

## Aula Prática Nº 6

### Objetivos

Acesso a ficheiros de texto

Acesso ao conteúdo de directorias

### Guião

1. Leia atentamente o código fonte `myCat.c`.

a) Compile e execute o programa passando-lhe o nome de um ficheiro de texto como argumento e interprete o resultado do programa.

b) Altere o programa anterior de modo a garantir o seu funcionamento quando o utilizador passa vários ficheiros como argumentos.

c) Adicione ao programa a capacidade de imprimir o número da linha do ficheiro que está a ser impressa, isto é, cada linha deve ser precedida do número de linhas desde o início do ficheiro.

d) Execute o programa que resultou da alínea c usando como argumento o ficheiro `myCat.c`. Deve notar algo estranho na forma como aparece a linha mais longa e na contagem de linhas. O que se passou? Altere o código de forma a resolver o problema.

2. Leia atentamente o código fonte `sortNumbers.c`.

a) Compile e execute o programa passando-lhe vários números como argumentos e interprete o resultado do programa.

b) Altere o programa de modo a ordenar o conteúdo de um ficheiro de texto contendo números inteiros, um por linha, cujo nome seja passado como argumento ao programa. Considere que os ficheiros a ordenar não têm mais do que 100 números. Crie 2 versões do programa:

i) usando a função `fgets()` para ler o ficheiro.

ii) usando a função `fscanf()` para ler o ficheiro.

c) Considere agora que não quer limitar o número de valores armazenados no ficheiro. Explore várias soluções para resolver este problema.

Sugestões: pode efetuar duas leituras ao ficheiro, uma para contar os valores outra para a leitura; pode considerar que o primeiro número contido no ficheiro é a indicação do número de valores, etc.

3. Leia atentamente o código fonte `myActions.c`.

a) Compile e execute o programa interagindo com ele escrevendo comandos (que podem incluir `pipes` e redirecionamentos) no terminal até introduzir a palavra “end”. Interprete o resultado do programa.

b) Altere o programa de modo a que o texto que introduziu e a hora em que ocorreu seja gravado num ficheiro de nome “`command.log`”. Sucessivas chamadas ao programa devem acrescentar conteúdo ao ficheiro.

4. Leia atentamente o código fonte `dirList.c`.

a) Compile e execute o programa passando-lhe como argumento o nome de uma diretoria e interprete o resultado do programa.

b) Altere o programa de modo a que, na listagem, as diretorias sejam precedidas por “`d` ” (d espaço) e os ficheiros por “” (2 espaços).

c) Complete o programa acrescentando a capacidade de fazer a listagem recursiva de todas as diretorias encontradas.

Sugestão: invoque recursivamente a função `listDir()` para as diretorias encontradas.