

1) Formule o problema, em termos de um algoritmo de pesquisa para solucionar um problema de equações algébricas (por exemplo  $x^2y^3 = 3 - xy$ ).

a) Indique, justificando, qual é o espaço de estados.

$\mathbb{R}^2$ , todos os números de reais para cada incógnita da equação

b) Indique qual ou quais os estados iniciais, operadores, e teste do objetivo entre outros elementos que entendam ser úteis para a compreensão da solução.

**Estado Inicial:** incógnitas  $x$  e  $y$  a 0

**Operadores:** incrementar ou decrementar as incógnitas  $x$  ou  $y$

**Teste de Objetivo:** verificar se Equação<sub>1</sub>  $\Leftrightarrow$  Equação<sub>2</sub>

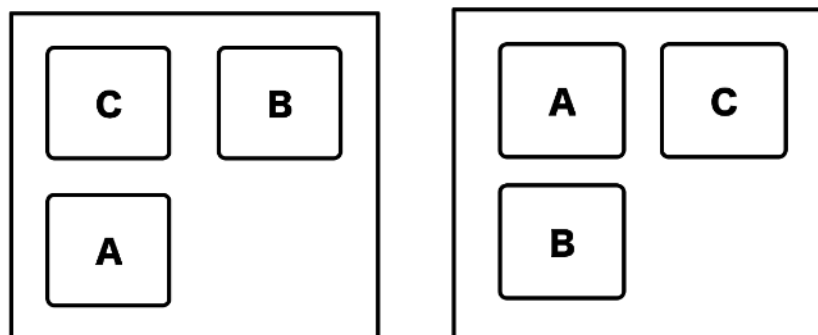
c) Especifique uma função custo para a utilização do algoritmo *Greedy Search* e explique porque considera que é uma boa função.

A função custo poderá ser o cálculo da diferença das expressões  $x^2y^3$  e  $3 - xy$  em módulo, assim sendo, podemos afirmar que a função custo é:

$$h(n) = |x^2y^3 + 3 - xy|$$

Assim sendo, considera-se uma função, uma vez que quando a diferença em módulo das duas expressões algébricas for 0 podemos afirmar que se encontrou a solução do problema e, quanto maior for a diferença maior será a distância da solução.

2) Considere o seguinte problema:



Pretende-se utilizar o algoritmo A com duas heurísticas:

$h1$  = soma das distancias de *Manhatam* (distancia em quadrículas para deslocar a peça do local onde está o seu lugar final)

$h2$  = numero de peças fora do lugar

Como regra está definido que as peças só movem na horizontal ou vertical e apenas para um espaço vazio

- a) Formule o problema considerando que mover uma peça tem custo 1.

Estado Inicial

C	B
A	0

Teste Objetivo → comparar o estado atual com o estado final/objetivo  
Operadores → mover o espaço vazio na horizontal ou na vertical  
Custo da Solução → 1 por cada movimento

- b) Desenha a árvore de pesquisa para a heurística  $h1$

- c) Desenha a árvore de pesquisa para a heurística  $h2$ .

- d) Discuta as diferenças entre heurísticas (vantagens e desvantagens de uma relativamente à outra).

Resolvendo o problema a partir da heurística  $H_2$ , o processo é muito mais lento, ocupa mais espaço em memória e ainda existe o risco de entrar num *loop*/repetição de estados, não chegando à solução final

- 3) No jogo do 4-em-linha o objetivo é criar uma sequência de 4 peças da mesma cor onde cada jogador usar uma cor e joga alternadamente com o seu oponente. O tabuleiro do jogo tem 7 colunas e 8 linhas e os jogadores colocam a sua peça na coluna que selecionam e a peça ficará na primeira linha disponível (por cima da última peça colocada nessa coluna).

- a) Formule o problema e explique como o pode resolver com o algoritmo minimax (use uma forma gráfica para melhor esclarecimento).

Para implementar um algoritmo minimax, é necessário representar a posição do tabuleiro e qual o próximo jogador a jogar (estado inicial). Para isso, vamos usar uma matriz de dimensão  $7 \times 6$ . Cada posição  $ij$  deverá conter a primeira inicial da cor de cada jogador (**P**reto ou **V**ermelho). Para além disso, deverá existir uma variável que contem a informação sobre o jogador a jogar, computador (*true*) ou utilizador (*false*). Uma vez que a peça tem que estar antes da anterior numa coluna, deverá existir um operador que verifique esta condição.

É importante a existência de uma função utilidade que verifique se o jogo já acabou e que atribua o resultado. Para isso, no fim de cada jogada, é necessário verificar se na coluna ou na linha em que a peça foi jogada, existem 4 cores iguais, apenas numa direção. Caso a matriz fique totalmente ocupada e nenhuma das cores se encontrar em linha/coluna 4 vezes seguida, a função utilidade deverá retornar 0 (empate). Caso a máquina ganhe deverá ser retornado -1, caso contrário 1.

**( esta está parcialmente mal, representar como na 1)**

- b) Especifique uma função de avaliação dos estados.

Função Avaliação → número de peças da mesma cor juntas (vertical/horizontal/diagonal) + número de possibilidades que existe de de juntar peças outra cor

- c) Explique como utilizaria os cortes Alpha-Beta para melhorar o desempenho do algoritmo.

Com base na função de avaliação, seria atribuir valores aos estados de modo a que com esses valores pudesse atribuir valores ao  $\alpha$  e ao  $\beta$ , simplificando e melhorando pesquisa de acordo com os parâmetros Min-max, sendo que *max* deverá ser o vencedor e o *min* o adversário.