

O metro de Paris é um serviço de transporte público que abrange cerca de 220 km da área metropolitana de Paris. Pode transportar diariamente cerca de 5.23 milhões de pessoas que percorrem as suas estações distribuídas por várias linhas¹. O presente projeto envolve a criação de uma biblioteca de classes, respetivos métodos e testes que permitam gerir **a rede de metro de Paris** e sugerir trajetos aos utentes em função dos seus objectivos.

Para tal dispõe-se da seguinte informação relativa à rede de metro Paris:

coordinates.csv –localização geográfica das estações

Nome_da_Estação, Latitude, Longitude

Lines_and_stations.csv – linha e respectivas estações

Código_da_Linha, Nome_da_Estação

connections.csv – conexões entre estações na mesma linha e o tempo médio de viagem entre as duas estações (em minutos).

Linha, Estação_1, Estação_2, Tempo

Usando a representação **map de adjacência** pretende-se que desenvolvam da forma **mais eficiente possível** as seguintes funcionalidades:

1. Criar o grafo da rede de metro de Paris.
2. Verificar se o grafo é conexo, se não for devolver os seus componentes.
3. Criar uma **classe Percurso** com toda a informação relativa a um percurso incluindo as linhas, as estações percorridas em cada linha e o instante em que passa em cada estação, dado um instante inicial do percurso.
4. Dada uma estação origem e uma estação destino encontrar o **caminho mínimo** entre estas **quanto ao número de estações**, devolvendo uma instância da classe Percurso.
5. Dada uma estação origem, uma estação destino e um instante inicial encontrar o **caminho mínimo** entre estas **quanto ao tempo a percorrer**, devolvendo uma instância da classe Percurso.

¹ <https://www.ratp.fr/en/plans-lignes/plan-metro>

6. Dada uma estação origem e uma estação destino encontrar o **caminho mínimo** entre estas quanto ao **número de mudanças de linha**, devolvendo uma instância da classe Percurso.
7. Dada uma estação origem, uma estação destino, um instante inicial e um conjunto de outras estações intermédias, encontrar o caminho mínimo, **quanto ao tempo a percorrer**, entre a estação inicial e a final que passe obrigatoriamente por todas as estações intermédias, devolvendo uma instância da classe Percurso.

Regras

- A avaliação do trabalho será feita principalmente em função das classes propostas, nomeadamente em termos da sua conformidade com o Paradigma da Programação por Objetos e **eficiência** das estruturas de dados usadas e funcionalidades solicitadas.
- O trabalho deverá ser realizado em **grupos de dois alunos**.
- O projeto tem de ser desenvolvido em Java e todas as funcionalidades testadas através de testes unitários e usando os ficheiros de teste disponibilizados.
- É obrigatório o uso da ferramenta de **controle de versões Git**.
- O relatório deverá ser elaborado para cada uma das partes do trabalho e deve servir de ferramenta de avaliação posterior à apresentação. Nele devem apresentar o digrama de classes, algoritmos dos métodos, análise de complexidade de todas as funcionalidades implementadas, melhoramentos possíveis.
- Cada Parte do trabalho deve ser submetida no Moodle até às **24 horas do dia 18 de Novembro**. A partir desta data a nota do trabalho será penalizada **10% por cada dia de atraso** e não se aceitam trabalhos **após dois dias** das datas indicadas.
- Na semana seguinte à data de entrega o professor das aulas PL fará a cada grupo de trabalho uma avaliação qualitativa do projeto submetido.