

# Problema do Metrô

Guilherme David Branco<sup>1</sup>, Gabriella de Oliveira Esteves<sup>1</sup>

<sup>1</sup>Departamento de Ciência da Computação  
Universidade de Brasília (UNB)  
Brasília – DF – Brazil

## 1. Introdução

O projeto do Problema do Metro apresenta a situação em que existem no mínimo uma estação de metro, no mínimo um passageiro e apenas um metro, onde o fluxo do percurso depende de certos fatores, como tempo espera do metro na estação, número de passageiros que deixam/saem da estação, etc. A Figura 1 faz uma simples representação do cenário.

O trajeto do metrô é circular e o numero de passageiros total em todas as estações e dentro do próprio metrô é constante, assim como o número de estações. As pessoas tem consciencia de decidirem entrar ou sair do metro, porem isto so pode ser feito quando existe um metro na estacao atual. O metro espera os passageiros por um tempo fixo e entao trava suas portas e vai a proxima estacao.

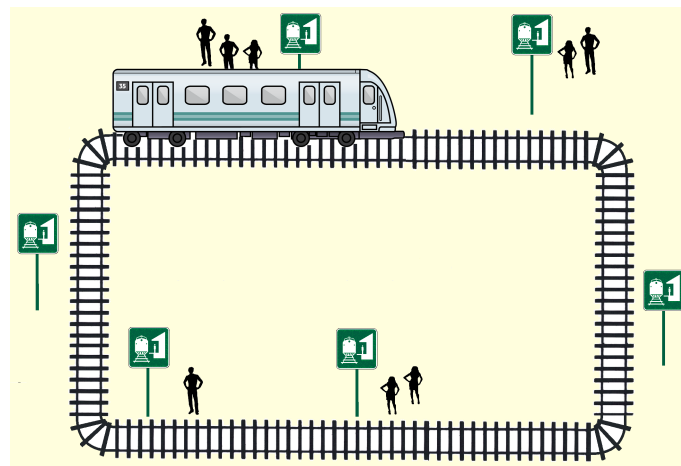


Figura 1. Cenário do Problema do Metrô.

## 2. Objetivo

Encontrar uma solucao para o Problema do Metro, enunciado acima, utilizando recursos de programação concorrente.

## 3. Materiais e Métodos

### 3.1. Descrição das Variáveis

Foram criadas três *structs* no programa, das quais as que se aplicam às pessoas e ao metrô serão representadas como threads.

A *struct estacao.t* possui a variável de condição *avisa*, que é utilizada para avisar aos passageiros que o metrô chegou na estação em questão, e a variável *id* do tipo inteiro,

de identidade que pode variar de 0 à n.

A *struct pessoa\_t* também possui uma variável *id*, que varia de 0 à m, uma variável *estacao\_destino* e uma *estacao\_atual*, que são geradas sempre no início do programa e sempre que o passageiro desce do metrô. Quando o passageiro está dentro do metrô, sua *estacao\_atual* é definida como número de estações + 1, significando que a pessoa esta dentro do metro. Todas as variáveis são do tipo inteiro.

A *struct metro\_t* possui um *mutex* chamado *porta*, que trava a porta do metrô quando seu tempo de espera na estação termina e deve ir a proxima estacao, para que as pessoas nao possam mais entrar. Também possui uma variável de condição, *dentro*, que utiliza para avisar aos passageiros de dentro do metrô que ele chegou em uma estação, representada por sua variável do tipo inteiro *estacao\_atual*. Possui dois semáforos *lotacao* e *avanca*, onde o primeiro lida com o número de passageiros dentro do metrô, e o segundo com o momento em que o metrô deve avançar. A variável *qtd\_pessoas* do tipo inteiro é utilizada somente para representar na saída do programa quantas pessoas estão dentro do metrô. Por fim, as *structs timeval*, *start* e *stop* determinam o intervalo de tempo em que o metrô está com as portas abertas.

Foram criadas cinco constantes que definirão a simulação do programa. MAX\_LOTACAO (número máximo permitido de pessoas no metrô), QTD\_PESSOAS (número total de pessoas), QTD\_ESTACOES (número total de estações), QTD\_METROS (número de metrôs), METRO (estação atual do passageiro que está dentro do metrô (QTD\_ESTACOES+1), duas variáveis que indicam o estado transitido do passageiro (entrar e sair do metrô), que são ESTADO\_ENTRAR e ESTADO\_SAIR, e finalmente TEMPO\_DE\_ESPERA (tempo em segundos que o metrô espera de portas abertas em uma estação). Para este trabalho, as constantes obtiveram os seguintes valores:

- ESTADO\_ENTRAR : 0
- ESTADO\_SAIR : 1
- MAX\_LOTACAO : 10
- QTD\_PESSOAS : 100
- QTD\_ESTACOES : 8
- QTD\_METROS : 1
- METRO : QTD\_ESTACOES+1
- TEMPO\_DE\_ESPERA : 5

### 3.2. Método de Concorrência

Primeiramente foram inicializadas todas as variáveis das três estruturas principais do programa e em seguida foram criadas todas as threads das pessoas, direcionadas para a função *Parada* e do metrô, direcionadas para a função *Viagem*.

A função *Parada* funciona da seguinte maneira : Possui uma maquina de estados, de modo que cada pessoa esteja tentando sair do ou entrar no metro. O estado de saida faz a pessoa sair ao chegar ao seu destino, caso ainda não tenha chegado ao seu destino ela deve esperar dentro do metro, e entao pensa num novo destino que deseja ir. Já o estado de entrar faz a pessoa esperar numa dada estacao atual até que o metro chegue na mesma.

A função *Parada* funciona da seguinte maneira : O metro chega a uma estação e avisa aos seus passageiros em qual estação ele esta para que eles possam sair caso necessário, também avisa às pessoas numa dada estação que ele chegou para que elas possam entrar. Nesta função também é dado um tempo limite para que ele possa avançar a próxima estação.

#### 4. Resultados e Conclusão

```
Metro chegou na estacao 2

Pessoa 82 esperando o metro chegar no destino 6
<<<< Pessoa 10 saiu do metro na estacao 2
>>>> Pessoa 88 entrou no metro na estacao 2 com destino 1
<<<< Pessoa 17 saiu do metro na estacao 2
>>>> Pessoa 93 entrou no metro na estacao 2 com destino 6
<<<< Pessoa 27 saiu do metro na estacao 2
>>>> Pessoa 2 entrou no metro na estacao 2 com destino 5
Pessoa 33 esperando o metro chegar no destino 5
Pessoa 39 esperando o metro chegar no destino 5
<<<< Pessoa 45 saiu do metro na estacao 2
>>>> Pessoa 49 entrou no metro na estacao 2 com destino 7
Pessoa 0 esperando o metro chegar no destino 6
Pessoa 76 esperando o metro chegar no destino 4
Pessoa 6 esperando o metro chegar no destino 3
Pessoa 88 esperando o metro chegar no destino 1
Pessoa 93 esperando o metro chegar no destino 6

Metro chegou na estacao 3

Pessoa 82 esperando o metro chegar no destino 6
Pessoa 49 esperando o metro chegar no destino 7
Pessoa 33 esperando o metro chegar no destino 5
Pessoa 39 esperando o metro chegar no destino 5
Pessoa 0 esperando o metro chegar no destino 6
Pessoa 76 esperando o metro chegar no destino 4
<<<< Pessoa 6 saiu do metro na estacao 3
>>>> Pessoa 60 entrou no metro na estacao 3 com destino 0
Pessoa 88 esperando o metro chegar no destino 1
Pessoa 93 esperando o metro chegar no destino 6
Pessoa 2 esperando o metro chegar no destino 5
Pessoa 60 esperando o metro chegar no destino 0

Metro chegou na estacao 4

Pessoa 49 esperando o metro chegar no destino 7
Pessoa 33 esperando o metro chegar no destino 5
Pessoa 39 esperando o metro chegar no destino 5
Pessoa 0 esperando o metro chegar no destino 6
<<<< Pessoa 76 saiu do metro na estacao 4
>>>> Pessoa 7 entrou no metro na estacao 4 com destino 7
```

Figura 2. Cenário do Problema do Metrô.

A solução está de acordo com o especificado e possui abertura para melhoras, como por exemplo adicionar mais de metrô a execução.