



Lista 04

Prazo para entrega: 02-08-2023

1 Instruções

- A entrega deve ser realizada presencialmente no início da aula.
- A lista deve ser feita à mão (caso você tenha um tablet com caneta stylus, você pode fazer usando o table e então entregar uma versão impressa da lista).
- A lista deve ser feita à caneta e sem resura.
- A lista é um trabalho individual.
- **Você deve entregar duas vias da sua lista de exercício (uma versão original e uma versão fotocopiada).**
- **Cada uma das vias deve estar grampeada separadamente.**

2 Exercícios

1. Uma fila de prioridades é uma fila na qual o elementos são processados de acordo com suas prioridades. As operações mais comuns de uma fila de prioridades são:

INSERIRFILA(Q, e, p) operação que insere um elemento e com prioridade p em uma fila de prioridades Q .

EXTRAIRMINFILA(Q) operação que extrai e retorna o elemento de menor prioridade da fila Q .

EHVAZIA(Q) operação que retorna Verdadeiro se a fila Q é vazia e Falso, caso contrário.

ATUALIZAPRIORIDADEFILA(Q, u, p) operação que atualiza a prioridade do elemento u para p na fila de prioridade Q .

PEGAPRIORIDADEFILA(Q, v) operação que retorna a prioridade do elemento v .

Fila de prioridade é um tipo abstrato de dados e, por conseguinte, pode ser implementado de diversas formas.

- (a) Assumindo que a estrutura matriz de adjacências foi utilizada para representar o grafo e que uma heap binária foi utilizada para representar a fila de prioridades, analise o tempo de execução do Algoritmo 1.

Algorithm 1 Algoritmo de Prim

```
1: Função PRIM( $G, w$ )  $\triangleright w: E(G) \rightarrow \mathbb{R}$  é uma função que dá custo às arestas. Qual estrutura de dados você usará para representar isso?
2:   Cria uma fila de prioridades  $Q$ 
3:   Para cada  $u \in V(G)$  faça
4:     INSERIRFILA( $Q, u, \infty$ )
5:      $pred[u] = Null$ 
6:   Seja  $s \in V(G)$ 
7:    $pred[s] = s$ 
8:   ATUALIZAPRIORIDADEFILA( $Q, s, 0$ )
9:   Enquanto não EHVAZIA( $Q$ ) faça
10:     $u = ExtrairMinFila(Q)$ 
11:    Para cada  $v \in N(u)$  faça
12:      Se  $v \in Q$  e  $w(uv) < PEGAPRIORIDADEFILA(Q, v)$  então
13:         $pred[v] = u$ 
14:        ATUALIZAPRIORIDADEFILA( $Q, v, w(uv)$ )
```

2. Analise o tempo de execução da função DFS(G, s).

```
1: Função DFS( $G, s$ )  $\triangleright$  onde  $G$  é um grafo e  $s \in V(G)$ 
2:   Para cada  $u \in V(G)$  faça
3:      $vis[u] = Falso$ 
4:      $pred[u] = Null$ 
5:    $pred[s] = s$ 
6:   DFS_SEARCH( $G, s$ )
7: Função DFS_SEARCH( $G, u$ )
8:    $vis[u] = Verdadeiro$ 
9:   Para cada  $v \in N(u)$  faça
10:    Se  $vis[v] == Falso$  então
11:       $pred[v] = u$ 
12:      DFS_SEARCH( $G, v$ )
```