Programação 2 (LTI), 2018/2019

**Projeto**

## **RelayStations**

# Grupo 12

1. **Membros:**
   1. André Miguel Tavares dos Ramos, nº53299;
   2. Manuel Maria Gomes Ferreira de Sá Machado, 53325;
2. A percentagem de trabalho realizado foi de 50% para ambos os membros;
3. Trabalho produzido e executado no Windows 10.

Das classes já contidas no ficheiro fornecido foi alterada a ***“Node”***, onde foi adicionado os atributos ***“ID”, “name”, “power”, “generation”*** e implementados *getters* bem como outros métodos ( *str* ) necessários á manipulação e visualização dos dados. A classe ***“DFS”*** foi também adaptada de forma a receber um argumento ***“bestTime”*** e ***“time”*** de forma a possibilitar a comparação de potencias dos diversos ***“Nodes***” e ***“Edges”*** de maneira a calcular tempos de comunicação uteis para indicar o percurso mais rápido.

Em termos de funções criadas e alteradas, temos ***“pathGeneration”*** que vai retornar apenas caminhos onde existe ***“Nodes”*** com ***“generation”*** superior aos 97. Função ***“search”*** que vai utilizar o dado ***“bestTime”*** passado pela função ***“DFS”*** para verificar se as estações que comunicam. Função ***“main”*** que é a responsável pela leitura e escrita dos ficheiros dados, bem como pela inicialização do programa em si. Na função ***“main”*** os nós e ligações do ficheiro que contêm a lista dos vários nós e suas estações, são colocados em listas para posteriormente serem instanciados pelas respetivas classes. Do segundo ficheiro que contem a estação origem e estação destino, é feito uma leitura linha a linha, em que cada linha é verificada a existência da estação de origem e destino através da função ***“findStation”***. Acrescentamos uma funcionalidade que mostra a percentagem de execução do teste e que mostra o total de ligações testadas.

Criamos também uma função **“testArgs”** que verifica o número de ficheiros dados na inicialização do programa, e que gera um erro caso não sejam os ficheiros necessários.