

Prática 1 - Grafos

Nesta atividade, vamos explorar o conceito de caminho mínimo utilizando algoritmos como Dijkstra, Bellman-Ford, Floyd. A atividade pode ser feita individual, dupla ou trio.

Vamos explorar três cenários de aplicação, duas em que um grafo já oferecido como entrada e outra em que você precisa montar um grafo a partir de um mapa/grid.

Em cada cenário vocês precisam dizer qual algoritmo foi usado e o motivo.

OBS: Deve ser escolhido um dos algoritmos mostrados em sala!

A implementação é livre, quanto à linguagem e paradigma, mas deve ser fornecido um comparativo do código implementado com o pseudocódigo apresentado em sala. (ver a parte de entrega)

Cenário 1: Determinando a estação central

Considere um grafo não-direcionado com pesos que representam pontos e conexões de metrô. Precisamos definir qual seria a estação central, ou o vértice central do grafo. O vértice central é o que consegue chegar a qualquer um dos outros vértices com o menor custo. Isso leva em conta tanto o somatório das distâncias do vértice em questão a cada um dos demais vértices.

A entrada para esse tipo de problema é um arquivo txt com a seguinte formatação:

```
<num_vertices> <num_arestas>
<vertice_inicial> <vertice_final> <custo> <- Repetição para cada aresta
```

Um exemplo está representado no arquivo *graph1.txt*, disponibilizado como anexo. Foi usado o TAB (\t) como separador.

O resultado deve ser:

- O nó que representa a estação central escolhida
- Um vetor com as distâncias da estação central até os demais vértices
- O vértice mais distante da estação central, junto com o valor de distância
- Uma matriz em que cada linha representa um vértice candidato à estação central e cada coluna é a distância mínima entre o vértice candidato e o vértice representante da coluna.

Cenário 2: Otimizando caminho com regeneração

Considere um carro elétrico com regeneração de bateria eficiente por freio-motor. O carro deve ir da origem ao destino minimizando a energia líquida da bateria (Wh). Trechos subida/arranque consomem energia (peso positivo). Trechos descida/frenagem regenerativa

devolvem energia à bateria (peso negativo).

Modelamos a malha viária como grafo direcionado (a energia não é simétrica por sentido) e o formato do grafo de entrada é similar ao do cenário anterior. O exemplo de entrada está presente no arquivo *graph2.txt*.

O resultado deve ser:

- O caminho mínimo, saindo sempre do vértice 0 até o vértice 6.
- Somatório do custo do caminho.

Cenário 3: Robô de armazém com obstáculos

Um robô de inventário precisa ir do ponto de recarga (S) até a docking station de coleta (G) dentro de um armazém. Há estantes (obstáculos), corredores livres e trechos de piso difícil (ex.: área com pallets baixos) com custo maior. A ideia é propor um mecanismo de busca em grafo que permita encontrar o caminho de menor custo do S ao G, desviando de obstáculos e preferindo corredores “baratos”.

O mapa é definido como um grid, seguindo as seguintes regras:

- '=' célula livre (custo 1)
- '#' obstáculo (intransponível)
- '~' piso difícil (custo 3)
- 'S' = início;
- 'G' = objetivo
- movimento em 4-direções (N,S,L,O)

O arquivo de grid exemplo começa com a definição do número de linhas e colunas, e então as linhas seguintes contêm os caracteres apresentados anteriormente. Ver arquivo *grid_example.txt* para referência.

Entrega

Apenas o link do repositório público no github ou gitlab, pode separar as implementações de cenários em pastas, se for melhor. Deverá conter um arquivo README.md na raiz do projeto com instruções de compilação/utilização. Além disso, deve ser feita uma comparação (pdf, wiki ou no README mesmo) do pseudocódigo do algoritmo apresentado na aula com o código que foi implementado.

- Funcionamento Cenário 1: 2.5 pontos
- Funcionamento Cenário 2: 2 pontos
- Funcionamento Cenário 3: 2.5 pontos
- Instruções: 1 ponto
- Comparativo: 2 pontos

Aviso: Quem não conhece controle de versão git, essa é a hora! Fica registrado quem deu os commits e quando deu. Se a atividade tiver sido feita em equipe, deve ter commit de

todos membros, não de apenas alguns! Isso será observado.