

INSTRUÇÕES

- A não entrega deste trabalho prático implica a reprovação à unidade curricular no ano letivo 2020/2021, **não sendo possível a sua realização em nenhuma outra época de avaliação.**
- O trabalho prático será realizado em Grupo com um **máximo de 2 alunos**;
- A data limite para a entrega do primeiro trabalho prático é o **dia 11 de Junho**. A submissão de trabalhos será feita apenas usando o formulário correspondente disponível no Moodle;
- A defesa do trabalho é realizada por **todos os elementos do grupo**;
- Para além da implementação do projeto, deverá ser preparado um **pequeno relatório** que explique de que forma o enunciado foi interpretado, e quais as decisões tomadas na sua implementação. Deve incluir uma discussão sobre a complexidade dos algoritmos usados.

AVALIADOR DE LIGAÇÕES AÉREAS

É comum a tarefa de seleção das melhores opções de viagem quando se pretende viajar de um destino para outro. Em particular, quando a viagem inclui uma ligação aérea, é feita uma seleção da melhor opção de voo possível com base em diversos critérios, tais como o número de paragens, a duração da viagem, a distância a percorrer, o tipo de companhia aérea (normal ou *lowcost*) ou o preço dos bilhetes. Atualmente existem diversas plataformas que oferecem acesso a este tipo de informação, tornando o processo de seleção mais simples.

O objetivo deste trabalho é desenvolver uma aplicação em C que auxilie na tarefa de seleção de ligações aéreas, tendo por base a construção de um modelo a partir do conjunto de dados disponibilizado.

Para o armazenamento destes dados deverão ser utilizadas as estruturas de dados mais eficientes para o tipo de dados e o género de resultados necessários. Sempre que possível, devem utilizar-se estruturas de dados dinâmicas, nomeadamente tabelas de dispersão e grafos.

Os dados existentes foram simplificados e resumidos em dois ficheiros de texto. Cada linha tem um conjunto de colunas separadas por um caractere de tabulação. A primeira linha de cada um dos ficheiros apresenta um cabeçalho, com a identificação de cada coluna.

1. `airports.tsv` (city, country, code)

Contém os dados dos aeroportos mundiais, indicando a sua cidade, o seu país e o código do aeroporto.

2. `flights.tsv` (aofd, distance, duration, type, aofa)

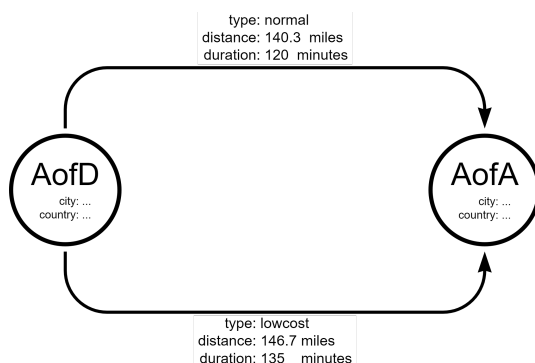
Contém as várias ligações aéreas entre o aeroporto de partida (*aofd*: *airport of departure*) e o

aeroporto de destino (*aofa: airport of arrival*), detalhando alguns parâmetros a ter em conta no cálculo do “custo” dessa ligação, nomeadamente: a distância da rota prevista, a duração prevista e o tipo de companhia aérea que assegura essa ligação (*lowcost* ou *normal*). De forma a integrar numa única variável uma métrica de “custo” de viagem entre dois aeroportos A e B , considera-se:

$$Custo(A, B) = 0.50 \times duration + 0.45 \times distance + 0.05 \times k,$$

em que $k = 0$ para o caso de um voo *lowcost*, e $k = 1$ para um voo normal.

Assume-se que entre dois aeroportos só podem existir, no máximo, duas ligações: uma *lowcost* e uma normal.



Pretende-se com este conjunto de dados calcular:

1. Todas as cidades de destino possíveis a partir de uma dada cidade de origem. O utilizador poderá indicar o número de escalas que está disposto a realizar (0, 1 ou 2). Por exemplo, caso o utilizador indique um máximo de uma escala, deverão ser apresentados destinos que possam ser visitados com voos diretos, ou com uma sequência de dois voos.
2. A cidade com mais partidas diferentes para cidades distintas (considerando apenas voos diretos).
3. A cidade com mais chegadas diferentes a partir de cidades distintas (considerando apenas voos diretos).
4. O aeroporto mais internacional de um dado país, ou seja, aquele que tem mais ligações (sem escalas) com outro país.
5. A melhor opção de voo entre dois aeroportos (com menor custo).
6. A melhor opção de voo entre duas cidades (com menor custo).

Para além dos dados lidos a partir dos ficheiros providenciados, devem existir opções para:

- Alterar distância e a duração de uma dada ligação;
- Adicionar ou remover um dado aeroporto de uma cidade específica;
- Adicionar uma nova ligação entre dois aeroportos;
- Adicionar uma nova cidade;

A implementação deverá usar os algoritmos e estruturas de dados adequados à resolução do problema tendo em vista a resposta às diferentes questões apresentadas, e tentando obter a maior eficiência possível.