



## PROGRAMAÇÃO 101 5 - LÓGICA E ARITMÉTICA



#### OPERADORES

AS OPERAÇÕES MAIS BÁSICAS QUE UM PROCESSADOR REALIZA SÃO OPERAÇÕES LÓGICAS E ARITMÉTICAS POR MEIO DOS OPERADORES:

- . MATEMÁTICOS: + \* / %
- . RELACIONAIS: > < >= <= == !=
- . LÓGICOS: && || ! (VEREMOS AMANHÃ)

#### OPERADORES MATEMÁTICOS





#### PRECEDENCIA

- . PRIMEIRO: PARÊNTESES
- . **SEGUNDO:** \*, /, %
- . TERCEIRO: +, -
- . QUANDO TEM A MESMA PRECEDÊNCIA: DA
- ESQUERDA PARA A DIREITA



#### EXEMPLO

```
#include<stdio.h>
 3
   int main (){
 5
       float resultado_1 = 6 * 2 / 3 + 10 % 2;
 6
       float resultado_2 = 6 * 2 / (3 + 10) % 2;
 8
 9
        printf("%lf\n", resultado 1);
        printf("%lf\n", resultado_2);
10
11
12
        return 0;
13
```

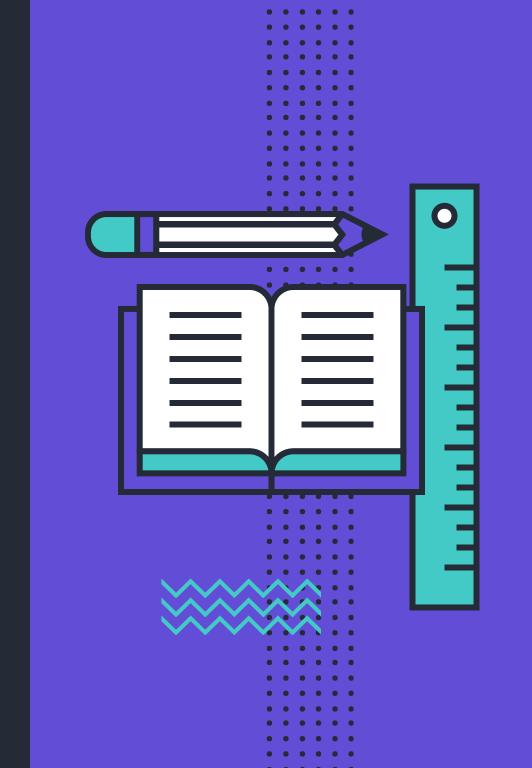
#### EXEMPLO

- 4.000000
- 0.000000

Process returned 0 (0x0) execution time : 0.022 s Press any key to continue.



# OPERADORES RELACIONAIS







OPERADORES LÓGICO-RELACIONAIS SÃO INTERPRETADOS
COMO PERGUNTAS CUJA RESPOSTA SEMPRE SERÁ

0, SE FALSO
1, SE VERDADEIRO

VALORES COMO ESSES SÃO CHAMADOS DE BOOLEANOS EXEMPLO: 3 < 5 = 1

```
#include<stdio.h>
 3
    int main (){
 4
 5
        int idade;
 6
 8
        idade = 17;
 9
        printf("Com %d anos pode tirar CNH: %d\n", idade, idade >= 18);
10
11
        idade = 35;
        printf("Com %d anos pode tirar CNH: %d\n", idade, idade >= 18);
12
13
        return 0;
14
15
```



Com 17 anos pode tirar CNH: 0

Com 35 anos pode tirar CNH: 1

Process returned 0 (0x0) execution time : 0.022 s

Press any key to continue.

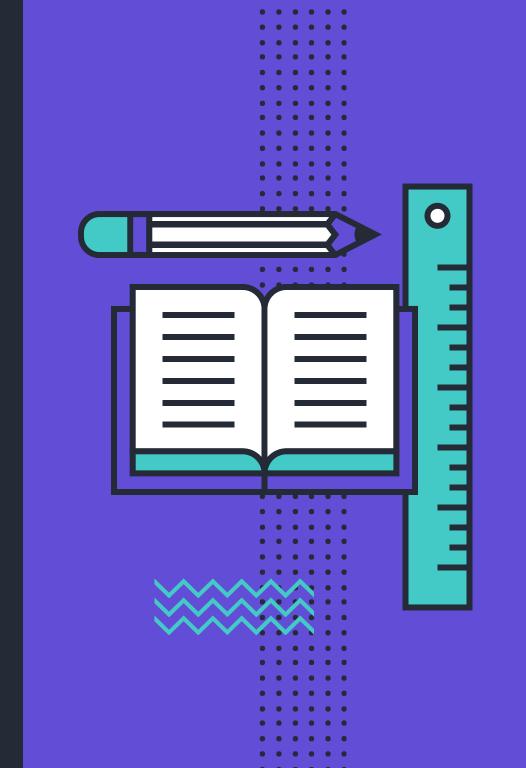


### PRECEDENCIA

- OPERADORES ARITMÉTICOS TEM PRECEDÊNCIA MAIOR QUE OS OPERADORES RELACIONAIS.
- 3 + 5 < 6 \* 2 É O MESMO QUE (3 + 5) < (6 \* 2)
- O QUE ACONTECE SE FIZERMOS 3 + (5 < 6) \* 2



OUTRAS OPERAÇÕES MATEMÁTICAS







ALÉM DA BIBLIOTECA STDIO.H, A LINGUAGEM C TEM VÁRIAS OUTRAS BIBLIOTECAS COM FUNÇÕES DE VÁRIAS NATUREZAS

ALGUMAS FUNÇÕES DA BIBLIOTECA MATH.H

- pow (double x, double y): retorna x elevado a y
- sqrt (double x): retorna raiz quadrada de x
- ceil (double x): arredonda x pra cima
- floor (double x): arredonda x pra baixo