

# Aula Prática 11

## Objetivos

Java Collections Framework.  
Java Stream API

## Problema 11.1

Utilizando o Java Collections Framework:

- a) Implemente novamente o programa *main* do problema 10.1 utilizando `ArrayList` e `LinkedList`.
- b) Desenvolva uma aplicação que leia um ficheiro de texto e conte o número de palavras diferentes existentes a implementação (Sugestão: utilize um `HashSet`).

```
Número Total de Palavras: 161
Número de Diferentes Palavras: 112
```

- c) Desenvolva uma aplicação que leia um ficheiro de texto, contabilize e imprima uma listagem com o número de ocorrências de cada palavra. (Sugestão: utilize um `HashMap`).

```
submission      1
não              1
apontarem       1
fases           1
notification     1
methods         2
.....
.....
elsevier        2
modelo         1
```

- d) Altere a aplicação anterior (c) para permitir uma listagem ordenada por palavra (Sugestão: Substituir `HashMap` por `TreeMap` com o devido `Comparator`).

```
abstract        1
acceptance      1
alguns          1
apontarem       1
artigo          4
autor           2
aveiro          1
.....
```

- e) Reimplemente as alíneas b, c e d utilizando Java 8 Stream API. (Sugestão: leia o conteúdo do ficheiro para uma estrutura de dados e depois utilize a stream).

## Problema 11.2

Implemente os métodos solicitados utilizando a nova Java 8 Stream API.

- a) Reimplemente o método *maiorFiguraJ7*.

```
private static Figura maiorFiguraJ7(List<Figura> figs) {
    Figura maior = figs.get(0);
    for (Figura f : figs) {
        if (f.compareTo(maior) >= 1)
            maior = f;
    }
    return maior;
}
```

- b) Reimplemente a alínea a) mas utilizando agora o perímetro como elemento de comparação (isto sem mexer na implementação de *Figura*).
- c) Implemente um método que retorne a soma das áreas das figuras da lista.

```
private static double areaTotalJ8(List<Figura> figs) {
    ...
}
```

- d) Altere a alínea c) para só calcular a área total do subtipo de *Figura* passado como argumento.

```
private static double areaTotalJ8(List<Figura> figs, String subtipoNome) {
    ...
}
```

## Problema 11.3

Tendo como base o trabalho desenvolvido na alínea 6.1 do guião número seis, transforme os diversos módulos desenvolvidos de forma a que estes utilizem o Java Collections Framework, nomeadamente para suporte das:

- a) Estruturas de dados utilizadas na classe *Prato* e *Ementa*;
- b) Estruturas de dados utilizada no *Menu* para armazenar os diversos ingredientes e pratos criados. Isto é, os ingredientes/pratos criados devem ser memorizados para eventual utilização futura na criação de novos pratos/ementas, respetivamente.