

Pensamento Computacional

Estruturas de Repetição



UNIVERSIDADE
CANDIDO
MENDES

EAD ■

■ Estruturas de repetição

■ Comando for:

- O comando for é uma estrutura que irá executar uma repetição de tarefas até que seja atingida uma determinada condição;

```
for <condição>:  
    <sequência de comandos>
```

■ Utilizando listas:

- O comando for irá executar uma série de interações até que seja atingida o final da lista:

```
lista = [10,11,12,13,14,15]  
for item in lista:  
    print(item)
```

10
11
12
13
14
15

```
cesta_de_frutas = ['banana','maça','pera','uva']  
for fruta in cesta_de_frutas:  
    print(fruta)
```

banana
maça
pera
uva



■ Estruturas de repetição

- Variáveis contadoras:
 - São utilizadas em conjunto com o comando de repetição (for ou while) para executar a contagem de repetições do loop;
 - Exemplo: contar_caracter
- Comando for:
 - Comando for em strings:
 - É possível interagir com variáveis do tipo string:

```
texto = 'Bem vindo a aula de Pensamento Computacional'
contar_caracter = 0
for caracter in texto:
    contar_caracter += 1
print('A frase: \'{}\'' possui {} caracteres (com espaços)'.format(texto, contar_caracter))
```

A frase: 'Bem vindo a aula de Pensamento Computacional' possui 44 caracteres (incluindo espaços)

```
texto = 'Pensamento Computacional'
for caracter in texto:
    print(caracter)
```

P
e
n
s
a
m
e
n
t
o

C
o
m
p
u
t
a
c
i
o
n
a
l



■ Estruturas de repetição

- Variáveis acumuladoras:
 - São utilizadas em conjunto com o comando repetição (for ou while) para executar somas de valores obtidos pelas repetições do loop;
 - Exemplo: soma_notas_turma
- Comando for:
 - Comando for utilizando a função range():
 - Utilizamos a função range() para executar a repetição por uma determinada quantidade de vezes;
 - range(start, stop step*);
 - start -> número que irá iniciar a interação;
 - stop -> número que irá finalizar a interação;
 - step -> * opcional; diferença entre os números da interação.
 - Exemplo: “calcular a média da nota de uma turma com 5 alunos”

```
soma_notas_turma = 0
notas_turma = 0
for num in range(1,6):
    notas_turma = float(input('Digite a nota do {}º aluno '.format(num)))
    soma_notas_turma = soma_notas_turma + notas_turma
media_notas_turma = soma_notas_turma/5
print('O somatório da nota dos alunos é igual à: {}'.format(soma_notas_turma))
print('A média da nota da turma é igual à: {}'.format(media_notas_turma))
```

```
Digite a nota do 1º aluno 8.5
Digite a nota do 2º aluno 9.0
Digite a nota do 3º aluno 7.0
Digite a nota do 4º aluno 6.5
Digite a nota do 5º aluno 5.5
O somatório da nota dos alunos é igual à: 36.5
A média da nota da turma é igual à: 7.3
```

```
for numero in range(0,10,2):
    print(numero, end=' ')
```

```
0 2 4 6 8
```



■ Estruturas de repetição

■ Comando while:

- O comando while é uma estrutura de repetição, assim como o comando for, que irá executar repetições de tarefas até que determinada condição seja atingida;
- Necessita do auxílio de uma variável contadora, que deve ser inicializada antes do comando while;
- Utilizando o comando while em listas:

```
lista = [5,6,7,8]
tamanho_lista = len(lista)
contador = 0
while (contador < tamanho_lista):
    print(lista[contador])
    contador += 1
```

5
6
7
8

```
cesta_de_frutas = ['banana', 'maça', 'pera', 'uva']
qtde_itens_cesta = len(cesta_de_frutas)
contador = 0
while (contador < qtde_itens_cesta):
    print(cesta_de_frutas[contador])
    contador += 1
```

banana
maça
pera
uva



■ Estruturas de repetição

■ Comando break e continue:

- O comando break, quando utilizado em uma estrutura de repetição, encerra o loop;
- O comando continue provoca uma interrupção no loop, mas sem sair do mesmo. Ele somente avança para a próxima interação da repetição;

```
contador = 1
while contador < 10:
    contador += 1
    if (contador%2 == 1):
        continue
    print(contador)
```

2
4
6
8
10

```
contador = 1
while contador < 20:
    contador += 1
    if (contador == 15):
        break
    if (contador%2 == 1):
        continue
    print(contador, end = ' ')
```

2 4 6 8 10 12 14



■ Estruturas de repetição

- Estruturas aninhadas:
 - Utilizar quando for necessário realizar condição ou repetição dentro de outra estrutura condicional ou de repetição:
 - Exemplo:
 - Um ano tem 12 meses -> 1º loop;
 - Cada mês contém uma certa quantidade de dias -> 2º loop;
 - Cada dia contém 24 horas -> 3º loop;
 - Exemplo 02:
 - Calcular o fatorial de 1 até 10:
 - Calcular a média da turma

```
for numero in range(0,11):
    fatorial = 1
    for valor in range(1,numero+1):
        fatorial *= valor
    print('Fatorial de {}: {}! = {}'.format(numero, numero, fatorial))
```

```
Fatorial de 0: 0! = 1
Fatorial de 1: 1! = 1
Fatorial de 2: 2! = 2
Fatorial de 3: 3! = 6
Fatorial de 4: 4! = 24
Fatorial de 5: 5! = 120
Fatorial de 6: 6! = 720
Fatorial de 7: 7! = 5040
Fatorial de 8: 8! = 40320
Fatorial de 9: 9! = 362880
Fatorial de 10: 10! = 3628800
```

```
media_ponderada_turma = 0
qtde_alunos = 0
qtde_alunos = int(input('Digite a quantidade de alunos: '))
for aluno in range(1,qtde_alunos+1):
    soma_notas_ponderadas = 0
    soma_peso = 0
    for prova in range(1,3):
        nota = float(input('Digite a nota da {}ª prova do {}º aluno: '.format(prova, aluno)))
        peso = int(input('Digite o peso da {}ª prova: '.format(prova)))
        soma_notas_ponderadas += (nota*peso)
        soma_peso += peso
    media_ponderada_turma = media_ponderada_turma + (soma_notas_ponderadas/soma_peso)
media_ponderada_turma = media_ponderada_turma/qtde_alunos
print('A média da turma é igual à: {}'.format(media_ponderada_turma))
```

```
Digite a quantidade de alunos: 2
Digite a nota da 1ª prova do 1º aluno: 9
Digite o peso da 1ª prova: 3
Digite a nota da 2ª prova do 1º aluno: 8
Digite o peso da 2ª prova: 5
Digite a nota da 1ª prova do 2º aluno: 7
Digite o peso da 1ª prova: 3
Digite a nota da 2ª prova do 2º aluno: 6
Digite o peso da 2ª prova: 5
A média da turma é igual à: 7.375
```





UNIVERSIDADE
CANDIDO
MENDES

EAD ■

