Instituto Superior Técnico

Análise e Síntese de Algoritmos

Ano Lectivo 2019/2020 — 2º Teste

1 de Julho de 2020 — Duração: 1h30m (09h00-10h30)

- As questões deste teste são geradas unicamente para o aluno número: **92433**
- O grupo I deve ser submetido no Fénix através do ficheiro `asa1920-t2-gI.json`
- As respostas dos restantes grupos devem ser efectuadas em folhas manuscritas, e as suas digitalizações submetidas no Fénix
- Certifique-se que o seu nome e número de aluno está legível na digitalização
- **Bom trabalho!**

I. (2,5 + 2,5 + 2,5 + 2,5 = 10,0 val.)

I.a) Considere o problema de compressão de dados de um ficheiro usando a codificação de Huffman. Indique o código livre de prefixo óptimo para cada carácter num ficheiro com 500 caracteres com a seguinte frequência de ocorrências: $f(a) = 13, f(b) = 4, f(c) = 8, f(d) = 47, f(e) = 16, f(f) = 12$. Quando constrói a árvore, considere o bit 0 para o nó com menor frequência.

Indique também o total de bits no ficheiro codificado.

	a	b	c	d	e	f
Codificação						

Total Bits	

I.b) Considere a maior sub-sequência comum entre as duas strings $ACABACB$ e $ABCAABBC$ e calcule a respectiva matriz de programação dinâmica $c[i, j]$ para este problema, em que o índice i está associado à string $ACABACB$. Indique os seguintes valores: $c[5, 3]$, $c[4, 4]$, $c[7, 4]$, $c[5, 7]$, $c[7, 8]$.

c[5,3]	c[4,4]	c[7,4]	c[5,7]	c[7,8]

I.c) Considere o algoritmo de Knuth-Morris-Pratt com o seguinte texto $T = ababaabaaba$ e o padrão $P = bbbaab$. Calcule a função de prefixo para o padrão P:

$P =$	b	b	b	a	a	b
i	1	2	3	4	5	6
$\pi[i]$						

Indique ainda todas as posições da sequência de índices i que percorre o padrão P, ao aplicar o algoritmo sobre o texto T apenas quando o índice que percorre o texto é incrementado.

i												

I.d) Considere o seguinte programa linear:

$$\begin{array}{rrrrrr} \min & -3x_1 & - \; x_2 & + \; x_3 & & \\ \text{s.a} & x_1 & + \; x_2 & + \; x_3 & \leq & 3 \\ & -2x_1 & -2x_2 & - \; x_3 & \geq & -7 \\ & x_1 & +2x_2 & +2x_3 & \leq & 4 \\ & & & x_1, x_2, x_3 & \geq & 0 \end{array}$$

Indique o valor da função objectivo e o respectivo valor das variáveis básicas e não-básicas após uma única operação de pivot.

Z	x_1	x_2	x_3	x_4	x_5	x_6

II. (2,5 + 2,5 + 2,5 + 2,5 = 10 val.)

II.a) O Eng. Caracol foi encarregado de apresentar uma proposta para a construção de postos de primeiros socorros ao longo da autoestrada AX, que começa no kilómetro 0 e termina no kilómetro k. O Eng. Caracol dispõe de uma lista de n locais candidatos. Cada local candidato, $1 \leq i \leq n$, é associado à sua distância ao kilómetro 0, d_i, e ao seu custo estimado de construção, c_i. Sabendo que dois postos de primeiros socorros consecutivos não podem estar a uma distância superior a D kilómetros, o objectivo do Eng. Caracol é determinar o conjunto de locais candidatos que satisfazem a restrição do problema pelo menor custo possível.

1. Seja $O(i)$ o custo da solução óptima para o troço da autoestrada AX entre o kilómetro 0 e o local candidato i, que atribui necessariamente um posto de primeiros socorros ao local candidato i. Defina $O(i)$ recursivamente, completando os campos abaixo.

$$O(i) = \begin{cases} \boxed{} & \text{se } i > 1 \\ \boxed{} & \text{caso contrário} \end{cases}$$

2. Complete o template de código em baixo que calcula a quantidade $O(i)$ para $1 \leq i \leq n$ e indique a respectiva complexidade assimptótica.

```
FindO(c[1..n], d[1..n])
   let O[1..n] be a new vector of size n
   O[1] = c[1]
   for i = 2 to n do
      [                    ]
   endfor
   return O
```

3. Explique como determinar o custo da melhor solução a partir do vector O.

II.b) Considere a seguinte variação do algoritmo de Rabin-Karp para emparelhamento de cadeias de caracteres sobre o alfabeto $\Sigma = \{a, b, c\}$, que usa a função de hash:

$$h(x_1...x_n) = (\,\alpha(x_1) + ... + \alpha(x_n)\,)\,\mathbf{mod}\ 5$$

onde: $\alpha(a) = 1$, $\alpha(b) = 2$ e $\alpha(c) = 3$. Seja $T = abab^2ab^3 \ldots ab^{n-1}ab^n$ a string de texto a processar. Calcule o número de *hits* espúrios, em função de n, gerados ao procurar os seguintes padrões em T (deve apresentar os cálculos): Nota: $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$

1. $P_1 = abb$
2. $P_2 = abc$

II.c)
Considere o seguinte programa linear:

$$\begin{array}{lrrcl} \max & 2x_1 & +x_2 & & \\ \text{s.a} & 5x_1 & +2x_2 & \leq & 60 \\ & 3x_1 & +2x_2 & \leq & 40 \\ & -3x_1 & +8x_2 & \leq & 40 \\ & & x_1, x_2 & \geq & 0 \end{array}$$

1. Desenhe o conjunto exequível e resolva geometricamente o programa linear (indique tanto o valor máximo como as coordenadas onde esse valor é atingido).
2. Formule o programa linear dual e calcule a respectiva solução a partir da solução do programa primal (indique tanto o valor mínimo como as coordenadas onde esse valor é atingido). Nota: pode obter o valor das coordenadas resolvendo o sistema de equações do problema dual colocando a 0 as variáveis que correspondem a restrições não activas.

II.d) Uma matriz de incompatibilidades é uma matriz quadrada cujas células guardam valores decimais entre 0 e 1. Intuitivamente, dada uma matriz de incompatibilidades M, $n \times n$, a célula M_{ij} guarda a incompatibilidade entre os índices i e j; M_{ij} é 0 se i e j são completamente compatíveis e $M_{ij} = 1$ se i e j são completamente incompatíveis. Dado um sub-conjunto de índices $I \subseteq \{1, ..., n\}$, o nível de incompatibilidade do conjunto é dado por: $\sum_{i,j \in I} M_{ij}$. O problema das incompatibilidades define-se formalmente da seguinte maneira:

$$\begin{aligned}\textbf{Incompat} = \{\langle M, k, v\rangle \mid\ & M \text{ contém um sub-conjunto de índices} \\ & \text{de tamanho } k \text{ e incompatibilidade igual ou inferior a } v\}\end{aligned}$$

1. Mostre que o problema **Incompat** está em **NP**.
2. Mostre que o problema **Incompat** é NP-difícil por redução a partir do problema **ISet**, que é sabido tratar-se de um problema NP-completo e que se define em baixo. Não é necessário provar formalmente a equivalência entre os dois problemas; é suficiente indicar a redução e a respectiva complexidade.
   Pista: Dado um grafo G indique como construir uma matriz de incompatibilidades cujos índices correspondem aos vértices de G tendo em conta o problema **ISet**.

Problema ISet: Seja $G = (V, E)$ um grafo não dirigido; dizemos que $V' \subseteq V$ é um conjunto de vértices independentes em G se e apenas se $\forall u, v \in V'.(u, v) \notin E$. O problema **ISet** define-se formalmente da seguinte maneira:

$$\textbf{ISet} = \{\langle G, k\rangle \mid G \text{ contém um conjunto de vértices independentes de tamanho } k\}$$