



UNIVERSIDADE DE ÉVORA

## 4º Trabalho

### Inteligência Artificial

**Professora:** Irene Rodrigues

**Realizado por:** Filipe Alfaiate (43315), Miguel de Carvalho (43108), João Pereira (42864)

1 de junho de 2021

## 1

### Vocabulário

Condições:

Eternas:

linha(L, C, C1) - A linha **L** com origem no local **C** e com o destino **C1**

Fluentes:

objLocal(O, L) - O objeto **O** está no local **L**

combLocal(C, L) - O comboio **C** está no local **L**

objComb(O, C) - O objeto **O** está no comboio **C**

Ações:

desloca(C, L) - O comboio **C** desloca-se para o local **L**

coloca(O, C) - O objeto **O** é colocado dentro do comboio **C**

tira(O, C) - O objeto **O** é retirado de dentro do comboio **C**

## 2

### Estado Inicial:

---

```
1 estado_inicial([
2     objLocal(1, porto),
3     objLocal(2, lisboa),
4     objLocal(3, lisboa),
5     objLocal(4, evora),
6     objLocal(5, evora),
7     combLocal(1, lisboa),
8     combLocal(2, lisboa),
9     linha(1, lisboa, porto),
10    linha(1, porto, lisboa),
11    linha(2, lisboa, evora),
12    linha(2, evora, lisboa)
13 ]).
```

---

### Estado Final:

---

```
1 estado_final1([
2     objLocal(1, evora),
3     objLocal(2, porto),
4     objLocal(3, evora),
5     objLocal(4, porto),
6     objLocal(5, lisboa)
7 ]).
```

---

## 3

Considerando o estado inicial da pergunta anterior até ao estado 1 escrito nesta pergunta a solução do pop seria:

---

```
1 [s1-inicial,
2   s5-desloca(1, porto),
3   s4-coloca(1, 1),
4   s427-desloca(1, lisboa),
5   s3-tira(1, 1),
6   s2-final]
```

---

Um exemplo de **ameaça**:

---

```
1 ameaca(s5, s4, s427, combLocal(1, porto))
```

---

Significa que o **s427** é ameaçado pelo **s5** e **s4**, o que vai originar uma ordenação topológica **s5 < s4 < s427**.

Tendo todas as **ameaças resolvidas** vamos obter uma ordenação topológica do resultado final **s1 < s5 < s4 < s427 < s3 < s2**.

## 4

A solução deste problema não pode ser obtida diretamente pois o programa excede o limite de memória. Para solucionar este problema, dividimos em sub-problemas, de forma a não exceder o limite de memória.

### 1º Sub-problema:

#### Estado Inicial:

---

```
1 estado_inicial([
2     objLocal(1, porto),
3     objLocal(2, lisboa),
4     objLocal(3, lisboa),
5     objLocal(4, evora),
6     objLocal(5, evora),
7     combLocal(1, lisboa),
8     combLocal(2, lisboa),
9     linha(1, lisboa, porto),
10    linha(1, porto, lisboa),
11    linha(2, lisboa, evora),
12    linha(2, evora, lisboa)
13 ]).
```

---

### Estado Final:

---

```
1 estado_final([
2     objLocal(1, lisboa),
3     objLocal(2, lisboa),
4     objLocal(3, lisboa),
5     objLocal(4, lisboa),
6     objLocal(5, lisboa)
7 ]).
```

---

Considerando os **estados iniciais e finais** referidos acima, a solução **pop** seria

---

```
1 [s1-inicial,
2   s1059-desloca(2, evora),
3   s5-desloca(1, porto),
4   s1009-coloca(4, 2),
5   s1104-coloca(5, 2),
6   s4-coloca(1, 1),
7   s1083-desloca(2, lisboa),
8   s427-desloca(1, lisboa),
9   s1096-tira(5, 2),
10  s3-tira(1, 1),
11  s428-tira(4, 2),
12  s2-final]
```

---

Um exemplo de **ameaça**:

---

```
1 ameaca(s1059, s1104, s1083, combLocal(2, evora))
```

---

Significa que o **s1083** é ameaçado pelo **s1059** e **s1104**, o que vai originar uma ordenação topológica **s1059 < s1104 < s1083**.

## 2º Sub-problema:

### Estado Inicial:

---

```
1 estado_inicial([
2     objLocal(1, lisboa),
3     objLocal(2, lisboa),
4     objLocal(3, lisboa),
5     objLocal(4, lisboa),
6     objLocal(5, lisboa),
7     combLocal(1, lisboa),
8     combLocal(2, lisboa),
9     linha(1, lisboa, porto),
10    linha(1, porto, lisboa),
11    linha(2, lisboa, evora),
12    linha(2, evora, lisboa)
13 ]).
```

---

### Estado Final:

---

```
1 estado_final([
2     objLocal(1, evora),
3     objLocal(2, porto),
4     objLocal(3, evora),
5     objLocal(4, porto),
6     objLocal(5, lisboa)
7 ]).
```

---

Considerando os **estados iniciais e finais** referidos acima, a solução pop seria

---

```
1 [s1-inicial,
2   s11938-coloca(1, 2),
3   s17552-coloca(2, 1),
4   s18110-coloca(3, 2),
5   s18153-coloca(4, 1),
6   s17129-desloca(2, evora),
7   s17967-desloca(1, porto),
8   s17551-tira(2, 1),
9   s18017-tira(3, 2),
10  s18152-tira(4, 1),
11  s3-tira(1, 2),
12  s2-final]
```

---

Um exemplo de **ameaça**:

---

```
1 ameaca(s1, s18153, s17967, combLocal(1, lisboa))
```

---

Significa que o s17967 é ameaçado pelo s18153 e s1, o que vai originar uma ordenação topológica  $s1 < s18153 < s17967$ .

## Solução do Problema:

### Estado Inicial:

---

```
1 estado_inicial([
2     objLocal(1, porto),
3     objLocal(2, lisboa),
4     objLocal(3, lisboa),
5     objLocal(4, evora),
6     objLocal(5, evora),
7     combLocal(1, lisboa),
8     combLocal(2, lisboa),
9     linha(1, lisboa, porto),
10    linha(1, porto, lisboa),
11    linha(2, lisboa, evora),
12    linha(2, evora, lisboa)
13 ]).
```

---

### Estado Final:

---

```
1 estado_final([
2     objLocal(1, evora),
3     objLocal(2, porto),
4     objLocal(3, evora),
5     objLocal(4, porto),
6     objLocal(5, lisboa)
7 ]).
```

---

Considerando os **estados iniciais e finais** referidos acima, a solução pop seria

---

```
1 [s1-inicial,
2   s1059-desloca(2, evora),
3   s5-desloca(1, porto),
4   s1009-coloca(4, 2),
5   s1104-coloca(5, 2),
6   s4-coloca(1, 1),
7   s1083-desloca(2, lisboa),
8   s427-desloca(1, lisboa),
9   s1096-tira(5, 2),
10  s6-tira(1, 1),
11  s428-tira(4, 2),
12  s11938-coloca(1, 2),
13  s17552-coloca(2, 1),
14  s18110-coloca(3, 2),
15  s18153-coloca(4, 1),
16  s17129-desloca(2, evora),
17  s17967-desloca(1, porto),
18  s17551-tira(2, 1),
19  s18017-tira(3, 2),
20  s18152-tira(4, 1),
21  s3-tira(1, 2),
22  s2-final]
```

---

Devido ao programa não executar num todo, não foi possível calcular as ameaças, contudo nos sub-problemas usados para calcular o **problema total** estão referidas algumas ameaças

Tendo todas as **ameaças resolvidas** vamos obter uma ordenação topológica do resultado final

---

```
1 s1 < s1059 < s5 < s1009 < s1104 < s4 < s1083 < s427 < s1096 < s6 < s428 <
   s11938 < s17552 < s18110 < s18153 < s17129 < s17967 < s17551 < s18017 <
   s18152 < s3 < s2
```

---