Teoria dos Grafos

Roteamento para Suporte a QoS em Redes IP Multisserviço

*Professor: Dr. Ivan Carlos Alcântara de Oliveira*

*Alunos: Raphael Vidal da Rosa Rgm: 125134-1*

*Eduardo Ferreira Santos Rgm: 126094-4*

*Daniel Marques Gonçalves Rgm: 1505161-7*

Sumário

[Introdução 3](#_Toc452320070)

[Diagrama de Caso de Uso 4](#_Toc452320071)

[Diagrama de Classe 6](#_Toc452320072)

[Descrição do sistema 7](#_Toc452320073)

[8](#_Toc452320074)

[Classe Minheap: 9](#_Toc452320075)

[Exemplos de Funcionamento do Sistema: 11](#_Toc452320076)

[Exemplo 1: 11](#_Toc452320077)

[Exemplo 2: 12](#_Toc452320078)

[Exemplo 3: 13](#_Toc452320079)

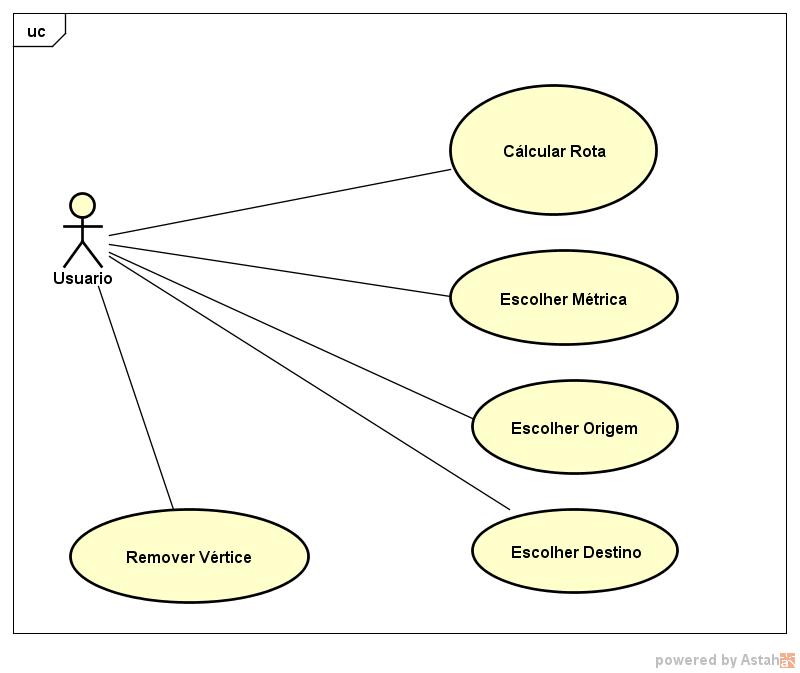
[Exemplo 4: 14](#_Toc452320080)

[Exemplo 5: 15](#_Toc452320081)

# Introdução

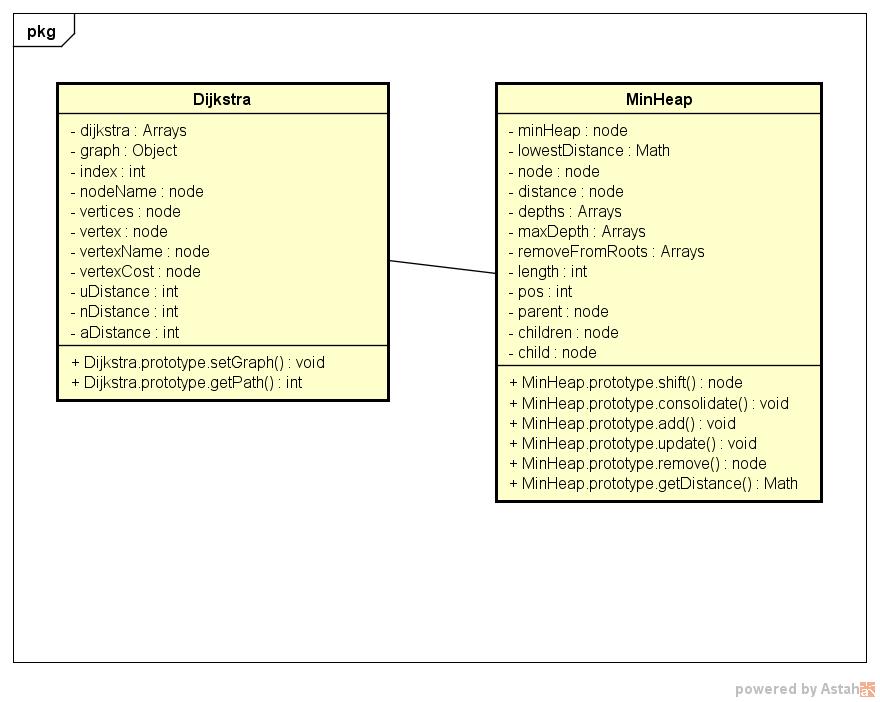
Este Projeto tem como objetivo solucionar problemas de rede de computadores, utilizando a teoria dos grafos e seus respectivos algoritmos de caminhos mínimos para se encontrar a melhor forma de se percorrer em uma rede, para entrega de pacotes, verificando dados como distancias Hops e custos dos caminhos, assegurando controle contra falhas de possíveis links que deixem de operar corretamente assim recalculando o caminho a ser percorrido para que a transmissões dos pacotes não sejam interrompidas.

# Diagrama de Caso de Uso

**

|  |  |
| --- | --- |
| *Nome do Caso de Uso* | Calcular Rota |
| *Sumario* | Usuário usa o sistema para calcular a rota. |
| *Ator Principal* | Usuário |
| *Fluxo Principal* | 1-Usuário escolhe métrica em qual quer calcular a rota.  2- Usuário escolhe a origem.  3- Usuários escolhe o Destino.  4- O sistema calcula a rota e mostra o melhor caminho. |
| *Fluxo Alternativo* | 5- Usuário Pode escolher um dos vértices para remover do caminho e recalcular a rota, sistema ira mostrar um caminho alternativo ao inicial sem o vértice removido. |

# Diagrama de Classe

**

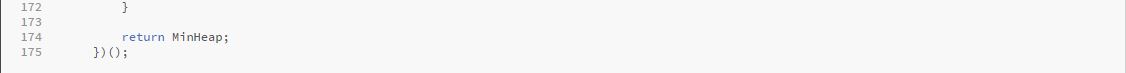
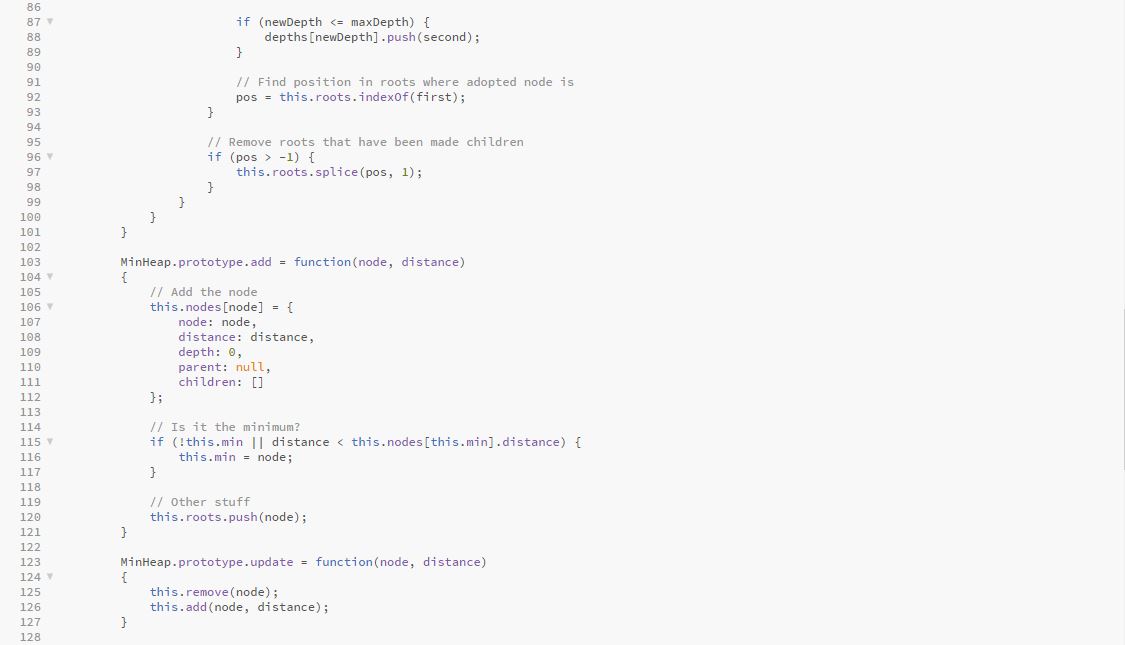
# Descrição do sistema

O sistema foi desenvolvido na linguagem Java Script com HTML5, e o algoritmo utilizado para realizar os cálculos de caminho mínimo foi o Djikstra em vez de utilizarmos matriz de adjacência, foi utilizado à estrutura de dados Heap para melhor eficiência do sistema.

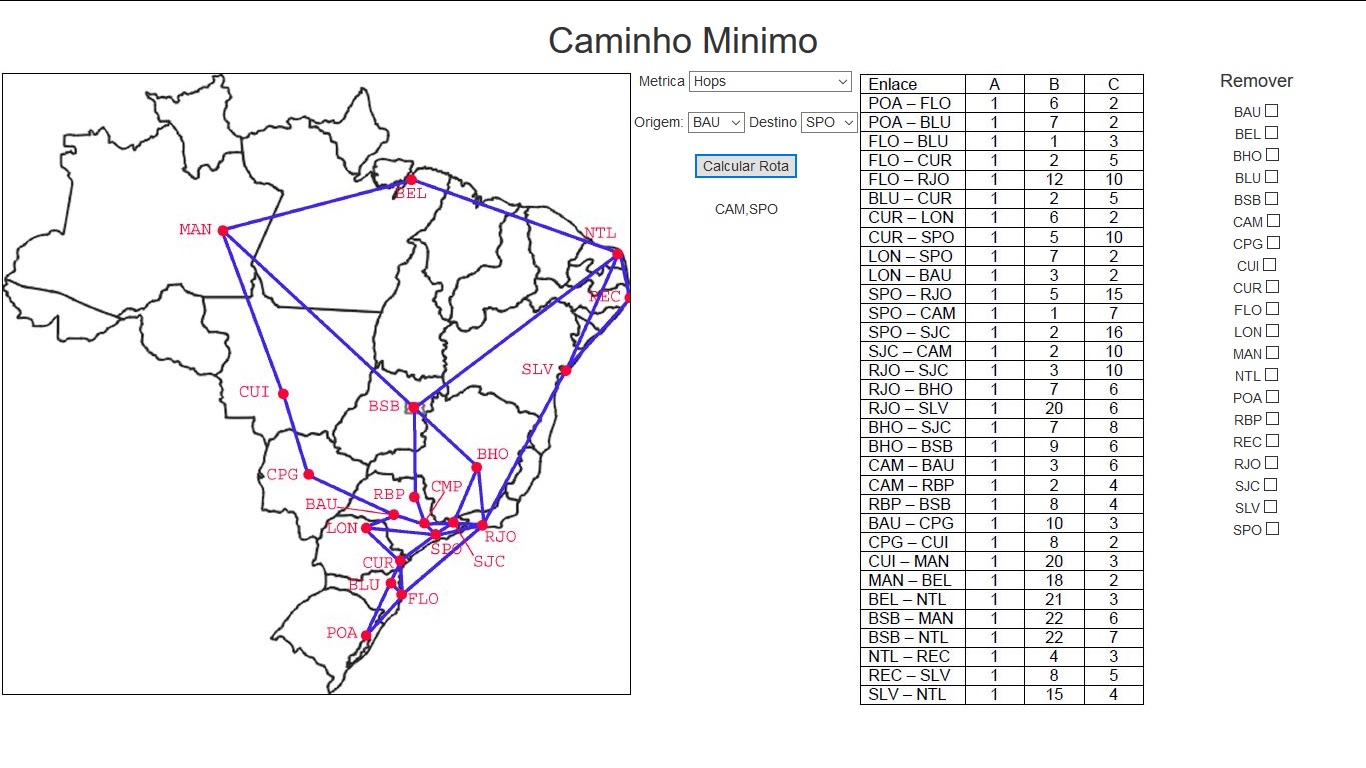
Abaixo imagens do código fonte: Classe Dijkstra.

# 

# Classe Minheap:

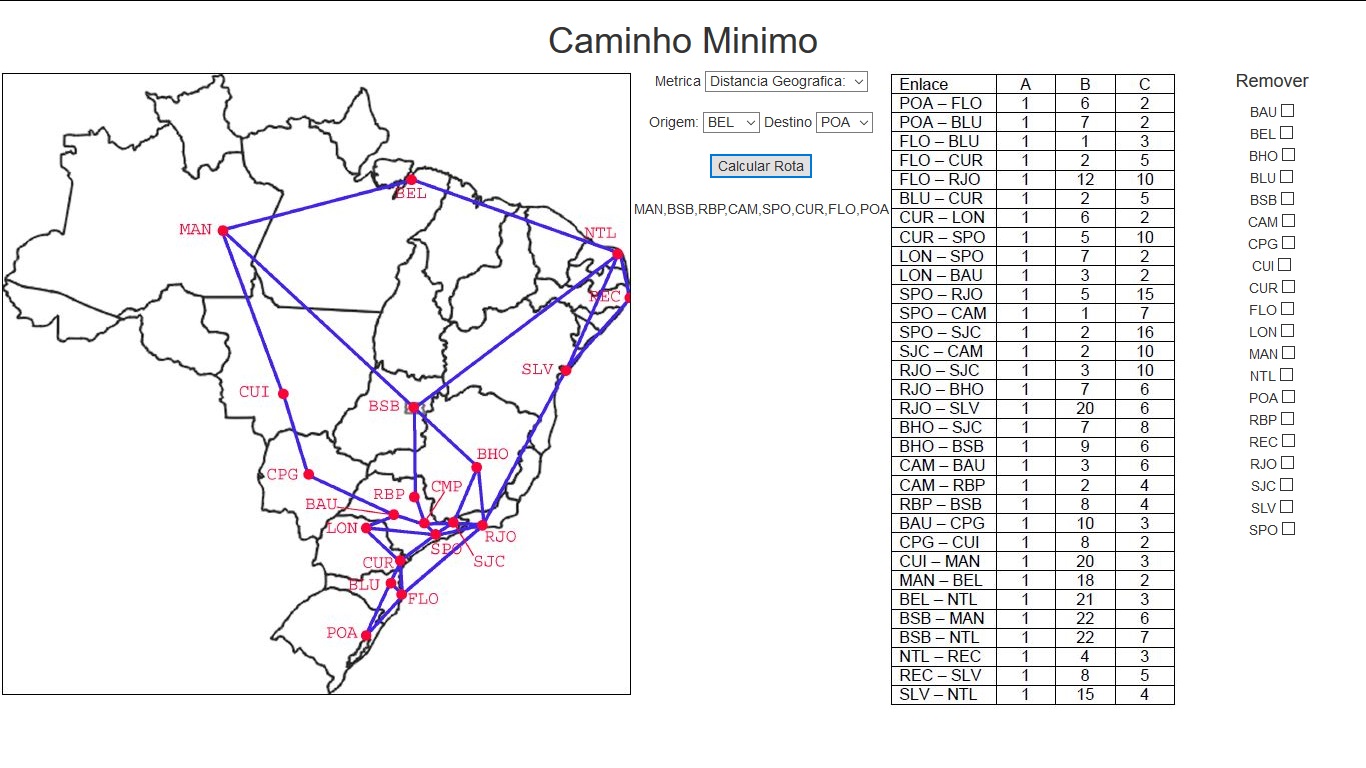


# Exemplos de Funcionamento do Sistema:

**

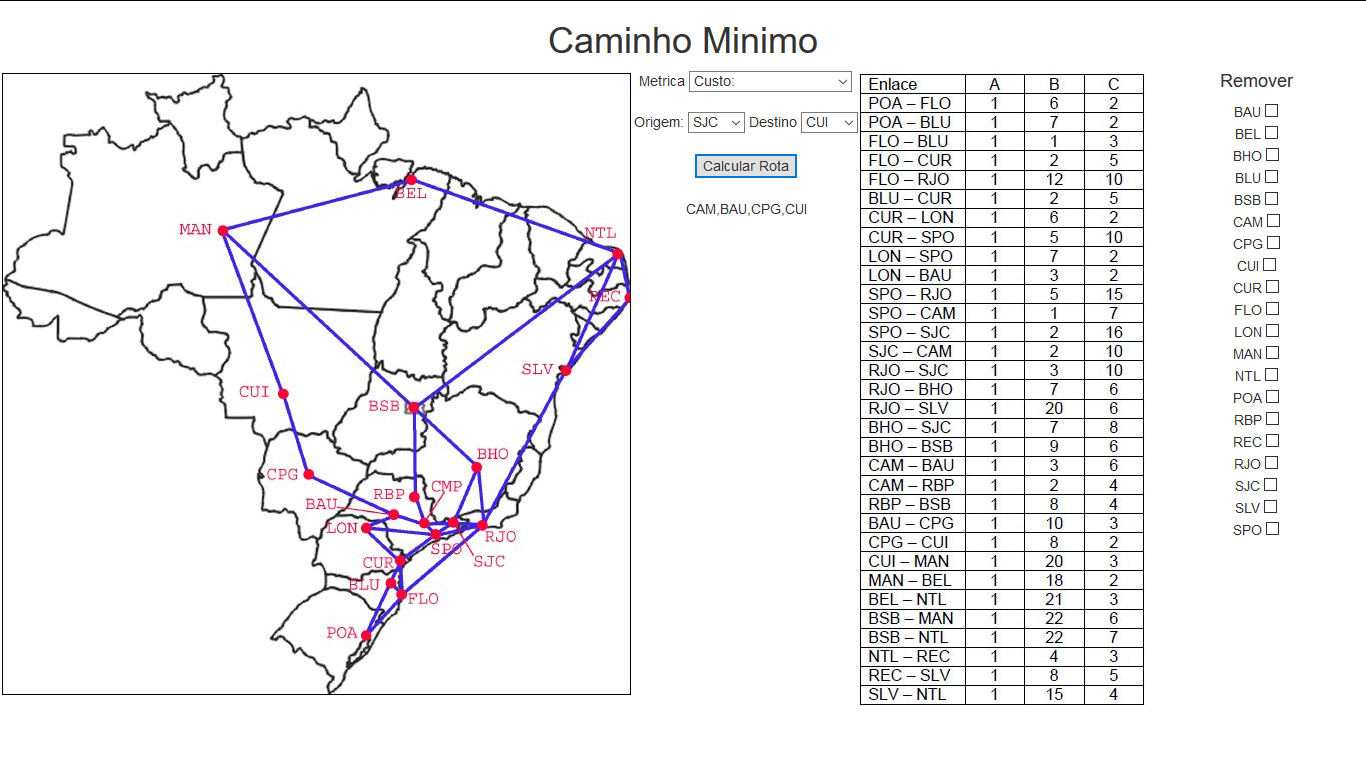
# Exemplo 1:

Escolhido a Métrica por hops e o ponto de origem: BAU e Destino: SPO, o algoritmo encontrou que a melhor rota é (CAM, SP).



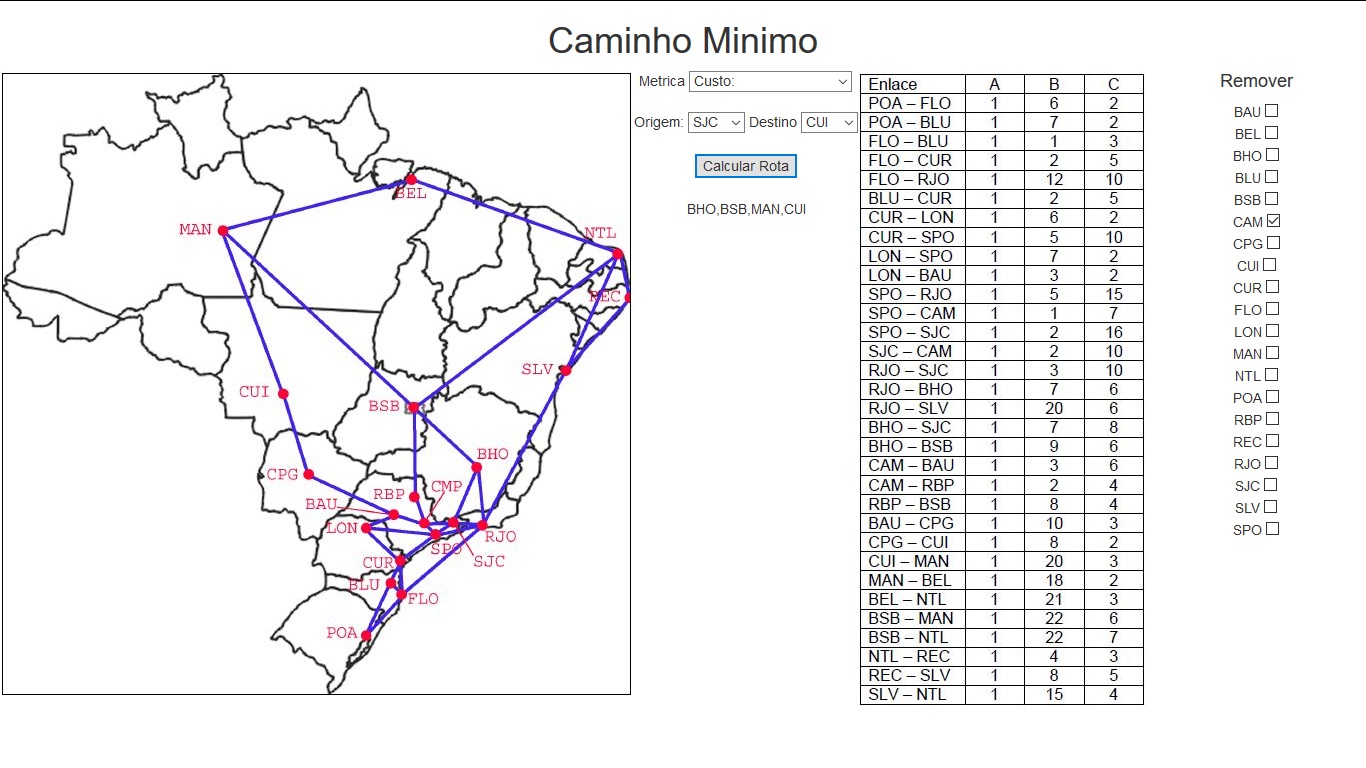
# Exemplo 2:

Escolhido a Métrica por Distancia geográfica e o ponto de origem: BEL e Destino: POA, o algoritmo encontrou que a melhor rota é (MAN, BSB, RBP, CAM, SPO, CUR, FLO, POA).



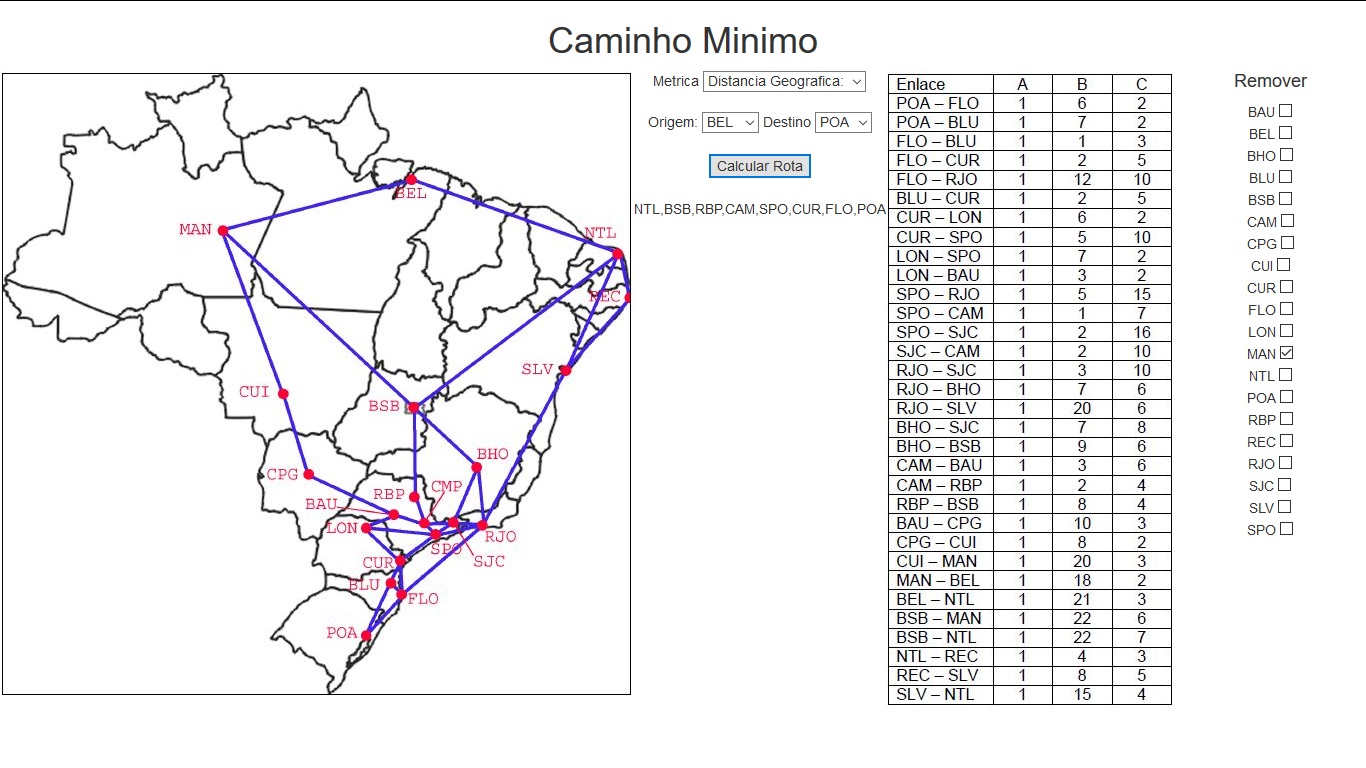
# Exemplo 3:

Escolhido a Métrica por custo e o ponto de origem: SJC e Destino: CUI, o algoritmo encontrou que a melhor rota é (CAM, BAU, CPG, CUI).



# Exemplo 4:

Escolhido a Métrica por custo e o ponto de origem: SJC e Destino: CUI, Só que agora foi removido o Vértice CAM e o algoritmo recalculou a rota e o mesmo escolheu (BHO, BSB, MAN, CUI).



# Exemplo 5:

Escolhido a Métrica por Distancia Geográfica e o ponto de origem: BEL e Destino: POA, Só que agora foi removido o Vértice MAN e o algoritmo recalculou a rota e o mesmo escolheu (NTL, BSB, RBP, CAM, SPO, CUR, FLO, POA).