

Departamento de Engenharia de Sistemas e Computação

Estrutura Básica de Programas

Sequência, entrada e saída, expressões simples

João Araujo

Algoritmos Computacionais I

Aula 06

Roteiro

- Programação estruturada: três blocos de construção (Cap. 3.1)
- **Sequência** (3.1.1): executar comandos em ordem
- Saída com `print`: separadores e formatação
- Entrada com `input` e conversão de tipos
- Expressões e precedência de operadores
- **Aula prática** no laboratório

Programação estruturada

Todo programa pode ser construído com apenas três **blocos**:

Sequência	Seleção	Iteração
Um comando depois do outro, de cima para baixo.	Decidir entre caminhos: `if`/`else` (Aula 7).	Repetir um bloco: `while`/`for` (Aula 8).

Hoje: **sequência**. A ordem importa: o computador nunca "adivinha" o que vem depois.

Sequência: a ordem dos comandos

```
base = 5
altura = 6
area_triang = (base * altura) / 2
print('A área do triângulo com base', base,
      'e altura', altura, 'é', area_triang)
```

```
A área do triângulo com base 5 e altura 6 é 15.0
```

O que acontece se invertermos as linhas 3 e 4?

`NameError`: a variável `area_triang` ainda não existe quando `print` é executado.

Fonte: `code/Cap2/prog.2.5_area_triangulo_v3.py`

Evolução do programa: três versões

```
# v1: um valor "mágico" (sem contexto)
print((5*6)/2)

# v2: guarda o resultado e explica a saída
area_triang = (5*6)/2
print('A área do triângulo é ', area_triang)

# v3: dados em variáveis com nomes claros
base = 5
altura = 6
```

- Cada versão é mais fácil de ler e modificar que a anterior
- Evitar números "soltos" no meio do código

Fonte: `code/Cap2/prog.2.3, prog.2.4, prog.2.5`

Saída: a função `print`

```
print("Olá", "mundo")              # Olá mundo
print("Olá", "mundo", sep="-")     # Olá-mundo
print("a", end=" ")                # não pula linha
print("b")                         # a b
print("Total:", 7 * 3)             # Total: 21
```

- Vários argumentos separados por vírgula; por padrão, separados por um espaço
- `sep` troca o separador; `end` troca o final (padrão: nova linha)
- `\n` = nova linha, `\t` = tabulação

Fonte: `code/Cap3/prog.3.15_contagem.py (usa end=' ')`

Saída formatada

```
import math
x = math.pi
print("Pi vale", x)
print("Pi vale {:.2f}".format(x))       # duas casas decimais
print(f"Pi vale {x:.4f}")                # f-string
print(f"{'Graus':>9}{'Seno':>9}")       # alinha à direita em 9 colunas
print(f"{30:>9.2f}{math.sin(math.radians(30)):>9.2f}")
```

```
Pi vale 3.141592653589793
Pi vale 3.14
Pi vale 3.1416
    Graus     Seno
    30.00     0.50
```

Fonte: code/Cap6/prog.6.17_tabela_senos_arquivo_f.py

Entrada: a função `input`

```
nome = input("Qual é o seu nome? ")
print("Olá,", nome)
idade = input("Qual a sua idade? ")
print(type(idade))              # <class 'str'>
```

`input` sempre devolve `str`!

Para fazer contas é preciso converter: `int(...)` ou `float(...)`.

- O texto entre parênteses é o *prompt* exibido ao usuário
- Se o usuário digitar algo inválido (`abc` para `int`), ocorre `ValueError`; trataremos na Aula 13

O erro clássico: somar textos

```
nota1 = input("Qual é a primeira nota? ")
nota2 = input("E a segunda nota? ")
nota_total = nota1 + nota2
print("A soma é ", nota_total)
```

```
Qual é a primeira nota? 7
E a segunda nota? 8
A soma é  78
```

- '7' + '8' $\Rightarrow$ '78' (concatenação de strings)
- **Correção:** `nota1 = float(input(...))`

Fonte: `code/Cap3/prog.3.2_testa_input.py`

Programa completo: área de um quadrado

```python
# lê o valor do lado
lado = float(input("Digite o valor do lado do quadrado: "))
# calcula a área
area = lado * lado
# imprime o resultado
print("Área do quadrado:", area)
```

Estrutura típica de um programa simples: **entrada** → **processamento** → **saída**.

Exercícios 3.1 e 3.2

Versão com `int` e versão com `float`: qual a diferença para o lado `2.5`?

Expressões e precedência

Ordem	Operadores
1	parênteses `( )`
2	potência `**`
3	`+x`, `-x` (sinal)
4	`*`, `/`, `//`, `%`
5	`+`, `-`
6	comparações `==`, `<`, ...
7	`not`, `and`, `or`

$2 + 3 \times 4 = 14$, mas $(2 + 3) \times 4 = 20$. Na dúvida, use parênteses.

Cuidado: $\frac{a+b}{2}$ é `(a + b) / 2`, e não `a + b / 2`.

Atividade prática no laboratório

Escreva e execute:

1. Programa que lê **base** e **altura** e imprime a área do triângulo (formato com 2 casas)
2. Programa que lê uma temperatura em Celsius e imprime em Fahrenheit: $F = \frac{9}{5}C + 32$
3. Programa que lê o número de segundos e mostra em horas, minutos e segundos (use `//` e `%`)
4. Programa que calcula a área de uma esfera de raio 5: $A = 4\pi r^2$ (Exercício 2.18)

Dica para o item 3

```
h = seg // 3600    resto = seg % 3600    m = resto // 60
```

Para praticar: exercícios do livro

Site do livro – Capítulo 2

- **2.9, 2.10** (área, segundos após a meia-noite)
- **2.17, 2.18** (hipotenusa, área da esfera)

Capítulo 3: `exercicios/cap3/` – **3.1, 3.2**.

Enunciados e soluções comentadas em
`https://www.araujo.eng.uerj.br/livro/exercicios/cap2/`

Exercícios adicionais (sem solução): `https://www.araujo.eng.uerj.br/livro/extras/`

Resumo da aula

- Um programa estruturado combina sequência, seleção e iteração
- `print` formata a saída; `input` lê *texto*, sempre
- Converta com `int()` e `float()` antes de calcular
- Parênteses tornam as expressões claras e seguras
- Próxima aula: estruturas de decisão (`if`, `elif`, `else`)