



# **Programação Funcional**

## Problema - Grafo Euleriano

Alunos: Natan Ribeiro  
Jefferson Benedito

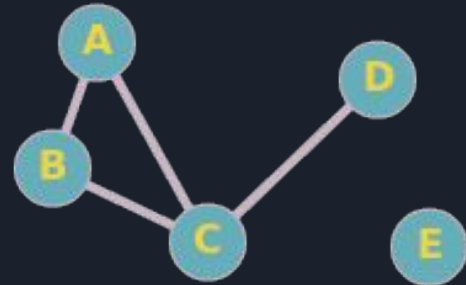
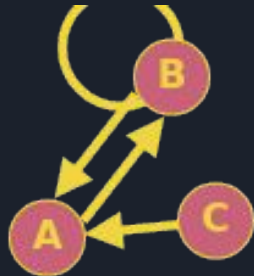


# Sobre o Problema...

1. Implementar um algoritmo que classifique um grafo como sendo Euleriano, semi-euleriano ou não tem caminho euleriano.

# Fundamentação Teórica

- Um **grafo** é um conjunto de vértices (nós) e um conjunto de arestas (pares de vértices distintos ligados entre si).
- Um **grafo simples**, é um grafo não direcionado, sem laços e sem arestas paralelas

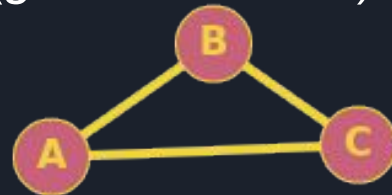


# Fundamentação Teórica


- **Caminho Euleriano** é um caminho em um grafo que visita cada aresta apenas uma vez (*grafo semi-euleriano*).



- **Circuito Euleriano** é um caminho Euleriano que começa e termina no mesmo vértice (*grafo euleriano*).



# Solução Proposta

- Assumimos que o algoritmo só reconhece **Grafos Simples**.
- Um vértice, é representado por um caracter. Ex: 'b'.
- Um vértice desconectado 'a', por exemplo, é representado por: ('a',' ') ou (' ','a') ou ('a','a'). 
- Dois vértices conectados entre si (uma aresta) 'a' e 'b', por exemplo, são representada por: ('a','b') ou ('b','a').



# Solução Proposta

- Um grafo é o conjunto de tuplas (arestas), distribuídas numa lista. Ex:  $[('a', 'b'), ('b', 'c')]$ .



- É contabilizada apenas uma aresta, grafos em situações deste tipo:  $[('a', 'b'), ('a', 'b')]$ ,  $[('a', 'b'), ('b', 'a')]$ ,  $[('a', 'b'), ('b', 'a'), ('b', ' ')]$ , etc.



- Arestas  $(' ', ' ')$ , são desconsideradas bem como vértices  $' '$ .



# Solução Proposta

```
*Main> main
```

```
"Inform the graph: (Ex: [('a', 'b'), ('b', 'c')])"  
[(' ', ' '), (' ', ' ')]
```

Then...

This graph has 0 edges.

This graph has 0 vertices.

Each vertex this graph, has the following degrees: []

It's not Eulerian graph



# Solução Proposta

```
*Main> main
```

```
"Inform the graph: (Ex: [('a', 'b'), ('b', 'c')])"
```

```
[('a', ' '), (' ', ' ')]
```

Then...



This graph has 0 edges.

This graph has 1 vertices.

Each vertex this graph, has the following degrees: [('a',0)]

It's not Eulerian graph





# Solução Proposta

\*Main> main

"Inform the graph: (Ex: [('a', 'b'), ('b', 'c')])"  
[('a', 'a'), ('b', 'b')]

Then...

This graph has 0 edges.

This graph has 2 vertices.

Each vertex this graph, has the following degrees: [('a',0),('b',0)]

It's not Eulerian graph



# Solução Proposta

\*Main> main

"Inform the graph: (Ex: [('a', 'b'), ('b', 'c')])"  
[('a', ' '), ('b', ' ')]

Then...

This graph has 0 edges.

This graph has 2 vertices.

Each vertex this graph, has the following degrees: [('a',0),('b',0)]

It's not Eulerian graph



# Solução Proposta

\*Main> main

"Inform the graph: (Ex: [('a', 'b'), ('b', 'c')])"  
[('a', 'b'), ('a', 'b'), ('b', 'a')]

Then...



This graph has 1 edges.

This graph has 2 vertices.

Each vertex this graph, has the following degrees: [('a',1),('b',1)]

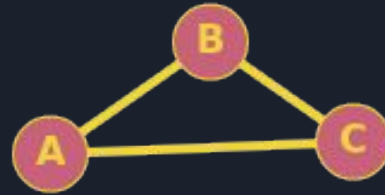
It's Semi-Eulerian graph

# Solução Proposta

\*Main> main

"Inform the graph: (Ex: [('a', 'b'), ('b', 'c')])"  
[('a', 'b'), ('c', 'b'), ('c', 'a')]

Then...



This graph has 3 edges.

This graph has 3 vertices.

Each vertex this graph, has the following degrees:

[('a',2),('b',2),('c',2)]

It's Eulerian cycle graph

# Solução Proposta

\*Main> main

"Inform the graph: (Ex: [('a', 'b'), ('b', 'c')])"  
[('a', 'b'), ('c', 'b'), ('c', 'a'), ('d', ' ')]

Then...

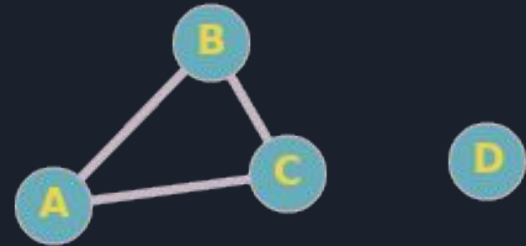
This graph has 3 edges.

This graph has 4 vertices.

Each vertex this graph, has the following degrees:

[('a',2),('b',2),('c',2),('d',0)]

It's not Eulerian graph



# Solução Proposta

\*Main> main

"Inform the graph: (Ex: [('a', 'b'), ('b', 'c')])"  
[('a', 'b'), ('c', 'b'), ('c', 'a'), ('d', 'e')]

Then...

This graph has 4 edges.

This graph has 5 vertices.

Each vertex this graph, has the following degrees:

[('a',2),('b',2),('c',2),('d',1),('e',1)]

It's not Eulerian graph

