

2º Trabalho de Inteligência Artificial

2020/2021

1. Considere o seguinte problema do tipo 'zebra':

Tem uma mesa rectangular de n (par) lugares: duas pessoas sentam-se na cabeceira e $(n - 2)/2$ de cada lado. Pretende-se impor restrições do tipo:

- 'A' senta-se à frente de 'B' – frente('A','B').
- 'A' senta-se ao lado de 'B' (à esquerda ou à direita) – lado('A','B').
- 'A' senta-se à esquerda de 'B' – esq('A','B').
- 'A' senta-se à direita de 'B' – dir('A','B').
- 'A' está na cabeceira – cabeceira('A').
- 'A' está na cabeceira esquerda – cabeceira_esq('A').
- 'A' está na cabeceira direita – cabeceira_dir('A').

A especificação do problema é feita com os seguintes predicados:

- lugares(N) – indica o número de lugares da mesa.

ex:

lugares(8).

```
  2 3 4
1      5
  8 7 6
```

- pessoas(lista de nomes) – lista com os nomes das pessoas que se devem sentar na mesa

(ex: pessoas(['Maria', 'Manuel', 'Madalena', 'Joaquim', 'Ana', 'Julio', 'Matilde', 'Gabriel'])).

- restricoes(lista de restrições) — lista de restrições a que deve obedecer.

ex:

```
restricoes([esq('Manuel','Maria'), frente('Joaquim','Maria'),
            lado('Joaquim','Matilde'), cabeceira('Gabriel') ]).
```

solução possível:

Manuel - 1

Joaquim - 2
 Matilde - 3
 Ana - 4
 Gabriel - 5
 Julio - 6
 Madalena - 7
 Maria - 8

- (a) Represente este problema como um problema de satisfação de restrições em prolog.

No relatório indique como é que representa os estados, as variáveis (nome, domínio e valor) e as restrições.

Também deve indicar no relatório como calcula o estado inicial e apresentar o operador sucessor.

Nota: Não se esqueça que neste problema não podem estar duas pessoas sentadas no mesmo lugar.

- (b) Resolva o problema com o algoritmo de backtracking.
 (c) Resolva o problema modificando o algoritmo anterior para que faça verificação para a frente (forward checking).
 (d) Modifique o algoritmo anterior como entender de forma melhorar a complexidade (temporal e espacial).
 (e) No relatório apresente os resultados para 4 exemplos diferentes:
- i. uma mesa com 4 pessoas.
 - ii. uma mesa com 6 pessoas
 - iii. uma mesa com 8 pessoas
 - iv. uma mesa com 12 pessoas

2. Considere o problema do sudoku.

Exemplo:

	1				8		7	3
			5		9			
7						9		4
					4			
				3	5		1	8
8			9					
			7					
2	6			4			3	
		5			3			

- (a) Represente este problema como um problema de satisfação de restrições em prolog.

No relatório indique como é que representa os estados, as variáveis (nome, domínio e valor) e as restrições.

Também deve indicar no relatório como calcula o estado inicial e apresentar o operador sucessor.

Nota: Não se esqueça de indicar como lida com as casas que já estão preenchidas.

- (b) Resolva o problema com o algoritmo de backtracking.
- (c) Resolva o problema modificando o algoritmo anterior para que faça verificação para a frente (forward checking).
- (d) Modifique o algoritmo anterior como entender de forma melhorar a complexidade (temporal e espacial).

3. Deve entregar o trabalho até às 17 horas do dia 26/4/2021.