Estrutura de Dados - 1º. período de 2018

Primeira Avaliação a Distância

1. (1,0) Escreva as seguintes funções em notação O:
   $\sqrt{n} + 2n^3;\ 2^n + n^5;\ n \log n + n^2;\ n + 10^{10};\ n! + 10^n$.

   Resposta: $O(n^3);\ O(2^n);\ O(n^2);\ O(n);\ O(n!)$.

2. (1,5) Sejam V_1 e V_2 dois vetores ordenados de tamanhos m e n, respectivamente. Escreva um algoritmo que intercale os dois vetores de forma que o vetor resultante V_3 esteja também ordenado. Os vetores V_1 e V_2 não podem ser alterados. Seu algoritmo deve ter complexidade $O(n + m)$.

   Resposta: O Algoritmo 1 efetua a operação desejada.

Algoritmo 1: *Intercala*(V_1, V_2).

Entrada: Vetores ordenados V_1 e V_2 de tamanhos $n > 0$ e $m > 0$, respectivamente.
Saída: Vetor ordenado V_3 com os elementos de V_1 e V_2.

```
Ocupar(V3, n + m);
i ← 1;
j ← 1;
atual ← 1;
enquanto i ≤ n  e  j ≤ m faça
    se V1[i] ≤ V2[j] então
        V3[atual] ← V1[i];
        i ← i + 1;
    senão
        V3[atual] ← V2[j];
        j ← j + 1;
    atual ← atual+1;
enquanto i ≤ n faça
    V3[atual] ← V1[i];
    i ← i + 1;
    atual ← atual+1;
enquanto j ≤ m faça
    V3[atual] ← V2[j];
    j ← j + 1;
    atual ← atual+1;
retorna V3;
```

3. Para cada item abaixo, responda "certo" ou "errado", justificando em ambos os casos:

   a. (0,5) Se a complexidade de caso médio de um algoritmo for $O(f)$, então o número de passos que o algoritmo efetua no melhor caso é $O(f)$.

      Resposta: Certo. A complexidade de melhor caso de um algoritmo é sempre menor ou igual a complexidade de caso médio do mesmo. Logo, se a complexidade de caso médio é limitada superiormente por f, então a complexidade de melhor caso também é limitada superiormente por f.

   b. (0,5) A complexidade de melhor caso de um algoritmo para um certo problema P é necessariamente maior que o limite inferior de P.

      Resposta: Errado. O limite inferior ℓ para um problema P diz respeito à complexidade de pior caso de qualquer algoritmo que resolve P, onde tal complexidade é assintoticamente maior ou igual a ℓ. Porém, nada se pode afirmar quanto à complexidade de melhor caso de uma algoritmo que resolve P. Por exemplo, a complexidade de melhor caso para o problema de ordenação de um vetor de tamanho n é $\theta(n)$, enquanto que o limite inferior para este problema é $\theta(n \log n)$.

   c. (0,5) Se um limite inferior para um problema P é n^2, então nenhum algoritmo ótimo para P pode ter complexidade de melhor caso n^3.

      Resposta: Errado. A existência de um limite inferior para P de complexidade n^2 não invalida a existência de um limite inferior para P de complexidade n^3. Neste caso um algoritmo que possua complexidades de melhor e pior casos iguais a $\theta(n^3)$ será ótimo.

4. Dado um vetor contendo os números $25, 18, 7, 3, 40, 12$, pede-se:

   a. (1,0) Desenhe todas as trocas de elementos que o *método de ordenação por seleção* efetua. **Exemplo:** se as trocas fossem "25 por 40", "7 por 3", "12 por 40" etc., você deveria desenhar a seguinte sequência de vetores:

$$40, 18, 7, 3, 25, 12$$
$$40, 18, 3, 7, 25, 12$$
$$12, 18, 3, 7, 25, 40$$

etc.

      Resposta: Trocas pela Ordenação por Seleção: são efetuadas 6 trocas.

25	18	7	3*	40	12	Vetor inicial
3	18	7*	25	40	12	
3	7	18	25	40	12*	
3	7	12	25	40	18*	
3	7	12	18	40	25*	
3	7	12	18	25	40*	
3	7	12	18	25	40	Vetor ordenado.

b. (1,0) Desenhe todas as trocas de elementos que o *método de ordenação da bolha* efetua. Utilize na resposta o mesmo sistema do item anterior.

Resposta: Trocas pelo Método da Bolha: são efetuadas 9 trocas.

25* 18 7 3 40 12 Vetor inicial
25 18* 7 3 40 12
18 25 7* 3 40 12
18 7* 25 3 40 12
7* 18 25 3 40 12
7 18 25 3* 40 12
7 18 3* 25 40 12
7 3* 18 25 40 12
3* 7 18 25 40 12
3 7 18 25 40* 12
3 7 18 25 40 12*
3 7 18 25 12* 40
3 7 18 12* 25 40
3 7 12* 18 25 40
3 7 12 18 25 40 Vetor ordenado

5. (2,0) Escreva um algoritmo que, a partir de uma lista simplesmente encadeada com nó cabeça L_1, crie outra lista encadeada com nó cabeça L_2, contendo apenas os elementos pares de L_1. A lista L_1 não pode ser alterada. Qual a complexidade do seu algoritmo? Justifique sua resposta.

Resposta: O Algoritmo 2 efetua tal operação. Como o algoritmo percorre cada posição de L_1 exatamente uma vez, então a complexidade do algoritmo é $\theta(n)$, onde n é o tamanho de L_1.

Algoritmo 2: *Extrai_Pares*(L_1, n).

Entrada: Lista encadeada L_1 e nó cabeça PT_1.
Saída: Lista encadeada L_2 com nó cabeça PT_2 e com os elementos pares de L_1.

$L_2 \leftarrow$ ocupar(PT_2);
$\mathrm{pt}_1 \leftarrow \mathrm{PT}_1$;
$\mathrm{pt}_2 \leftarrow \mathrm{PT}_2$;
enquanto $\mathrm{pt}_1 \neq \lambda$ **faça**
  se $\mathrm{pt}_1 \uparrow$.info $\mod 2 = 0$ **então**
    ocupar(pt);
    pt$\uparrow$.info $\leftarrow \mathrm{pt}_1 \uparrow$.info;
    pt$\uparrow$.prox $\leftarrow \lambda$;
    $\mathrm{pt}_2 \uparrow$.prox $\leftarrow$ pt;
  $\mathrm{pt}_1 \leftarrow \mathrm{pt}_1 \uparrow$.prox;
retorna $\mathrm{PT}\ _2$;

6. Considere um vetor V que armazena duas pilhas P_1 e P_2, que compartilham V da seguinte forma: P_1 se desenvolve sequencialmente da extremidade esquerda de V para a direita, enquanto que P_2 ocupa as posições a partir da extremidade direita e se desenvolve, em sequência, para a esquerda. Para inserirmos um dado x em uma pilha P ($P = P_1$ ou $P = P_2$), usamos o comando $I(P, x)$; para removermos um dado, usamos o comando $R(P)$. Pede-se:

   a. (1,0) Considerando que V tem 5 posições e que, inicialmente, P_1 e P_2 estão vazias, desenhe V após cada comando, para a seguinte sequência de comandos: $I(P_1, a)$, $I(P_1, b)$, $I(P_1, c)$, $I(P_2, d)$, $I(P_2, e)$, $R(P_1)$, $R(P_1)$, $R(P_2)$, $I(P_2, f)$, $I(P_2, g)$.

   Resposta: Estado do vetor V a cada operação.
   $- \quad - \quad - \quad - \quad -$ Vetor inicialmente vazio.
   $a \quad - \quad - \quad - \quad -$
   $a \quad b \quad - \quad - \quad -$
   $a \quad b \quad c \quad - \quad -$
   $a \quad b \quad c \quad - \quad d$
   $a \quad b \quad c \quad e \quad d$
   $a \quad b \quad - \quad e \quad d$
   $a \quad - \quad - \quad e \quad d$
   $a \quad - \quad - \quad - \quad d$
   $a \quad - \quad - \quad f \quad d$
   $a \quad - \quad g \quad f \quad d$ Vetor final.

   b. (1,0) Determine as condições de overflow e underflow para cada pilha.

   Resposta: Para que P_1 (resp. P_2) esteja em overflow, é necessário que o topo de P_1 (resp. P_2) seja igual ao topo de P_2 menos uma unidade (igual ao topo de P_1 mais uma unidade) e seja feita uma inserção em P_1 (resp. P_2). Para que P_1 (resp. P_2) esteja em underflow, é necessário que o topo de P_1 (resp. P_2) seja igual a 0 (resp. 6) e seja feita uma remoção.