

## Relatório da Atividade A1 - Grafos

### Participantes:

Caetano Colin Torres

Matheus Henrique Schaly

Wesley da Costa Silva

### 1. Representação

Estrutura de Dados Escolhida: Lista de Adjacências. É usado um dicionário  $d1$  contendo o índice do vértice como chave e um dicionário  $d2$  como valor. O dicionário  $d2$  representa as arestas conectadas com a chave de  $d1$ , onde as chaves de  $d2$  são os vértices que conectam-se com a chave de  $d1$  e o valor das chaves de  $d2$  são os pesos de tais arestas. Nesse caso as complexidades esperadas seriam:  $O(1)$  para encontrar o grau de um vértice,  $O(1)$  para encontrar os vizinhos,  $O(1)$  para encontrar uma aresta,  $O(1)$  para encontrar o peso de uma aresta, e  $O(2|V| + 3|E|)$  para o armazenamento. Além disso o grafo possui uma lista de tuplas de vértices  $E$ , uma lista de conjuntos de vértices  $E\_set$  e uma lista de rótulos  $V$ . O uso ou não desta estrutura de dados dependerá do problema que questão.

### 2. Buscas

Estrutura de Dados Escolhida: Lista de Adjacências. As operações para encontrar vizinhos é  $O(1)$  enquanto em uma matriz de adjacência seria de  $O(n)$ . Além disso, caso tenhamos uma matriz esparsa, estaríamos usando menos espaço se comparado a uma matriz de adjacência.

### 3. Ciclo Euleriano

Estrutura de Dados Escolhida: Lista de Adjacências. As operações para encontrar vizinhos é  $O(1)$  enquanto em uma matriz de adjacência seria de  $O(n)$ . Caso tenhamos uma matriz esparsa, estaríamos usando menos espaço se comparado a uma matriz de adjacência. Além disso, o grafo utilizando a lista de adjacências contém o atributo  $E\_set$  que faz com que tenhamos apenas uma aresta representando uma aresta bidirecional, ajudando na resolução do problema.

### 4. Dijkstra

Estrutura de Dados Escolhida: Lista de Adjacências. As operações para encontrar vizinhos é  $O(1)$  enquanto em uma matriz de adjacência seria de  $O(n)$ . O algoritmo recebe um arquivo como entrada com um grafo e um vértice  $s$ , e como saída o mesmo apresenta o menor caminho a partir do vértice  $s$  até os outros vértices do grafo e o peso deste caminho.

### 5. Floyd-Warshall

Estrutura de Dados Escolhida: Usamos a Matriz de Adjacências, a complexidade do algoritmo com a matriz de adjacências é de  $O(|qtd\_vertices|^3)$ . Escolhemos a matriz porque é mais trivial e eficiente para resolver o problema. O algoritmo recebe um arquivo de entrada com um grafo e mostra as distâncias dos caminhos mínimos para cada par de vértices. Como usamos grafos dirigidos as posições  $[u][v]$  e  $[v][u]$  da matriz devem ser iguais.