

Programação III

Semestre de Inverno de 2021-2022

2º Trabalho prático, Parte I

LEIRT

Data de Entrega: 11 de Dezembro de 2021

Objetivos: Praticar a utilização de *streams* de texto, o mecanismo de tratamento de exceções, a construção e utilização de tipos e métodos genéricos, a utilização de interfaces funcionais e expressões lambda, e os contentores da *framework* de coleções da linguagem Java.

Nota: Para cada método realizado, deverá incluir testes unitários que permitam validar a sua correção. Pode ser útil definir métodos auxiliares à realização dos métodos propostos.

1. Realize a classe `StreamUtils` com os seguintes métodos estáticos e públicos:

a) `int copyWithReplace(Reader in, Writer pw) throws IOException;`

Copia do *stream in* para o *stream out* substituindo cada sequência de caracteres dígito (valor numérico) por uma nova sequência contendo o separador ponto entre cada 3 dígitos a contar da direita. O método retorna o número de caracteres ponto acrescentados. Exemplos:

Reader in	Writer pw
123	123
the value is 1234	the value is 1.234
12345678 persons	12.345.678 persons
123456789	123.456.789

b) `int copyWithReplace(String filenameIn, Writer pw) throws IOException;`

Este método é uma variante do método `copyWithReplace` da alínea a) que recebe agora o nome do ficheiro que contém o texto a copiar, com substituição das sequências de dígitos, para a *stream* `pw`.

Nota: na realização deste método, utilize o método `copyWithReplace` da alínea a).

c) `String contentWithReplace(String filenameIn) throws IOException;`

Este método recebe por parâmetro o nome de um ficheiro de texto, e usa o método da alínea b) para retornar uma `String` com o texto resultante da substituição das sequências de dígitos.

Nota: na realização deste método considere a utilização da classe `StringWriter`.

d) `void printWithReplace(String filenameIn) throws IOException`

Recebe por parâmetro o ficheiro de nome `filenameIn`, e usa o método da alínea b) de forma a que a cópia com substituição seja apresentada no *standard output*.

e) `void copyWithReplace(String fileIn, String fileOut) throws IOException;`

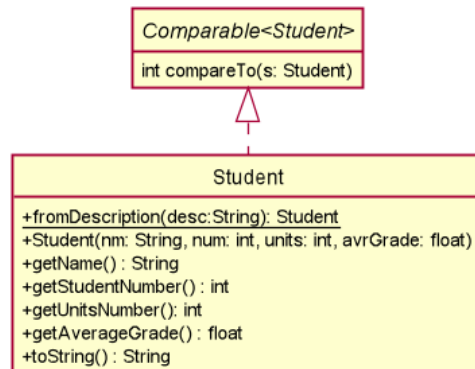
Recebendo por parâmetro o nome do ficheiro de texto com nome `fileIn`, usa o método da alínea b) para copiar o texto resultante da substituição das sequências de dígitos para o ficheiro de texto com nome `fileOut`.

2. Realize o método estático público da classe `AlgorithmUtils` com a seguinte assinatura:

```
public static <V> int forEachIf(BufferedReader in, Function<String, V> mapper,
                                Predicate<V> pred, Consumer<V> action) throws IOException;
```

que por cada linha do *stream* `in` obtenha o valor através da aplicação da função `mapper`, e caso o valor obedeça ao predicado definido em `pred`, execute a ação definida em `action` passando-lhe o valor retornado pela função `mapper`. O método `forEachIf` retorna o número de valores que foram passados ao *consumer* `action`.

3. Implemente a classe `Student` ao lado, a usar nas questões seguintes, para representar um estudante de uma escola de engenharia. Os estudantes são comparados pela média obtida (`averageGrade`) nas disciplinas com aprovação (`unitsNumber`). O método estático `fromDescription` cria um `Student` a partir de uma *string* com o seguinte formato:



`<student_name> ':' <student_number> ':' <total_units> :<average_grade>`

onde `<total_units>` indica o total de disciplinas realizadas e `<average_grade>` a média obtida.

4. Realize o método da classe `Course`

```
public static Student[]
getKBestStudents( int k, int semester, String dirName) throws IOException;
```

que tira partido do método `forEachIf`, realizado na questão 2, para determinar os `k` estudantes de uma escola que obtiveram, num determinado semestre, a melhor média de notas nas disciplinas realizadas nesse semestre, tendo realizado pelo menos 5 disciplinas, para lhes ser atribuída uma bolsa de estudos para o semestre seguinte.

Em caso de empate de notas, será utilizado o número de aluno, dando prioridade aos alunos com menor número. Os alunos estão distribuídos por turmas, estando cada turma descrita num ficheiro de texto em que cada linha contém os dados de um aluno no formato indicado na questão 3.

Os estudantes estão organizados em turmas de um dado semestre, sendo cada turma representada por um ficheiro cujo nome (`class_name`) segue o seguinte formato:

```
<class_name> ::= <course_name> <semester> <class_number> <regime> <extension>
<course_name> := <word>
<regime> ::= 'D' | 'N'
<semester> ::= '1'..'6'
<class number> ::= '1'..'9'
<extension> ::= '.dat'
```

LT51D, LT51N e LEIRT31D são exemplos de nomes de turmas.

O parâmetro `dirName` indica o nome da pasta onde constam os ficheiros das turmas dos vários semestres.

5. Tirando partido do método desenvolvido na questão 4, realize uma aplicação que recebendo por parâmetro o nome da diretoria, o semestre, e o número de bolsas, liste na consola os alunos a quem são atribuídas as bolsas.

Bom trabalho,

*Manuela Sousa,
Jorge Martins*