Programação Imperativa LEI + LCC (1º ano)

8ª Ficha Prática

Ano Lectivo de 2006/07

Objectivos

O objectivo desta ficha é treinar o aluno em algoritmos de manipulação de listas ligadas dinâmicas—caso geral e casos particulares da stack e da queue.

Exercícios

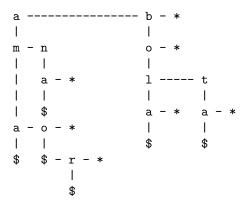
1. Desenvolva, na linguagem C, um programa para implementar *lista das Mensagens Recebidas* num telemóvel Nokia.

Cada mensagem, além do texto respectivo, está associada a um nome ou número de telefone, a uma data e hora e tem ainda um estado (lida, não-lida). —deve ser inserido ordenadamente na lista.

Cada nova mensagem que chega é inserida na frente da lista; cada vez que uma mensagem é lida passa para a frente das mensagens lidas. Além das operações de inserção de nova mensagem e de leitura da primeira, o programa deve permitir seleccionar uma mensagem por posição, ou por nome de remetente, possibilitando então apagá-la ou lê-la.

No fim de cada utilização, o programa deve gravar a lista em ficheiro, para ser recuperada na invocação seguinte. O nome do ficheiro a usar, para recuperação (caso já exista) e salvaguarda dos dados, deve ser passado como argumento na linha de comandos, aquando da invocação do programa.

- 2. Desenvolva, na linguagem C, um programa para implementar a lista de voos que vão partir nas próximas horas de um aeroporto.
 - Sobre cada *voo* conhece-se o número, a companhia, o destino e a hora de partida. O programa deve permitir inserir novos voos, colocando-os por ordem de partida, remover voos que partiram e alterar a hora de partida, corrigindo a posição na lista.
- 3. Desenvolva, na linguagem C, um programa para implementar as operações básicas sobre matrizes (inversão, multiplicação por um escalar, adição e multiplicação), adaptado a *matrizes esparsas* (em que mais de 80% das entradas são nulas) implementadas como um array de listas ligadas de valores não-nulos.
- 4. Desenvolva, na linguagem C, um programa para implementar um dicionário com as palavras para apoio a um corrector ortográfico, armazenado-as na forma de uma lista generalizada de caracteres, em que cada um aponta para o caracter seguinte, na ordem alfabética e para a lista de todas as palavras que o contém esse caracter nessa posição. Por exemplo: ama, amo, amor, ana, bola, bota ficariam armazenados na forma:



O seu programa deve construir o dicionário, na memória, lendo um texto (passado como parâmetro na linha de comandos) e inserindo todas as novas palavras com 2 ou mais caracteres. Depois de construir o dicionário, o programa deve: listar todas as palavras por ordem alfabética; e, dada uma palavra, verificar se ela existe no dicionário.