

MAC 110 – Introdução à Ciência da Computação

Aula 2

Nelson Lago

BMAC – 2024



Previously on MAC 110...

Programar envolve

- ❶ **Compreender um problema em termos computacionais**
- ❷ **Definir como esse problema pode ser solucionado (*algoritmo*)**
 - ▶ O algoritmo é *abstrato* (como a planta de um prédio ou uma receita de bolo)
- ❸ **Implementar o algoritmo em uma linguagem de programação**
 - ▶ Gerando um *programa* que pode ser *executado* para solucionar o problema
- ❹ **Testar o programa**

Para ser útil, um programa geralmente

❶ **Obtém dados**

❷ **“Faz alguma coisa” com esses dados**

- ▶ Gerando um resultado

❸ **“Faz alguma coisa” com esse resultado**

- ▶ Mostra para o usuário

- ▶ **Utiliza como dado para fazer outra coisa**

Expressões em python

- A maioria das coisas em python são *expressões*
 - ▶ (expressões são coisas que têm um *valor*)
 - ▶ Exemplo: 47
 - ▶ Exemplo: 2 + 3
 - ▶ Exemplo: "Oi galera!"
- Expressões podem ser combinadas ou utilizadas como partes de outras expressões
 - ▶ Exemplo: $\frac{2+3+7}{6}$
 - ▶ Exemplo: 2 + 3 + 7 < 0
 - ▶ Exemplo: 2 + 3 + 7 < 0 **and** 5 > 4



```
print(type(2))  
print(type(False))  
print(type(2.0))  
print(type("Olá"))
```

```
<class 'int'>  
<class 'bool'>  
<class 'float'>  
<class 'str'>
```

Exercícios no eDisciplinas

- **Bem-vindo ao python**
- **Bem-vindo a MAC110**
- **Brincando com expressões**

Nomes (variáveis)

- Ao programar, preferimos pensar no problema a ser resolvido e não nas idiossincrasias do computador
- Linguagens de programação de alto nível procuram oferecer os recursos para isso
- Uma das coisas mais importantes para esse fim é utilizar *nomes*

Nomes (variáveis)

$x \leftarrow 5$ (atribuição)

Há um número finito de caracteres no teclado, então fazemos atribuição em python com “=”:

```
x = 5
x = "Olá, galera!"
x = y
x = x + 1
```

Nomes (variáveis)

$$ax^2 + bx + c = 0$$

Nomes (variáveis)

$$ax^2 + bx + c = 0$$

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Nomes (variáveis)

$$ax^2 + bx + c = 0$$

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$-x^2 + 3x + 4 = 0$$

Nomes (variáveis)

$$ax^2 + bx + c = 0$$

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$-x^2 + 3x + 4 = 0$$

```
import math
print((-3 + math.sqrt(3**2 - 4 * -1 * 4)) / -2,
      (-3 - math.sqrt(3**2 - 4 * -1 * 4)) / -2)
```

Nomes (variáveis)

$$ax^2 + bx + c = 0$$

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$-x^2 + 3x + 4 = 0$$

```
import math
print((-3 + math.sqrt(3**2 - 4 * -1 * 4)) / -2,
      (-3 - math.sqrt(3**2 - 4 * -1 * 4)) / -2)
```

-1.0 4.0

Nomes (variáveis)

$$ax^2 + bx + c = 0$$

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$-x^2 + 3x + 4 = 0$$

```
import math
print((-3 + math.sqrt(3**2 - 4 * -1 * 4)) / -2,
      (-3 - math.sqrt(3**2 - 4 * -1 * 4)) / -2)
```

-1.0 4.0



Nomes (variáveis)

```
import math
a = -1
b = 3
c = 4
delta = b**2 - 4 * a * c
raiz1 = (-1 * b + math.sqrt(delta)) / 2 * a
raiz2 = (-1 * b - math.sqrt(delta)) / 2 * a
print("As raízes são", raiz1, "e", raiz2)
```

Nomes (variáveis)

```
import math
a = -1
b = 3
c = 4
delta = b**2 - 4 * a * c
raiz1 = (-1 * b + math.sqrt(delta)) / 2 * a
raiz2 = (-1 * b - math.sqrt(delta)) / 2 * a
print("As raízes são", raiz1, "e", raiz2)
```

As raízes são -1.0 e 4.0

Nomes (variáveis)

```
# Calcula raízes da equação de segundo grau  
import math  
a = -1  
b = 3  
c = 4  
delta = b**2 - 4 * a * c # Não pode ser menor que zero!  
raiz1 = (-1 * b + math.sqrt(delta)) / 2 * a  
raiz2 = (-1 * b - math.sqrt(delta)) / 2 * a  
print("As raízes são", raiz1, "e", raiz2)
```

As raízes são -1.0 e 4.0

Exercícios no eDisciplinas

- **Valor da função**
- **Cálculo da distância**
- **Área do triângulo**

and now for something completely different

Nomes (variáveis)

- **Por que o título destes slides é “Nomes (variáveis)”?**

Nomes (variáveis)

- **Por que o título destes slides é “Nomes (variáveis)”?**
 - ▶ Não há muita graça em escrever programas como os que vimos acima, em que os dados são sempre os mesmos

- **Por que o título destes slides é “Nomes (variáveis)”?**
 - ▶ Não há muita graça em escrever programas como os que vimos acima, em que os dados são sempre os mesmos
 - ▶ Um dos principais usos de nomes é representar valores que *variam* (basicamente, alguma informação “real” que está em algum lugar na memória do computador)
 - » *Como na matemática!*

- **Por que o título destes slides é “Nomes (variáveis)”?**
 - ▶ Não há muita graça em escrever programas como os que vimos acima, em que os dados são sempre os mesmos
 - ▶ Um dos principais usos de nomes é representar valores que *variam* (basicamente, alguma informação “real” que está em algum lugar na memória do computador)
 - » *Como na matemática!*
 - ▶ Por isso, chamamos esses nomes de “variáveis”

Nomes (variáveis)

- **Por que o título destes slides é “Nomes (variáveis)”?**
 - ▶ Não há muita graça em escrever programas como os que vimos acima, em que os dados são sempre os mesmos
 - ▶ Um dos principais usos de nomes é representar valores que *variam* (basicamente, alguma informação “real” que está em algum lugar na memória do computador)
 - » *Como na matemática!*
 - ▶ Por isso, chamamos esses nomes de “variáveis”
 - ▶ Acima, definimos o valor dos nomes (variáveis) com comandos de atribuição ($a = -1$) fixos. Como fazer se queremos valores que não sejam fixos?

Nomes (variáveis)

- **Por que o título destes slides é “Nomes (variáveis)”?**
 - ▶ Não há muita graça em escrever programas como os que vimos acima, em que os dados são sempre os mesmos
 - ▶ Um dos principais usos de nomes é representar valores que *variam* (basicamente, alguma informação “real” que está em algum lugar na memória do computador)
 - » *Como na matemática!*
 - ▶ Por isso, chamamos esses nomes de “variáveis”
 - ▶ Acima, definimos o valor dos nomes (variáveis) com comandos de atribuição ($a = -1$) fixos. Como fazer se queremos valores que não sejam fixos?
 - ▶ `input()`

Primeiro programa – bully

```
idade = input("Informe sua idade: ")
```

Primeiro programa – bully

```
idade = input("Informe sua idade: ")  
  
print("Você tem só", idade, "anos?!?!")  
print("Nossa, você aparenta ter", 2 * idade, "anos!")
```

Primeiro programa – bully

```
idade = input("Informe sua idade: ")  
  
print("Você tem só", idade, "anos?!?!")  
print("Nossa, você aparenta ter", 2 * idade, "anos!")
```

Informe sua idade: 23

Primeiro programa – bully

```
idade = input("Informe sua idade: ")  
  
print("Você tem só", idade, "anos?!?!")  
print("Nossa, você aparenta ter", 2 * idade, "anos!")
```

Informe sua idade: 23

Você tem só 23 anos?!?!

Primeiro programa – bully

```
idade = input("Informe sua idade: ")  
  
print("Você tem só", idade, "anos?!?!")  
print("Nossa, você aparenta ter", 2 * idade, "anos!")
```

Informe sua idade: 23

Você tem só 23 anos?!?!

Nossa, você aparenta ter 2323 anos!

Primeiro programa – bully

```
idade = input("Informe sua idade: ")  
  
print("Você tem só", idade, "anos?!?!")  
print("Nossa, você aparenta ter", 2 * idade, "anos!")
```


Primeiro programa – bully

```
idade = int(input("Informe sua idade: "))  
  
print("Você tem só", idade, "anos?!?!")  
print("Nossa, você aparenta ter", 2 * idade, "anos!")
```

Primeiro programa – bully

```
idade = input("Informe sua idade: ")

print("Você tem só", idade, "anos?!?!")
print("Nossa, você aparenta ter", 2 * int(idade), "anos!")
```

Primeiro programa – bully

```
idade = input("Informe sua idade: ")
idade = int(idade)
print("Você tem só", idade, "anos?!?!")
print("Nossa, você aparenta ter", 2 * idade, "anos!")
```

Exercícios no eDisciplinas

- **Quadrado de um número**
- **Quadrado da diferença**