

Curso de

## **Licenciatura em Engenharia Informática**

Cadeira de Algoritmia

Projecto de avaliação 2010/2011: Gestor de Tráfego Aéreo

Este projecto pretende ser um auxiliar na gestão de tráfego aéreo de uma região ou pequeno país. O programa só lida com tráfego *incoming* (a entrar na região), não com aquele que atravessa ou levanta voo dentro da região e vai para fora.

Quando um novo avião entra no radar desta região, ele deve ser registado no sistema (com os dados de entrada no sistema). Também podem ser removidos aviões conforme eles aterram.

O programa deve arrancar com uma mensagem de boas-vindas e ficar à espera de comandos. Eu registo um voo (avião) que surge no radar, escrevendo o comando:

\* `companhia passageiros aeroporto-de-destino  
tempo-de-combustivel tempo-ate-aterrar`

Onde:

`companhia`

É uma *string* (sem espaços) com o nome ou abreviatura da companhia aérea do voo (e.g.: TAP, AirFrance, etc.).

`passageiros`

É um inteiro com o número de passageiros no voo.

`aeroporto-de-destino`

*String* (sem espaços) com o nome ou abreviatura do aeroporto para o qual o voo se dirige (e.g.: LX, Porto).

`tempo-de-combustivel`

Inteiro com a quantidade de minutos de combustível que o voo tem.

`tempo-ate-aterrar`

Inteiro com a quantidade de minutos que o voo demora a chegar ao aeroporto.

Atenção que `tempo-de-combustivel` e `tempo-ate-aterrar` são válidos no momento de introdução dos comandos. Se eu listar os voos um minuto depois, estes valores devem estar menores por 1, e assim por diante.

Exemplo:

\* TAP 320 Lx 340 250

Descreve um voo da TAP para Lisboa que leva 320 passageiros.  
O voo ainda tem 340min de combustível, mas espera aterrar  
em 250 minutos.

Comandos que não começam com "\*" são comandos introduzidos pelo operador para saber informações sobre a situação actual de tráfego.

Existem os seguintes comandos:

ajuda

Exibe ajuda sobre os comandos

listar companhias

Exibe todas as companhias que têm voos em processo de aterragem,  
por ordem alfabética crescente.

listar aeroportos

Exibe todos aeroportos que têm voos em processo de aterragem,  
por ordem alfabética crescente.

listar voos

Exibe todos os voos ordenados por ordem de aterragem.  
É apresentado um número ao pé de cada voo (pode não estar  
ordenado este número, mas serve para identificar o voo).

listar companhia

Exibe voos de uma companhia aérea específica, ordenados por  
ordem de aterragem.

listar aeroporto

Exibe voos ordenados por ordem de aterragem num aeroporto  
específico.

listar emergencias

(sem acentuação) Deve listar todos os voos onde o tempo até  
ao aeroporto  $\leq 0$ , ou o tempo de combustível  $\leq 10$ .  
A ordem é por ordem crescente de tempo de combustível.

tempo minutos

Finge que minutos minutos já passaram, reduzindo os tempos  
de combustível e de aterragem dos voos sejam reduzidos na  
quantidade correspondente.

### apagar **VOO**

Apaga um voo particular (i.e.: já aterrou). **VOO** é um dos números exibidos pelo comando "listar voos". Este comando **não** invalida a numeração dos restantes voos existentes!

A ordem de aterragem é determinada por:

1. Menor tempo até ao aeroporto (ordem crescente de tempo).
2. Voos que estejam a 10min uns dos outros em relação ao aeroporto, devem ser ordenados por ordem crescente de tempo até ao combustível acabar.
3. Mesmo que estejam separados por mais de 10min, um voo não pode aterrar no mesmo aeroporto depois de outro, se o combustível dele acaba antes do tempo que leva ao outro a chegar ao aeroporto.
4. Depois dos critérios acima, os voos são ordenados por quantidade de passageiros (ordem decrescente).

Critérios de avaliação:

1. Capacidade de introduzir e listar voos sem trocar campos e sem limite na quantidade de voos a introduzir.
2. Capacidade de exibir ajuda e detectar comandos mal escritos, informando o utilizador da existência de um erro.
3. Os comandos "listar voos" e "apagar **VOO**" funcionam bem, existindo uma numeração de voos que não depende da ordem dos mesmos nas estruturas de dados em memória.
4. Os comandos "listar companhias" e "listar aeroportos" funcionam correctamente.
5. Os comandos "listar **companhia**" e "listar **aeroporto**" funcionam correctamente.
6. O comando "tempo **minutos**" funciona bem, e os comandos que listam voos exibem os tempos reduzidos quer devido a este comando, quer devido à passagem do tempo.
7. O comando "listar **emergencias**" funciona correctamente.
8. A ordem de aterragem dos comandos de listagem (em particular dos que envolvem voos ou emergências) está correcta.
9. Uso de estruturas de dados adequadas ao funcionamento eficiente da aplicação, i.e., rápida e ocupando a quantidade mínima de memória, sem ter limites na quantidade de dados que é capaz de operar enquanto o sistema tiver memória para a aplicação.
10. Critério livre, subjectivo.

O projecto deve ser realizado e apresentado em grupos de 2 a 4 alunos. A nota do projecto pode ser diferente para cada elemento do grupo. Um elemento do grupo que não apareça ao mesmo tempo que os restantes para a discussão terá nota zero para a apresentação oral do mesmo, e nota reduzida para os restantes critérios.

Boa Sorte! Pedro Freire