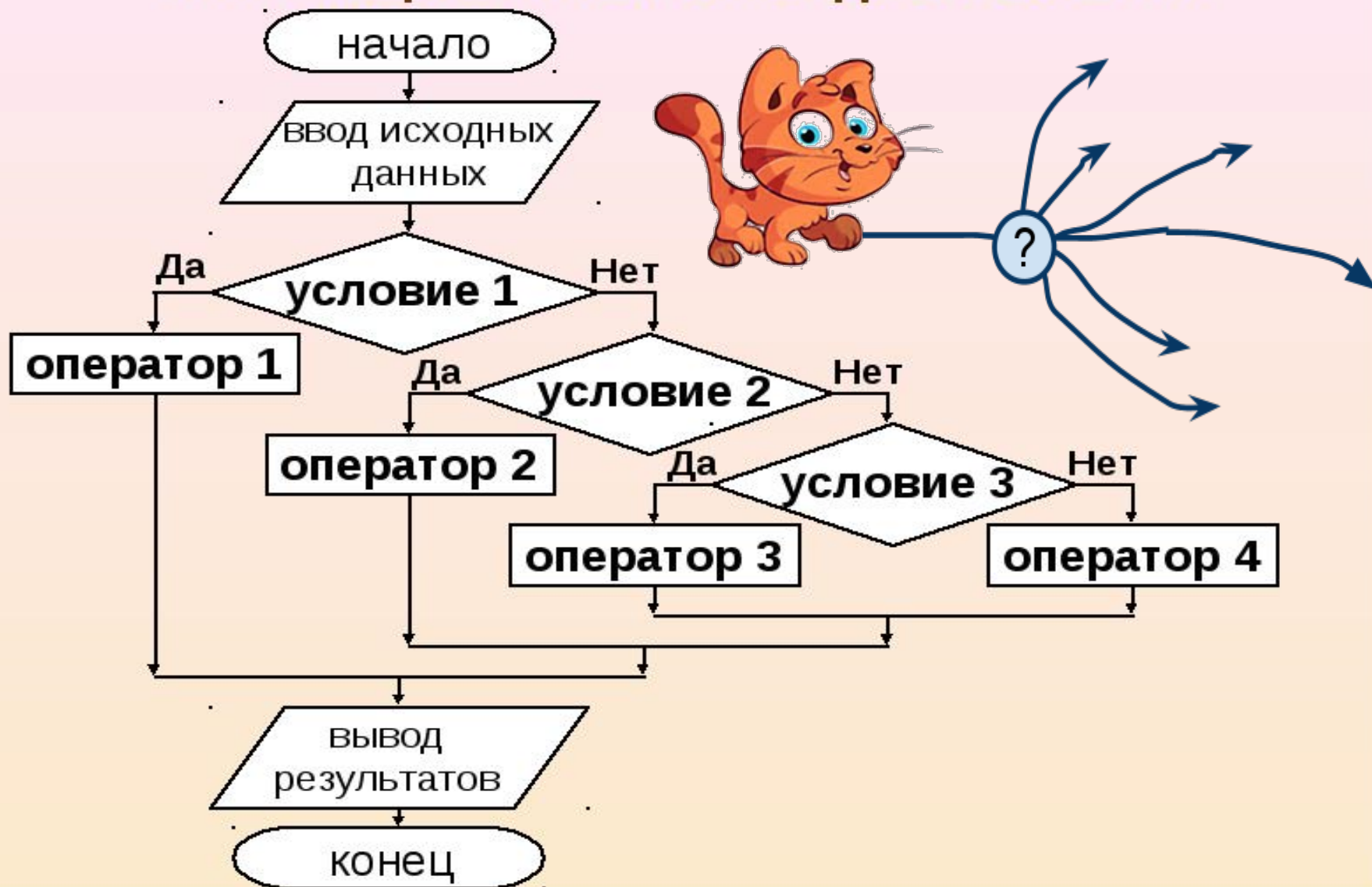
- Алгоритм ветвления
- Логические выражения и операции отношения
- Логические операции и порядок выполнения
- Условный оператор IF
- Оператор выбора CASE

Литература:

И. Г. Семакин. Основы программирования, с. 57 – 64.

Алгоритм ветвления

Многовариантная команда ветвления



Логические выражения

- **Выражением отношения** называется словосочетание языка, в котором два выражения связаны знаком операции отношения.
- Выражение отношения определяет истинность или ложность результата.
- Операции отношения выполняют сравнение двух операндов и определяют, истинно значение выражения или ложно.
- Сравнимые логические значения обозначаются служебными словами **False** (ложь) и **True** (истина), а идентификатор данных логического типа – **Boolean**.
- В языке Pascal логические операции отношения важны при написании программ разветвляющихся алгоритмов.

Операции отношения

<i>Операция</i>	<i>Название</i>	<i>Выражение</i>	<i>Результат</i>
=	Равно	$A=B$	True, если A равно B
<>	Не равно	$A<>B$	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	$A>=B$	True, если A больше или равно B
<=	Меньше или равно	$A<=B$	True, если A меньше или равно B
In	Принадлежность	$A \text{ in } M$	True, если A находится в списке M

Логические операции

<i>Логическая операция</i>	<i>Название операции</i>	<i>Запись</i>	<i>Результат операции</i>
Not	Не	Not F1	Значение, противоположное F1
And	И	F1 and F2	Сложное условие выполняется, если выполняются условия F1 и F2
Or	Или	F1 or F2	Сложное условие выполняется, если выполняется хотя бы одно из условий F1 или F2
Xor	Исключающее «Или»	F1 xor F2	Сложное условие выполняется, если F1 не равно F2

Сложное условие

- **Условия** бывают **простые** и **сложные**.
- **Сложное условие** – это условие, построенное из нескольких простых условий с помощью логических операций.
- **Правило 1:** При записи сложного условия простые условия заключаются в скобки!
Пример: записать условие $3 \leq x < 8$
 $(x \geq 3) \text{ and } (x < 8)$
- **Правило 2:** Перед **Else (иначе)** точка с запятой не ставится
- **Правило 3:** Если вместо одного оператора необходимо выполнить несколько, используются операторные скобки **Begin <операторы> end;**

Порядок выполнения операций

Для определения старшинства операций имеются четыре основных правила:

- Операнд, находящийся между двумя операциями с разными приоритетами, связывается с операцией, имеющей более высокий приоритет.
- Операция, находящаяся между двумя операциями с равными приоритетами, связывается с той операцией, которая находится слева.
- Выражение, заключенное в скобки, перед выполнением вычисляется как отдельный операнд.
- Операции с равным приоритетом производятся слева на право с возможным регулированием порядка выполнения скобками.

Порядок выполнения операций

<i>Операция</i>	<i>Приоритет</i>	<i>Вид операции</i>
@, not	Первый	Унарная операция
*, /, div, mod, and	Второй	Операция типа умножения
+, -, or, xor	Третий	Операция типа сложения
=, <>, <, >, <=, >=, in	Четвертый	Операция отношений

C := D or B and not (K and D)

4 3 2 1

Условный оператор IF

- Условный оператор позволяет проверить некоторое условие и в зависимости от результатов проверки выполнить то или иное действие.

- **Полное ветвление**

**IF(если) <условие> THEN(то) <оператор 1>
ELSE(иначе) <оператор 2>;**

- **Неполное ветвление**

IF(если) <условие> THEN(то) <оператор >;

Задача

- Составить программу, которая вычисляет частное двух целых чисел. В связи с тем, что на ноль делить нельзя, организуем контроль ввода данных. Для контроля вводимых значений делителя используем оператор условного перехода `If...then...else`.

```
Program Control;  
Uses CRT;  
Var A, B: Integer; R: Real;  
Begin  
ClrScr;  
Writeln('Введите значение делимого A');  
  Readln(A);  
Writeln('Введите значение делителя B');  
  Readln(B);  
If B=0 {контроль ввода} then  
  Writeln('на ноль делить нельзя!')  
  {условие выполнено}  
Else {условие не выполнено}  
  Begin {начало составного  
    оператора}  
R := A/B;  
Writeln('частное чисел =', R:5:2);  
End; {конец составного оператора}  
Readln;  
End.
```

Вложенный условный оператор

- **Вложенный условный оператор** – если используется проверка нескольких условий и «условный оператор» **if** используется несколько раз, переход между условиями связывается оператором «иначе» **else**.

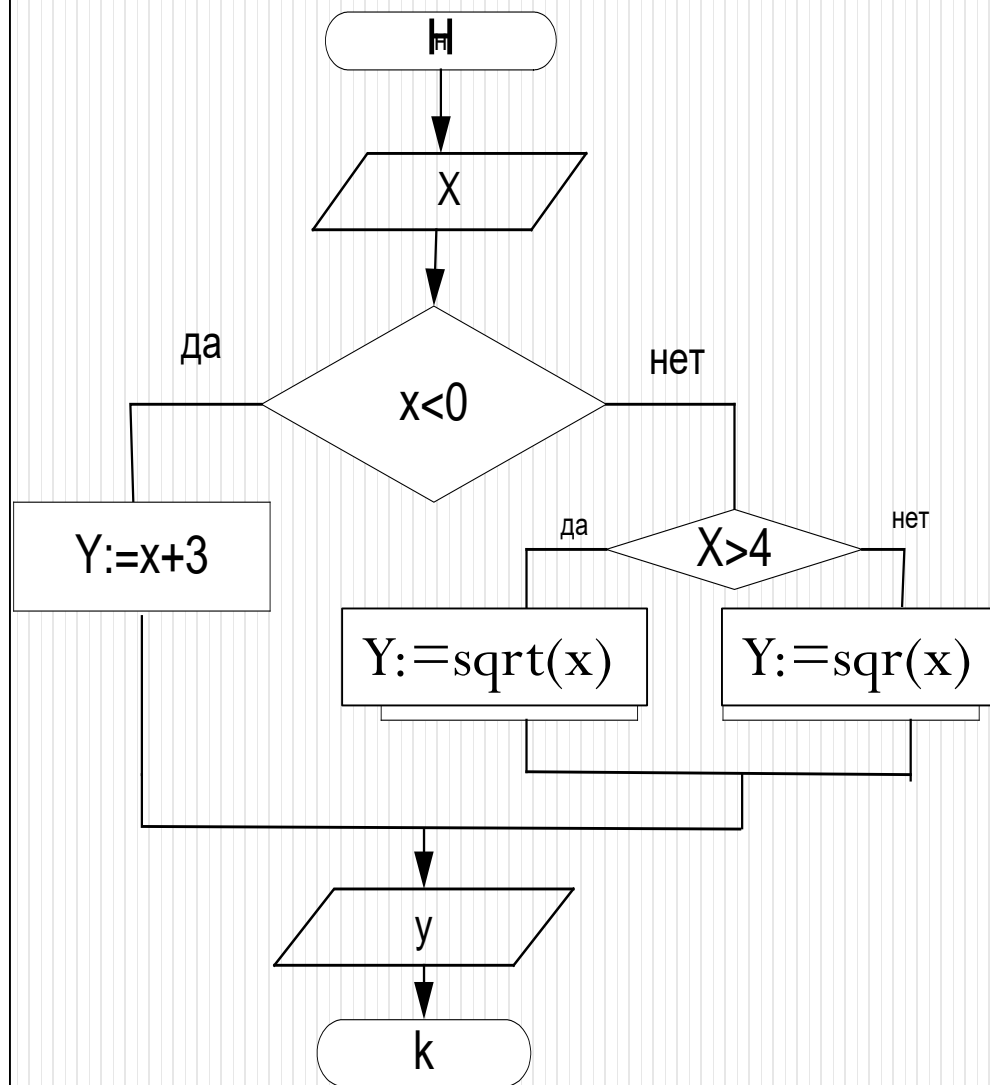
- **Пример:**

Создайте программу, определяющую, какой сейчас сезон по введённой температуре воздуха:

- *от 0 до -30 °C – зима;*
- *от +20 до +40 °C – лето;*
- *иначе (от +1 до +19 °C) - весна или осень*

Задача на вложенный условный оператор

```
Program primer2;  
Uses crt;  
Var x, y: Real;  
Begin  
  Clrscr;  
  Writeln('ВВЕДИТЕ x');  
  Readln(x);  
  If x<0 then Y:=x+3  
else  
  If x>4 then Y:=sqrt(x)  
else y:=sqr(x);  
  Writeln('Y=',y:6:2);  
  Readln;  
End.
```



Составной условный оператор

Если по условию задачи необходимо использовать несколько операторов или выражений, при выполнении или невыполнении условия применяют составной оператор.

Запись сложного условия:

```
If (A>3) and (A<7)  
then  
writeln ('точка на отрезке  
(3,7)');
```

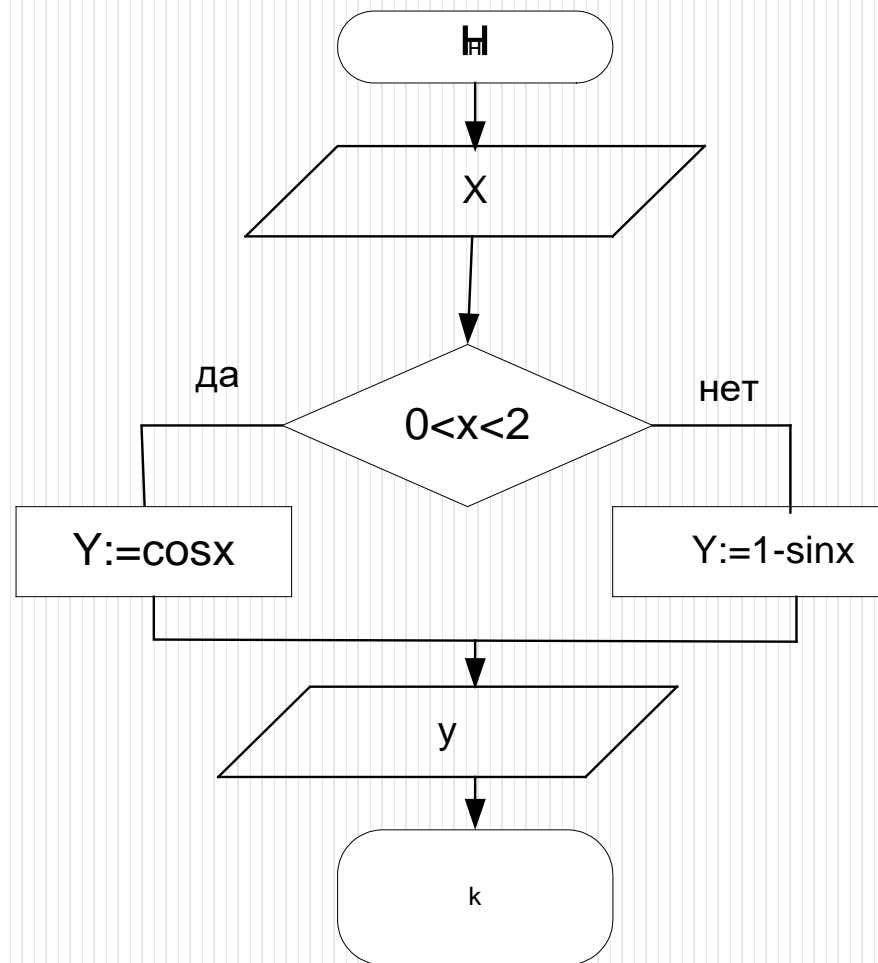
Запись составного оператора:

```
If (a<c) and (a>3) then  
Begin  
a:=c; k:=k+1; end;  
else  
Begin  
c:=a-c; m:=m+1; end;  
If (a>c) or (a<7) then  
else  
Begin  
X:=X+1; Y:=abs(A); end;  
else A:=A+1; end;
```

Составной условный оператор IF

```
Program primer1;  
Uses Crt;  
Var x, y: Real;  
Begin  
  Clrscr;  
  Writeln('введите x'); Readln(x);  
  If (x>0) and (x<2) then  
    Begin  
      Y:=sqr(cos(x));  
    Writeln(y:6:2);  
    End;  
  Else  
    Begin  
      Y:=1-sqr(sin(x));  
    Writeln(y:6:2);  
    End;  
  Readln;  
End.
```

Алгоритм расчета функции



Оператор выбора условия CASE

- Условный оператор IF при выполнении программы позволяет выбрать одно из двух возможных действий. Если же необходимо выполнить много взаимоисключающих проверок, то удобнее воспользоваться оператором выбора нужного варианта CASE.
- Выполнение оператора варианта начинается с вычисления селектора (выражения скалярного типа). Затем для исполнения выбирается оператор, одна из меток которого совпадает с полученным значением, затем управление передается оператору, следующему за оператором варианта. Если же значение селектора не совпало ни с одной из меток, то выполняется оператор, стоящий после служебного слова ELSE.

Структура оператора выбора

```
Case <выражение-селектор> of  
<список меток варианта 1> : <оператор 1>;  
<список меток варианта 2> : <оператор 2>;  
...  
<список меток варианта N> : <оператор N>;  
Else <оператор N+1>;  
End;
```

Пример записи

- Полная форма

Case S of

C1 : <оператор1>;

C2 : < оператор 2>;

...

CN : < оператор N>;

Else < оператор N+1>;

end;

- Неполная форма

Case S of

C1 : <оператор1>;

C2 : < оператор 2>;

...

CN : < оператор N>;

end;

Типичные формы записи оператора Case

<i>Селектор</i>	<i>Форма записи</i>
Интервального типа	Case k of 1..10 : writeln('число', k:4, 'в диапазоне 1-10'); 11..20 : writeln('число', k:4, 'в диапазоне 11-20'); 21..30 : writeln('число', k:4, 'в диапазоне 21-30'); else writeln('число', k:4, 'вне пределов контроля') end;
Целочисленного типа	Case k of 1 : Z := k+10; 2 : Z := k+100; 3 : Z := k+1000; end;
Перечисляемого пользовательского типа	Case N of 'W' : writeln('winter'); 'S' : writeln('string'); 'M' : writeln('summer'); 'A' : writeln('autumn'); End;

Задача 1:

Программа, которая по введенному номеру выводит на экран его название

```
Program mes;
```

```
Use CRT;
```

```
Var M : byte;
```

```
Begin
```

```
ClrScr;
```

```
Writeln ('введите оценку-число:');
```

```
Readln (M);
```

```
Case M of {вычисление значения селектора и выбор}
```

```
  2 : writeln('неудовлетворительно');
```

```
  3 : writeln('удовлетворительно');
```

```
  4 : writeln('хорошо');
```

```
  Else writeln ('отлично');
```

```
  end;
```

```
Readln;
```

```
End.
```

Задача 2:

Программа получения значения R , в зависимости от выбора варианта S и ввода двух любых целых чисел A и B .

```
Program primer;  
Use CRT;  
Var S : String; A, B: Integer; R: Real;  
Begin  
  ClrScr;  
  Writeln ('введите A='); Readln (A);  
  Writeln ('введите B='); Readln (B);  
  Writeln ('введите S='); Readln (S);  
  Writeln;  
  Case S of           {вычисление значения селектора и выбор}  
    '+' : R:=A+B;  
    '-' : R:=A-B;  
    '*' : R:=A*B;  
    '/' : R:=A/B;  
  Else Writeln('неверный знак операции');  
  end;  
  Writeln('R=', R:4:1);  
  Readln;  
End.
```

Домашнее задание

Выполнить задания

с.315 п.6.2.2 №1,9

с.319 №13



Успехов в освоении данной темы!