

Operadores

Prof. Fernando Figueira
(adaptado do material do Prof. Rafael Beserra Gomes)

UFRN

Material compilado em 27 de agosto de 2025.

Licença desta apresentação:



<http://creativecommons.org/licenses/>

- ▶ Combinação de **valores**, **variáveis**, **operadores** e **chamadas de funções**.
- ▶ São resolvidas durante a compilação ou durante a execução

Operadores realizam uma funcionalidade específica com os **operandos**.

Categorias dos operadores desta disciplina:

- ▶ Operadores aritméticos
- ▶ Operadores de atribuição
- ▶ Operadores relacionais
- ▶ Operadores lógicos

Operadores aritméticos

- ▶ +, -, *, /, %
- ▶ todos esses são binários (dois operandos numéricos)
- ▶ Exemplo:

```
1 #include <stdio.h>
2
3 int main() {
4
5     printf("%d\n", 2+3);
6     printf("%d\n", 5*3);
7     printf("%d\n", 5/2);
8     printf("%d\n", 17%3);
9
10    return 0;
11 }
```

Operadores de atribuição

- ▶ `<variável> = <expressão>`
- ▶ `<variável> += <expressão>` (ou `-=`, `*=`, `/=`)
- ▶ Exemplo:

```
1 #include <stdio.h>
2
3 int main() {
4
5     int x;
6     x = 3;
7     printf("Valor da variavel x: %d\n", x);
8     x = 5;
9     printf("Valor da variavel x: %d\n", x);
10    x -= 1;
11    printf("Valor da variavel x: %d\n", x);
12
13    return 0;
14 }
```

Operadores lógicos: atua em **proposições**

- ▶ Uma proposição é uma sentença que pode ser avaliada como **verdadeira** ou **falsa**
- ▶ Exemplos de proposições:
 - ▶ p: O céu é azul
 - ▶ q: Natal é a capital da Paraíba

Exemplos de sentenças que não são proposições:

- ▶ r: Um evento fora da cidade

- ▶ Em C declare as proposições como **int** (chamamos de variável lógica ou booleana)
- ▶ $= 0 \leftrightarrow$ **falso**
- ▶ $\neq 0 \leftrightarrow$ **verdadeiro**
- ▶ Exemplo:

```
1 int oCeuEhAzul = 1;  
2 int natalCapitalPB = 0;
```

- ▶ Por que variáveis lógicas?
 - ▶ p: O usuário clicou no botão x
 - ▶ q: O CPF digitado não é um CPF válido

A **Tabela Verdade** exibe todos os resultados possíveis de um **operador lógico** em função dos valores dos operandos.

Vamos ver 3 operadores lógicos:

- ▶ negação
- ▶ disjunção (ou)
- ▶ conjunção (e)

Operador de negação (unitário): !

| p | $!p$ |
|-----|------|
| F | V |
| V | F |

Por exemplo:

- ▶ p : O céu é azul
- ▶ seja $p = V$, $!p = F$

Operador de disjunção (ou): \parallel

| p | q | $p \parallel q$ |
|----------|----------|-----------------------------------|
| F | F | F |
| V | F | V |
| F | V | V |
| V | V | V |

Por exemplo:

- ▶ p : O céu é azul
- ▶ q : Natal é a capital da Paraíba
- ▶ seja $p = V$ e $q = F$, $p \parallel q = V$

Operador de conjunção (e): $\&\&$

| p | q | p $\&\&$ q |
|----------|----------|--------------------------------------|
| F | F | F |
| V | F | F |
| F | V | F |
| V | V | V |

Por exemplo:

- ▶ p: O céu é azul
- ▶ q: Natal é a capital da Paraíba
- ▶ seja $p = V$ e $q = F$, $p \&\& q = F$

Operadores relacionais:

- ▶ resultam em valores lógicos (1 para verdadeiro ou 0 para falso)
- ▶ `==`: compara se os dois operandos são iguais
- ▶ `!=`: compara se os dois operandos são diferentes
- ▶ `>`, `<`, `>=`, `<=`: o mesmo significado matemático para $>$, $<$, \geq , \leq

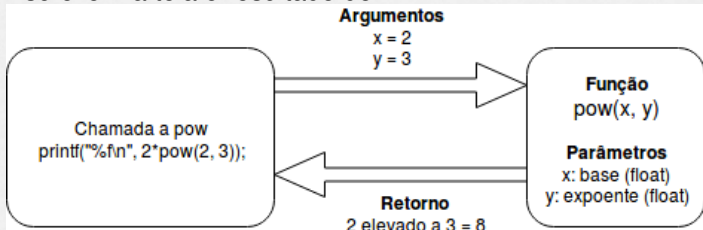
Observações:

- ▶ Os operadores `=>` ou `=<` não existem
- ▶ Por que $a > b > c$ não realiza o seu respectivo significado matemático?

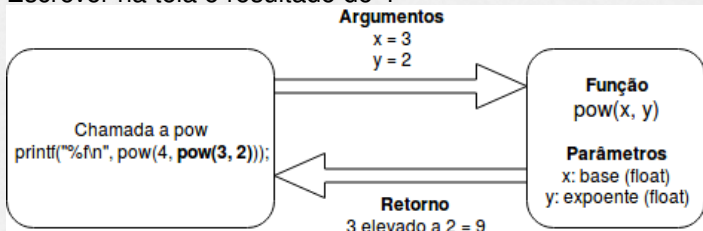
Precedência dos operadores vistos:

| precedência | operadores |
|-----------------------------|---|
| maior (avaliado primeiro) | <p>()</p> <p>* / %</p> <p>+ -</p> <p>< <= >= ></p> <p>== !=</p> <p>&&</p> <p> </p> |
| menor (avaliado por último) | <p>= *= += -=</p> |

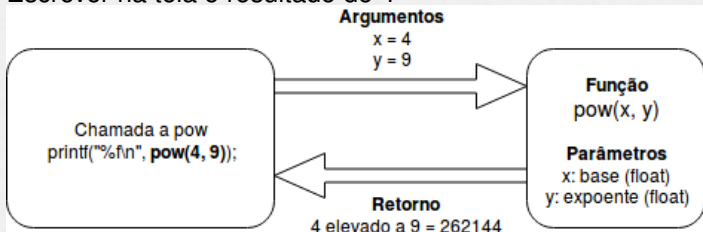
Escrever na tela o resultado de 2×2^3



Escrever na tela o resultado de 4^{3^2}



Escrever na tela o resultado de 4^3

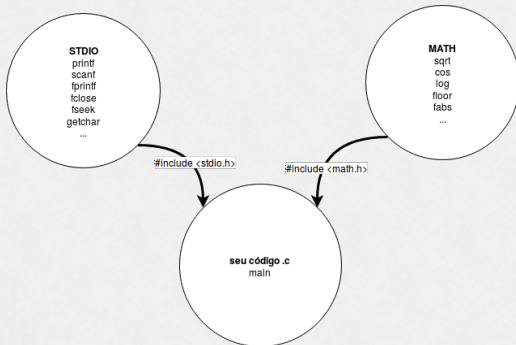


Já utilizamos (*chamamos*) várias funções:

- ▶ `pow(x, y)`: **retorna** x^y
- ▶ `sqrt(x)`: **retorna** a raiz quadrada de x
- ▶ `printf(...)`: escreve na saída padrão e **retorna** a quantidade de caracteres escritos
- ▶ `scanf(...)`: lê da entrada padrão e **retorna** a quantidade de elementos lidos com sucesso
- ▶ `srand(x)`: altera a semente para a geração de números aleatórios

Quais funções há em C e como utilizá-las?

- ▶ as funções estão organizadas em **bibliotecas**
- ▶ verifique a **assinatura** da função
exemplo: **double sqrt(double x);**
- ▶ consulte **man 3 nomefuncao** ou
<http://www.cplusplus.com/reference/clibrary/>



Você pode incluir outras bibliotecas, exemplo:

- ▶ gtk+: para criar interfaces gráficas
- ▶ opengl: para gráficos 3D
- ▶ ...dentre tantas outras

Provavelmente exigirá a instalação da biblioteca (exemplo, para usar o gtk+ pode fazer no ubuntu: *sudo apt-get install libgtk-3-dev*)

