

Aula Prática 7

Objetivos

Utilização do Sistema de Entrada e Saída (I/O) de Java, Serialização e JARs

Problema 7.1

No dossier da disciplina encontra, para além deste guião, dois ficheiros de texto: *voos.txt* e *companhias.txt*.

O primeiro representa os voos que chegaram ao aeroporto do Porto no dia 24 de Maio. A estrutura deste ficheiro é a seguinte (os campos são separados por *tab*):

Hora	Voo	Origem	Atraso
00:50	TP 1944	Lisboa	
07:00	AEA1147	Madrid	
07:35	IB 8720	Madrid	00:25
...			

O segundo contém uma tabela com as siglas e os nomes de cada companhia:

Sigla	Companhia
A5	HOP!
AE	Air Europa
DT	TAAG
EV	Evelop Airlines
EZ	EasyJet Airlines
FR	Ryanair
...	

Construa um programa que leia estes dois ficheiros para estruturas adequadas. Crie a classe *Voo*, por exemplo, use conjuntos para armazenar os voos em memória, bem como outras estruturas/algoritmos que ache necessários para cada uma das alíneas seguintes.

- a) Apresente no ecrã a lista de voos com informação mais completa, tal como consta da tabela seguinte:

Hora	Voo	Companhia	Origem	Atraso	Obs
00:50	TP 1944	TAP Portugal	Lisboa		
07:00	AEA1147	Air Europa	Madrid		
07:35	IB 8720	Iberia	Madrid	00:25	Previsto: 8:00
07:35	TO 3408	Transavia France	Paris, Orly		
07:40	FR 5451	Ryanair	Faro		
07:55	EZY3771	EasyJet Airlines	Paris, Ch. de Gaulle	00:33	Previsto: 8:28
08:05	TP 459	TAP Portugal	Paris, Orly	00:13	Previsto: 8:18
08:15	FR 1386	Ryanair	Brussels, Charleroi		
08:20	FR 4171	Ryanair	Frankfurt, Hahn		
08:30	S4 370	Sata Internacional	Toronto, Pearson	00:35	Previsto: 9:05
...					

- b) Guarde a tabela no ficheiro *Infopublico.txt*.
c) Calcule a média dos atrasos por companhia e apresente no ecrã uma tabela (Companhia, Atraso médio) ordenada por ordem crescente de atraso médio.
d) Guarde no ficheiro *idades.txt* uma tabela com informação com o total de

chegadas de cada cidade origem. Exemplo (ordenação por número de voos):

Origem	Voos
Lisboa	11
Madrid	9
Paris, Orly	8

...

- e) Repita a alínea b) mas agora armazenando os dados num ficheiro binário (*Infopublico.bin*). Construa também um método para ler o ficheiro e apresentar a tabela no ecrã. Use como base a classe *RandomAccessFile*.

```
RandomAccessFile file = new RandomAccessFile("Infopublico.bin", "rw");
```

Problema 7.2

Pretende-se desenvolver uma aplicação que permita descodificar e processar o conteúdo de um ficheiro de imagem em formato BMP (**não comprimido**). A aplicação deve responder aos seguintes requisitos:

- Receber a localização de um ficheiro BMP, abri-lo e imprimir toda a informação relativa a esse ficheiro, i.e. número de linhas, colunas, número de cores, etc.
- Permitir a extração do “pixel data” e guardar num ficheiro “.raw”. Tente visualizar este ficheiro com visualizador de imagens apropriado (Irfanview, Photoshop, etc)
- Permitir a redução da dimensão da imagem para $\frac{1}{4}$. Por exemplo, uma imagem com 512x512 pixels passaria a ter 256x256. O resultado final deverá ser guardado num novo ficheiro BMP.
- Permitir inverter a imagem (*flip*) na vertical ou horizontal. O resultado final deverá se guardado num novo ficheiro BMP.

Nota: Toda a informação necessária para conhecer a estrutura de dados deste tipo de formato (BMP) é disponibilizada em documento de texto fornecido como anexo juntamente com uma figura de teste (aula7_material.zip).