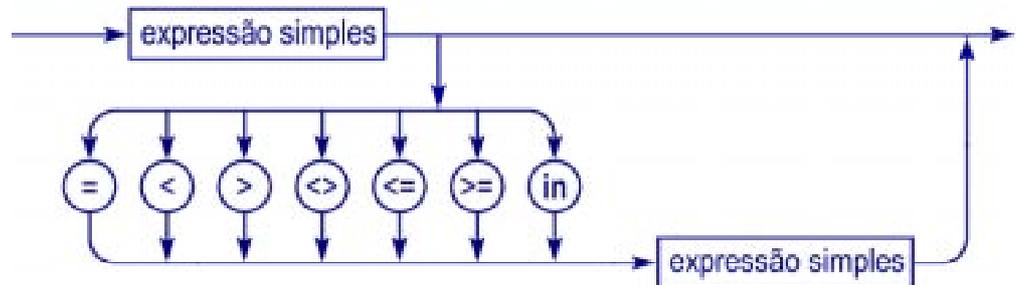
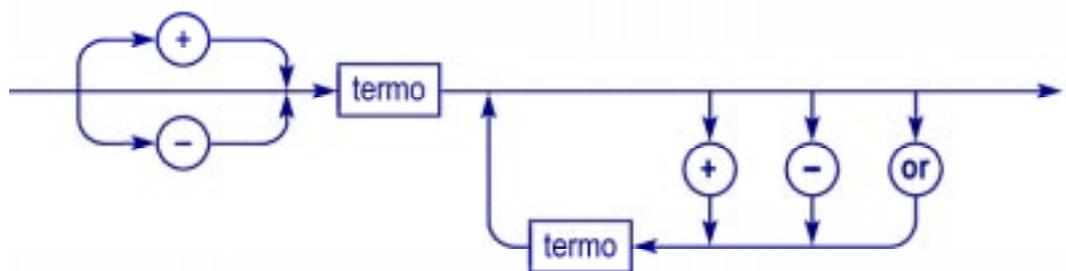


8. Expressões (Regras para o Cálculo de Valores):

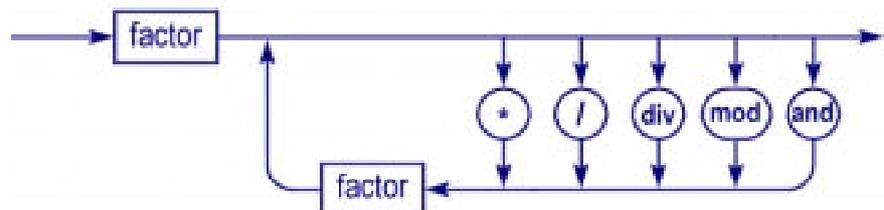
Expressão:



Expressão Simples:



Termo:



Factor:



A Sintaxe da Gramática da Linguagem Pascal estabelece as seguintes **Regras** para o Cálculo de Expressões:

- **Existe uma Ordem de Prioridade entre os Operadores, segundo a qual são calculados:**

Operadores	Prioridade
not	4
* / div mod and	3
+ - or	2
= <> > >= < <=	1

- **Para Operadores da mesma Prioridade, o Cálculo efectua-se da Esquerda para a Direita.**
- **A Utilização de Parêntesis sobrepõe-se às Regras anteriores.**

Observações:

- Todas as Variáveis que ocorrem numa Expressão têm já um valor atribuído;
- Dois Operadores não podem aparecer juntos;
- O Operador Multiplicação (*) não pode ser omitido;
- Em caso de dúvida, quanto a Prioridades, usar Parêntesis;
- Parêntesis em excesso não provocam erro e podem tornar a expressão mais legível.

9. A Instrução de Atribuição:



Deve ler-se: “Variável toma o valor de Expressão”!

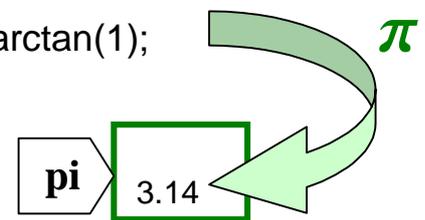
A Variável foi Declarada.

A Expressão é calculada e o seu valor final é registado na Célula de Memória associada à Variável.

```
var pi : real;
```

...

```
pi := 4* arctan(1);
```



- A Variável e a Expressão são do **mesmo Tipo**, excepto o caso da Conversão Implícita de Inteiros para Reais.

Exemplo: Como Trocar os Valores de duas Variáveis?

```
var a, b, aux : integer;
```

...

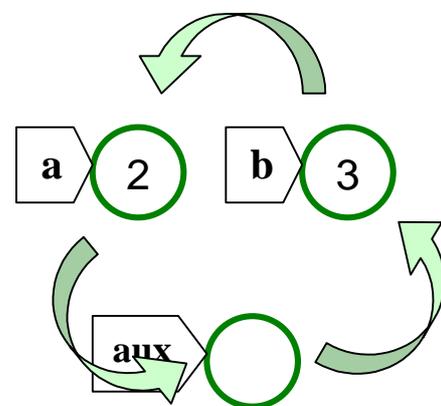
(* aqui a=2 e b=3 *)

```
aux := a;
```

```
a := b;
```

```
b := aux;
```

(* aqui a=3 e b=2 *)



- Os valores dos Dados devem aparecer no Ficheiro de Entrada pela mesma ordem que as correspondentes Variáveis na Instrução de Leitura.
- O valor de cada Dado deve ser do mesmo Tipo que a correspondente Variável.
- Não é permitida a Leitura de Dados do Tipo Lógico ou de Tipos Definidos por Enumeração.
- Na Leitura de Informação Numérica, espaços e mudanças de linha são ignorados.

Exemplo:

```
var x, y, z : real;  
    i, j : integer;  
    c, d : char;  
...
```

```
begin read(c, d, x, y, z);  
      readln(i, j);
```

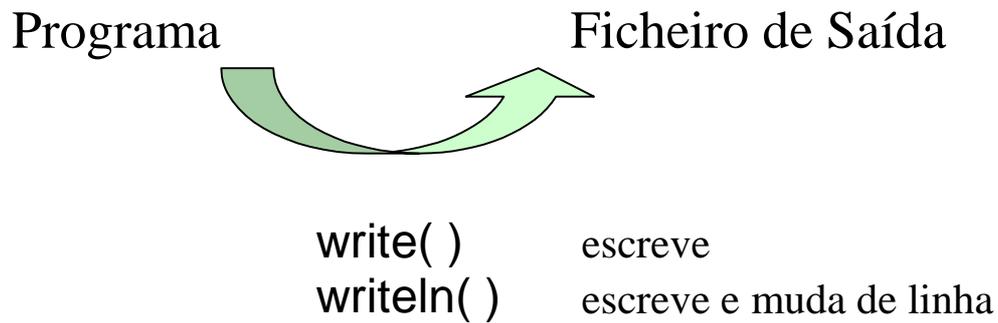
Ficheiro de Entrada:

```
AB  3.1E4    2  
   4.5  10  11  12  13
```

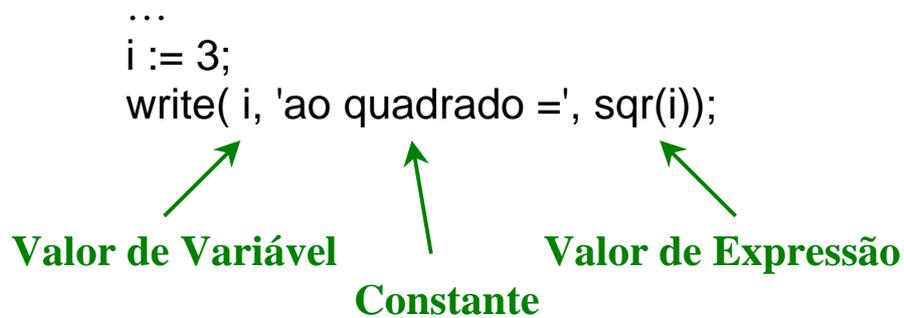
Tem o mesmo efeito que:

```
c:= 'A';  
d:= 'B';  
x:= 3.1E4; (*Saltou espaços*)  
y:= 2.0;   (*Converteu para Real*)  
z :=4.5;   (*Saltou espaços e linha*)  
i := 10;   (*Continuou na mesma linha*)  
j := 11;  
  
(*Os Restantes Dados foram ignorados*)
```

10.2. Instruções de Escrita:



Exemplo1:



Escreve:

3 ao quadrado = 9

Exemplo2:

```

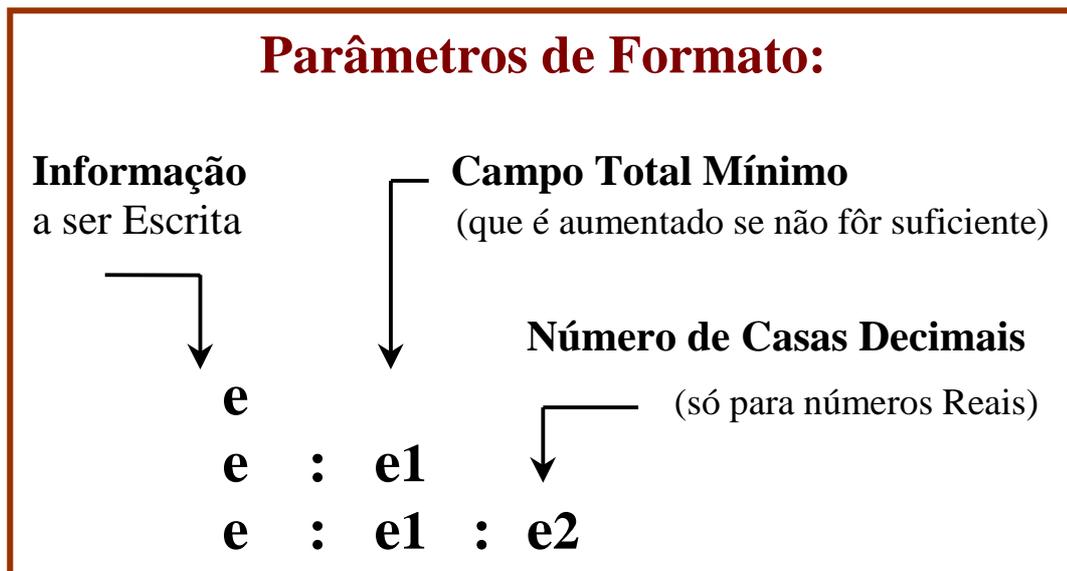
...
x := 2.5;
y := 2;
write('X=',x,'Y=',y);
writeln('SOMA=',x+y);

```

Escreve:

X=2.500000E+00Y=2.000000E+00SOMA=4.500000E+00

... mas será conveniente?



Exemplo3:

```

const digitos = '0123456789';
var a, b : integer;
    x, y : real;
...
writeln(digitos,digitos,digitos);
a := 20; b := 32000;
writeln('A =', a:7);
writeln('B = ' :5, b:2);
x := -2.5; y := 22;
writeln;
writeln('X =',x:7:2,'Y =':8,y:8:5);
writeln('Soma =',x+y:21:5);
...

```

Escreve:

```

012345678901234567890123456789
A =          20
  B =32000

X =   -2.50          Y =22.00000
Soma=                    19.50000

```