Aprendizagem e Decisão Inteligente		**2017/2018**
Filipe Rafael Soares	76543	*Grupo 59*
Ana Luísa Santo	79758	HW-2

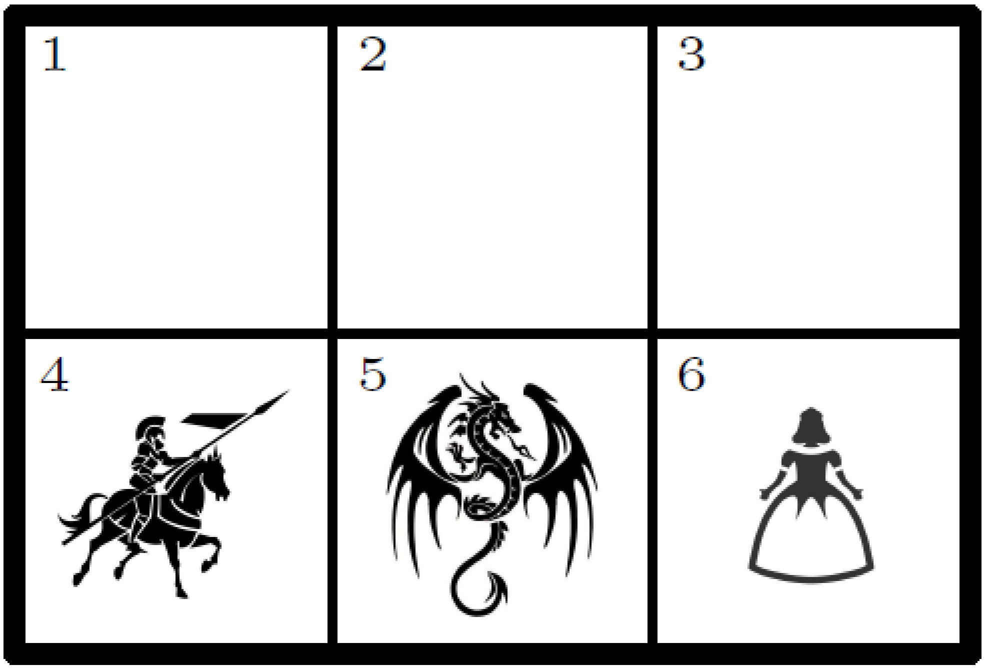

Figura 1 – Mundo/grelha do cenário em discussão neste documento.

1)

De modo a tornar mais simples a representação em termos de código do espaço de estados ($\boldsymbol{\chi}$), consideremos a Figura 1, uma matriz 2 por 3, tendo como índice inicial o zero em vez de 1. No que diz respeito ao espaço de ações ($\mathbf{A}$), estas dizem respeito apenas ao cavaleiro que é a única entidade que se movimenta, podendo efetuas as ações cima (**U**), baixo (**D**), esquerda (**L**) e direita (**R**).

Assim sendo obtemos o seguinte:

$$\chi = \begin{Bmatrix} (0,0) & (0,1) & (0,2) \\ (1,0) & (1,1) & (1,2) \end{Bmatrix} \qquad \mathbf{A} = \{U,\ D,\ L,\ R\}$$

2)

Para cada possível ação tem de constar uma matriz de transição associada à mesma. Assim sendo temos uma matriz de transição associada à ação cima (**P_U**), baixo (**P_D**), esquerda (**P_L**) e direita (**P_R**).

```
A matriz de transicao associada a acao U e:
[[ 0.8  0.1  0.   0.1  0.   0. ]
 [ 0.1  0.7  0.1  0.   0.1  0. ]
 [ 0.   0.1  0.8  0.   0.   0.1]
 [ 0.6  0.   0.   0.3  0.1  0. ]
 [ 0.   0.6  0.   0.1  0.2  0.1]
 [ 0.   0.   0.6  0.   0.1  0.3]]

A matriz de transicao associada a acao L e:
[[ 0.8  0.1  0.   0.1  0.   0. ]
 [ 0.6  0.2  0.1  0.   0.1  0. ]
 [ 0.   0.6  0.3  0.   0.   0.1]
 [ 0.1  0.   0.   0.8  0.1  0. ]
 [ 0.   0.1  0.   0.6  0.2  0.1]
 [ 0.   0.   0.1  0.   0.6  0.3]]

A matriz de transicao associada a acao D e:
[[ 0.3  0.1  0.   0.6  0.   0. ]
 [ 0.1  0.2  0.1  0.   0.6  0. ]
 [ 0.   0.1  0.3  0.   0.   0.6]
 [ 0.1  0.   0.   0.8  0.1  0. ]
 [ 0.   0.1  0.   0.1  0.7  0.1]
 [ 0.   0.   0.1  0.   0.1  0.8]]

A matriz de transicao associada a acao R e:
[[ 0.3  0.6  0.   0.1  0.   0. ]
 [ 0.1  0.2  0.6  0.   0.1  0. ]
 [ 0.   0.1  0.8  0.   0.   0.1]
 [ 0.1  0.   0.   0.3  0.6  0. ]
 [ 0.   0.1  0.   0.1  0.2  0.6]
 [ 0.   0.   0.1  0.   0.1  0.8]]
```

Figura 2 – Representação matricial em Python das matrizes de transição P_U, P_D, P_L e P_R.

Para finalizar este exercício definiu-se a matriz de custo como consta na Figura 3. A definição da matriz de custo foi com base no seguinte pensamento:

1. O objetivo é chegar à princesa evitando o dragão. O custo é calculado tendo em conta a posição onde esta cavaleiro no tabuleiro e as ações que podem tomar nessa posição.
2. Para o caso da posição do dragão, o custo associado para qualquer ação tomada, deverá ser o máximo dos custos da matriz, neste caso definimos como 1.
3. O mesmo pensamento é aplicado na casa da princesa, em que o custo deverá ser o mínimo dos custos da matriz, e por isso, definimos como 0.
4. Todas as outras casas deverão ter um custo superior a 0 e inferior a 1 (definidos como o custo da posição da princesa e do dragão, respetivamente). Neste caso o valor será arbitrário, e por isso definimos como 0.5. Este custo é uniforme face às ações tomadas e às posições pois o ambiente não é determinístico – existe probabilidades inerentes as ações tomadas.

```
[[ 0.5  0.5  0.5  0.5]
 [ 0.5  0.5  0.5  0.5]
 [ 0.5  0.5  0.5  0.5]
 [ 0.5  0.5  0.5  0.5]
 [ 1.   1.   1.   1. ]
 [ 0.   0.   0.   0. ]]
```

Figura 3 – Representação matricial em Python da matriz de custos (X por A), mantendo a ordem dos espaços apresentados.

3)

Para cálculo da função cost-to-go, aplicou-se a fórmula 1 de acordo com a política pedida, obtendo-se os valores apresentados na Figura 4.

$$J^{\pi} = (I - \gamma P_{\pi})^{-1} C_{\pi} \quad \textbf{(1)}$$

```
[ 5.08360511  5.12162162  4.79477327  5.13848316  5.62162162  4.23989522]
```

Figura 4 – Representação do resultado obtido da fórmula 1.