

Programação I

Folha de Exercícios 7

António J. R. Neves
João Rodrigues
Osvaldo Pacheco
Arnaldo Martins

2018/19/20

Folha de exercícios 7

Resumo:

- Acesso a ficheiros de texto.
- Construção de programas envolvendo ficheiros de texto e manipulação de diretórios.

Em todos os programas desenvolvidos até ao momento, a informação manipulada era perdida sempre que terminamos os programas. Isto deve-se ao facto de as variáveis que declaramos reservarem espaço na memória do computador, que depois é libertada quando o programa termina.

Para armazenarmos permanentemente informação gerada pelos nossos programas, temos que a guardar no disco rígido do computador (ou em qualquer outro dispositivo de memória de massa). Isto é possível através da utilização de ficheiros.

Para esta aula deve utilizar (**import**) os módulos **java.util.Scanner**, **java.io.File** e **java.io.PrintWriter** da biblioteca do Java

7.1 Problemas para resolver

Exercício 7.1

Escreva um programa que mostre o conteúdo de um ficheiro de texto no terminal. O nome do ficheiro deve ser pedido ao utilizador e introduzido através do teclado. Valide o nome do ficheiro: tem de ser um ficheiro normal (método **.isFile()**), com permissão de leitura (método **.canRead()**). Se isso não se verificar, deve indicar a razão e voltar a pedir o nome.

Exercício 7.2

Altere o programa 6.2 para ler a lista de números de um ficheiro (deve parar quando ler 100 números positivos ou quando chegar ao fim do ficheiro e deve ignorar os negativos). O nome do ficheiro deve ser passado como *argumento na linha de comandos*. Assim, para testar o programa deve usar uma linha de comando deste género:

java Ex6_3 FicheiroComNumeros.txt

Nota: Num programa Java os argumentos são recebidos no parâmetro da função **main**, que é um array de Strings (geralmente **String[] args**). (No exemplo acima, **args[0]** teria o valor "**FicheiroComNumeros.txt**").

Exercício 7.3

Modifique o programa 6.3 de modo a incluir mais três operações:

1. ler uma sequência de números de um ficheiro de texto;
2. adicionar números à sequência existente;
3. gravar a sequência atual de números num ficheiro.

Os nomes dos ficheiros envolvidos devem ser pedidos ao utilizador.

Exercício 7.4

Implemente um programa que: 1) leia uma turma de alunos de um ficheiro de texto, com o formato indicado, para 2 arrays, um com o nome e outro de 2 dimensões com o nºmec e as notas. 2) peça o nº mec de um aluno e liste a informação respetiva: nmec, nome, nota1, nota2. Implemente estas dois passos como funções

Ficheiro:

```
Antonio    1233 12    15
Maria      2314 13    14
Xico       1115 8     11
... .
```

Exercício 7.5

Construa um programa que leia um ficheiro de texto correspondente a um programa em Java e que copie o seu conteúdo para um novo ficheiro filtrando todos os comentários nele existentes (ou seja, o novo ficheiro não deve conter nenhum dos comentários existentes no ficheiro original). Primeiro considere apenas comentários que comecem com *// comentário*. Depois melhore o seu programa para considerar também comentários do tipo */* comentários aqui */*.

Faça duas funções: uma, **find()**, que dados dois strings como argumentos devolva a primeira posição em que encontrar o string2 no string1 ou -1 se não encontrar; outra, **substring()**, que dado um string e dois índices inteiros como argumentos devolve o substring correspondente ao intervalo dado pelos índices.

A função da classe String **charAt(i)** devolve o caracter na posição i do string (começa em 0). A função **length()** devolve o nº de caracteres do string.

```
String frase "aveiro";
char letra = frase.charAt(2); → 'e'
           frase.length(); → 6
```

Exercício 7.6

Desenvolva um programa que copie um ficheiro de texto. O nome do ficheiro original e o novo nome devem ser dados como argumentos na linha de comandos. Assim a execução do programa com os argumentos **Texto1.txt Texto2.txt** deve criar um ficheiro **Texto2.txt** com um conteúdo igual ao do ficheiro **Texto1.txt** (ver nota do Ex. 7.2).

Nota: Torne o programa robusto. Para isso, deve verificar: que o ficheiro a copiar é um ficheiro comum; que pode ser lido; que o ficheiro a criar ainda não existe. Caso alguma destas condições não se verifique, deve terminar com uma mensagem de erro apropriada.

Exercício 7.7

Desenvolva um programa que leia um ficheiro de texto e imprima o seu conteúdo com todas as letras transformadas em minúsculas à exceção da primeira letra a seguir a um ponto final, que deve ser colocada em maiúscula. A distância entre as letras minúsculas e maiúsculas é 32 e pode ser calculada assim: `int distancia = 'a' - 'A'`; passar de minúsculas para maiúsculas é subtrair 32 e ao contrário é somar 32 ao código do caracter respetivo.

Pode eventualmente usar as funções desenvolvidas do exercício 7.5