

Algoritmos

Exame Final — 2 de Junho de 2005

1 – Uma possível estratégia para efectuar a ordenação de um vector por **Inserção Linear** consiste basicamente, para cada passo, em inserir um elemento num sub-vector já ordenado.

Considere, sem perda de generalidade, vectores cujos elementos são números inteiros.

- 2.0 a) Construa uma função iterativa que insira o elemento $v[esq]$ no sub-vector ordenado $v[esq + 1, dir]$.
- 2.5 b) Faça uma análise completa do número de *atribuições* e de *comparações* — associadas a elementos do vector — efectuadas pelo algoritmo da alínea anterior.
- 2.0 c) Construa uma função que implemente a estratégia de ordenação por **Inserção Linear** proposta.
- 2.5 d) Faça uma análise completa do número de *atribuições* e de *comparações* — associadas a elementos do vector — efectuadas pelo algoritmo da alínea anterior.

~~1.5~~ 2 – Considere o tipo abstracto de dados **Árvore Binária de Inteiros**, em cujos nós é possível armazenar um número inteiro.

Considere também que os números inteiros se encontram registados “em-ordem” crescente.

Elabore funções **eficientes** que permitam:

- 1.5 a) Obter um ponteiro para o nó contendo o maior elemento armazenado numa dada árvore.
- 2.5 b) Listar, em ordem crescente, todos os elementos armazenados numa dada árvore e que pertençam ao intervalo $[a, b]$.
- 3.0 c) Calcular a média de todos os elementos armazenados numa dada árvore.

4.0 3 – Considere o tipo abstracto de dados **Digrafo**, definido usando uma estrutura de dados dinâmica que representa um dado digrafo $G(V, E)$, com n vértices e m arestas, armazenando a **lista dos vértices** do digrafo e associando a cada elemento dessa lista (i.e., a cada vértice) a correspondente **lista de adjacências**.

Considere também que os n vértices de um digrafo se encontram identificados pela sequência de números inteiros $0, 1, \dots, (n - 1)$.

Dado um vértice $v_i \in V$, pretende-se listar todos os vértices v_j alcançáveis a partir de v_i .

Essa listagem deverá ser iniciada pelo vértice v_i e ser ordenada — de modo não-decrescente —, de acordo com a distância associada ao caminho mais curto de v_i para cada v_j (i.e., o número de arcos que o definem).

Por exemplo, para um dado digrafo e para o caso dos vértices alcançáveis a partir do vértice 1, obter-se-ia:

No	1	-	Distancia	0
No	2	-	Distancia	1
No	3	-	Distancia	1
No	4	-	Distancia	2
No	5	-	Distancia	2
No	6	-	Distancia	3

Desenvolva uma função que, dado um digrafo e um seu vértice v_i , lhe permita obter a listagem desejada.

Atenção:

- O digrafo pode conter ciclos.
- Assuma que estão definidos os tipos abstractos **Pilha** e **Fila**; não é necessário implementá-los.
- Desenvolva eventuais funções auxiliares de que possa necessitar.