

# Aula Prática 7

## Resumo:

- Funções e estruturas de dados recursivas.

### Exercício 7.1

Escreva um programa que processe um ficheiro de texto e imprima primeiro as linhas com menos de 20 caracteres, depois as linhas com 20 a 40 caracteres e finalmente as linhas mais longas. Para cumprir o objectivo, o programa poderá ler e armazenar as linhas de texto, divididas em três conjuntos distintos, imprimindo-os no fim. Não se conhecendo a priori o número de linhas existentes no ficheiro, sugere-se a utilização de três listas para guardar as frases separadamente. Use para esse efeito a classe `LinkedList` do pacote `p2utils` em anexo.

### Exercício 7.2

A classe `p2utils.LinkedList` define o tipo *lista ligada* com os métodos essenciais para inserção e remoção de elementos. Analise os métodos `print()` e `contains()`. Repare que ambos recorrem a funções auxiliares privadas e recursivas, que usam um parâmetro extra de tipo `Node`.

Acrescente à classe os seguintes métodos, implementado-os também de forma recursiva.

- `clone()` - devolve uma nova lista com a mesma sequência de elementos.
- `reverse()` - devolve uma nova lista com os mesmos elementos por ordem inversa.
- `get(pos)` - devolve o elemento na posição `pos` da lista, em que `pos` varia entre 0 e `size()-1`.
- `concatenate(lst)` - devolve uma nova lista com os elementos da lista (em que o método é chamado) seguidos dos elementos da lista dada no argumento.
- `contains(e)` - devolve `true` caso o elemento dado exista na lista, e `false` caso contrário
- `remove(e)` - remove da lista a primeira ocorrência do elemento dado no argumento.

Teste a classe com o programa P72.

### **Exercício 7.3**

No exercício 6.5 fez uma função para listar recursivamente o conteúdo de um diretório. Escreva agora uma função **recListFiles** que em vez de imprimir a lista, devolva o resultado numa lista ligada de elementos de tipo **File**.

### **Exercício 7.4**

Considere agora um problema parecido com o do exercício 7.1. Neste caso, o programa deve imprimir primeiro as linhas com menos de 20 caracteres, por ordem inversa daquela em que estavam no ficheiro, seguidas das linhas mais longas, pela ordem original. Note que pode resolver o problema com apenas uma lista, se inserir elementos pelas duas extremidades.