

# Lógica de Programação – variáveis, operadores, constantes e tipos de dados



- ✓ À medida em que codificamos um problema do mundo real na forma de um programa de computador, há a necessidade de armazenar dados, que podem ser fornecidos pelo usuário, ou que podem ser produzidos no decorrer da execução do código;
- ✓ As variáveis são compostas por um identificador, com um nome. Os dados colocados em uma variável, e armazenados na memória RAM, são perdidos após a execução da aplicação terminar, ou quando estes dados são substituídos por outros, dentro do código.

# Atribuição de valores a variáveis



- ✓ Na linguagem algorítmica ou de pseudocódigo, sempre que for necessário indicar que estamos inserindo ou alterando (variando) um dado em uma variável, usamos o operador de atribuição =
- ✓ Exemplo:
  - Idade = 48
  - Nome = "Maria das Graças"
  - TemCarro = falso
- ✓ Cada linguagem de programação tem normas específicas para darmos nomes a constantes e variáveis (identificadores);
- ✓ Notar que, para toda variável ou constante que armazena valores do tipo texto (string ou char), estes valores devem estar dentro de aspas ("" ) ou apóstrofos (').

# Constantes



- ✓ **Como o próprio nome sugere, constante é um identificador dentro da Lógica de Programação que, uma vez atribuído um valor, não pode ser modificado, até o término da execução do código;**
- ✓ **Afora esta diferença, o uso de uma constante comporta-se exatamente igual ao uso de uma variável.**

# Tipos de dados

P

- ✓ Embora não seja obrigatório em todas as linguagens de programação, a maioria delas exige que, antes de usarmos uma variável ou constante, devemos declará-la e associá-la a um determinado tipo de dado;
  - ✓ Tipos de dados mais comuns na Lógica de Programação:
    - **Números inteiros (int)**: definem valores positivos e negativos para uma variável ou constante;
    - **Números de ponto flutuante (float ou double)**: definem valores reais para uma variável ou constante;
    - **Cadeia de caracteres (string)**: definem uma sequência de caracteres (texto);
    - **Caracter único (char)**: dados compostos por um só caracter;
    - **Valores lógicos (boolean)**: dados compostos por verdadeiro ou falso.
-

# Declaração de variáveis e constantes



- ✓ Como dito anteriormente, a maioria das linguagens de programação exigem que uma variável ou constante seja declarada antes de poder ser utilizada no código;
- ✓ Declarar uma variável ou constante nada mais é do que colocar o tipo de dado que a variável irá armazenar antes do seu nome. Também, é possível, ao mesmo tempo em que declaramos uma variável ou constante, já a inicializarmos com um valor específico. Exemplo:

- **Int** idade;
- **Boolean** temCarro
- **Boolean** podeVotar = true
- **String** nomeDoAluno
- **Char** digitoVerificador = "7"
- **Double** saldoBancario = 10789.78
- **Const Int** ano

# Operadores em Lógica de Programação

P

- ✓ Em Lógica de Programação, operadores são símbolos especiais que dizem ao nosso código o que fazer com um ou mais valores de uma variável. Estas operações podem ser operações matemáticas, lógicas, de comparação, etc.;
- ✓ Os operadores podem trabalhar nos dados de duas ou mais variáveis (operadores binários) ou com os dados de uma só variável (operadores unários). Exemplo:
  - ✓ Operador binário: idade1 – idade2
  - ✓ Operador unário: – raizQuadrada
- ✓ Tipos de operadores mais comuns:
  - a) **Aritméticos;**
  - b) **De atribuição;**
  - c) **Lógicos;**
  - d) **De comparação.**

# Operadores aritméticos

- ✓ **Definem operações matemáticas sobre os operandos. Podem ser utilizados tanto em variáveis quanto em constantes. Aqui, os principais:**
- a) **Adição (+) soma duas variáveis;**
  - b) **Subtração (-) subtrai o valor de duas variáveis;**
  - c) **Multiplicação (\*) multiplica os valores de duas variáveis;**
  - d) **Divisão real (/) divide o valor de duas variáveis;**
  - e) **Módulo (%) obtém o resto de uma divisão entre duas variáveis – resto inteiro.**

# Operadores de atribuição



- ✓ **Definem operações de associação de uma variável ou constante a um determinado valor;**
- ✓ **O símbolo mais comum é o sinal de igual, embora isso dependa linguagem de programação utilizada;**
- ✓ **O símbolo indica que o valor do lado direito da atribuição será armazenado na constante ou variável que está do lado esquerdo da atribuição. Exemplo:**
  - ✓ **Const Int Temperatura = 13.7**
  - ✓ **Idade = 47**
  - ✓ **Saldo3 = Saldo1 + Saldo2**
  - ✓ **Endereco = "Rua das Camélias, 760 – centro – Florianópolis"**
  - ✓ **RestoDaDivisao = Saldo3 % 12**



# Operadores compostos de atribuição

- ✓ São usados em conjunto com operadores matemáticos e simplificam operações dentro do código. São eles:
- ✓ **+=** soma valores e atribui
- ✓ **-=** diminui valores e atribui
- ✓ **\*=** multiplica valores e atribui
- ✓ **/=** divide valores e atribui
- ✓ **%=** calcula o resto de uma divisão e atribui.
- ✓ **Exemplos:**
  - **A += B** é equivalente a **A = A + B**
  - **A %= B** é equivalente a **A = A % B**
  - **A -= B** é equivalente a **A = A - B**
  - **A /= B** é equivalente a **A = A / B**
  - **A \*= B** é equivalente a **A = A \* B**

# Operadores lógicos

- ✓ São usados, principalmente em estruturas condicionais, e fazem comparações lógicas entre variáveis ou constantes. O resultado final de um operador lógica é sempre o valor verdadeiro (true) ou falso (false);
- ✓ São eles:
  - E lógico (**AND** ou **&&**): retorna o valor verdadeiro para a variável ou expressão somente se **TODAS** as variáveis contiverem o valor verdadeiro. Falso, caso contrário;
  - OU lógico (**OR** ou **||**): retorna o valor verdadeiro para uma variável ou expressão lógica sempre que houver, pelo menos, **UMA** variável com o valor verdadeiro. Retorna falso se **TODAS** as variáveis contiverem o valor falso;
  - Não lógico (**NOT** ou **!**): inverte o valor lógico de uma variável. Se a mesma contém true e aplicarmos este operador, seu valor seguinte será falso, e vice-versa.

# Operadores relacionais ou de comparação

- ✓ São símbolos matemáticos para comparar valores de grandeza entre variáveis ou constantes. Estes operadores, ao final da comparação, **sempre produzem um valor booleano verdadeiro ou falso**. São eles:
  - ✓ Igual (**==** ou **===**): verdadeiro se os valores das duas variáveis forem iguais;
  - ✓ Diferente (**!=** ou **!==**): retorna verdadeiro se e somente se os valores das duas variáveis forem diferentes;
  - ✓ Maior que (**>**): retorna verdadeiro somente se o valor de uma variável for maior que o valor da outra;
  - ✓ Maior ou igual a (**>=**): retorna verdadeiro somente se o valor de uma variável for maior ou igual ao valor de outra variável. Falso, caso contrário;
  - ✓ Menor que (**<**): retorna verdadeiro se, e somente se, o valor de uma variável for menor que o valor da outra. Falso, caso contrário;
  - ✓ Menor que ou igual a (**<=**): retorna verdadeiro se, e somente se, o valor de uma variável for menor que ou igual a o valor da outra. Falso, caso contrário.