Aprendizagem e Decisão Inteligente		**2017/2018**
Filipe Rafael Soares	76543	*Grupo 59*
Ana Luísa Santo	79758	HW-1

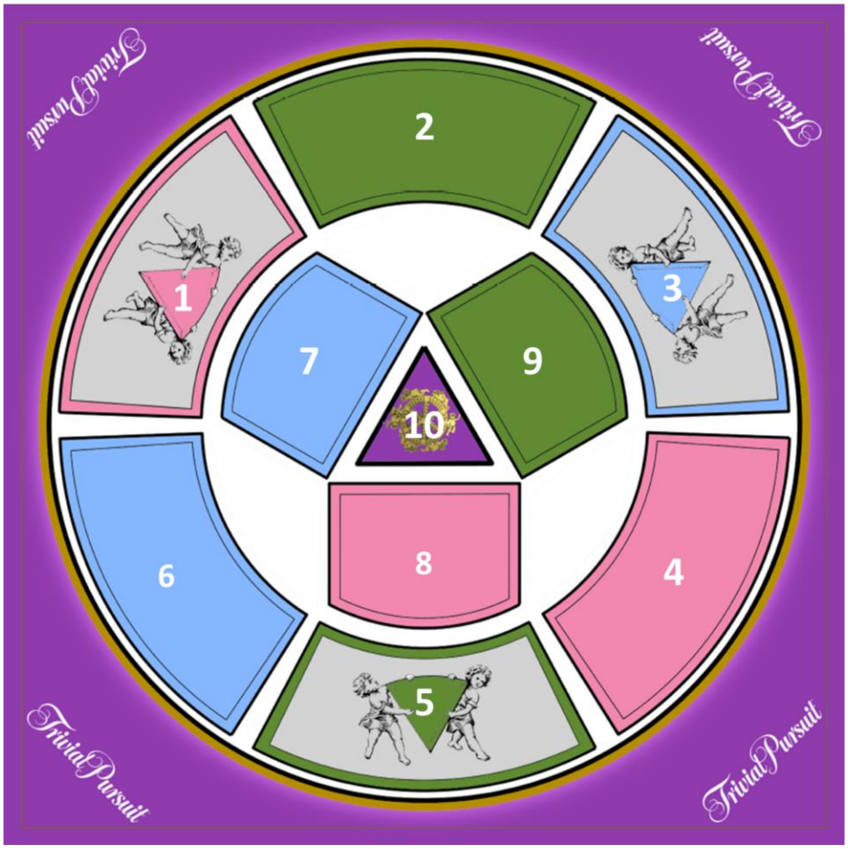

Figura 1 – Versão simplificada do Trivial Pursuit, com as casas numeradas de acordo a matrix

1)

De modo a representar uma Cadeia de Markov é necessário referir o Espaço de Estados (**χ**) e a Matriz de Probabilidade de Transição (**P**) entre os mesmos. Neste caso queremos representar uma Cadeia de Markov para o sistema representado na Figura 1, jogado com um dado não modificado numerado entre 1 e 6. É nos mencionado que o jogador tem a mesma probabilidade de escolher qualquer uma das casas, isto é, na casa 1 da Figura 1 este escolhe a entre as casas 6,7 e 2 com a mesma probabilidade e se tivesse na casa 2 escolhia entre 1 e 3 com a mesma probabilidade.

A matriz P irá resultar da soma de 6 matrizes, que representam a probabilidade de calhar os números entre 1 e 6 do dado. Após construção da matriz para o caso de calhar 1 e 2, reparamos que a matriz P poderia ser obtida com a seguinte expressão:

$$P = \frac{1}{6}\sum_{n=1}^{6} P_1^n \ (1)$$

Na formula 1, P_1 representa a matriz de transação entre o Espaço de Estados e ter saído 1 no dado. Podemos constatar, como o jogador pode andar livremente pelo tabuleiro, esta matriz elevada ao número calhado no dado, concede a matriz de transação para cada valor que possa sair no dado. A matriz P1 pode ser vista na Figura 2.

Assim sendo, a somatório de todas as matrizes P1 elevado aos possíveis valores do dado, resulta na matriz P não normalizada, sendo então normalizado pela constante 1/6.

```
[
[0, 1 / 3, 0, 0, 0, 1 / 3, 1 / 3, 0, 0, 0],
[1 / 2, 0, 1 / 2, 0, 0, 0, 0, 0, 0, 0],
[0, 1 / 3, 0, 1 / 3, 0, 0, 0, 1 / 3, 0, 0],
[0, 0, 1 / 2, 0, 1 / 2, 0, 0, 0, 0, 0],
[0, 0, 0, 1 / 3, 0, 1 / 3, 0, 0, 1 / 3, 0],
[1 / 2, 0, 0, 0, 1 / 2, 0, 0, 0, 0, 0],
[1 / 2, 0, 0, 0, 0, 0, 0, 0, 0, 1 / 2],
[0, 0, 1 / 2, 0, 0, 0, 0, 0, 0, 1 / 2],
[0, 0, 0, 0, 1 / 2, 0, 0, 0, 0, 1 / 2],
[0, 0, 0, 0, 0, 0, 1 / 3, 1 / 3, 1 / 3, 0],
]
```

Figura 2 – Representação matricial em Python, da matriz P1 (lista de listas).

Concluindo, aplicando a formula 1, obtemos a <u>matriz **P**</u> representada na Figura 3. O Espaço de Estados χ são todas as casas assinaladas numericamente na Figura 1.

```
[0.19 0.12 0.1  0.04 0.1  0.12 0.12 0.04 0.04 0.1 ]
[0.19 0.12 0.19 0.08 0.06 0.08 0.08 0.08 0.04 0.06]
[0.1  0.12 0.19 0.12 0.1  0.04 0.04 0.12 0.04 0.1 ]
[0.06 0.08 0.19 0.12 0.19 0.08 0.04 0.08 0.08 0.06]
[0.1  0.04 0.1  0.12 0.19 0.12 0.04 0.04 0.12 0.1 ]
[0.19 0.08 0.06 0.08 0.19 0.12 0.08 0.04 0.08 0.06]
[0.19 0.08 0.06 0.04 0.06 0.08 0.12 0.08 0.08 0.19]
[0.06 0.08 0.19 0.08 0.06 0.04 0.08 0.12 0.08 0.19]
[0.06 0.04 0.06 0.08 0.19 0.08 0.08 0.08 0.12 0.19]
[0.1  0.04 0.1  0.04 0.1  0.04 0.12 0.12 0.12 0.19]
```

Figura 3 – Representação matricial em Python da matriz P (lista de listas) no instante t=0.

2)

Se pretendemos saber que onde estará o jogador ao fim de 3 jogadas, tendo em conta que começou o jogo numa situação inicial (t=0), é apenas necessário multiplicar a matriz P 3 vezes por ela mesma, por outras palavras, elevar a matriz P ao número de jogadas pretendidas ($P^t \rightarrow P^3$). Assim sendo, a matriz que representa a probabilidade ao fim de 3 jogadas encontra-se representada na figura 4.

```
[0.13 0.08 0.12 0.08 0.12 0.08 0.08 0.08 0.08 0.12]
[0.13 0.09 0.13 0.08 0.12 0.08 0.08 0.08 0.08 0.12]
[0.12 0.08 0.13 0.08 0.12 0.08 0.08 0.08 0.08 0.12]
[0.12 0.08 0.13 0.09 0.13 0.08 0.08 0.08 0.08 0.12]
[0.12 0.08 0.12 0.08 0.13 0.08 0.08 0.08 0.08 0.12]
[0.13 0.08 0.12 0.08 0.13 0.09 0.08 0.08 0.08 0.12]
[0.13 0.08 0.12 0.08 0.12 0.08 0.09 0.08 0.08 0.13]
[0.12 0.08 0.13 0.08 0.12 0.08 0.08 0.09 0.08 0.13]
[0.12 0.08 0.12 0.08 0.13 0.08 0.08 0.08 0.09 0.13]
[0.12 0.08 0.12 0.08 0.12 0.08 0.08 0.08 0.08 0.13]
```

Figura 4 – Representa matricial da matriz P elevado a 3 em Python (lista de listas)