

Exame de CAL - 2012-13 - epoca normal

1)

a)

Começando no ponto inicial percorre os hotéis e vai comparando os valores na função $(300-x)^2$. Escolhe o proximo hotel para parar.
Recomeça a partir da ultima paragem até chegar a an.

b)

```
Inicializa array custo[n] = -1
custo[0] = 0
FOR(int i = 1; i < n+1; i++)
    custo[i] = min(
        FOR (int j = i; j >= 0; j--)
            custo[i-j] + (300 - (ai - a(i-j)))^2 //Apenas guardar o minimo valor
        FIM_FOR
    )
FIM_FOR
Devolve custo[n]
```

c)

Complexidade temporal: $O(n^2)$
Complexidade espacial: $S(n)$

2)

Aplicar Dijkstra à rede de grafos.

Percorrer a partir de 'g' os path's e guardar todos os nós numa PriorityQueue ordenada por dist até que o path seja null.

O resultado é o conteúdo dessa PriorityQueue

Passos:

Primeiro estamos em a, e analisamos todas as suas arestas.

Analisando primeiro a aresta que liga A a D, atualiza-se o valor de dist (6) e o path (A) do vértice D, pois o vértice ainda não tem valores.

Analisando depois a aresta de A para B, atualizamos o dist (3) e o path (A) do vértice B. Estes vértices atualizados estão agora guardados.

Analisamos então o vértice com menos dist, e este deixa de estar guardado, neste caso seria o B, e analisa-se as suas arestas, da maneira que as arestas do outro vértice foram analisadas.

O vértice não é atualizado, e o vértice C é atualizado com dist de (13) e path (B), este vértice está agora guardado.

O próximo vértice a deixar de estar guardado e a ser analisado é o vértice D, analisando as suas arestas, começando pela D - C, verificamos que o dist em C é maior que o dist em D + o peso da aresta D - C, então o dist e o path de C são atualizados, e o vértice C é atualizado na estrutura de dados onde está guardado.

3)

a)

Não é possível pois para haver um circuito num grafo dirigido é necessário que todos os nós tenham o mesmo grau de saída e de entrada , oque não se verifica.

b)

Lisboa -> Paris
Paris -> Lisboa
Lisboa -> Atenas

Atenas -> Paris
Paris -> Atenas
Atenas -> Lisboa
Lisboa -> Berlim
Berlim -> Londres
Londres -> Berlim
Berlim -> Lisboa
Lisboa -> Londres

c)

Trata-se de um grafo não biconexo tendo como pontos de articulação Lisboa e Berlim. Analisando-o chega-se à conclusão que a banda apenas teria que fazer 5 voos ao contrário dos 11 da alínea anterior, ou seja, não é uma boa estratégia de negócio.

No entanto, havendo qualquer problema num fdos aeroportos correspondentes aos pontos de aticulação, a banda não pode viajar para mais cidades para além daquelas duas.

4)

a)

Começando em C:

C -> D -> A -> B -> G -> H -> E -> F

b)

Não pois não pode visitar um Nó já visitado.

c)

Em primeiro lugar a diminuição desse peso pode ou não fazer com que M continue a ser a árvore de expansão mínima.

Adiciona-se uma nova aresta identica à árvore de expansão mínima retirando posteriormente a aresta maior do ciclo criado. A árvore restante é a nova árvore de expansão mínima.

5)

a)

Para cada família ou sub-conjunto C de S devemos ou não incluir esse subconjunto para a união resultar no universo?

b)

SCP está em NP porque se eu tiver um conjunto S para verificar o problema só se tem que fazer a união dos subconjuntos para verificar se é 'U'.

Se os elementos forem convertidos para arestas e cada subconjunto for o conjunto das arestas que incidem num vértice, então se se conseguir resolver a cobertura dos conjuyntos em tempo polinomial verifica-se que se consegue converter a solução de SCP para uma solução de VCP.

6)

NÃO DEMOS