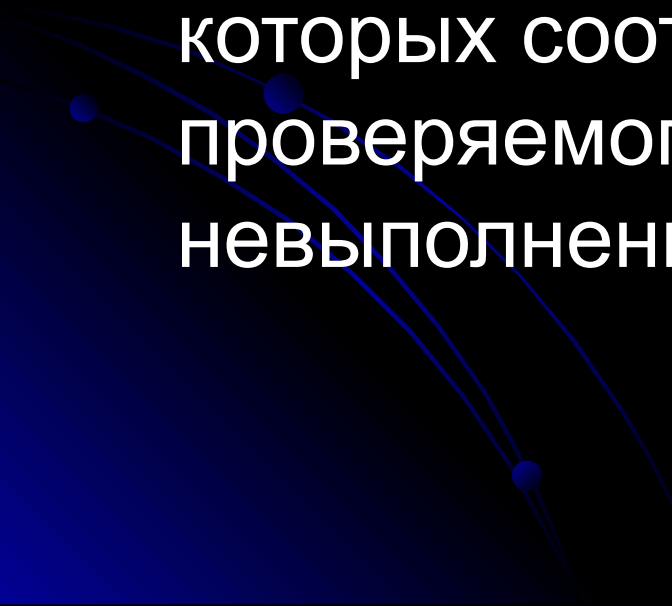
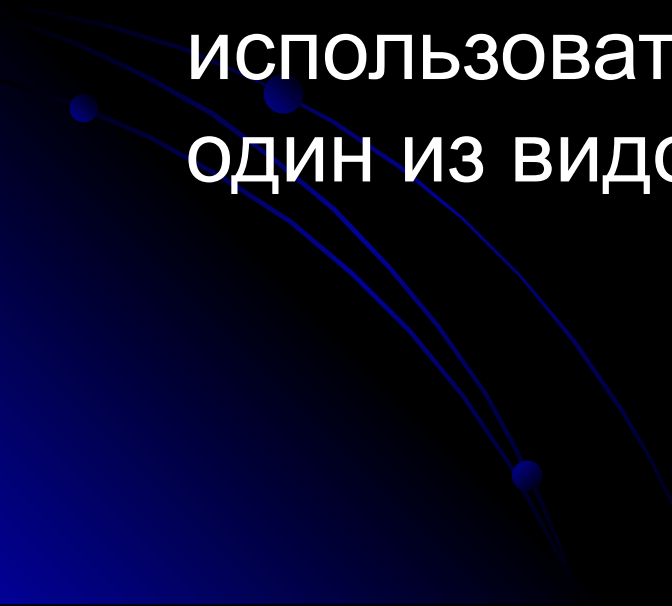
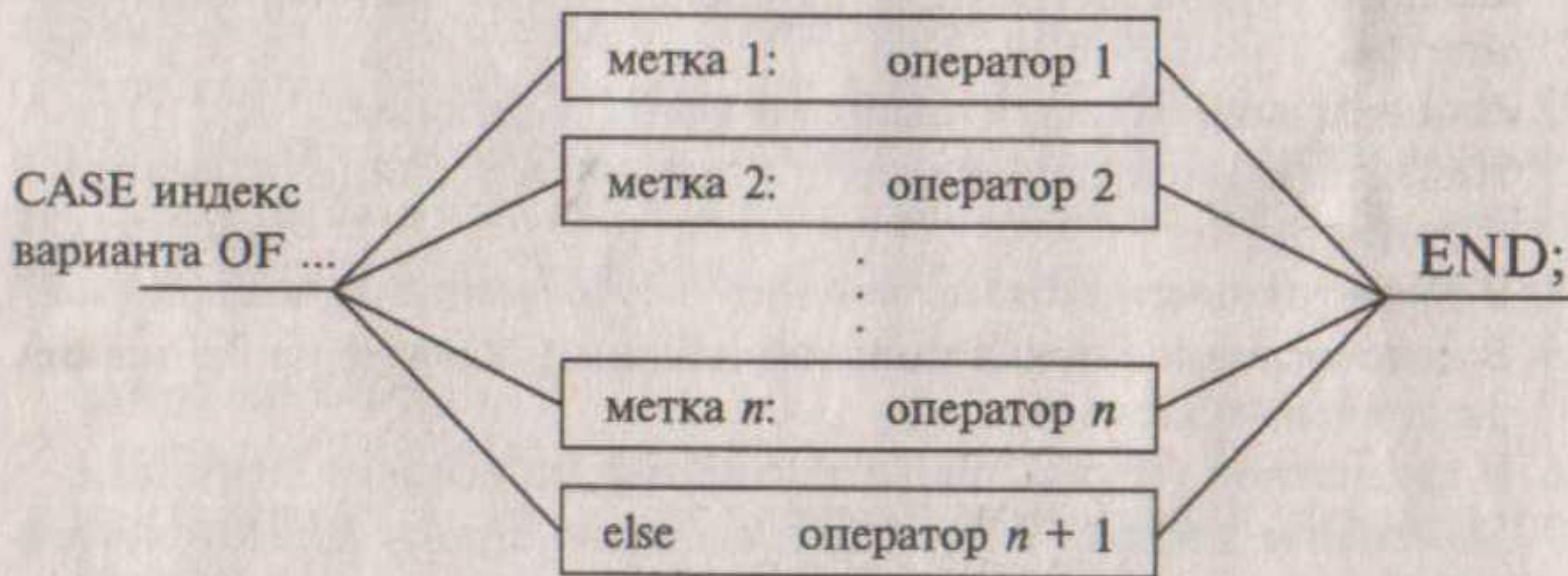


Условный оператор. Сокращенная структура выбора. Графическая схема, операторная реализация. Пример нахождения наибольшего из трех заданных чисел.

***Условный оператор*** позволяет осуществить ветвление программы только по двум направлениям, одно из которых соответствует выполнению проверяемого условия, а другое - невыполнению этого же условия.

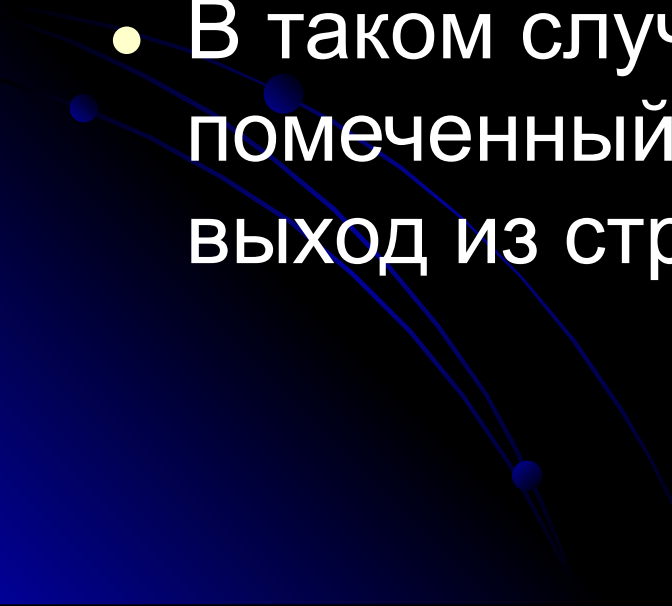


- Если для переменной необходимо осуществить ряд действий, зависящих от других условий, то надо записывать либо вложенные условные операторы, либо несколько таких операторов подряд.
  - Для такой ситуации удобно использовать **оператор варианта** – один из видов алгоритма ветвления.
- 

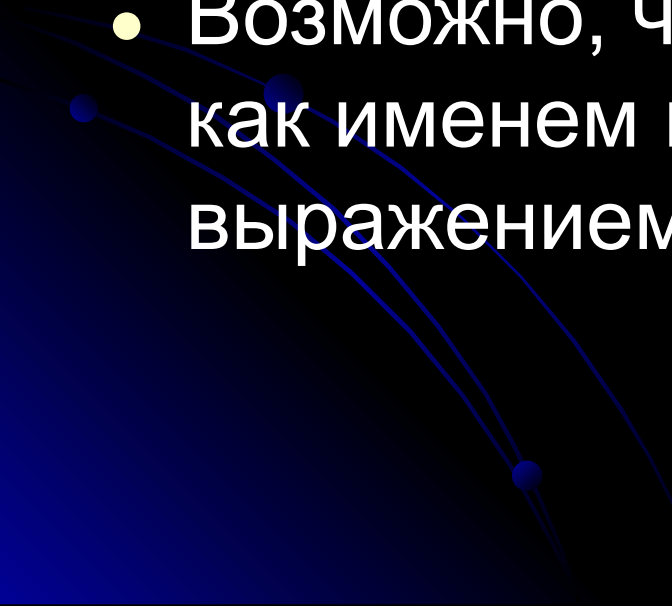



*Рис. 35. Структура оператора варианта*

- Данная структура называется также **переключателем**.

- Вход в структуру содержит вычисление ил ранее полученное значение переменной (индекса варианта).
  - Это значение может совпасть с меткой, стоящей перед оператором на одной из ветвей переключателя.
  - В таком случае выполняется оператор, помеченный этой меткой, и происходит выход из структуры.
- 

- Оператор бывает простым или составным, ограниченным операторными скобками **begin...end**;
- Если значение индекса варианта не совпало ни с одной из меток, то выполняется оператор с номером **n+1** из строки **else**;
- Если оператор варианта содержит строку **else**, то это полная форма оператора;
- Если такой строки нет, то используется сокращенная форма оператора варианта.

- Метки оператора варианта могут быть константами любого типа.
  - Их тип должен совпадать с переменной индекса варианта.
  - Возможно, чтобы индекс варианта был как именем переменной, так и выражением соответствующего типа.
- 

- На языке Паскаль оператор варианта имеет вид:
    - **Case** индекс варианта **of**
    - Метка1: оператор1;
    - Метка2: оператор2;
    - ...
    - Метка n: оператор n;
    - **Else** оператор n+1;
    - **End**;
- 

# Примеры программ, содержащих оператор варианта.

- **Program** seasons;
  - `var k:integer;`
- **begin**
  - `writeln ('введите номер месяца');`
  - `readln (k);`
  - **Case** k of
    - 1,2,12: `writeln ('зима');`
    - 3,4,5: `writeln ('весна');`
    - 6,7,8: `writeln ('лето');`
    - 9,10,11: `writeln ('осень');`
  - `end;`
  - `readln`
- **end.**



# Примеры программ, содержащих оператор варианта.

- Дано:
  - Молярная масса вещества  $M$ ,
  - Плотность данного вещества  $\rho$ ,
  - Масса  $R$ ,
  - Объем заданного тела  $V$ ;
- Требуется найти число молекул  $K$ :
  1. В единице массы вещества;
  2. В теле с заданной массой;
  3. В единице объема вещества;
  4. В теле с заданным объемом.

# Примеры программ, содержащих оператор варианта.

Для решения задачи воспользуемся формулой:

$$K=(NA/M)*R$$

где  $NA=6,022*10^{23}$  г/моль – число Авогадро.

На основании этой формулы получаем расчетные формулы:

1.  $K=NA/M;$
2.  $K=NA*R/M;$
3.  $K=NA*P*V/M;$
4.  $K=NA*P/M;$

# Программа имеет вид:

```
program E20;  
const NA = 6.022 E20;  
var N: integer; M, R, P, V, K: real;  
begin  
  writeln ('зная постоянную Авогадро, плотность P данного  
вещества');  
  writeln ('и его молярную массу M, можно найти число мо-  
лекул в');  
  writeln ('1. в единице массы вещества');  
  writeln ('2. в теле массой R');  
  writeln ('3. в единице объема');  
  writeln ('4. в теле объемом V');  
  write ('введите номер решаемой задачи');
```

```
readln (N);
write ('введите исходные данные: M = ');
readln (M);
case N of
1: K: = NA / M;
2: begin
    write ('R = ');
    readln (R);
    K: = NA *R/M;
    end;
3: begin
    write ('плотность вещества P = ');
    readln (P);
    write ('V = ');
    readln (V);
    K: = NA *P*V/M;
    end;
4: begin
    write ('плотность вещества P = ');
    readln (P);
    K: = NA *P/V
    end;
end;
write ('число молекул K = ', K)
end.
```