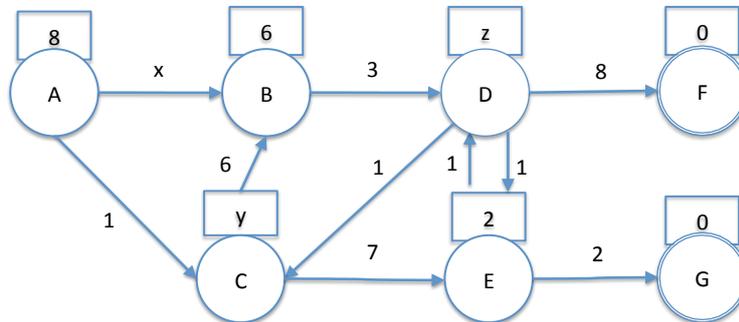


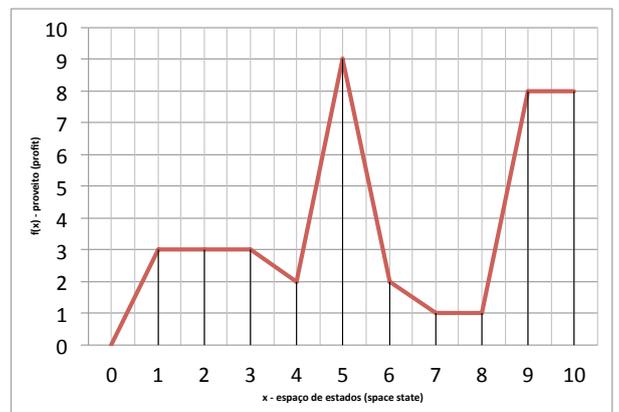
GRUPO I

I.1) Considere o seguinte grafo de estados de um problema de procura. Os valores apresentados nos arcos correspondem ao custo do operador (acção) respectivo, enquanto os valores nos rectângulos correspondem ao valor da heurística. Os estados objectivo são o F e o G. Não se representam os nomes dos operadores, correspondendo cada arco a um operador distinto. Indique os intervalos de valores para x , y e z de forma a que a heurística seja admissível, justificando.



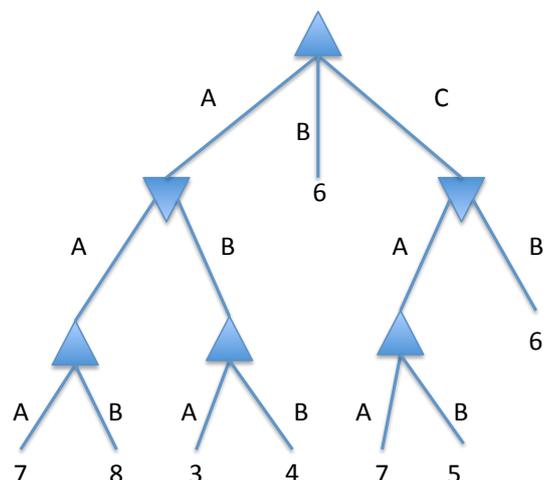
I.2) Considere novamente o grafo da questão 1. Explique como se comporta o algoritmo de procura A* em grafos, partindo do estado inicial A e assumindo que $x=4$, $y=1$ e $z=0$. Deve explicitar os conteúdos das estruturas de dados auxiliares ao longo das iterações do algoritmo, colocando entre parêntesis o valor da função de avaliação para cada nó na lista. Indique a solução obtida, assim como o seu custo, e diga se a solução encontrada é óptima.

I.3) Considere o espaço de estados contínuo $[0,10]$ e a função $f(x)$ apresentada na figura, que se pretende maximizar. Em abcissas encontram-se os estados e em ordenadas os valores de $f(x)$. Suponha que a configuração inicial é $x=2$ e com vizinhança de 1 unidade. Justifique se é possível com o algoritmo trepa colinas atingir o máximo local quando se permite um máximo de 3 movimentos (laterais) no planalto. Em caso afirmativo, apresente a sequência de transições efetuadas pelo algoritmo.



I.4) Seja $S = \{\sim a \vee b, \sim b \vee \sim c, \sim a \vee \sim c, b \vee d\}$ um conjunto de cláusulas booleanas. Pretende-se recorrer aos algoritmos de satisfação de restrições para saber se S é satisfazível. Apresente o grafo de restrições para o problema e indique qual a primeira variável a atribuir valores, de acordo com os critérios estudados.

I.5) Apresenta-se ao lado a árvore de um jogo com 2 adversários. O primeiro jogador é um jogador MAX. Recorrendo à árvore indicada, mostre que a ordem de travessia da árvore de jogo é importante para a eficiência do algoritmo $\alpha-\beta$.



GRUPO II

Uma empresa de produção de moldes reserva algum do tempo de produção para a execução de pedidos especiais dos seus clientes. Para cada dia determina o número de horas de produção disponíveis para dedicar aos pedidos especiais. Cada pedido é caracterizado por um código, o prazo limite para a sua entrega, a duração necessária à produção do molde, assim como a penalização para a empresa caso não seja possível realizar o pedido. O objetivo deste problema é determinar um agendamento dos pedidos a realizar assegurando que a empresa tenha a menor penalização possível. Todos os parâmetros são inteiros positivos, incluindo os dias que são sequencialmente numerados a partir de 1. No mesmo dia poderão ser atendidos mais do que um pedido desde que haja disponibilidade.

Uma instância concreta do problema encontra-se representada nas tabelas abaixo. Por exemplo, o agendamento do pedido A para o dia 2, do pedido C para o dia 1 e descartando os pedidos B, D e E resulta numa solução com penalização total de 8 unidades = 1 (do pedido B) + 5 (do pedido D) + 2 (do pedido E).

Disponibilidade

Dia	Tempo disponível (h)
1	3
2	4
3	1
4	1

Pedidos

Código	Prazo (dia limite)	Duração (h)	Penalização
A	2	4	1
B	1	2	1
C	4	3	2
D	2	1	5
E	4	2	2

III.1) Indique uma outra solução para a instância do problema e que tenha custo inferior ao da solução apresentada.

III.2) Formule claramente o problema para ser resolvido recorrendo a algoritmos de procura em espaço de estados, indicando o estado inicial, teste de estado objectivo e função que devolve os sucessores de um estado, não esquecendo de indicar o custo dos operadores. A formulação deve funcionar para qualquer problema deste tipo e não apenas para a instância em concreto.

NOTA: Não é necessário apresentar qualquer código Java, desde que fique perfeitamente claro e sem ambiguidades como definiria e implementaria cada um dos itens da alínea.

III.3) Indique a dimensão máxima do espaço de estados em função dos prazos $p(1), \dots, p(n)$ dos n pedidos.

III.4) Considere cada uma das seguintes funções heurísticas descritas abaixo. Indique quais delas garantem a obtenção de uma solução óptima pelo algoritmo A* de procura em árvores para a classe de problemas descrita. Justifique sucintamente a sua resposta, quer para os casos de garantia quer para as situações de não garantia.

- a) O número de dias com tempo disponível.
- b) O número de pedidos por atender subtraído do número total de horas ainda disponíveis, caso este valor seja positivo. O valor da heurística será nulo caso contrário (subtração não positiva).
- c) A soma das penalizações de pedidos por atender.
- d) O máximo de **b)** e **c)**

III.5) Caso pretendesse resolver o problema através de um algoritmo genético indique qual o operador que seria à partida mais difícil de implementar, justificando.

FIM