**Algoritmia Avançada**



Modelação de Rede de Metro de Paris

Carlos Miguel, 1120287

Simão Ribeiro, 1110322

Índice

[ Introdução 3](#_Toc408781419)

[ Desenvolvimento 4](#_Toc408781420)

[Base de conhecimento 4](#_Toc408781421)

[Gerar Dados 4](#_Toc408781422)

[Percurso 5](#_Toc408781423)

[Pontos de Interesse 6](#_Toc408781424)

[ Conclusão 8](#_Toc408781425)

[ Bibliografia 8](#_Toc408781426)

## Introdução

O presente trabalho foi desenvolvido no âmbito da disciplina de Algoritmia Avançada (ALGAV), onde foi pedido o desenvolvimento de uma aplicação da modelação da rede de metro de paris. Para a mesma, foi utilizada a linguagem de programação PROLOG e utilizados métodos de pesquisa lecionados durante o semestre.

Para facilitar o desenvolvimento foram fornecidos vários sites com informação relativa as linhas de metro, para assim, ser possível criar uma base de conhecimento o mais semelhante com a rede de metro.

Além da modelação da linha, foi pedido a criação de geradores de percursos e guias. Para os percursos foi pedido a criação de percurso mais rápido, menos trocas e menos percursos a pé. Relativamente aos guias, era pedido a criação de guias em que o utilizador escolhia os locais a visitar e era sugerido um guia utilizando a linha de metro e também um guia que começasse e terminasse no mesmo local.

## Desenvolvimento

## Base de conhecimento

Para a modelação da rede de metro foi criada uma base de conhecimento, onde foram criados vários termos, para guardar tanto as linhas, horários e os pontos de interesse (PDI). No ficheiro *metro\_paris.pl*, encontra-se a base de conhecimento.

* Linha (nº linha, [Lista de Estacoes]), este predicado é utilizado para a criação de linhas, onde é guardado o número da linha e uma lista com as estações da mesma.
* Horário (nº da linha, Origem, Fim, Inicio, Fim, horário Dia, Horário Noite, Horário Sabado, Horário Domingo), o predica exemplificado represente um horários de uma linha onde é diferenciado a direção da linha;
* pontoDeInteresse (Nome,[Estações Próximas],Hora Abertura, Hora Fecho, Duração),

por último, a representação de um PDI, onde é guardado as estações próximas e os horários do mesmo;

## Gerar Dados

Depois de a base de conhecimento completa, foram criados predicados para gerar dados, optamos assim para tornar o trabalho mais organizado e evitar uma base de conhecimento extensa.

Os dados mais importantes gerados foram os cruzamentos, em que são criados predicados que representam os cruzamentos das linhas de metro.

* estacoes\_linhas(Nome Estação,[linhas que cruzam]);

Para facilitar a utilização dos métodos de pesquisa foi criado também o predicado liga, que facilita a pesquisa de ligações entre estações.

* liga(Nº de Linha, Estacão Origem, Estação Destino);

Por último, também é gerado para uma ligação entre 2 linhas, o primeiro e ultimo horário da mesma.

* valida\_horario(Origem, Destino, Dia, Primeiro, Final).

## Percurso

Em termos de funcionalidades, foi implementado o cálculo de percursos mais rápido, com menos trocas e com menos percurso a pé. Para o cálculo da mesma é utilizado o método de pesquisa Branch and Bound, em que é utilizado como custo o tempo de duração da viagem. Depois de gerados os percursos são avaliado consoante a escolha do utilizador.

Para o cálculo do percurso tal como estava referido no enunciado é pedido ao utilizador a estação de origem, a estação de destino, a hora desejada e o tipo de dia.



Figura 1 - Menu para seleção de percurso

É de referenciar que todas as inserções de dados são validadas (existência de estações, horas possíveis e opções corretas).

Depois de finalizada a configuração do percurso é impresso o percurso de acordo com a escolha do utilizador.



Figura 2 - Percurso menos trocas entre Nation-Lena

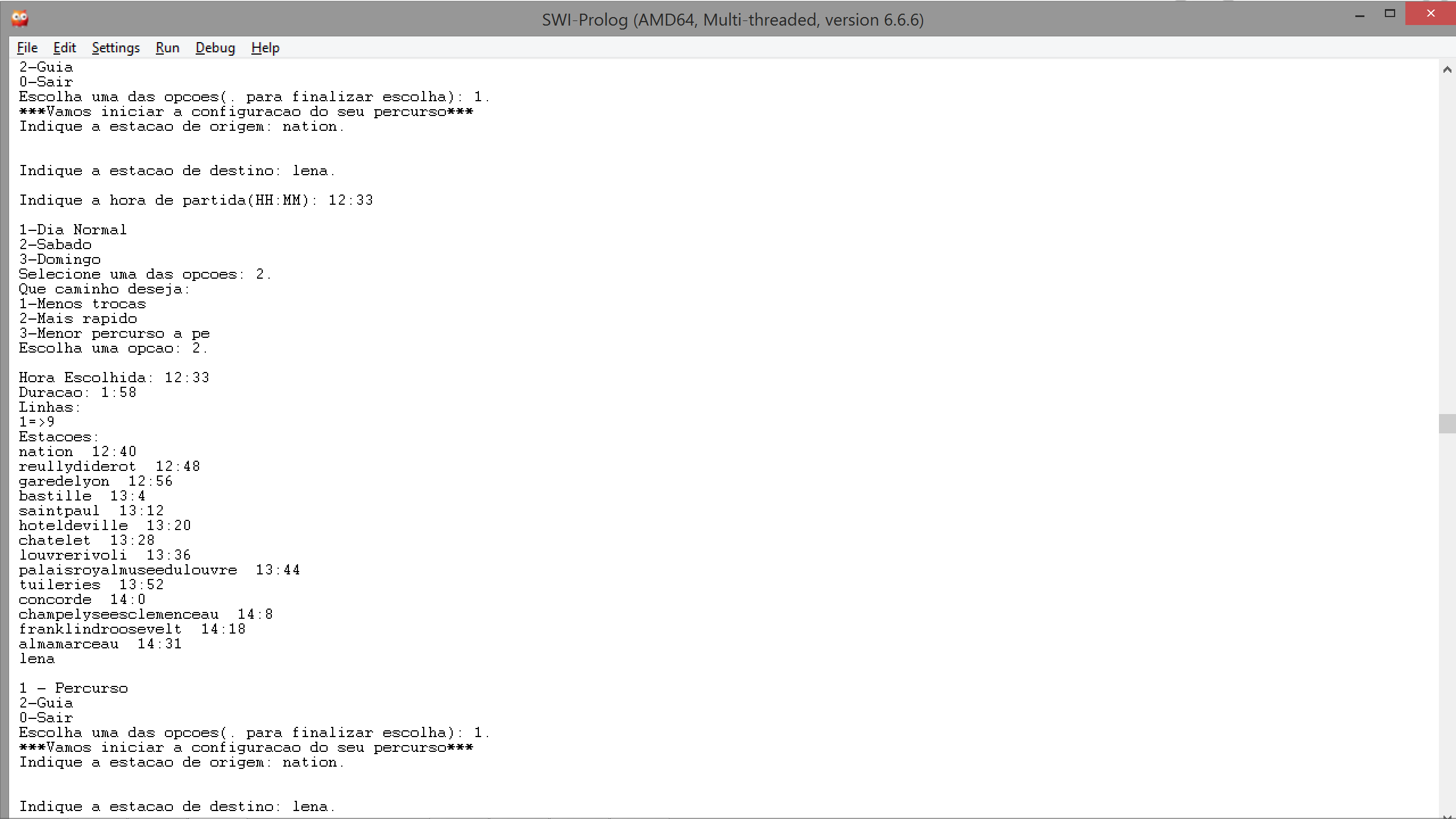


Figura 3 - Percurso mais rápido Nation-Lena

Relativamente ao cálculo não foi possível validar os valores do horário uma vez que tornava o cálculo dos percursos demorado. A sua funcionalidade esta implementada mas a não ser utilizada.

Para o cálculo de percurso de menos percurso a pé, foi utilizado a mesma forma de cálculo do percurso menos trocas, uma vez não ser possível ter os valores ou de tempos, ou de distância dentro das paragens.

## Pontos de Interesse

Para o cálculo do guia o utilizador tem que escolher primeiro o número de pontos de interesse e de seguida dizer quais são. O pograma pega nessa lista de pontos de interesse e aplica-a, por ordem como esta apresentada, os seguintes predicados:

* todos\_guias(Lista, R) – que me devolve todas a possibilidades de percursos entre os pontos de interesse. Podias obter o meu resultado se usasse-mos o algoritmo em largura, mas desta forma ocupa menos espaço na *stack*.
* listas\_pdis\_estacoes(R1,R2) – associa os pontos de interesse às estações, retornando também o tempo de visita em cada ponto.
* somar\_todos(R2,Guia)- associa para cada percurso o total de tempo necessário para percorre-lo.

No fim deste processo o programa pega nas cabeça da lista devolvida pelo predicado somar\_todos(R2,Guia), ordena-a por ordem decrescente e imprimia.

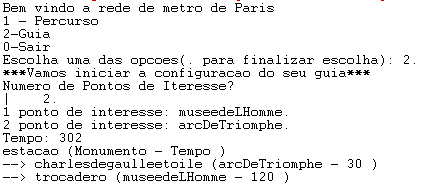


Figura Escolher percurso

## Conclusão

Este trabalho permitiu aprofundar o conhecimento na área de métodos de pesquisa e como os utilizar numa situação real. Aqui era proposto a modelação do metro de paris que nos dava uma grande base de conhecimento, acabamos por optar por não utiliza-la por completo e trabalhar só com algumas linhas. Caso decidíssemos o contrário estaríamos a prejudicar os testes ao programa, pois estes demorariam imenso tempo.

Não conseguimos perceber o conceito de guias de 5h e de 8h, por isso ao pedir um guia ele vai mostrar aquele que esta dentro do intervalo de 5 a 8. Mesmo com a nossa base de conhecimento, tivemos dificuldades na validação dos horários, pois este processo demorava muito.

Infelizmente não conseguimos fazer o último ponto proposto, que era a exportação de um percurso para um ficheiro de texto. O ponto não era muito complicado, mas por falta de tempo não o conseguimos realizar, tendo consciência bastaria só reformular o cálculo de um guia e trabalhar no formato do ficheiro.

## Bibliografia

* Informação relativa as linhas: <http://www.evous.fr/Metros-a-Paris>: