



# ОСНОВЫ программирования

---

Учитель информатики и ИКТ  
ГОУ г.Москвы СОШ №310  
«У Чистых прудов»  
Цыбикова Т.Р.



Тема 8.

# ОПЕРАТОР ВАРИАНТА

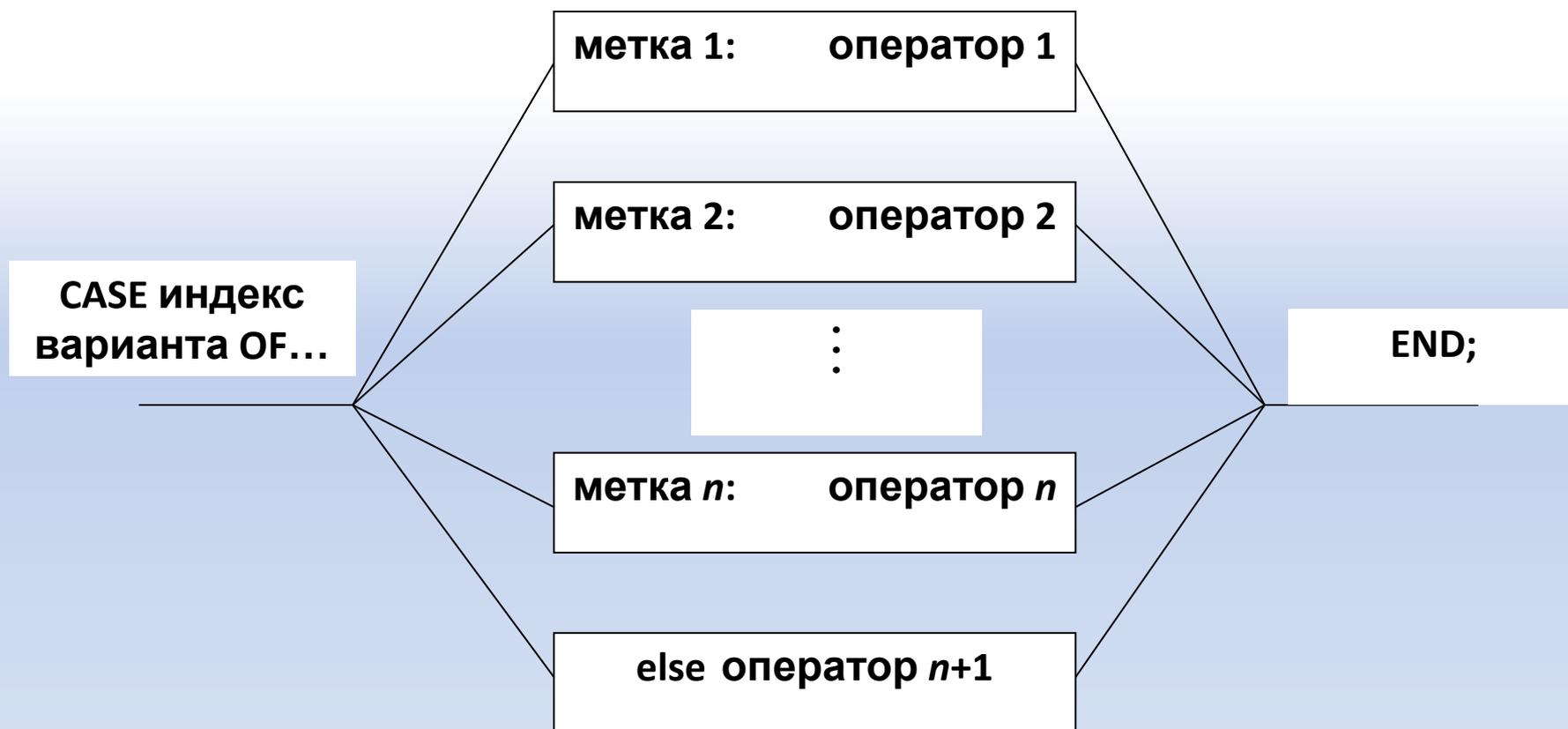




# Удобно использовать оператор варианта

- 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**. Он соответствует структуре, изображенной на рисунке 35.

## Рис.35. Структура оператора варианта





# Данная структура называется также переключателем

- Данная структура называется также переключателем и выполняется следующим образом.
- Вход в структуру содержит вычисление или 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 варианта бывает простым или СОСТАВНЫМ

- Оператор бывает простым или составным, ограниченными 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

- **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: **оператор 1;**

метка 2: **оператор 2;**

...

метка n: **оператор n;**

**else** оператор n+1

**end;**



# Приведем пример программы

- Приведем пример программы, содержащей оператор варианта.
- Одной из распространенных задач раздела физики «Молекулярно-кинетическая теория» является **задача, связанная с расчетом числа молекул в единице объема в теле заданной массы и в теле известного объема.**
- Для такой задачи можно построить программу-решатель.



# Задача 1

## Задача 1.

Для решения задачи воспользуемся формулой:

**Дано:** молярная масса вещества  $M$ , плотность данного вещества  $P$ , масса  $R$  или объем данного 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.

$$K = (N_A / M) R,$$

где  $N_A = 6,022 \cdot 10^{23}$  г/моль – число Авогадро.

На основании этой формулы получаем расчетные формулы для программы:

- 1)  $K = N_A / M$ ;
- 2)  $K = N_A R / M$ ;
- 3)  $K = N_A P V / M$ ;
- 4)  $K = N_A P / M$ .



# Программа имеет вид:

```
program E20;  
  const NA=6.022E20;  
  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 молекул V');  
  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 =
```

```
  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.
```



```
program E20;
const NA=6.022E20;
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 молекул 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 N = ');
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.
```



# Литература

- **А.А.Кузнецов, Н.В.Ипатова**  
«Основы информатики», 8-9 кл.:
  - Раздел 3. ОСНОВЫ ПРОГРАММИРОВАНИЯ,  
С.120-122