



Curso de Tecnologia em Sistemas de Computação  
Disciplina: Estrutura de Dados e Algoritmos  
Gabarito da AP1 - Primeiro Semestre de 2012

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) Complexidade de pior caso.

Resposta: Sejam  $A$  um algoritmo,  $E = \{E_1, \dots, E_n\}$  o conjunto de todas as entradas possíveis de  $A$  e  $t_i$  o número de passos efetuados por  $A$ , quando a entrada for  $E_i$ . A complexidade de pior caso é definida por  $\max_{E_i \in E} \{t_i \mid E_i \in E\}$ .

- b. (1,0) Algoritmo ótimo.

Resposta: Um algoritmo é ótimo quando sua complexidade de pior caso é igual ao limite inferior para o problema.

- c. (1,0) Árvore estritamente binária.

Resposta: Uma árvore é estritamente binária se cada nó possui 0 ou 2 filhos.

- d. (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.

2. Dado um vetor contendo os números 3, 8, 11, 0, 5, 9, pede-se:

- a. (1,0) Desenhe todas as trocas de elementos que o *método de ordenação por seleção* efetua. **Exemplo:** se as trocas fossem “3 por 8”, “5 por 9”, “0 por 3” etc., você deve desenhar a seguinte sequência de vetores:

8, 3, 11, 0, 5, 9

8, 3, 11, 0, 9, 5

8, 0, 11, 3, 9, 5

etc.

Resposta:

0, 8, 11, 3, 5, 9

0, 3, 11, 8, 5, 9

0, 3, 5, 8, 11, 9

0, 3, 5, 8, 11, 9

0, 3, 5, 8, 9, 11

0, 3, 5, 8, 9, 11

- 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: A bolha é marcada com \*.

3\*, 8, 11, 0, 5, 9

3, 8\*, 11, 0, 5, 9

3, 8, 11\*, 0, 5, 9

3, 8, 11, 0\*, 5, 9

3, 8, 0\*, 11, 5, 9

3, 0\*, 8, 11, 5, 9

0\*, 3, 8, 11, 5, 9

0, 3, 8, 11, 5\*, 9

0, 3, 8, 5\*, 11, 9

0, 3, 5\*, 8, 11, 9

0, 3, 5, 8, 11, 9\*

0, 3, 5, 8, 9\*, 11

3. Considere as estruturas de dados PILHA e FILA.

- a. (1,0) Escreva os algoritmos de inserção e remoção de um elemento na PILHA.

Resposta:

Inserção:

se  $topo \neq M$  então

$topo := topo + 1$

$P[topo] := novo\_valor$

senão *overflow*

Remoção:

se  $topo \neq 0$  então  
     $valor\_recuperado := P[topo]$   
     $topo := topo - 1$   
senão *underflow*

- b. (1,0) Escreva os algoritmos de inserção e remoção de um elemento na FILA.

Resposta:

Inserção:

$prov := (r \bmod M) + 1$   
se  $prov \neq f$  então  
     $r := prov$   
     $F[r] := novo\_valor$   
    se  $f = 0$  então  $f := 1$   
senão *overflow*

Remoção:

se  $f \neq 0$  então  
     $valor\_recuperado := F[f]$   
    se  $f = r$  então  
         $f := r := 0$   
    senão  
         $f := (f \bmod M) + 1$   
senão *underflow*

4. (1.0) Escreva um algoritmo que realiza a seguinte tarefa: Dada uma lista simplesmente encadeada  $L$ , contar quantos nós da lista têm seu campo de informação igual a  $x$ .

Resposta: Seja  $L$  uma lista encadeada com nó cabeça.

$total := 0$   
 $pt := ptlista \uparrow .prox$   
enquanto  $pt \neq \lambda$  faça  
    se  $pt \uparrow .info = x$  então  
         $total := total + 1$   
     $pt := ptlista \uparrow .prox$

5. (1.0) Suponhamos que um NÓ seja um elemento composto por três campos: um campo *info* para armazenar uma informação qualquer, e dois campos *pt1* e *pt2* que são ponteiros para outros elementos do tipo NÓ. Explique como utilizar elementos do tipo NÓ para implementar uma *árvore binária*.

Resposta:

Em uma árvore binária, dado um nó  $N$ , seu ponteiro *pt1* deve ser utilizado para apontar para o filho esquerdo de  $N$ , e seu ponteiro *pt2*, para apontar para o filho direito de  $N$ . Assim,  $N \uparrow .pt1$ , por exemplo, referencia o nó que é filho esquerdo de  $N$  na árvore. Caso  $N$  não tenha filho esquerdo, então o ponteiro *pt1* de  $N$  é igual a  $\lambda$ . (O mesmo se aplica ao filho direito.)