

Universidade Federal de Viçosa
Departamento de Informática
Centro de Ciências Exatas e Tecnológicas

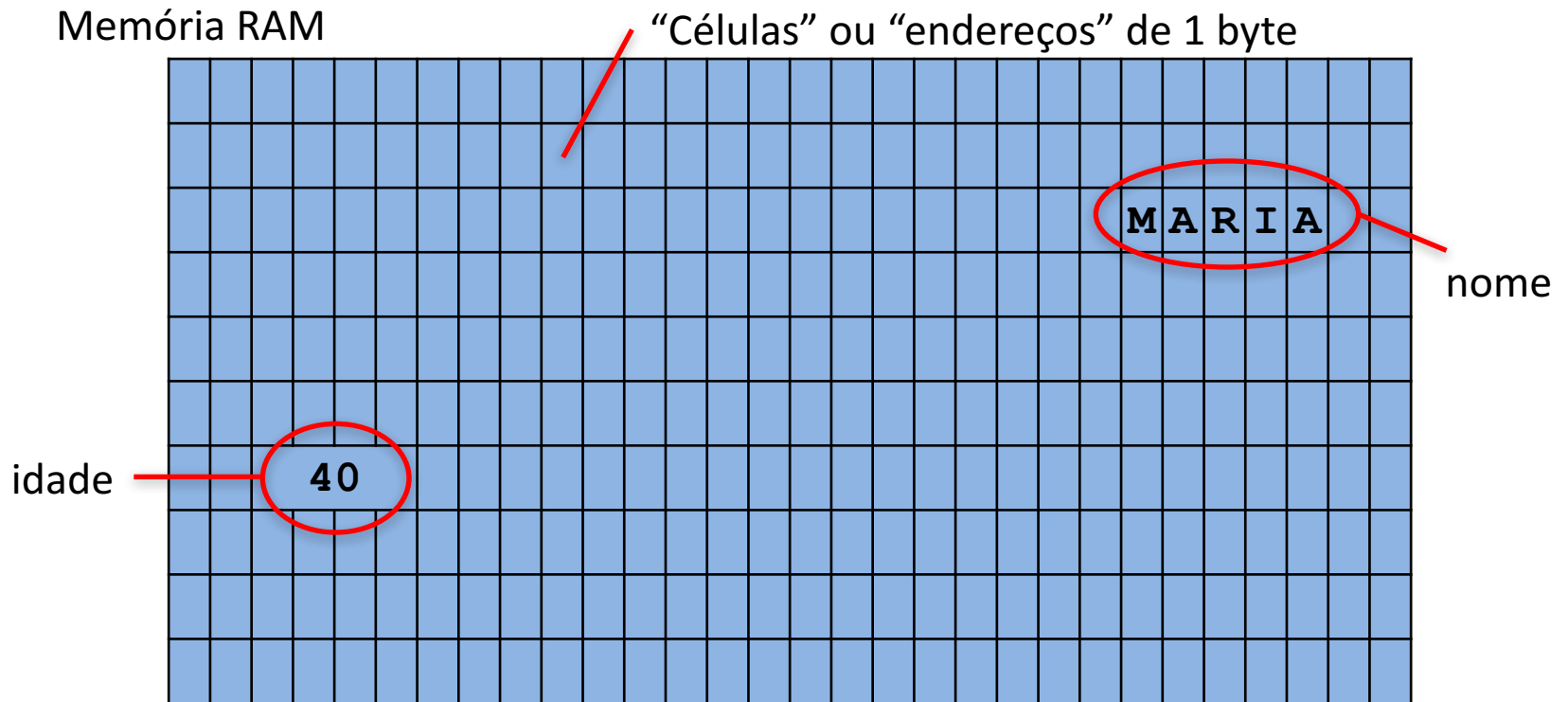


INF 100 – Introdução à Programação

Variáveis, Operadores,
Entrada de Dados

Variáveis

São nomes que usamos dentro dos programas para nos referirmos a algum espaço da memória RAM onde é guardado algum dado.



Variáveis

- Conceito semelhante, mas um pouco diferente do usado na matemática.
- Ex.: $ax^2 + bx + c = 0$
- Na matemática, a equação acima tem apenas uma variável: x . Os valores a , b e c não são vistos como variáveis, e sim parâmetros ou constantes quaisquer de uma equação de 2º grau.
- Na programação de computadores, x , a , b e c são vistos como variáveis, pois podem representar um valor numérico que pode variar dependendo da equação que se deseja resolver.



Variáveis

- Na matemática, podemos usar letras gregas ou outros símbolos para representar constantes ou variáveis:
$$= (r_1 \quad r_2) \text{ fator de impacto}$$
- Na programação, todo identificador ou nome de variável precisa iniciar com A..Z, a..z, _, e pode conter depois os caracteres A..Z, a..z, 0..9, _. Não são permitidos quaisquer outros símbolos, nem mesmo espaços em branco.

`delta = PI* (r1 - r2)*fator_de_impacto`



Variáveis

- Algumas palavras são “reservadas” na linguagem e não podem ser usadas como nomes de variáveis.
- Em Python, a lista de palavras reservadas pode ser obtida assim:

```
>>> import keyword
>>> keyword.kwlist
['False', 'None', 'True', 'and', 'as', 'assert',
'break', 'class', 'continue', 'def', 'del',
'elif', 'else', 'except', 'finally', 'for',
'from', 'global', 'if', 'import', 'in', 'is',
'lambda', 'nonlocal', 'not', 'or', 'pass',
'raise', 'return', 'try', 'while', 'with',
'yield']
```



Variáveis

- Exemplos de identificadores válidos para variáveis:
 - x
 - nome
 - x7
 - y_max
 - NomeGrande
- Exemplos de identificadores não válidos para variáveis:
 - 8y
 - π
 - and
 - for
 - Nome Grande



Variáveis

- Atribuindo valores iniciais para variáveis:

n=0

idade = 40

fator = 3.4

sexo = 'M'

nome = 'Carlos Alberto'



Escrevendo da tela o valor (conteúdo) de Variáveis

```
idade = 40
sexo = 'M'
nome = 'Carlos Alberto'
cabo = "Pirelli's antichama"
espessura = '20"'
print('Nome:', nome )
print('Sexo:', sexo, '      Idade:', idade )
print( cabo, espessura )
```



Expressões aritméticas

- Principais operadores para a realização de cálculos e criação de expressões aritméticas:

Operador	Nome/Papel do operador
()	Parênteses
+	Adição
-	Subtração
*	Multiplicação
/	Divisão
%	Módulo (resto da divisão inteira)
//	Quociente (da divisão inteira)
**	potenciação (x elevado a y)



Expressões aritméticas

- Comando de atribuição: armazena o valor da expressão do lado direito na variável que está do lado esquerdo.

```
resultado = 2+6  
x = 2*resultado  
pi = 3.14159  
y = (x+3)*x/3  
y = x+y  
x = y = 0
```



Expressões aritméticas (Observações Importantes)

```
>>> pi = 3.14159
>>> r = 2
>>> area = pi * r**2
>>> area
12.56636
>>> r = 10
>>> area
12.56636
>>> area = pi * r**2
>>> area
314.159
>>> pi = -1
>>> area
314.159
>>> area = pi * r**2
>>> area
-100
>>>
```

Esta expressão **não define 'area'**. Ela simplesmente efetua o cálculo da expressão à direita do '=' e atribui o resultado à variável à esquerda do '='.

Alterar o valor de 'r' **não afeta** o valor de 'area'.

Para recalcular a área, é preciso repetir o cálculo.

O nome 'pi' foi apenas um nome que demos para o valor 3.14159. O nome em si não significa nada para o computador. Ele não sabe que 'pi', neste contexto, deveria se referir à constante π . Apenas facilita a **nossa** compreensão do programa.



Expressões aritméticas

- **Precedência dos operadores** (em ordem da maior para a menor precedência):

Operadores	Nomes dos operadores
()	Parênteses
**	Potenciação
*, /, //, %	Multiplicação, Divisão, Módulo
+, -	Adição, Subtração



Expressões aritméticas

- Exemplos:

Expressão	Resultado
$8 + 3 * 2$	
$(8 + 3) * 2$	
$2 + 4 \% 3$	
$(2 + 4) \% 3$	
$5 - 3 + 1$	
$5 - (3 + 1)$	
$1 + 4 / 2 + 2 * 2$	



Expressões aritméticas

- Exemplos:

Expressão	Resultado
$8 + 3 * 2$	14
$(8 + 3) * 2$	22
$2 + 4 \% 3$	3
$(2 + 4) \% 3$	0
$5 - 3 + 1$	3
$5 - (3 + 1)$	1
$1 + 4 / 2 + 2 * 2$	7

Na dúvida, use
parênteses!



Entrada Básica de Dados

- Comando **input**: permite recebermos qualquer texto digitado pelo teclado, que é o dispositivo padrão de entrada.



Entrada de Dados

- Ao encontrar o comando **input**, o programa aguarda até que o usuário digite alguma coisa pelo teclado e tecele ENTER. O valor digitado é então transferido para dentro da variável:

```
nome = input('Entre com seu nome: ')\nprint('Bom dia,', nome, '!')
```



Entrada de Dados

- Para receber valores numéricos, é preciso converter o texto digitado para um número:

```
idade = int (input('Qual sua idade? '))  
altura = float (input('Sua altura (m)? '))  
print('Idade:', idade )  
print('Altura:', altura, 'm')
```



Entrada de Dados

- `int(x)`
 - Converte x para um número inteiro.
 - Exemplos:
 - `idade = int('18')`
 - `x = int(3.14)`
- `float(x)`
 - Converte x para um número real (*floating point number*, ou número de ponto flutuante).
 - Exemplos:
 - `altura = float('1.75')`



Exercício

- Faça um programa que leia do teclado três valores reais a , b e c , e depois escreva na tela a média desses valores.



Exercício

- Versão 1:

```
media = (a + b + c) / 3
a = float (input('a = '))
b = float (input('b = '))
c = float (input('c = '))
print('Média:', media )
```

Isso funciona?



Exercício

- Versão 2:

```
a = b = c = 0
media = (a + b + c) / 3
a = float (input('a = '))
b = float (input('b = '))
c = float (input('c = '))
print('Média:', media )
```

Isso funciona?



Exercício

- Versão 3:

```
a = float (input('a = '))  
b = float (input('b = '))  
c = float (input('c = '))  
media = (a + b + c) / 3  
print('Média:', media )
```

Isso funciona?

