

Algoritmos e Complexidade

LMCC

13 de Janeiro 2006

1

Questão 1 Com base no seguinte código C:

```
typedef struct node {
    int num;
    struct node *esq, *dir;
} *BTree;

void swap (BTree h)
{
    BTree aux = h->esq;
    h->esq = h->dir;
    h->dir = aux;
}

BTree insert (int x, BTree h)
{
    if (!h) {
        BTree new = malloc (sizeof(struct node));
        new -> num = x;
        new -> esq = new -> dir = NULL;
        return new;
    }
    if (x < h->num) {
        h->dir = insert(h->num, h->dir);
        swap(h);
        h->num = x;
    }
    else {
        h->dir = insert(x, h->dir);
        swap(h);
    }
    return h;
}
```

1. Utilizando uma recorrência, efectue a análise assintótica do tempo de execução da função `insert`.
2. Desenhe a árvore que resulta de se inserir consecutivamente os números 10, 20, 30, 22, 21, 35, 40, 24 e 27 numa árvore vazia. As árvores construídas por inserções sucessivas (usando a função `insert`) possuem todas uma determinada propriedade de ordem. Identifique esta propriedade.
3. Defina uma função em C que funde duas árvores com a propriedade que identificou, numa única árvore que possua ainda essa propriedade.

Nome: _____

Número: _____ Curso: _____

Algoritmos e Complexidade

LMCC

13 de Janeiro 2006

2

Questão 2 A função seguinte pesquisa a primeira ocorrência de um elemento num vector:

```
int procura (int vector[], int a, int b, int k)
{
    int i=a;
    while ((i<=b) && (vector[i]!=k)) i++;
    if (i>b) return -1;
        else return i;
}
```

1. Prove as propriedades de inicialização e preservação do invariante:

No início de cada iteração, k não se encontra nas posições $[a..i]$ do vector.

Mostre ainda (terminação) de que forma o invariante garante a correcção do algoritmo.

2. Efectue a análise do tempo de execução assintótico no pior e no melhor caso deste algoritmo. Exprima os resultados utilizando a notação apropriada.
3. Escreva uma versão alternativa *recursiva* desta função e compare o comportamento assintótico das duas versões.

Nome: _____

Número: _____ Curso: _____

Algoritmos e Complexidade

LMCC

13 de Janeiro 2006

3

Questão 3

1. Considere o grafo

$$G = (\{1, 2, 3, 4, 5, 6, 7, 8, 9\}, \{(1, 2), (2, 3), (2, 5), (3, 6), (4, 5), (5, 6), (5, 8), (6, 9), (7, 8), (8, 9)\})$$

Comece por desenhar o grafo; classifique-o de acordo com todos os critérios que conhece. Desenhe a árvore produzida por uma travessia em profundidade com início no nó 1.

2. Considere o problema da construção de uma *ordenação topológica* para um grafo ordenado acíclico. Selecciona um dos algoritmos que conhece para resolver este problema (baseado numa travessia em largura ou em profundidade). Ilustre a sua execução completa sobre o grafo da alínea 1 (justifique todos os passos).

Nome: _____

Número: _____ Curso: _____