

Algoritmos e Estrutura de Dados

Lista 6

Regras básicas:

1. A atividade deve ser feita individualmente ou em dupla;
2. Deve ser entregue até 18/02 às 23h59 no Classroom da disciplina;
3. Projetos entregues atrasados terão nota zerada.

Orientações:

1. **Localize uma base de dados pública que tenha informações suficientes para resolver este problema. Não pode haver repetição de base de dados nas turmas, logo, cada equipe deve colocar o nome da base de dados que está usando na planilha compartilhada.**
2. A base precisa ter pelo menos >90 vértices e > 100 arestas. **OBS:** A base necessita ter pesos, então, caso você opte por uma sem pesos ou com pesos iguais, você terá que criar uma função que modifique a base de dados adicionando pesos diversos.
3. Você pode construir sua base de dados. Sejam criativos. Mas há a exigência de que a base seja real. Os dados precisam ser acessíveis por qualquer pessoa (públicos) e precisam ser reais.
 - a. Exemplos: Links entre páginas do Wikipedia, seguidores no Twitter, distância entre locais no Google Maps, etc.

Sobre os algoritmos:

- a) **[6 vagas] Utilizar o algoritmo de Dijkstra de menor caminho.** • No seu programa, você deve dar como entrada o Grafo, o vértice que você deseja partir e o vértice que você deseja chegar. O programa deve dar como saída o menor caminho, e seus pesos, entre seus dois vértices.
- b) **[6 vagas] Utilizar o algoritmo de Bellman-Ford de menor caminho.** • A entrada para seu programa deve ser o grafo, o vértice de início, o vértice fim, e a saída o menor menor caminho, e seus pesos, entre esses vértices.

c) [6 vagas] Utilizar o algoritmo de Prim de árvore geradora mínima; •
Você deve passar o grafo como entrada assim como o vértice na qual a busca pela árvore geradora mínima será iniciada.

d) [6 vagas] Utilizar o algoritmo de Kruskal de árvore geradora mínima;
• *A entrada para seu programa deve ser o grafo e a saída a árvore geradora mínima.*

ATENÇÃO: Não pode haver repetição de base de dados. A preferência será por ordem de chegada. Preencher a planilha:

https://docs.google.com/spreadsheets/d/118QT_QqEXPgsGY2dNLcwlF2H_qvVI0-LUgVjicV0roA/edit?usp=sharing

Exemplo de implementação: você pode encontrar a base de aeroportos do Brasil, que tenha os nomes dos aeroportos e a distância entre eles. Cada aeroporto pode ser modelado como vértice de um grafo, e cada aresta pode ser a conexão entre um aeroporto e outro, tendo a distância entre eles como o peso dessa aresta. Dessa forma, seu problema pode ser modelado em como ir de um aeroporto a outro pelo menor caminho, se sua dupla for usar Dijkstra ou Bellman-Ford. Mas se for usar o algoritmo de Prim ou Kruskal, pode resolver o problema de criar a árvore geradora mínima de tal forma que possa conectar o grafo com os menores pesos.

Idéias de base de dados podem ser encontradas aqui:

<http://snap.stanford.edu/data/>

<http://www-personal.umich.edu/~mejn/netdata/>

<https://deim.urv.cat/~alexandre.arenas/data/welcome.htm>

<http://socialnetworks.mpi-sws.org/data-imc2007.html>

<https://lod-cloud.net>

<http://networkrepository.com/networks.php>

<https://www.cs.cmu.edu/~./enron/>

Pontuação extra:

1. Se criar um visualizador de grafos que mostre a saída do seu programa usando alguma biblioteca do Python, terá 0.5 ponto extra na média da unidade.
2. Se criar interface gráfica para seu programa, terá 1 ponto extra na média da unidade.

Dicas:

1. Crie um projeto no github e atualize ele com commits pelo git a cada funcionalidade nova adicionada, assim, os monitores poderão ver a evolução do projeto

OBS: Link explicando sobre branches, como fazer commits pelo git e explicando sobre os commits semânticos

<https://blog.geekhunter.com.br/o-que-e-commit-e-como-usar-commits-semanticos/>)

2. Insira capturas do projeto no readme.md e insira o conteúdo relatório nele.

Entrega (pelo classroom; apenas uma pessoa por dupla):

1. Apenas uma pessoa por dupla entrega, a outra só marca como concluída;
2. Link do projeto postado no github de forma pública.
3. Relatório simples explicando brevemente o contexto do problema (qual a base de dados, o que ela contém), qual algoritmo foi utilizado, quais bibliotecas foram utilizadas (se tiver utilizado), prints do programa rodando, etc;
4. Template do relatório:

<https://docs.google.com/document/d/1g05Snpb621cFySvY87IT4wWSbKdKYD6VWrJDT6tqfm4/edit?usp=sharing>

Advertências:

- É proibido utilizar bibliotecas que implementam funções/algoritmos que estão sendo vistos nesta disciplina.
- Em caso de plágio, a nota do projeto será zerada.