



Problema do Caixeiro Viajante usando Algoritmos Genéticos

Gabriel de Paula Meira - 222050003

O Problema do Caixeiro Viajante consiste em encontrar a rota mais curta que permita a um viajante visitar todas as cidades exatamente uma vez e retornar ao ponto de partida.



Algoritmo

O problema não possui um algoritmo determinístico conhecido que resolve o problema em tempo polinomial, portanto, uma estratégia é utilizar de heurísticas capazes de encontrar uma solução aproximada.

A proposta do trabalho é utilizar algoritmos genéticos, que são técnicas de otimização inspiradas na seleção natural, utilizando operadores genéticos, como seleção, cruzamento e mutação, para encontrar soluções aproximadas em problemas complexos de busca e otimização.



Rodando o programa

[> Ver main.cpp <](#) [> Ver graph.hpp <](#) [> Ver file.hpp <](#) [> Ver tsp.hpp <](#)

Para compilar o programa basta possuir o compilador G++ para o código-fonte que está na linguagem C++.

Utilize o comando:

```
g++ main.cpp -o main
```

Em seguida execute com um arquivo de teste, informando o arquivo de entrada:

```
./main tests/co04_dist.txt
```



Testes de Execução

[> Ver pasta ./tests <](#)

A entrada do programa é um arquivo de texto contendo a matriz de adjacências do grafo a ser analisado. Note que o problema exige que a diagonal principal seja composta apenas por zeros.

0	11	6	9
7	0	8	2
7	3	0	3
10	9	4	0

Saída

A primeira linha contém a distância total da melhor solução encontrada pelo algoritmo, seguido da representação do caminho com o índice dos vértices conectados por -> .

(21)

0 -> 2 -> 1 -> 3 -> 0

O exemplo acima é gerado a partir do arquivo de teste [> tests/co04_dist.txt <](#)