

# Estrutura de repetição `while`

## Aula 7

Diego Padilha Rubert

Faculdade de Computação  
Universidade Federal de Mato Grosso do Sul

Algoritmos e Programação

- 1 Motivação
- 2 Estrutura de repetição `while`
- 3 Exercícios

- ▶ Com o que sabemos, como podemos imprimir os 10 primeiros números inteiros positivos?

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

print(1)
print(2)
print(3)
print(4)
print(5)
print(6)
print(7)
print(8)
print(9)
print(10)

exit(0)
```

- ▶ Com o que sabemos, como podemos imprimir os 10 primeiros números inteiros positivos?

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

print(1)
print(2)
print(3)
print(4)
print(5)
print(6)
print(7)
print(8)
print(9)
print(10)

exit(0)
```

- ▶ e se quiséssemos imprimir os 100 primeiros números inteiros positivos? Ou os 1000 primeiros?
- ▶ computadores são construídos para que possamos delegar a eles as tarefas mais repetitivas
- ▶ na linguagem Python temos duas estruturas de repetição diferentes ao nosso dispor
- ▶ estudaremos primeiro a estrutura de repetição `while`

- ▶ e se quiséssemos imprimir os 100 primeiros números inteiros positivos? Ou os 1000 primeiros?
- ▶ computadores são construídos para que possamos delegar a eles as tarefas mais repetitivas
- ▶ na linguagem Python temos duas estruturas de repetição diferentes ao nosso dispor
- ▶ estudaremos primeiro a estrutura de repetição `while`

- ▶ e se quiséssemos imprimir os 100 primeiros números inteiros positivos? Ou os 1000 primeiros?
- ▶ computadores são construídos para que possamos delegar a eles as tarefas mais repetitivas
- ▶ na linguagem Python temos duas estruturas de repetição diferentes ao nosso dispor
- ▶ estudaremos primeiro a estrutura de repetição `while`

- ▶ e se quiséssemos imprimir os 100 primeiros números inteiros positivos? Ou os 1000 primeiros?
- ▶ computadores são construídos para que possamos delegar a eles as tarefas mais repetitivas
- ▶ na linguagem Python temos duas estruturas de repetição diferentes ao nosso dispor
- ▶ estudaremos primeiro a estrutura de repetição **while**



# Estrutura de repetição `while`

Formato geral:

```
while condição:  
    :  
    :  
    bloco de instruções  
    :  
    :
```

A indentação é **necessária** para definir o trecho de código que compõe o `while`.

# Estrutura de repetição `while`

Exemplo:

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

# Mostra os 100 primeiros números inteiros positivos

numero = 1
while numero <= 100:
    print(numero)
    numero = numero + 1
exit(0)
```

# Estrutura de repetição `while`

Simulação da execução passo a passo de uma simplificação do programa anterior:

```
01  #!/usr/bin/env python3
02  # -*- coding: utf-8 -*-
03
04  numero = 1
05  while numero <= 10:
06      print(numero)
07      numero = numero + 1
08  exit(0)
```

# Exercícios

1. O programa abaixo pretende resolver o seguinte problema computacional: dado um número inteiro positivo  $n$ , somar os  $n$  primeiros inteiros positivos.

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

n = int(input("Informe n: "))
soma = 0
numero = 1
while numero <= n:
    soma = soma + numero
    numero = numero + 1
print("Soma dos %d primeiros inteiros é %d" % (n, soma))
exit(0)
```

Verifique se tal solução está correta. Faça algumas simulações passo a passo da execução do programa.

# Exercícios

2. Dado um número inteiro positivo  $n$ , imprimir os  $n$  primeiros naturais ímpares.

Exemplo:

Para  $n = 4$  a saída deverá ser 1, 3, 5, 7.

Faça ao menos uma simulação passo a passo da execução de sua solução.

3. O **fatorial** de um número inteiro  $n$ , denotado por  $n!$ , é dado pela seguinte fórmula:

$$n! = n \cdot (n - 1) \cdot (n - 2) \cdots 2 \cdot 1.$$

Dessa forma,  $5! = 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1 = 120$ . Por definição,  $0! = 1$ .

Dado um número inteiro não-negativo  $n$ , escreva uma função que calcule e devolva  $n!$  com a seguinte interface:

```
def fatorial(n):
```

2. Dado um número inteiro positivo  $n$ , imprimir os  $n$  primeiros naturais ímpares.

Exemplo:

Para  $n = 4$  a saída deverá ser 1, 3, 5, 7.

Faça ao menos uma simulação passo a passo da execução de sua solução.

3. O **fatorial** de um número inteiro  $n$ , denotado por  $n!$ , é dado pela seguinte fórmula:

$$n! = n \cdot (n - 1) \cdot (n - 2) \cdots 2 \cdot 1.$$

Dessa forma,  $5! = 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1 = 120$ . Por definição,  $0! = 1$ .

Dado um número inteiro não-negativo  $n$ , escreva uma função que calcule e devolva  $n!$  com a seguinte interface:

```
def fatorial(n):
```

4. Dado um número inteiro positivo  $n$ , imprimir as  $n$  primeiras potências de 2.

Exemplo:

Para  $n = 5$  a saída deverá ser 1, 2, 4, 8, 16.

Faça ao menos uma simulação passo a passo da execução de sua solução.

5. Dado um número inteiro positivo  $n$  e uma sequência de  $n$  inteiros, somar esses  $n$  números.

Exemplo:

Para  $n = 5$  e a sequência 5, -3, 6, 0, 12 a saída deve ser 20(=  $5 + (-3) + 6 + 0 + 12$ ).

Faça ao menos uma simulação passo a passo da execução de sua solução.

4. Dado um número inteiro positivo  $n$ , imprimir as  $n$  primeiras potências de 2.

Exemplo:

Para  $n = 5$  a saída deverá ser 1, 2, 4, 8, 16.

Faça ao menos uma simulação passo a passo da execução de sua solução.

5. Dado um número inteiro positivo  $n$  e uma sequência de  $n$  inteiros, somar esses  $n$  números.

Exemplo:

Para  $n = 5$  e a sequência 5, -3, 6, 0, 12 a saída deve ser 20(=  $5 + (-3) + 6 + 0 + 12$ ).

Faça ao menos uma simulação passo a passo da execução de sua solução.



6. Dado um número inteiro positivo  $n$  e uma sequência de  $n$  números inteiros, determinar a soma dos números inteiros positivos da sequência.

Exemplo:

Se  $n = 7$  e a sequência é 6, -2, 7, 0, -5, 8, 4, a saída deve ser 25.

Faça ao menos uma simulação passo a passo da execução de sua solução.

7. Dado um número inteiro positivo  $n$  e uma sequência de  $n$  inteiros positivos, somar os números pares e os números ímpares.

Exemplo:

Se  $n = 7$  e a sequência de números inteiros é 6, 1, 3, 14, 4, 22, 7 a saída deve ser 46(= 6 + 14 + 4 + 22) e 11(= 1 + 3 + 7).

Faça ao menos uma simulação passo a passo da execução de sua solução.

6. Dado um número inteiro positivo  $n$  e uma sequência de  $n$  números inteiros, determinar a soma dos números inteiros positivos da sequência.

Exemplo:

Se  $n = 7$  e a sequência é 6, -2, 7, 0, -5, 8, 4, a saída deve ser 25.

Faça ao menos uma simulação passo a passo da execução de sua solução.

7. Dado um número inteiro positivo  $n$  e uma sequência de  $n$  inteiros positivos, somar os números pares e os números ímpares.

Exemplo:

Se  $n = 7$  e a sequência de números inteiros é 6, 1, 3, 14, 4, 22, 7 a saída deve ser 46(= 6 + 14 + 4 + 22) e 11(= 1 + 3 + 7).

Faça ao menos uma simulação passo a passo da execução de sua solução.

8. Durante os 7 dias de uma determinada semana foram tomadas as temperaturas médias diárias de Campo Grande, MS. Determinar o número de dias dessa semana com temperaturas abaixo de zero. Faça ao menos uma simulação passo a passo da execução de sua solução.
9. Dado um número inteiro positivo  $n$  e uma sequência de  $n$  números inteiros, determinar quantos números da sequência são positivos e quantos são não-positivos. Um número é não-positivo se é negativo ou se é igual a 0 (zero).

Exemplo:

Se  $n = 6$  e a sequência de números inteiros é 6, -1, 0, 16, -5, 0 a saída deve ser 2 e 4.

Faça ao menos uma simulação passo a passo da execução de sua solução.

8. Durante os 7 dias de uma determinada semana foram tomadas as temperaturas médias diárias de Campo Grande, MS. Determinar o número de dias dessa semana com temperaturas abaixo de zero. Faça ao menos uma simulação passo a passo da execução de sua solução.
9. Dado um número inteiro positivo  $n$  e uma sequência de  $n$  números inteiros, determinar quantos números da sequência são positivos e quantos são não-positivos. Um número é não-positivo se é negativo ou se é igual a 0 (zero).

Exemplo:

Se  $n = 6$  e a sequência de números inteiros é 6, -1, 0, 16, -5, 0 a saída deve ser 2 e 4.

Faça ao menos uma simulação passo a passo da execução de sua solução.

10. Dado um número inteiro positivo  $n$  e uma sequência de  $n$  números inteiros positivos, determinar quantos números da sequência são pares e quantos são ímpares.

Exemplo:

Se  $n = 6$  e a sequência de números inteiros é 28, 5, 4, 9, 720, 566 a saída deve ser 4 e 2.

Faça ao menos uma simulação passo a passo da execução de sua solução.

11. Uma loja de discos anota diariamente durante uma determinada semana a quantidade de discos vendidos. Determinar em que dia dessa semana ocorreu a maior venda e qual foi a quantidade de discos vendida nesse dia. Faça ao menos uma simulação passo a passo da execução de sua solução.

10. Dado um número inteiro positivo  $n$  e uma sequência de  $n$  números inteiros positivos, determinar quantos números da sequência são pares e quantos são ímpares.

Exemplo:

Se  $n = 6$  e a sequência de números inteiros é 28, 5, 4, 9, 720, 566 a saída deve ser 4 e 2.

Faça ao menos uma simulação passo a passo da execução de sua solução.

11. Uma loja de discos anota diariamente durante uma determinada semana a quantidade de discos vendidos. Determinar em que dia dessa semana ocorreu a maior venda e qual foi a quantidade de discos vendida nesse dia. Faça ao menos uma simulação passo a passo da execução de sua solução.

12. Dados o número  $n$ , inteiro positivo, de estudantes de uma turma de Algoritmos e Programação e suas notas de primeira prova, determinar a maior e a menor nota obtidas por essa turma, onde a nota mínima é 0 e a nota máxima é 100. Faça ao menos uma simulação passo a passo da execução de sua solução.

13. Observe o algoritmo abaixo, que dada uma frase determina quantos espaços a frase contém.

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
frase = input("Informe uma frase: ")
esp = i = 0
while i < len(frase):
    if frase[i] == ' ':
        esp += 1
    i += 1
print("A frase tem %d espaços" % esp)
exit(0)
```

Escreva um programa que, dada uma cadeia de caracteres, conte a quantidade de letras minúsculas, letras maiúsculas, dígitos, espaços e símbolos de pontuação que essa cadeia possui (considere que se um caractere não é letra, nem dígito e nem espaço, então é uma pontuação). Dica: digite `help(str)` no console do Python e procure funções que ajudem nesse exercício.



13. Observe o algoritmo abaixo, que dada uma frase determina quantos espaços a frase contém.

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
frase = input("Informe uma frase: ")
esp = i = 0
while i < len(frase):
    if frase[i] == ' ':
        esp += 1
    i += 1
print("A frase tem %d espaços" % esp)
exit(0)
```

Escreva um programa que, dada uma cadeia de caracteres, conte a quantidade de letras minúsculas, letras maiúsculas, dígitos, espaços e símbolos de pontuação que essa cadeia possui (considere que se um caractere não é letra, nem dígito e nem espaço, então é uma pontuação). Dica: digite `help(str)` no console do Python e procure funções que ajudem nesse exercício.

14. Dado um número inteiro positivo  $n$  e uma sequência de  $n$  números inteiros, verificar se a sequência está em ordem crescente.

Exemplo:

Se  $n = 6$  e a sequência é 1, 5, 9, 12, 13, 26, dizemos que a sequência está em ordem crescente.

15. Dado um número inteiro positivo  $n$ , verificar se este número contém dois dígitos consecutivos iguais.

Exemplo:

Se  $n = 23667$ , então  $n$  contém dois dígitos consecutivos iguais (66).

16. Dado um número inteiro positivo  $n$ , verificar se o primeiro e o último dígito deste número são iguais.

Exemplo:

Se  $n = 5185$ , então  $n$  tem o primeiro e o último dígito iguais.

14. Dado um número inteiro positivo  $n$  e uma sequência de  $n$  números inteiros, verificar se a sequência está em ordem crescente.

Exemplo:

Se  $n = 6$  e a sequência é 1, 5, 9, 12, 13, 26, dizemos que a sequência está em ordem crescente.

15. Dado um número inteiro positivo  $n$ , verificar se este número contém dois dígitos consecutivos iguais.

Exemplo:

Se  $n = 23667$ , então  $n$  contém dois dígitos consecutivos iguais (66).

16. Dado um número inteiro positivo  $n$ , verificar se o primeiro e o último dígito deste número são iguais.

Exemplo:

Se  $n = 5185$ , então  $n$  tem o primeiro e o último dígito iguais.

14. Dado um número inteiro positivo  $n$  e uma sequência de  $n$  números inteiros, verificar se a sequência está em ordem crescente.

Exemplo:

Se  $n = 6$  e a sequência é 1, 5, 9, 12, 13, 26, dizemos que a sequência está em ordem crescente.

15. Dado um número inteiro positivo  $n$ , verificar se este número contém dois dígitos consecutivos iguais.

Exemplo:

Se  $n = 23667$ , então  $n$  contém dois dígitos consecutivos iguais (66).

16. Dado um número inteiro positivo  $n$ , verificar se o primeiro e o último dígito deste número são iguais.

Exemplo:

Se  $n = 5185$ , então  $n$  tem o primeiro e o último dígito iguais.

17. Dado um número inteiro positivo  $n$  e dois números naturais não nulos  $i$  e  $j$ , imprimir em ordem crescente os  $n$  primeiros naturais que são múltiplos de  $i$  ou de  $j$  ou de ambos.

Exemplo:

Para  $n = 6, i = 2$  e  $j = 3$  a saída deverá ser 0, 2, 3, 4, 6, 8.

18. Dados um número inteiro  $n > 0$  e  $n$  sequências de números inteiros, cada qual terminada por 0, determinar a soma dos números pares de cada sequência.

Exemplo:

Se  $n = 3$  e as sequências são

3, 1, 8, 7, 0

6, 4, 0

3, 12, 1, 6, 17, 9, 0

então a saída deverá ser 8, 10 e 18, respectivamente.

17. Dado um número inteiro positivo  $n$  e dois números naturais não nulos  $i$  e  $j$ , imprimir em ordem crescente os  $n$  primeiros naturais que são múltiplos de  $i$  ou de  $j$  ou de ambos.

Exemplo:

Para  $n = 6, i = 2$  e  $j = 3$  a saída deverá ser 0, 2, 3, 4, 6, 8.

18. Dados um número inteiro  $n > 0$  e  $n$  sequências de números inteiros, cada qual terminada por 0, determinar a soma dos números pares de cada sequência.

Exemplo:

Se  $n = 3$  e as sequências são

3, 1, 8, 7, 0

6, 4, 0

3, 12, 1, 6, 17, 9, 0

então a saída deverá ser 8, 10 e 18, respectivamente.

19. Para  $n > 0$  estudantes de uma determinada turma são dadas 3 notas de provas. Calcular a média aritmética das provas de cada estudante, a média da turma, o número de aprovados e o número de reprovados, onde o critério de aprovação é média  $\geq 5.0$ .
20. Dado um natural  $n$ , calcular e imprimir o valor da seguinte soma

$$\frac{1}{n} + \frac{2}{n-1} + \frac{3}{n-2} + \dots + \frac{n}{1}.$$

19. Para  $n > 0$  estudantes de uma determinada turma são dadas 3 notas de provas. Calcular a média aritmética das provas de cada estudante, a média da turma, o número de aprovados e o número de reprovados, onde o critério de aprovação é média  $\geq 5.0$ .
20. Dado um natural  $n$ , calcular e imprimir o valor da seguinte soma

$$\frac{1}{n} + \frac{2}{n-1} + \frac{3}{n-2} + \dots + \frac{n}{1}.$$



21. (a) Escreva uma função com a seguinte interface:

```
def mdc(a, b):
```

que receba dois números inteiros positivos  $a$  e  $b$  e calcule e devolva o máximo divisor comum entre eles utilizando o algoritmo de Euclides. Exemplo:

$$\begin{array}{c|c|c|c|c} & 1 & 1 & 1 & 2 \\ \hline 24 & 15 & 9 & 6 & 3 \\ \hline 9 & 6 & 3 & 0 & \end{array} = \text{mdc}(24,15)$$

- (b) Usando a função do item anterior, escreva um programa que receba  $n \geq 1$  números inteiros positivos e calcule o máximo divisor comum entre todos eles.

22. Um número inteiro  $a$  é dito ser **permutação** de um número inteiro  $b$  se os dígitos de  $a$  formam uma permutação dos dígitos de  $b$ . Exemplo: 5412434 é uma permutação de 4321445, mas não é uma permutação de 4312455.

*Observação:* considere que o dígito 0 (zero) não ocorre nos números.

- (a) Escreva uma função com a seguinte interface:

```
def conta_digitos(n, d):
```

que receba dois números inteiros  $n$  e  $d$ , com  $0 < d \leq 9$ , devolva um valor que representa o número de vezes que o dígito  $d$  ocorre no número  $n$ .

- (b) Usando a função do item anterior, escreva um programa que leia dois números inteiros positivos  $a$  e  $b$  e responda se  $a$  é permutação de  $b$  (pense em como a função **conta\_digitos** pode te ajudar).

23. Considere o seguinte processo para gerar uma sequência de números. Comece com um número inteiro positivo  $n$ . Se  $n$  é par, divida por 2. Se  $n$  é ímpar, multiplique por 3 e some 1. Repita esse processo com o novo valor de  $n$ , terminando quando  $n = 1$ . Por exemplo, a sequência de números a seguir é gerada para  $n = 22$ :

22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1

É conjecturado que este processo termina com  $n = 1$  para todo inteiro  $n > 0$ . Os números gerados nessa sequência são chamados de **ciclo de  $n$** . Ademais, para um número  $n$ , o **comprimento do ciclo de  $n$**  é o número de elementos gerados na sequência. No exemplo acima, o comprimento do ciclo de 22 é 16.

## 23. (continuação)

- (a) Escreva uma função com a seguinte interface:

```
def comprimento(n):
```

que receba um número inteiro  $n \geq 1$ , imprima o ciclo de  $n$  e devolva o comprimento do ciclo de  $n$ .

- (b) Escreva um programa que leia um número inteiro  $k > 0$  e uma sequência de  $k$  números inteiros positivos e, para cada um deles, mostre seu ciclo e seu comprimento do ciclo. Use a função do item (a).

24. Dizemos que um número natural  $n$  é **palíndromo** se lemos o número da esquerda para direita e também da direita para esquerda e obtemos o mesmo número.  
Exemplos: 567765 e 32423 são palíndromos, mas 567675 não é.  
(a) Escreva uma função com a seguinte interface:

```
def quebra(n):
```

que receba um número inteiro  $n > 0$  e devolva três números inteiros: o primeiro dígito de  $n$ , o último dígito de  $n$  e um inteiro que represente o número  $n$  sem seu primeiro e último dígitos.

Exemplo:

| valor inicial de $n$ | primeiro dígito | último dígito | miolo de $n$ |
|----------------------|-----------------|---------------|--------------|
| 732                  | 7               | 2             | 3            |
| 14738                | 1               | 8             | 473          |
| 78                   | 7               | 8             | 0            |
| 7                    | 7               | 7             | 0            |

## 24. (*continuação*)

- (b) Usando a função do item (a), escreva um programa que receba um número inteiro  $n > 0$  e verifique se  $n$  é palíndromo. Suponha que  $n$  não contém o dígito 0.