

Representação: Crie um tipo estruturado de dados ou uma classe que represente um grafo não-dirigido ponderado $G(V, E, w)$, no qual V é o conjunto de vértices, E é o conjunto de arestas e $w: E \rightarrow \mathbb{R}$ é a função que mapeia o peso de cada aresta $\{u, v\} \in E$. As operações/métodos contemplados para o grafo deverão ser:

- qtdVertices():** retornar a quantidade de vértices; $- \text{int}$
- qtdArestas():** retorna a quantidade de arestas; $- \text{int}$
- grau(v):** retorna o grau do vértice v ; \rightarrow lista de adj
- rotulo(v):** retorna o rótulo do vértice v ;
- vizinhos(v):** retorna os vizinhos do vértice v ;
- haAresta(u, v):** se $\{u, v\} \in E$, retorna verdadeiro; se não existir, retorna falso;
- peso(u, v):** se $\{u, v\} \in E$, retorna o peso da aresta $\{u, v\}$; se não existir, retorna um valor infinito positivo;
- ler(arquivo):** deve carregar um grafo a partir de um arquivo no formato especificado ao final deste documento.

IMPORTANTE: As operações/metodos deverão ter complexidade de tempo computacional $O(1)$ quando possível.

Em caso de dúvidas, consulte o professor da disciplina.

2. [Buscas] (2,0pts) Crie um programa que receba um arquivo de grafo e o índice (de 1 a n) vértice s como argumentos. O programa deve fazer uma busca em largura³ a partir de s e deverá imprimir a saída na tela, onde cada linha deverá conter o nível seguido de “.” e a listagem de vértices encontrados naquele nível. O exemplo abaixo trata de uma saída, na qual a busca se iniciou pelo vértice s no nível 0, depois prosseguiu nos vértices 3, 4 e 5 para o próximo nível. No nível seguinte, a busca encontrou os vértices 1, 2, 6 e 7.

```
0: 8
1: 3,4,5
```

vector X

arr. / Hash

mapa < destino, peso >

1 0

2 1

3 2

$X[1][0]$