Curso de Tecnologia em Sistemas de Computação
Disciplina: Estrutura de Dados e Algoritmos
AP1 - Primeiro Semestre de 2014

Nome -
Assinatura -

Observações:

1. Prova sem consulta e sem uso de máquina de calcular.
2. Use caneta para preencher o seu nome e assinar nas folhas de questões e nas folhas de respostas.
3. Você pode usar lápis para responder as questões.
4. Ao final da prova devolva as folhas de questões e as de respostas.
5. Todas as respostas devem ser transcritas nas folhas de respostas. As respostas nas folhas de questões não serão corrigidas.

---

1. Defina:

   (a) (1,0) Árvore binária completa.

   Resposta: Uma árvore é binária completa se ela é binária (cada nó possui no máximo 2 filhos) e é cheia até o penúltimo nível.

   (b) (1,0) Algoritmo ótimo.

   Resposta: Sendo ℓ o limite inferior do problema, um algoritmo é ótimo se sua complexidade é $O(\ell)$.

   (c) (1,0) Algoritmo recursivo.

   Resposta: É um algoritmo que contém ao mínimo uma chamada a si mesmo. Todo algoritmo recursivo possui um procedimento não-recursivo em sua composição.

2. Para os itens abaixo, recorde que existem dois métodos clássicos para ordenação de listas lineares: a ORDENAÇÃO POR SELEÇÃO (OS) e a ORDENAÇÃO PELO MÉTODO DA BOLHA (OB).

   (a) (1,5) Desenhe um vetor de 5 elementos que leve a OS a realizar o MAIOR número possível de trocas de elementos, demonstrando quantas e quais trocas foram feitas.

   Quantidade de trocas: 4

Início	3	4	5	1	2
$3 \leftrightarrow 1$	1	4	5	3	2
$4 \leftrightarrow 2$	1	2	5	3	4
$5 \leftrightarrow 3$	1	2	3	5	4
$5 \leftrightarrow 4$	1	2	3	4	5

   (b) (1,5) Desenhe um vetor de 5 elementos que leve a OB a realizar o MAIOR número possível de trocas de elementos, demonstrando quantas e quais trocas foram feitas.

   Quantidade de trocas: 10

Início	5	4	3	2	1
$5 \leftrightarrow 4$	4	5	3	2	1
$5 \leftrightarrow 3$	4	3	5	2	1
$5 \leftrightarrow 2$	4	3	2	5	1

$5 \leftrightarrow 1$	4	3	2	1	5
$4 \leftrightarrow 3$	3	4	2	1	5
$4 \leftrightarrow 2$	3	2	4	1	5
$4 \leftrightarrow 1$	3	2	1	4	5
$3 \leftrightarrow 2$	2	3	1	4	5
$3 \leftrightarrow 1$	2	1	3	4	5
$2 \leftrightarrow 1$	1	2	3	4	5

3. Considere as seguintes estruturas de dados:

   - Lista sequencial não ordenada
   - Lista sequencial ordenada (crescentemente)
   - Lista encadeada não ordenada
   - Lista encadeada ordenada (crescentemente)

   Para cada operação a seguir, diga qual (ou quais) das estruturas acima é (são) a(s) mais eficiente(s) para aquela operação. Justifique.

   (a) (0,5) Remoção do menor elemento (desconsiderando o tempo da busca)

   Resposta: Lista encadeada ordenada. Apenas com algumas operações sobre ponteiros, atualizando o início da lista e liberando a memória do primeiro nó (de menor valor), podemos fazer a remoção em um tempo $O(1)$. Esta remoção dispensa uma busca prévia e o deslocamento dos demais elementos.

   (b) (0,5) Inserção de um elemento qualquer (desconsiderando o tempo da busca)

   Resposta: Lista sequencial não ordenada e lista encadeada não ordenada. Na primeira, basta inserirmos o novo elemento na posição $n + 1$, e atualizarmos o valor de n. Na segunda, basta inserirmos o novo elemento como primeiro da lista. Para ambas, o número de passos necessários é $O(1)$.

   (c) (0,5) Busca de um elemento qualquer

Resposta: Lista sequencial ordenada. Neste caso, podemos usar busca binária, e o pior caso da busca terá tempo $O(\log n)$. Todas as demais estruturas, no pior caso, necessitarão de um tempo $O(n)$.

4. Seja uma estrutura de dados E composta por duas pilhas P_1 e P_2, que compartilham a mesma área de tamanho correspondente a n nós. No caso, P_1 e P_2 compartilham o mesmo vetor de n elementos, com P_1 se desenvolvendo seqüencialmente da extremidade esquerda do vetor para a direita, enquanto que P_2 ocupa as posições a partir da extremidade direita e se desenvolve, em seqüência, para a esquerda. Pede-se:

   (a) (1,5) Formular um algoritmo para inserir dados nesta estrutura. A entrada deste algoritmo consiste da estrutura E, do dado a ser inserido e da informação em qual pilha P_1 ou P_2 deve ser realizada a inserção.

   Resposta: Sejam $topo1$ e $topo2$ as variáveis que indicam os topos das pilhas P_1 e P_2, respectivamente. Inicialmente, temos $topo1 = 0$ e $topo2 = n + 1$, indicando que P_1 e P_2 estão vazias. Seja b uma variável booleana que indica em qual pilha o dado será inserido. $b = verdadeiro$ indica inserção na pilha P_1.

   Algoritmo:

   se $topo2 = (topo1 + 1)$ então
   overflow
   senão
   se b então
   $topo1 := topo1 + 1$
   $E[topo1] :=$ *novo-valor*
   senão
   $topo2 := topo2 - 1$
   $E[topo2] :=$ *novo-valor*

   (b) (1,0) Descrever as condições de overflow e underflow na estrutura.

   Resposta: Ocorre overflow quando $topo2 = topo1 + 1$ e tentamos inserir um dado em P_1 ou em P_2. Ocorre underflow quando

$topo1 = 0$ e tentamos remover um dado de P_1, ou quando $topo2 = n + 1$ e tentamos remover um dado de P_2.