

Algoritmos

Exame Final (1ª chamada) — 5 de Junho de 2007 — Duração 2h30m

1 – O algoritmo *Insertion Sort* é um dos métodos possíveis para efectuar a ordenação de uma sequência de elementos e consiste, basicamente, em inserir um elemento numa sequência já ordenada. Considere uma sequência, cujos elementos são números inteiros, com possíveis elementos repetidos, e que se pretende ordená-la de modo **não-decrescente**.

[2.0] a) Implemente uma função repetitiva que insira o elemento $\text{seq}[\text{esq}]$ na sequência ordenada $\text{seq}[0, \text{esq}-1]$.

[2.5] b) Faça uma análise completa do número de comparações – entre elementos da sequência – efectuadas pelo algoritmo da alínea anterior.

[2.0] c) Implemente uma função que, usando a função anterior, implemente a estratégia de ordenação por inserção linear.

[2.5] d) Faça uma análise completa do número de comparações – entre elementos da sequência – efectuadas pelo algoritmo de ordenação da alínea anterior.

$$\text{Nota: } \sum_{i=1}^n \left(\frac{1}{i} \right) \approx \ln(n)$$

2 – Considere o tipo abstracto de dados Árvore Binária de Pesquisa, em cujos nós é possível armazenar um número inteiro. Considere também que os números inteiros se encontram armazenados “**em-ordem**” crescente.

[1.5] a) Implemente uma função recursiva que obtém um ponteiro para o menor número inteiro armazenado na árvore.

[2.0] b) Implemente uma função recursiva que determine a soma dos números inteiros armazenados na árvore.

[2.5] c) Implemente uma função repetitiva que determine a soma dos números inteiros armazenados na árvore.

Atenção: Assuma que estão definidos os tipos abstractos de dados **Fila** (*Queue*) e **Pilha** (*Stack*), pelo que não é necessário implementá-los.

3 – Considere o tipo abstracto de dados **Digrafo**, definido usando uma estrutura de dados dinâmica que representa um digrafo com V vértices e A arestas, armazenando a lista de vértices do digrafo e associando a cada elemento dessa lista, ou seja, a cada vértice, a correspondente lista de adjacências.

Considere também que os V vértices do digrafo se encontram identificados pela sequência de números inteiros $1, 2, \dots, V$.

Dado um vértice existente no digrafo pretende-se determinar os vértices alcançáveis a partir desse vértice.

[2.5] a) Implemente uma função repetitiva que determine os vértices alcançáveis a partir de um dado vértice do digrafo.

[2.5] b) Implemente uma função recursiva que determine os vértices alcançáveis a partir de um dado vértice do digrafo.

Atenção:

- O digrafo pode conter ciclos.
- Assuma que estão definidos os tipos abstractos de dados **Fila** (*Queue*) e **Pilha** (*Stack*), pelo que não é necessário implementá-los.
- Desenvolva eventuais funções auxiliares de que possa necessitar.