

Relatório de implementação

relayStations



Grupo 70

Francisco do Ó - 53340

Mário Gil Poiares Rodrigues de Oliveira – 49269

Programação II (LTI)

Departamento de Informática

Faculdade de Ciências da Universidade de Lisboa

Sistemas operativos usados:

Este software foi testado em **Windows 10**, **macOS Mojave 10.14** e **Ubuntu 19.04**.

O programa corre corretamente em todos os sistemas operativos usados.

Linhas essenciais da estrutura e funcionamento do software:

O software recebe o seguinte comando de entrada:

```
relayStations.py stations.txt requests.txt output.txt
```

De acordo com as especificações do projeto e devolve o ficheiro com os tempos decorrentes da ligação, ou em caso de ligação falhada, “out of the network” ou “do not communicate” respetivamente para os casos em que a estação não exista ou não haja comunicação entre as duas estações.

Este programa usa a lógica que assenta sobre a representação da rede como um grafo bidirecional em que cada torre é um nó e cada associação de torres um arco.

Numa lógica resumida e não de todo extensiva eis o que o programa faz:

- Cria um objeto da classe Graph, subclasse de Digraph que corresponde ao grafo;
- Cada torre corresponde a um objeto da classe Node, adicionado ao Graph;
- Cada ligação unidirecional entre torres corresponde a um objeto da classe Edge que é adicionado ao Graph em cada um dos sentidos, evitando duplicados;
- Para cada pedido de ligação o programa verifica se as torres existem e são compatíveis, e em caso positivo executa um algoritmo que segue a abordagem DFS para pesquisa do caminho mais rápido entre as duas torres, atendendo aos requisitos da especificação do projeto;
- Os resultados são transpostos para uma lista, os elementos da qual são inseridos no ficheiro de saída.

Contribuição de cada elemento do grupo para a resolução

Os módulos e as funções que não requeriam um grande esforço na sua conceção, tendo em conta que são semelhantes aos de projetos anteriores foram divididos pelos elementos do grupo, sendo que o Francisco se focou nas funções de escrita e o Mário Gil nas de leitura. As funções restantes, que se prendiam com o cálculo dos tempos e a adaptação dos algoritmos de DFS, bem como o *debugging* final foram feitos em conjunto.

Funcionalidades que ficaram por implementar

Nenhuma funcionalidade requerida ficou por implementar.

Indicação de erros conhecidos

Os autores deste projeto não têm conhecimento de erros no programa.

Critérios usados na seleção dos itens de teste

- Casos de comunicação entre estações da mesma geração (a diferentes distâncias no grafo) – deve mostrar o tempo adequado;
- Casos de comunicação entre estações compatíveis de gerações diferentes (99G e 98G) – deve mostrar o tempo adequado;
- Casos de comunicação entre estações de gerações incompatíveis (99G-97G e 09G-97G) – deve mostrar o erro correspondente;
- Casos em que o emissor ou o recetor correspondem a estações que não estejam na lista de entrada – deve mostrar o erro correspondente.

Para cada um destes casos, podem existir vários exemplos para averiguar que o tempo no qual o programa decorre é aceitável independentemente do número de ligações entre torres adjacentes testados.