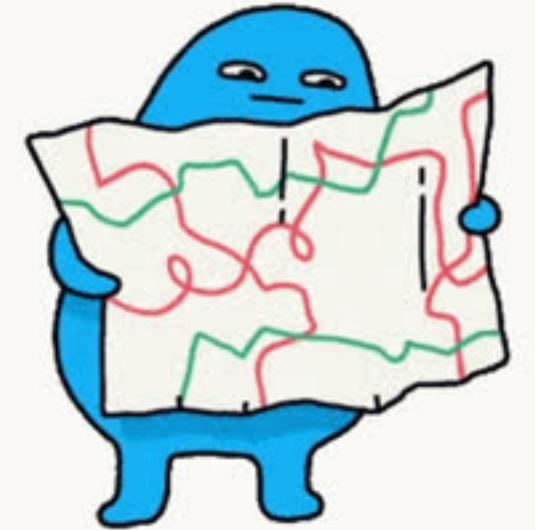


Problema dos mapas



O **Problema dos Mapas**, no contexto da Inteligência Artificial e da Ciência da Computação, refere-se ao **desafio de encontrar caminhos ótimos** em redes geográficas, como estradas, rotas de transporte ou conexões entre localidades.

Algumas aplicações que giram em torno desse problema são:

- Aplicativos de GPS;
- Logística e entrega de mercadorias;
- Jogos digitais;



A linguagem utilizada na implementação foi o **Rust**, escolhida por seu **alto desempenho** e excelente **gerenciamento de memória**, características comparáveis ao C++.

Atualmente, Rust já é adotada em partes do **kernel do Linux**, o que demonstra sua **eficiência e robustez**.

O principal foco da implementação foi a **estrutura de entrada do programa**. As entradas são representadas por **duas strings**, correspondendo às **idades de origem e destino** para a criação do grafo.



Estruturas de Dados e Funções

O Rust **não segue** o paradigma tradicional de **orientação a objetos**, mas oferece formas eficientes de modelagem de dados. As principais estruturas utilizadas no programa foram:

- **City**: representa uma cidade (nome, coordenadas, etc.);
- **GraphStructure**: representa o grafo como um todo;
- **Map**: representa o mapa completo;
- **BoundingBox**: representa a área delimitada entre as cidades de origem e destino;
- **SearchResults**: armazena os resultados e métricas retornadas pelos algoritmos.

As principais funções implementadas, além dos algoritmos de busca, foram:

- **calculate_distance_value()**: calcula a distância entre duas cidades;
- **calculate_level()**: determina o "nível" de cada cidade no grafo;
- **get_city_coordinates()**: obtém o nome, latitude e longitude de uma cidade;
- **get_all_cities_from_bounding_box()**: retorna todas as cidades entre a origem e o destino;
- **calculate_path_distance()**: retorna a distância em linha reta de um caminho percorrido.

Para tentar simular com maior fidelidade a distância real entre as cidades, foi adicionada uma **margem de variação** de 6% a 15% sobre a distância **entre cada cidade** para tentar representar a distância real entre elas

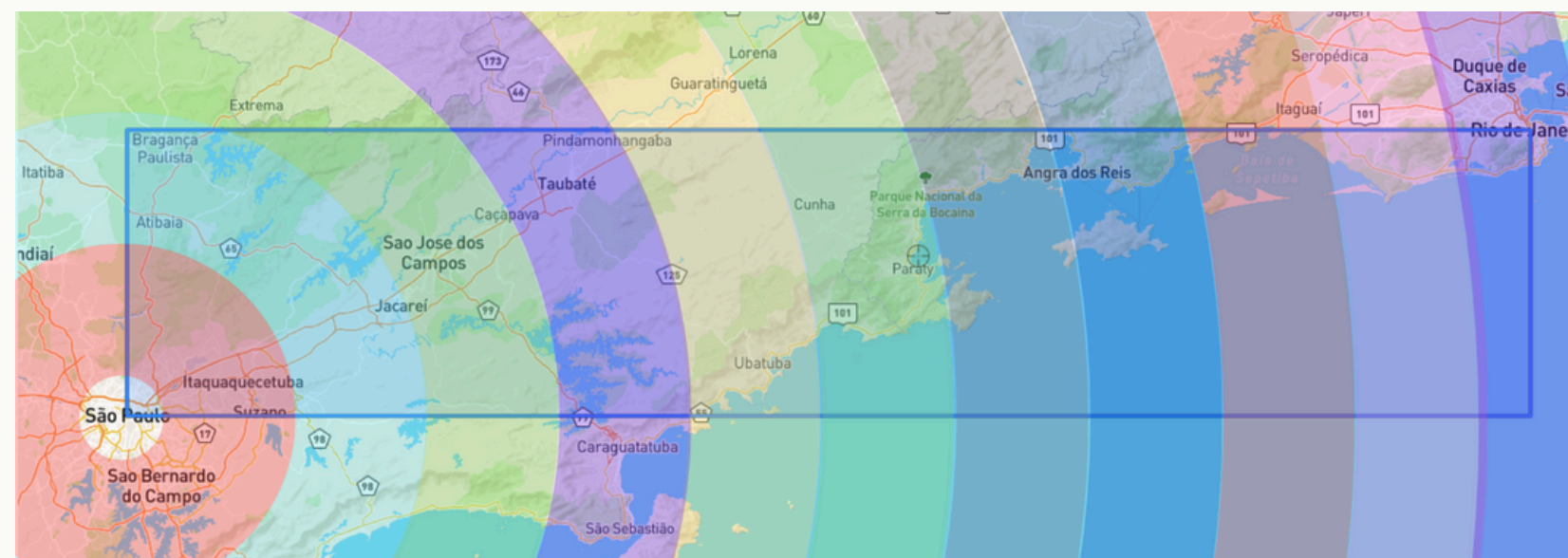


Dificuldades Encontradas

O principal desafio foi definir uma forma eficaz de construir o grafo com cidades reais **mantendo-o conexo**. Em um mapa real, **é possível** se deslocar entre quaisquer cidades por terra, água ou ar.

Assim, a solução adotada foi **delimitar uma área entre as cidades** de origem e destino, com uma **margem de erro controlada**, e capturar todas as cidades dentro dessa região.

Em seguida, as cidades foram **organizadas em níveis de proximidade**, a partir da cidade de origem até a cidade de destino. Cada cidade de um determinado nível é conectada a **todas as cidades** do nível seguinte, formando um grafo acíclico e conexo, conforme ilustrado na imagem ao lado:



Exemplo de execução:

Pegando **duas cidades** como exemplo:

- Cidade de Origem: Juiz de Fora
- Cidade de Destino: Rio de Janeiro

Distância em linha reta entre as cidades – 136KM

Distância por estradas ----- 187KM

Busca Ordenada

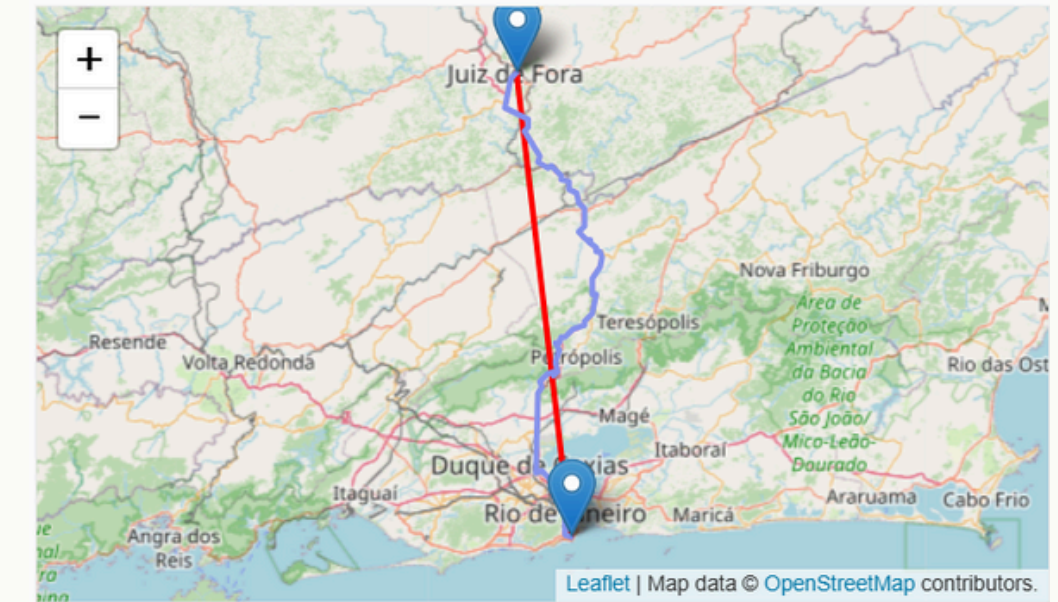
Estatísticas da busca:

- Distância do caminho: 170.25 KM
- Nós visitados: 52
- Nós expandidos: 51
- Profundidade máxima: 6
- Fator de ramificação médio: 4.29
- Tempo de execução: 0.0006 segundos

Busca IDA*

Estatísticas da busca:

- Distância do caminho: 202.52 KM
- Nós visitados: 34
- Nós expandidos: 27
- Profundidade máxima: 6
- Fator de ramificação médio: 4.59
- Tempo de execução: 0.0003 segundos



Qual é a distância entre Juiz De Fora - MG, Brazil e Rio De Janeiro - , Brazil?

Distância em linha reta: **136 Km**
Distância por estradas: **187 Km**

GOSTOU DESTA FERRAMENTA?

Juiz De Fora - MG, Brazil

Latitude: -21.765319
Longitude: -43.348729

Rio De Janeiro - , Brazil

Latitude: -22.976734
Longitude: -43.195084

Links usados para ilustrações e exemplos

[bboxfinder](#) – usado no slide 4

[Distância entre as cidades](#) – usado no slide 5

[Documentação do código](#)

[Repositório do github](#)

Obrigado!

Lukas Freitas de Carvalho – 202376033