

1 Laboratório 1

O laboratório consiste na implementação de uma biblioteca para manipulação de grafos. Mais especificamente, você deve implementar três funções no arquivo `grafo.c`. A função `leGrafo(Grafo** g, int n)`, a função `Dijkstra(Grafo* g)` e a função `destroiGrafo(Grafo** g)`.

A função `leGrafo(Grafo** g, int n)` deve ler um grafo da entrada padrão. O número n de vértices do grafo é passado como parâmetro para essa função, portanto, n não deve ser lido da entrada padrão dentro da função `leGrafo`.

A função `Dijkstra(Grafo* g)` deve calcular a distância mínima entre o primeiro vértice do vetor de vértices e os demais vértices do grafo g , ou seja, essa função deve implementar o algoritmo de Dijkstra. Ao final, essa função deve imprimir a distância mínima entre o primeiro vértice e todos os outros vértices do grafo.

A função `destroiGrafo(Grafo** g)` deve desalocar o espaço utilizado pelo grafo.

Algoritmo de Dijkstra

```
dados grafo(V,A) e vértice s inicial
  declare os vetores dist, prec e aberto;
  atribua dist[s] = 0 e dist[i] = INF para todos outros vértices;
  inicie o vetor de precedentes com um valor inválido;

  enquanto houver vértice aberto
    seja k o vértice aberto com o menor custo (menor distância)
    feche k
    para todo vértice aberto j que seja sucessor de k
      se ( dist[k] + d[k,j] < dist[j] )
        dist[j] = dist[k] + d[k,j]
        prec[j] = k;
```

Alocação e desalocação dinâmica de matrizes

```
int **matriz;
int tam1, tam2;
matriz = new int *[tam1];

for(int i = 0; i < tam1; i++)
  matriz[i] = new int [tam2];

for(int i = 0; i < tam1; i++)
  delete matriz[i]

delete *matriz;
```