

Teoria dos Grafos

Berilhes

Ordenação Topológica

Ordenação Topológica

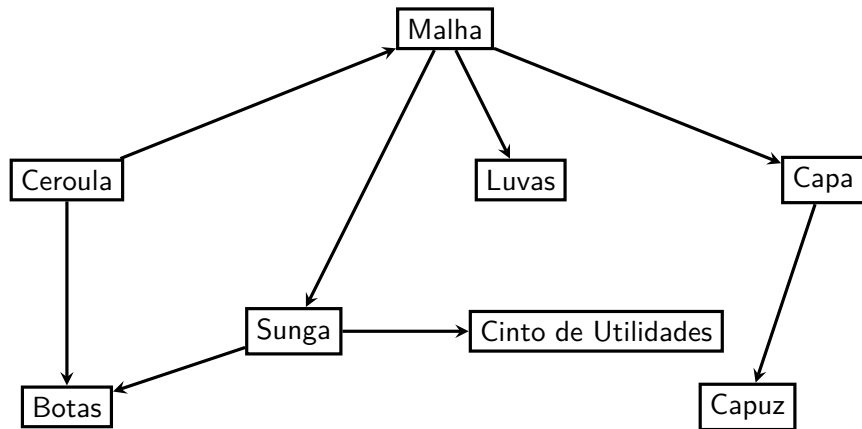
- Problema do Batman

Ordenação Topológica

■ Problema do Batman



Ordenação Topológica



Ordenação Topológica

Ordenação Topológica

- A ordenação topológica de um grafo dirigido G é uma ordem total $<$ sobre os vértices tal que $u < v$ para toda aresta (u, v) .

Ordenação Topológica

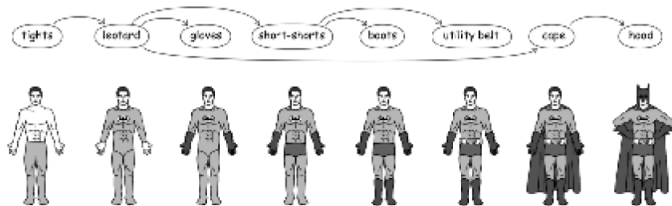
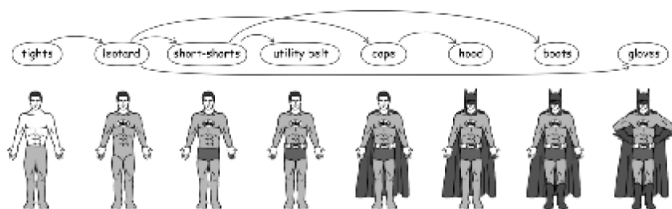
- A ordenação topológica de um grafo dirigido G é uma ordem total $<$ sobre os vértices tal que $u < v$ para toda aresta (u, v) .
- Uma ordenação topológica arranja os vértices ao longo de uma linha horizontal de modo que toda aresta aponte da esquerda para a direita.

Ordenação Topológica

- A ordenação topológica de um grafo dirigido G é uma ordem total $<$ sobre os vértices tal que $u < v$ para toda aresta (u, v) .
- Uma ordenação topológica arranja os vértices ao longo de uma linha horizontal de modo que toda aresta aponte da esquerda para a direita.
- Uma fonte é qualquer vértice com nenhuma aresta incidente, um sorvedor, por sua vez, é um vértice com nenhuma aresta saindo.

Ordenação Topológica

Ordenação Topológica



Ordenação Topológica

Ordenação Topológica

ORDENACAO_TOPOLOGICA1(V, E)

```
1   $n \leftarrow |V|$ 
2  for  $i \leftarrow 1$  to  $n$ 
3       $v \leftarrow$  qualquer vértice fonte em  $G$ 
4       $S[i] \leftarrow v$ 
5      remova  $v$  e todas as arestas saindo de  $v$  do grafo
6  return  $S$ 
```

Ordenação Topológica

ORDENACAO_TOPOLOGICA1(V, E)

```
1   $n \leftarrow |V|$ 
2  for  $i \leftarrow 1$  to  $n$ 
3       $v \leftarrow$  qualquer vértice fonte em  $G$ 
4       $S[i] \leftarrow v$ 
5      remova  $v$  e todas as arestas saindo de  $v$  do grafo
6  return  $S$ 
```

ORDENACAO_TOPOLOGICA2(V, E)

```
1   $n \leftarrow |V|$ 
2  for  $i \leftarrow n$  downto 1
3       $v \leftarrow$  qualquer vértice sorvedor em  $G$ 
4       $S[i] \leftarrow v$ 
5      remova  $v$  e todas as arestas incidentes em  $v$  do grafo
6  return  $S$ 
```

Ordenação Topológica

- A correção desses algoritmos segue indutivamente da observação que remover um vértice não pode criar um ciclo.

Ordenação Topológica

- A correção desses algoritmos segue indutivamente da observação que remover um vértice não pode criar um ciclo.
- O principal problema com esses algoritmos é que encontrar um vértice fonte (sorvedor) parece demandar tempo $O(|V|)$ no pior caso,

Ordenação Topológica

- A correção desses algoritmos segue indutivamente da observação que remover um vértice não pode criar um ciclo.
- O principal problema com esses algoritmos é que encontrar um vértice fonte (sorvedor) parece demandar tempo $O(|V|)$ no pior caso,
- o que faz com que o tempo de execução de cada algoritmo seja $O(|V|^2)$.

Ordenação Topológica

Algoritmo mais eficiente

Lema

Para qualquer grafo acíclico dirigido G , o primeiro vértice marcado finalizado pela pesquisa primeiro em profundidade deve ser um sorvedor.

É suficiente listar os vértices em ordem reversa em que eles são finalizados pela pesquisa primeiro em profundidade.

Ordenação Topológica

Ordenação Topológica

TOPOLOGICO(G)

```
1  for todo vértice  $v \in G.V$ 
2       $v.visitado \leftarrow \text{FALSO}$ 
3       $v.finalizado \leftarrow \text{FALSO}$ 
4       $v.numero \leftarrow 0$ 
5   $num \leftarrow n$ 
6  for todo vértice  $v \in G.V$ 
7      if  $v.visitado == \text{FALSO}$ 
8          DFS-TOP( $v, num$ )
```

Ordenação Topológica

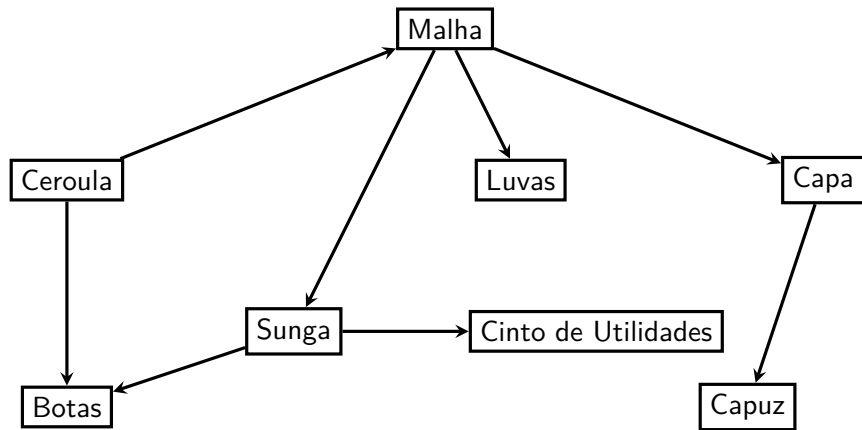
TOPOLOGICO(G)

```
1  for todo vértice  $v \in G.V$ 
2       $v.visitado \leftarrow \text{FALSO}$ 
3       $v.finalizado \leftarrow \text{FALSO}$ 
4       $v.numero \leftarrow 0$ 
5   $num \leftarrow n$ 
6  for todo vértice  $v \in G.V$ 
7      if  $v.visitado == \text{FALSO}$ 
8          DFS-TOP( $v, num$ )
```

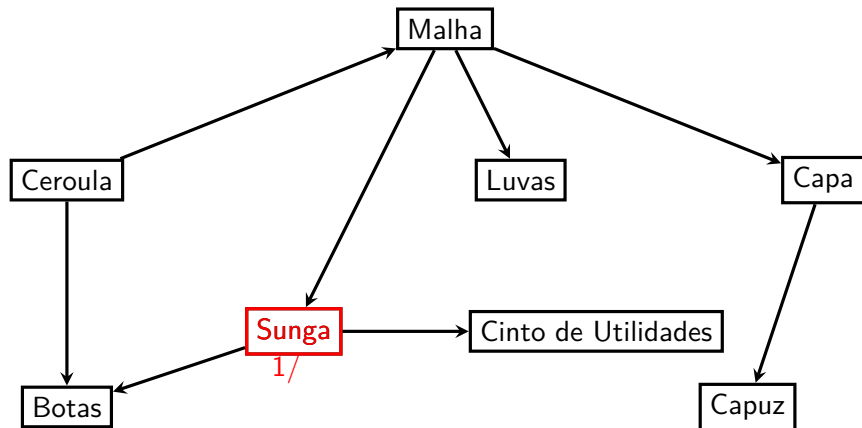
DFS-TOP(v, num)

```
1   $v.visitado \leftarrow \text{VERDADEIRO}$ 
2  for cada vértice  $w \in Adj[v]$ 
3      if  $w.visitado == \text{FALSO}$ 
4          DFS-TOP( $w, num$ )
5      elseif  $v.finalizado == \text{FALSO}$ 
6          PRINT(Erro - grafo cíclico)
7   $v.finalizado \leftarrow \text{VERDADEIRO}$ 
8   $v.numero \leftarrow num$ 
9   $num \leftarrow num - 1$ 
```

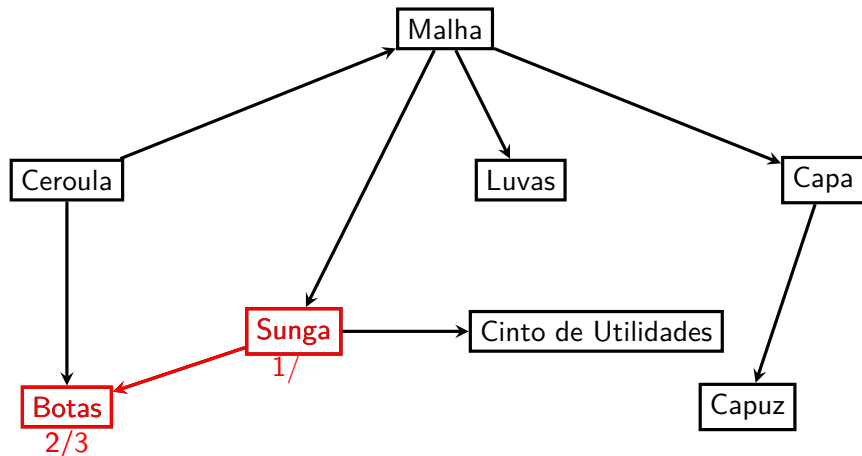
Ordenação Topológica



Ordenação Topológica

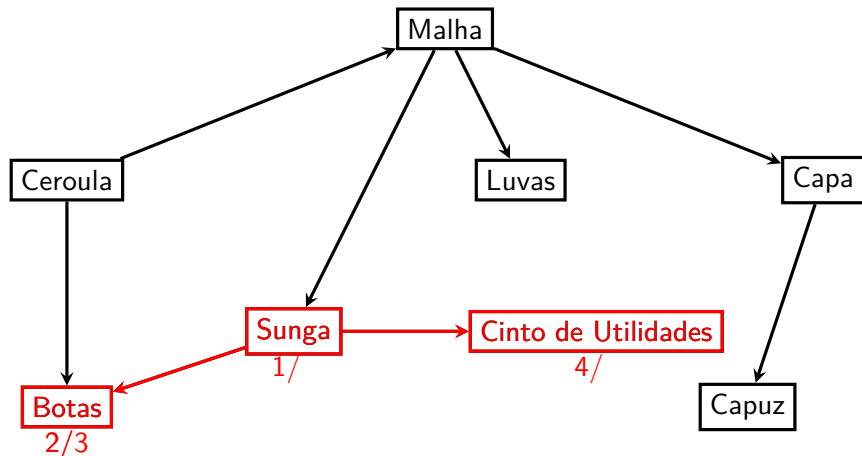


Ordenação Topológica



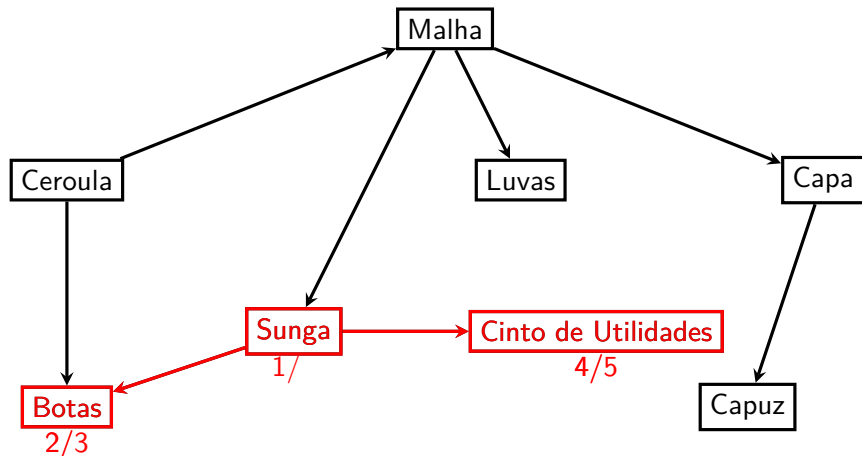
Botas

Ordenação Topológica



Botas

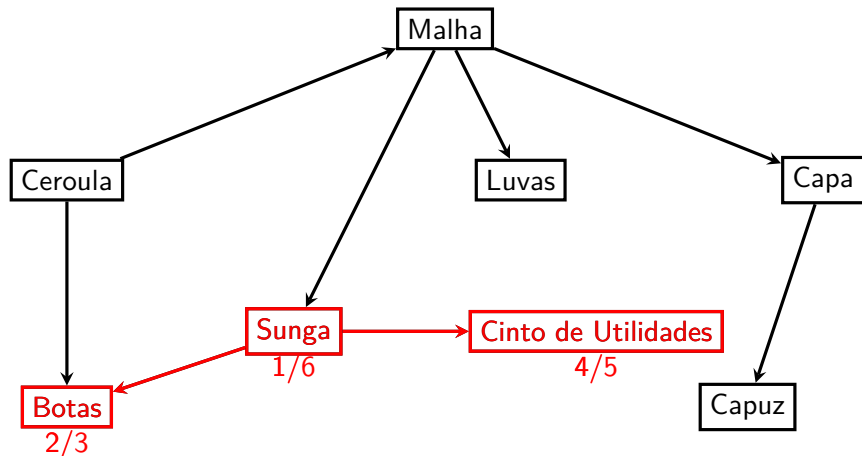
Ordenação Topológica



Cinto

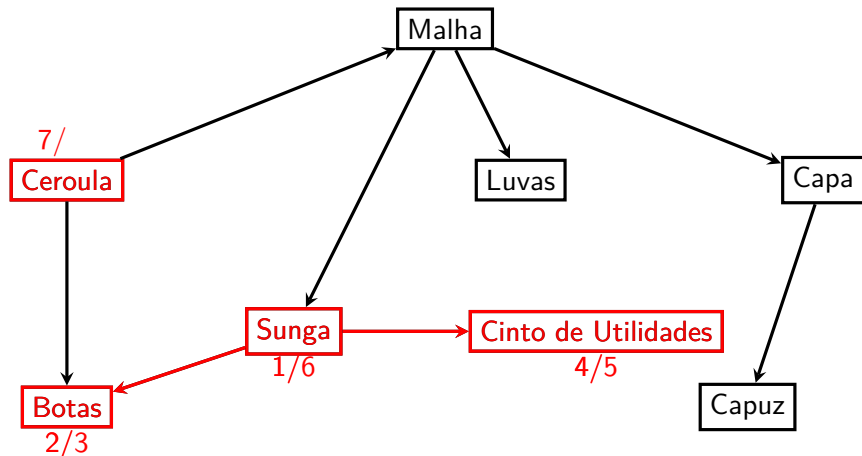
Botas

Ordenação Topológica



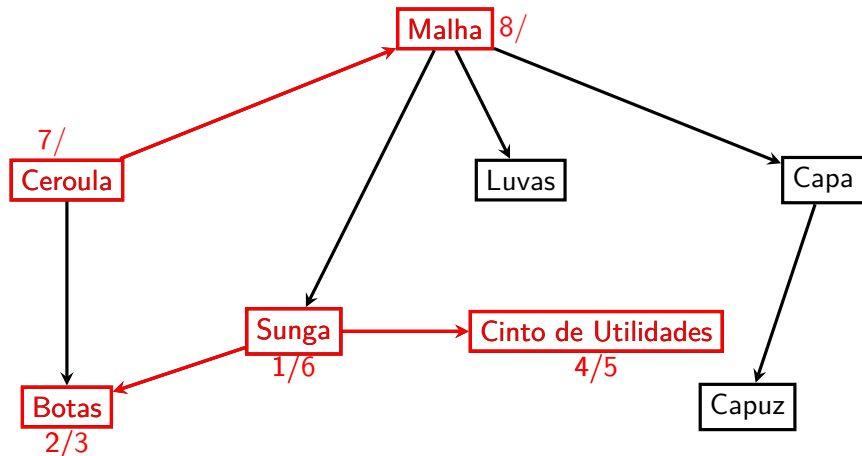
Sunga Cinto Botas

Ordenação Topológica



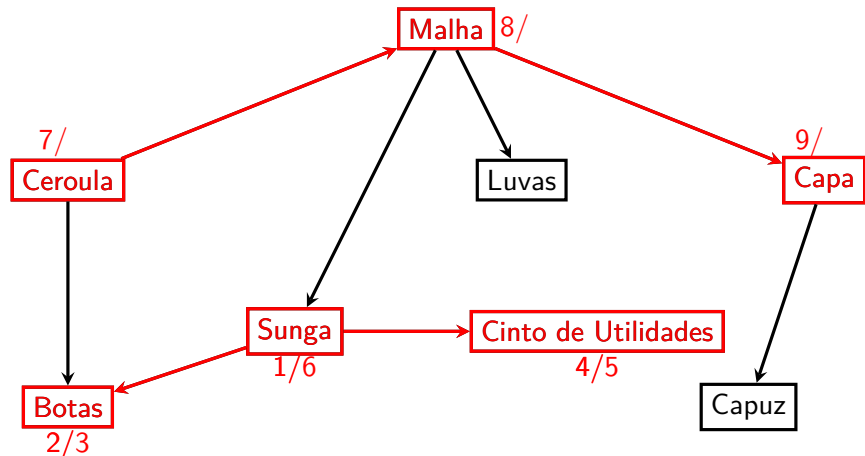
Sunga Cinto Botas

Ordenação Topológica

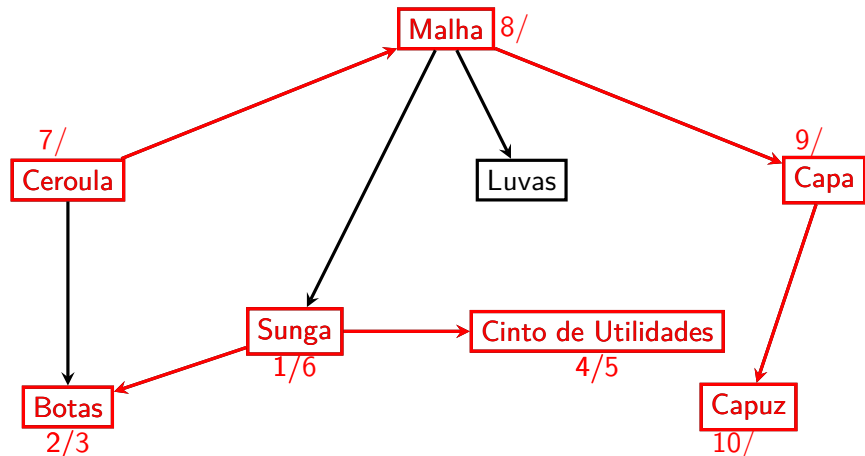


Sunga Cinto Botas

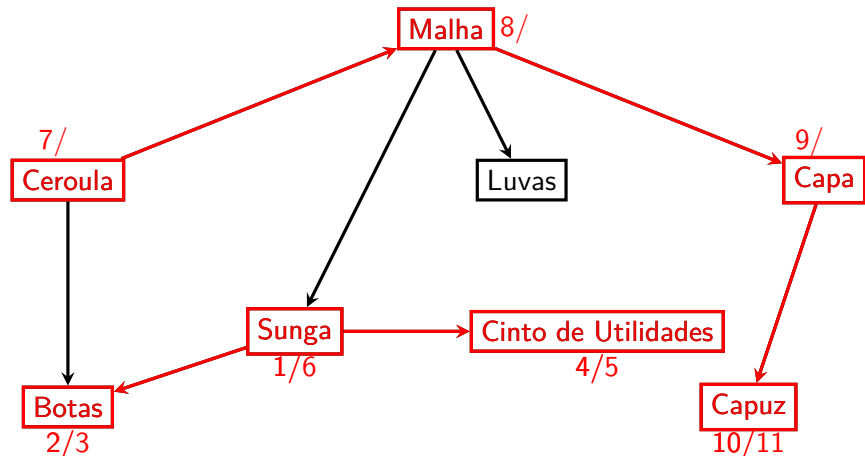
Ordenação Topológica



Ordenação Topológica

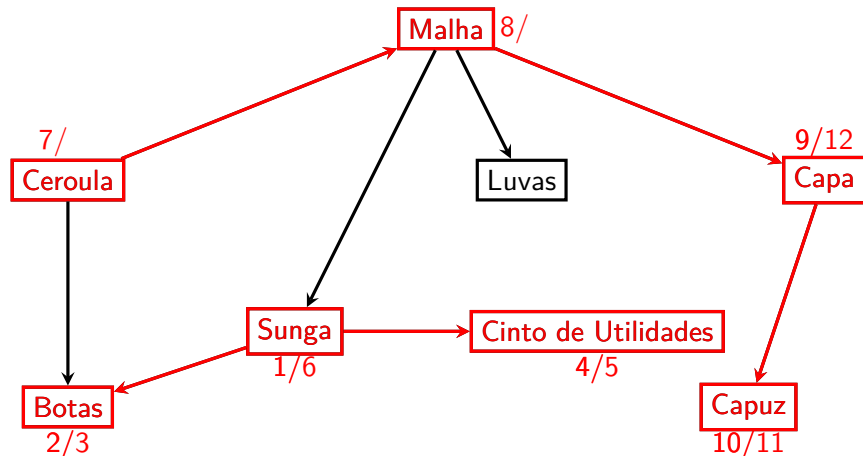


Ordenação Topológica



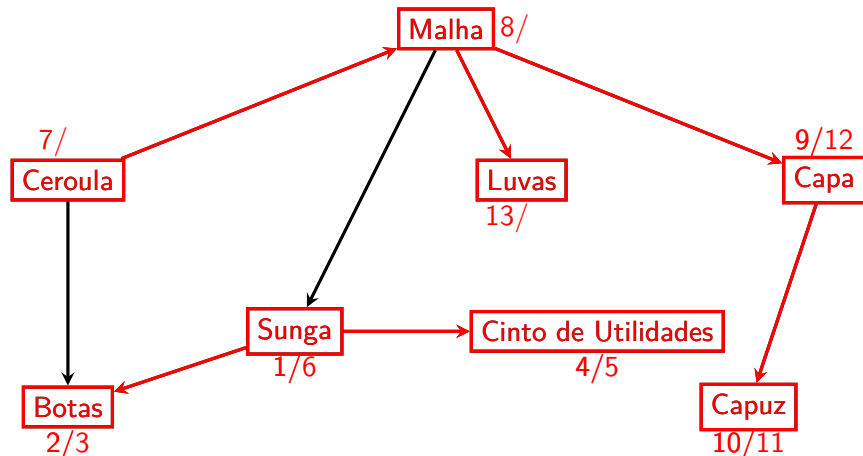
Capuz Sunga Cinto Botas

Ordenação Topológica



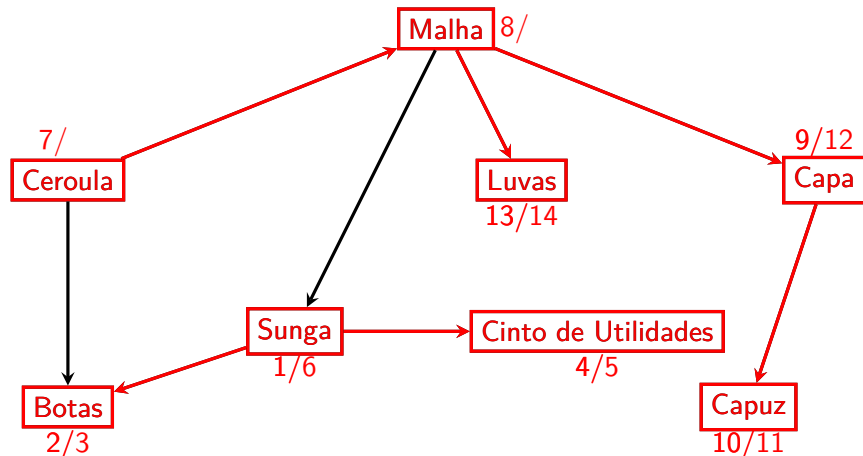
Capa Capuz Sunga Cinto Botas

Ordenação Topológica



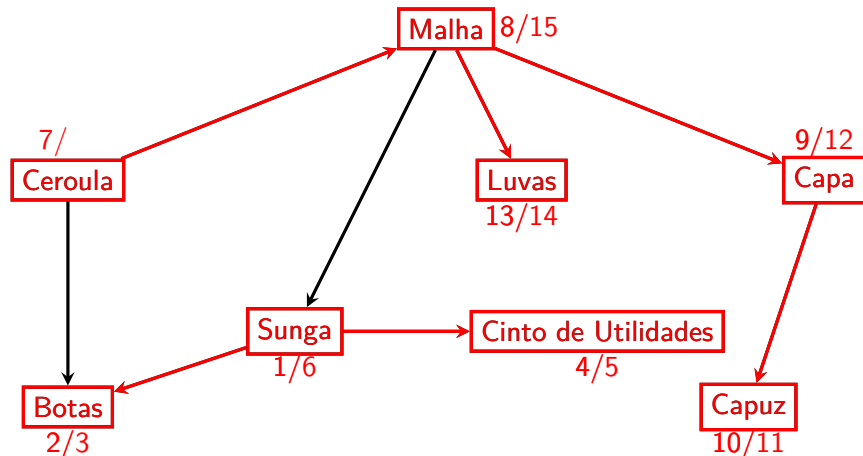
Capa Capuz Sunga Cinto Botas

Ordenação Topológica



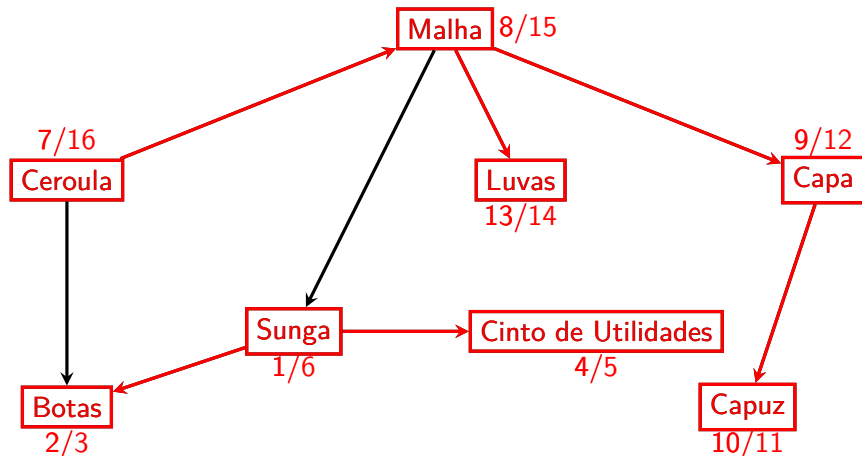
Luvas Capa Capuz Sunga Cinto Botas

Ordenação Topológica



Malha Luvas Capa Capuz Sunga Cinto Botas

Ordenação Topológica



Ceroula Malha Luvas Capa Capuz Sunga Cinto Botas

Código Sage

```
D = DiGraph({1: [2, 6], 2: [5, 3, 7],  
.....: 5: [8], 3: [6,4]})  
D.topological_sort()
```

[1, 2, 3, 4, 5, 8, 7, 6]

```
D = DiGraph({1: [2, 6], 2: [5, 3, 7],  
.....: 5: [8], 3: [6,4]})  
list(D.topological_sort_generator())
```

[(1, 2, 3, 5, 4, 6, 8, 7), (1, 2, 3, 4, 6, 7, 5, 8), ..., (1, 2, 3, 5, 4, 6, 7, 8)]

Ordenação Topológica

Conclusões

- Ordenação topológica tem aplicações em várias áreas:

Ordenação Topológica

Conclusões

- Ordenação topológica tem aplicações em várias áreas:
 - gerência de projeto,

Ordenação Topológica

Conclusões

- Ordenação topológica tem aplicações em várias áreas:
 - gerência de projeto,
 - escalonamento,

Ordenação Topológica

Conclusões

- Ordenação topológica tem aplicações em várias áreas:
 - gerência de projeto,
 - escalonamento,
 - avaliação de circuito, etc.

Ordenação Topológica

Conclusões

- Ordenação topológica tem aplicações em várias áreas:
 - gerência de projeto,
 - escalonamento,
 - avaliação de circuito, etc.
- Os vértices de um DAG podem ser ordenados em tempo linear ($O(|V| + |E|)$) por uma versão modificada do algoritmo DFS.