

Relatório Atividade 1 de Grafos (INE5413)

Paulo Almeida, Pedro Silva, Wesly Ataíde

O seguinte relatório foi pedido como questão 6 da atividade 1. O trabalho foi desenvolvido em linguagem python, e portanto as estruturas de dados fornecidas pela linguagem foram de bastante utilidade. A seguir serão comentadas cada estrutura de dados usada em cada exercício:

No exercício 1, foi-se pedido que os alunos criassem uma classe que representasse um grafo não-dirigido e ponderado. Utilizamos a representação de lista de adjacência, sendo implementada em dois dicionários python, um para armazenar as adjacências de cada vértice, isto é, para armazenar as arestas, e outro para os pesos referentes a cada aresta. Também se utilizou uma lista python para armazenar os rótulos referentes a cada vértice do grafo.

No exercício 2, foi-se implementado um algoritmo de busca em largura. Implementamos a fila pedida pelo algoritmo utilizando uma lista python, removendo sempre o elemento da posição inicial e colocando sempre um novo elemento na posição final. Foram usadas também duas outras listas, para marcar vértices já visitados, e para armazenar as distâncias do vértice origem ao vértice visitado. Além destas estruturas de dados, fez-se necessária a utilização de um tipo string para realizar a saída a função.

O exercício 3 pede que um algoritmo que determine se há um e qual é esse ciclo euleriano no grafo de entrada. Foi-se implementado o algoritmo de Hierholzer. Utilizamos uma lista python para armazenar a lista de vizinhos (cada lista de adjacência) de cada vértice, e uma lista para armazenar os vértices do caminho vértices para determinar o caminho atual.

O exercício 4 pede a implementação do algoritmo Bellman-Ford ou de Dijkstra. Optamos pela implementação do algoritmo de Dijkstra, utilizando uma lista para os ancestrais (vértices que já foram testados), uma lista para os vértices já visitados, uma lista para as distâncias dos caminhos. Também foi utilizada uma string para a saída do programa.

Por fim, foi realizada a implementação do algoritmo de Floyd-Warshall no exercício 5. Foram-se utilizadas apenas duas matrizes para o problema, de modo a economizar memória na resolução. As duas matrizes foram concatenadas numa terceira matriz para o laço principal do algoritmo, e o controle de qual matriz é avaliada é realizado utilizando um operador modular no laço mais externo. Também utilizamos strings para a saída do programa.