

Trabalho prático 2 - Transportes Aéreos

Neste trabalho, a sua tarefa é implementar um sistema capaz de providenciar ajuda para quem quer usar a rede de voos das companhias aéreas de todo o mundo.



Para o ajudar, terá acesso a dados reais indicando os aeroportos existentes (bem como a sua localização), os voos (indicando aeroporto de partida e chegada e companhia aérea) e uma lista das companhias aéreas existentes. No final do enunciado poderá encontrar uma descrição detalhada dos dados fornecidos, que estão acessíveis a partir de um ficheiro [dataset.zip](#). A sua primeira tarefa será ler os dados fornecidos e com eles **criar um ou mais grafos sobre os quais poderá efetuar as operações pedidas**.

O sistema desenvolvido deverá ter um pequeno menu, expondo as funcionalidades que implementou de uma maneira o mais amigável possível. A tarefa mais fundamental e o requisito mínimo do trabalho é:

- **Dados dois locais, indicar a “melhor” maneira de voar de um para o outro (menor número de voos).**

Para obter pontuação máxima, o seu programa deverá ser capaz indicar um percurso a seguir para obter esse mínimo e de implementar diferentes versões do que significa um “local” e de que parte da rede pode ser usada:

- **“Local”** – tanto pode ser indicado diretamente um aeroporto em específico, como uma cidade (devendo nesse caso considerar todos os aeroportos que partem da cidade) ou uma localização (i.e., as coordenadas, devendo nesse caso considerar todos os aeroportos a menos do que X kms da localização).
- **“Rede” de voos** – um utilizador pode querer usar qualquer companhia aérea, somente uma dada companhia aérea ou um dado conjunto de companhias aéreas.

O segundo objetivo é ser capaz de dar **informações sobre um aeroporto**:

- Quantos voos existem a partir de um dado aeroporto? De quantas companhias aéreas diferentes? Para quantos destinos diferentes? De quantos países diferentes? Quantos aeroportos, cidades ou países são atingíveis usando um máximo de Y voos?

A sua implementação deverá fazer uso pelo menos uma vez de uma **pesquisa em largura (BFS)**. Deverá também usar **tabelas de hash** (`unordered_set` ou `unordered_map`) para guardar aeroportos e cidades e suportar a sua pesquisa em $\mathcal{O}(1)$.

Se quiser incluir distâncias no seu programa, deverá usar a [fórmula de Haversine](#) para calcular a distância entre dois pontos dadas as suas latitudes e longitudes ([ver exemplo de implementação](#)).

Deverá **documentar** o seu código usando Doxygen e indicar para cada método de carácter mais algorítmico qual a **complexidade temporal** associada.

Como fatores de valorização extra, para além das diferentes variações atrás expostas e outras listagens que imagine, poderá por exemplo fazer algumas das tarefas seguintes (são apenas indicações):

- Se existir mais do que uma possibilidade com o menor nº de voos, indicar isso mesmo e quais são todas essas possibilidades (incluindo que companhias aéreas se devem usar);
- Possibilitar a pesquisa de percursos que usem só uma companhia aérea e/ou mudem o menor número de vezes de companhias aéreas (e/ou incluir critérios de desempate no caso de várias soluções mínimas)
- Calcular estatísticas globais da rede, de um país ou de uma companhia (nº aeroportos? nº voos? nº companhias? diâmetro? top- k de aeroportos com mais voos e/ou companhias? ...)
- Calcular os pontos de articulação existentes na rede (toda a rede ou só de um subconjunto de companhias)
- (...)

Dados fornecidos para criação do sistema:

Os dados estão disponíveis em [dataset.zip](#) sob a forma de ficheiros **csv** (*comma separated values*) e foram obtidos diretamente a partir do site [OpenFlights](#) (tendo sido recolhidos em Junho de 2014).

- Ficheiro **airports.csv** (3019 aeroportos)

(primeiras 6 linhas)

```
Code,Name,City,Country,Latitude,Longitude
CDG,Charles De Gaulle,Paris,France,49.012779,2.550000
ORY,Orly,Paris,France,48.725278,2.359444
LGA,La Guardia,New York,United States,40.777245,-73.872608
JFK,John F Kennedy Intl,New York,United States,40.639751,-73.778925
LHR,Heathrow,London,United Kingdom,51.477500,-0.461389
```

A primeira linha são os cabeçalhos. Cada uma das linhas seguintes indica o código IATA do aeroporto (que é único e sem repetições), o nome do aeroporto, a cidade onde está, o país e onde está localizado (latitude e longitude).

- Ficheiro **airlines.csv** (444 companhias aéreas)

(primeiras 6 linhas)

```
Code,Name,Callsign,Country
IBE,Iberia Airlines,IBERIA,Spain
KLM,KLM Royal Dutch Airlines,KLM,Netherlands
SWR,Swiss International Air Lines,SWISS,Switzerland
AAL,American Airlines,AMERICAN,United States
RJR,Ryanair,RYANAIR,Ireland
```

A primeira linha são os cabeçalhos. Cada uma das linhas seguintes indica o código ICAO da companhia aérea (que é único e sem repetições), o seu nome “oficial”, a “alcunha” porque é chamada (se não existir alcunha, este campo contém somente “_”) e o país de origem.

- Ficheiros **flights.csv** (63832 voos)

(primeiras 6 linhas)

```
Source,Target,Airline
CDG,JFK,AAL
JFK,CDG,AAL
ORY,LHR,IBE
ORY,LHR,BAW
OPO,LIS,TAP
```

A primeira linha são os cabeçalhos. Cada uma das linhas seguintes indica o código IATA do aeroporto de partida, o código IATA do aeroporto de chegada e o código ICAO da companhia aérea. Note que as ligações são dirigidas e que podem existir vários voos com o mesmo destino e origem, se forem de diferentes companhias.

Imagem que ilustra os dados fornecidos:

A imagem seguinte foi obtida a partir dos dados fornecidos, e ilustra as posições dos aeroportos e os voos entre eles:



Como os dados que recebeu são reais, pode em qualquer altura verificar o resultado do seu programa e ver se fazem sentido os resultados que obteve (note contudo que são dados de 2014).