



Universidade Federal de Alagoas (UFAL)  
Campus Arapiraca



Ciência da  
Computação  
UFAL Arapiraca

# Algoritmos e Programação de Computadores (APC)

11

Tuplas e Listas

**Alexandre de Andrade Barbosa**

alexandre.barbosa@arapiraca.ufal.br

# Objetivos

## Objetivos

- Descrever os conteúdos:
  - Tuplas
  - Listas

# Tuplas e Listas

## Introdução

- Tuplas e listas também podem ser usados para armazenar mais de um valor
  - Tuplas são bastante similares a Listas, mas são imutáveis
  - Listas armazenam valores que podem ser de tipos diferentes

# Tuplas e Listas

## Listas

### Listas

- armazenam valores que podem ser de tipos diferentes
- os elementos armazenados podem ser acessados através de um índice (posição)
- são mutáveis:
  - os elementos armazenados podem ser alterados
  - é possível adicionar e remover elementos

# Tuplas e Listas

## Listas

### Arquivo: exemplo.py

```
1 lista = []  
2 lista = ["ufal", 4, 10.3]  
3 print(lista)  
4 print(lista[1])  
5 print(lista[0])  
6 print(lista[-2])  
7 print(len(lista))
```

### Saída esperada

```
1 ['ufal', 4, 10.3]  
2 4  
3 'ufal'  
4 4  
5 3
```

# Tuplas e Listas

## Listas

### Arquivo: exemplo.py

```
1 lista = ["ufal", 4, 10.3]
2 print(lista)
3 lista[1] = 5
4 print(lista)
```

### Saída esperada

```
1 ['ufal', 4, 10.3]
2 ['ufal', 5, 10.3]
```

# Tuplas e Listas

## Listas

### Listas - métodos

- `append(item)` - adiciona o 'item' no final da lista
- `extend(lista2)` - adiciona os elementos de 'lista2' no final da lista
- `remove(item)` - remove o 'item' da lista
- `pop()` - remove e retorna o último elemento da lista
- `reverse()` - inverte a ordem dos elementos da lista
- `sort()` - ordena os elementos da lista
- `count(item)` - retorna a quantidade de 'item' na lista
- `index(item)` - retorna a posição de 'item' na lista
- `len(lista)` - retorna a quantidade de elementos da lista
- `x in lista` - retorna 'True' se 'x' está presente na lista

# Tuplas e Listas

## Listas

### Arquivo: exemplo.py

```
1 lista = ["ufal", 4, 10.3]
2 item = lista.pop(1)
3 print(lista)
4 del lista[0]
5 print(lista)
6 lista.remove(10.3)
7 print(lista)
```

### Saída esperada

```
1 ['ufal', 10.3]
2 [10.3]
3 []
```



# Tuplas e Listas

## Listas

A instrução `for` é utilizada para iterar sobre uma sequência (obs.: execução em Python 2)

```
1 l = range(10)
```

Produz uma lista:

```
1 [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

# Tuplas e Listas

## Listas

A instrução `for` é utilizada para iterar sobre uma sequência (lista)

```
1 for <variavel> in <sequencia>:  
2     <instrução 1>  
3     <instrução 2>  
4     <instrução n>  
5 <próxima instrução após o for>
```

# Tuplas e Listas

## Listas

for itera sobre sequências

```
1 for num in [1, 2, 3, 4, 5]:  
2     print(num)
```

```
1 for palavra in ["UFAL", "ASA", "Arapiraca"]  
2     print(palavra)
```

# Tuplas e Listas

## Listas

### Listas - slices (:)

- `lista[inicio:final]` - retorna os elementos armazenados nas posições do intervalo `[inicio, final)`
- `lista[inicio:]` - omitindo o valor 'final', o slice irá até o último elementos da lista
- `lista[:final]` - omitindo o valor 'inicio', o slice iniciará do primeiro elemento da lista

# Tuplas e Listas

## Listas

### Arquivo: exemplo.py

```
1 lista = ["ufal", 4, 10.3]
2 print(lista[:3])
3 print(lista[1:3])
4 print(lista[2:])
5 print(lista[:])
6 print(lista[::-1])
```

### Saída esperada

```
1 ['ufal', 4, 10.3]
2 [4, 10.3]
3 [10.3]
4 ["ufal", 4, 10.3]
5 [10.3, 4, 'ufal']
```

# Tuplas e Listas

## Tuplas

### Tuplas

- Armazenam valores que podem ser de tipos diferentes
- Os elementos armazenados podem ser acessados através de um índice (posição)
- São imutáveis:
  - os elementos armazenados NÃO podem ser alterados
  - NÃO é possível adicionar e remover elementos

# Tuplas e Listas

## Tuplas

Os elementos das tuplas são separados por vírgula

```
1 # definindo uma lista
2 lista = [1,2,3]
3 print(lista)
4
5 # definindo uma tupla com parenteses
6 tupla = (1,2,3)
7 print(tupla)
8
9 # definindo uma tupla sem parenteses
10 tupla = 1,2,3
11 print(tupla)
```

Obs.: por convenção deve-se representar as tuplas usando parênteses

# Tuplas e Listas

## Tuplas

```
1 # definindo uma tupla
2 tupla = 1,2,3
3 print(tupla)
4
5 # acesso aos valores da tupla
6 print(tupla[1])
7 print(tupla[-1])
8
9 # slices também são permitidos
10 print(tupla[:2])
```



# Tuplas e Listas

## Tuplas

Uma Tupla que contém um único elemento deve ser finalizada com uma virgula

```
1 tupla = ("ufal")  
2 print(type(tupla)) # <type 'str'>  
3  
4 tupla = ("ufal", )  
5 print(type(tupla)) # <type 'tuple'>
```

# Tuplas e Listas

## Tuplas

### Tuplas são imutáveis

```
1 tupla = ("ufal", 4, 10.3)
2 tupla[0] = "universidade"
3
4 Traceback (most recent call last):
5   File "<stdin>", line 1, in <module>
6   TypeError: 'tuple' object does not support item assignment
```

# Exercícios

## Exercício

- 1 Escreva uma função que receba uma lista e retorne uma tupla com seus elementos

# Tuplas e Listas

## Exercícios

### Exercício

- 1 Leia números e salve estes em uma lista. Ao receber o valor 0 (zero) finalize a leitura de números. Ao final da leitura informe:
  - (a) o maior valor fornecido
  - (b) a soma de todos os valores armazenados
- 2 Dada uma lista conte a quantidade de vezes que um elemento (fornecido pelo usuário) aparece (ex. [1, 2, 3, 1, 4, 5, 6], e = 1, aparece 2 vezes)

# Tuplas e Listas

## Listas e Tuplas

### Construtores

```
1 lista = list((1,2))
2 lista.append(3)
3 print(lista)
4
5 tupla = tuple((1,2,3))
6 print(tupla)
```

# Resumo

## Resumo

Tipo	Declaração	Mutável	Duplicatas	Acesso
Lista	[ ]	Sim	Sim	Índices inteiros
Tupla	( )	Não	Sim	Índices inteiros
String	" " ou ' '	Não	Sim	Índices inteiros

# Exercícios

## Exercício

- 1 Crie um programa para realizar matrículas em uma disciplina. Discentes possuem um nome e uma matrícula, a matrícula é única para cada discente. Deve ser apresentada também a opção de listar os discentes matriculados. O programa permanece em execução até que o usuário deseje finalizar.

# Leituras recomendadas



Python tutorial

5.3. Tuples and Sequences

[docs.python.org/3/tutorial/datastructures.html#  
tuples-and-sequences](https://docs.python.org/3/tutorial/datastructures.html#tuples-and-sequences)




Python tutorial


Lists


[docs.python.org/3/tutorial/introduction.html#lists](https://docs.python.org/3/tutorial/introduction.html#lists)



## Vídeos recomendados

 [Curso em Vídeo \(Canal no Youtube\)](#)  
[Curso Python n16 - Tuplas](#)  
<https://www.youtube.com/watch?v=0LB3FSfjvao>

 [Curso em Vídeo \(Canal no Youtube\)](#)  
[Curso Python 17 - Listas \(Parte 1\)](#)  
<https://www.youtube.com/watch?v=N1hTsbW50eM>

 [Curso em Vídeo \(Canal no Youtube\)](#)  
[Curso Python 17 - Listas \(Parte 2\)](#)  
[https://www.youtube.com/watch?v=YV\\_JQmZNFsk](https://www.youtube.com/watch?v=YV_JQmZNFsk)

# Perguntas?

**Alexandre de Andrade Barbosa**  
[alexandre.barbosa@arapiraca.ufal.br](mailto:alexandre.barbosa@arapiraca.ufal.br)