

Algoritmos e Programação Estruturada

Operações e expressões

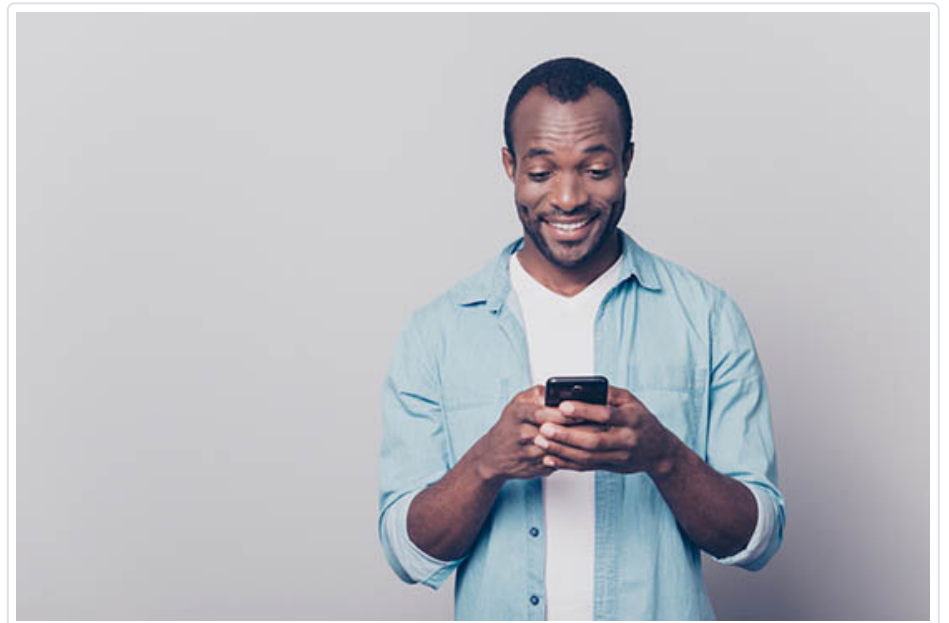
Você sabia que seu material didático é interativo e multimídia? Isso significa que você pode interagir com o conteúdo de diversas formas, a qualquer hora e lugar. Na versão impressa, porém, alguns conteúdos interativos ficam desabilitados. Por essa razão, fique atento: sempre que possível, opte pela versão digital. Bons estudos!

Nesta webaula temos como objetivo apresentar os operadores matemáticos, relacionais e lógicos.

Sistemas computacionais

Desde o momento em que você liga um computador (ou *tablet*, ou *smartphone*), centenas de processos são inicializados e passam a competir pelo processador para que possam ser executados e fazer a “mágica” do mundo digital acontecer.

Todos os resultados desses sistemas são obtidos através do processamento de dados e nesta webaula começaremos a estudar os recursos que lhe permitirão implementar soluções com processamento.



Fonte: Shutterstock.

Os sistemas computacionais são construídos para resolver os mais diversos problemas. Todos esses sistemas, independentemente da sua aplicação, são construídos em três partes: entrada, processamento e saída.

Nos três casos, a leitura dos dados é feita para um único fim: processamento e geração de informações e essa etapa é construída a partir da combinação de operações aritméticas, relacionais, lógicas e outras técnicas de programação.

Operadores aritméticos

Vamos começar a aprimorar nossos algoritmos através das operações aritméticas. Os operadores aritméticos podem ser classificados em unários ou binários (MANZANO, 2015).

Operadores aritméticos unários

Operador	Descrição	Exemplo	Resultado
++	Pós-incremento	$x ++$	$x + 1$
++	Pré-incremento	$++ x$	$x + 1$
--	Pós-decremento	$y --$	$y - 1$
--	Pré-decremento	$-- y$	$y - 1$

Fonte: elaborado pela autora.

Quando trabalhamos com operadores, a ordem de precedência é muito importante. Segundo Soffner (2013), os operadores aritméticos possuem a seguinte ordem de execução:

- 1° Parênteses.
- 2° Potenciação e radiciação.
- 3° Multiplicação, divisão e módulo.
- 4° Soma e subtração.

Operadores relacionais

Faz parte do processamento fazer comparações entre valores e, a partir do resultado, realizar novas ações. Em programação, para compararmos valores usamos operadores relacionais. Os operadores relacionais são utilizados para construir expressões booleanas, ou seja, expressões que terão como resultado verdadeiro ou falso.

Operadores aritméticos unários		
Operador	Descrição	Exemplo
==	igual a	x == y
!=	diferente de	X != y
>	maior que	X > y
<	menor que	X < y
>=	maior ou igual que	X >= y
<=	menor ou igual que	X <= y

Fonte: adaptado de Manzano (2015, p. 82).

Observe o código a seguir, de acordo com as entradas **n1=5**, **n2=10** e **n3=5**

- A instrução `(n1 == n2) && (n1 == n3)`, mostrará se **n1** é igual a **n2** E(&&) **n1** é igual a **n3**. No caso será impresso o valor 0 (falso), pois **n1** e **n2** são diferentes.
- A instrução `(n1 == n2) || (n1 == n3)`, mostrará se **n1** é igual a **n2** OU(||) **n1** é igual a **n3**. No caso será impresso o valor 1 (verdadeiro), pois, os valores de **n1** e **n3** são iguais.
- A instrução `(n1 < n3) || (n1 > n2)`, *mostrará se* **n1** é menor que **n3** OU (||) **n1** é maior que **n2**. No caso será impresso o valor 0 (falso), pois, os valores de **n1** e **n3** são iguais e **n1** é maior que **n2**.

Para visualizar o vídeo, acesse seu material digital.

Operadores lógicos

Além dos operadores relacionais, outro importante recurso para o processamento é a utilização de operadores lógicos, que possuem como fundamento a lógica matemática clássica e a lógica booleana (GERSTING, 2017).

Operadores lógicos		
Operador	Descrição	Exemplo
!	negação (NOT - NÃO)	!(x == y)
&&	conjunção (AND - E)	(x > y) && (a == b)
	disjunção (OR - OU)	(x > y) (a == b)

Fonte: adaptado de Soffner (2013, p. 35).

Observe no código a seguir, que na linha 14 `if (strcmp(login, "aluno01")==0 && strcmp(senha, "teste01")==0)` utilizamos os operadores “==” e “&&”.

- O operador “==” compara se as strings de entradas em login é igual a “aluno01” e senha é igual a “teste01”.
- O operador “&&” verifica se o login e a senha correspondem os valores de entradas.

Para visualizar o vídeo, acesse seu material digital.

[Online GDB](#)

Funções predefinidas

Para facilitar o desenvolvimento de soluções em software, cada linguagem de programação oferece um conjunto de funções predefinidas que ficam à disposição dos programadores. Entende-se por função “*um conjunto de instruções que efetuam uma tarefa específica.*” (MANZANO, 2015, p. 153).

Algumas bibliotecas e funções na linguagem C		
Biblioteca	Função	Descrição
<stdio.h>	printf()	Imprime na tela.
	scanf()	Faz leitura de um dado digitado.
	fgets(variavel, tamanho, fluxo)	Faz a leitura de uma linha digitada.
<math.h>	pow(base,potencia)	Operação de potenciação.
	sqrt(numero)	Calcula a raiz quadrada.
	sin(angulo)	Calcula o seno de um ângulo.
	cos(angulo)	Calcula o cosseno de um ângulo.

Biblioteca	Função	Descrição
<string.h>	strcmp(string1, string2)	Verifica se duas <i>strings</i> são iguais.
	strcpy(destino, origem)	Copia uma <i>string</i> da origem para o destino.
<stdlib.h>	malloc(tamanho)	Aloca dinamicamente espaço na memória.
	realloc(local,tamanho)	Modifica um espaço já alocado dinamicamente.
	free(local)	Libera um espaço alocado dinamicamente.

Fonte: adaptado de Tutorials point (2020).

Com esta webaula, exploramos as formas de armazenar temporariamente os dados em diversos tipos de variáveis e como podemos utilizar os operadores para realizar o processamento dos dados. Não se esqueça de recorrer ao livro didático para aprofundar os seus estudos.

Para visualizar o vídeo, acesse seu material digital.