Aluno(a):\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Matrícula:\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |  |
| --- | --- | --- |
| 1a) | 1.0 |  |
| 2a) | 2.5 |  |
| 3a) | 2.5 |  |
| 4a) | 2.5 |  |
| 5a) | 1,5 |  |
|  | 10.0 |  |

1. A prova é individual e sem consulta.
2. A interpretação faz parte da questão. Em caso de dúvida escreva a dúvida e a sua interpretação na resposta.
3. O tempo de prova é 1:45 h.
4. As respostas devem seguir as questões. Caso precise de rascunho use o verso da folha.
5. A prova pode ser feita a lápis.
6. (1.0 pontos)
   1. (0,5 pontos) No pior caso, qual a altura de uma árvore de busca binária com *n* nós. Explique e ilustre a sua resposta.
   2. (0,5 pontos) No melhor caso, qual a altura de uma árvore de busca binária com *n* nós. Qual a relação entre *h* e *n* para atingir o melhor caso. Explique e ilustre a sua resposta.

Resp:

1. (2.5 pontos) O *índice* de uma árvore AVL é definido como o maior número de nós que devem ser visitados para localizar uma chave na árvore. Sem utilizar estruturas auxiliares (pilhas, filas, etc...), implemente de forma não recursiva uma função que calcule o índice de uma AVL, visitando o menor número possível de nós. A função deve ter o seguinte protótipo:

int indice(AvlNode\* node)**;**

Utilize a estrutura:

typedef struct \_avl\_node AvlNode;

struct \_avl\_node {

void\* info;

int bf; /\* balance factor = h\_right - h\_left \*/

AvlNode\* parent;

AvlNode\* left;

AvlNode\* right;

};

Resp:

1. a) (1.5 pontos) Em cada um dos casos abaixo indique qual ou quais rotações devem ser feitas para rebalancear a árvore AVL após a sequencia de operações indicada. Explique a sua resposta, usando os fatores de balanceamento.

(i) (0,5 ponto) Operação: **Inserção da chave 5**.

(ii) (0,5 ponto) Operação: **Remoção da chave 6**.

(iii) (0,5 ponto) Operações: **Remoção da chave 6**, seguida de **Remoção da chave 11**.

1. **(1.0 ponto) Qual estratégia você sugeriria para diminuir o número de rotações em uma árvore AVL, à luz dos exemplos acima? Explique sua resposta.**

Resp:

1. (2.5 pontos) Remova a chave 70 da árvore B abaixo. Explique e ilustre como a remoção foi processada. Use e explique decisões ótimas em cada passo.

10

20

60

90

100

120

**70**

50

80

Resp:

1. (1.5 pontos) Considere uma família de árvores definida de forma semelhante a árvores B, exceto que:

* Os nós possuem tamanho fixo em 136 bytes
* As chaves possuem tamanho variável, entre 4 bytes e 12 bytes
* Os ponteiros ocupam 4 bytes
* Em cada nó, há um campo a mais, de 4 bytes, indicando o número de chaves que o nó efetivamente armazena
  1. (0.5 pontos) Qual será o maior número de chaves, todas de menor tamanho, que é possível acomodar em uma árvore de altura 2? Explique cuidadosamente sua resposta.
  2. (1.0 ponto) Qual será o número mínimo de chaves, todas de maior tamanho, que uma árvore de altura 2 armazena? Explique cuidadosamente sua resposta.

Resp: