

7ª Série de exercícios – Teoria dos Grafos
Busca em grafos e os Algoritmos BFS e DFS

1) A busca em grafos é uma das tarefas mais importantes em computação pois nos fornece maneiras de percorrer um grafo G acessando seus elementos (vértices) em uma determinada “ordem”. Responda:

a) Tendo como base o pseudo-código a seguir, explique o funcionamento do algoritmo de Busca em Largura (BFS). Comente sobre as estruturas de dados bem como o papel das primitivas empregadas.

```

BFS( $G, s$ )
1  for each vertex  $u \in G.V - \{s\}$ 
2       $u.color = WHITE$ 
3       $u.d = \infty$ 
4       $u.\pi = NIL$ 
5   $s.color = GRAY$ 
6   $s.d = 0$ 
7   $s.\pi = NIL$ 
8   $Q = \emptyset$ 
9  ENQUEUE( $Q, s$ )
10 while  $Q \neq \emptyset$ 
11      $u = DEQUEUE(Q)$ 
12     for each  $v \in G.Adj[u]$ 
13         if  $v.color == WHITE$ 
14              $v.color = GRAY$ 
15              $v.d = u.d + 1$ 
16              $v.\pi = u$ 
17             ENQUEUE( $Q, v$ )
18      $u.color = BLACK$ 

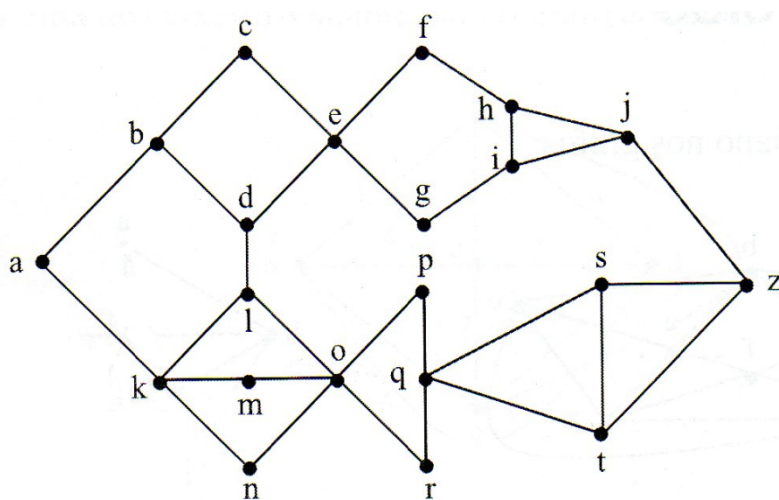
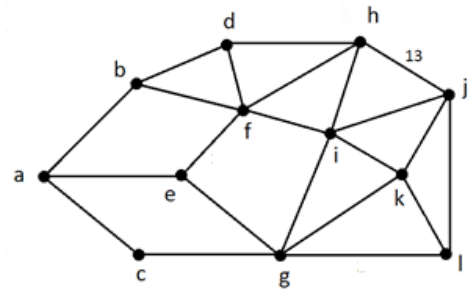
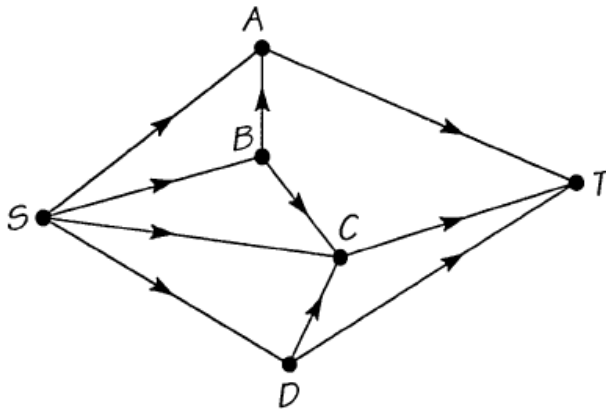
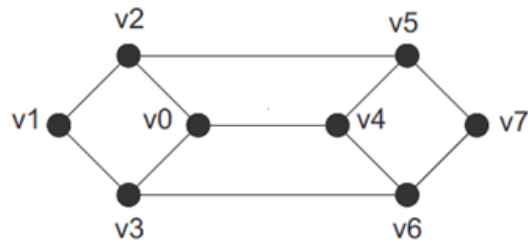
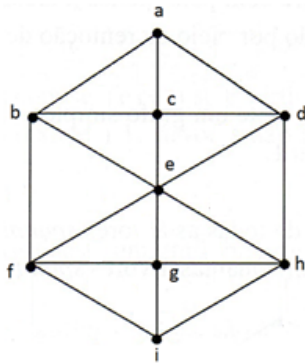
```

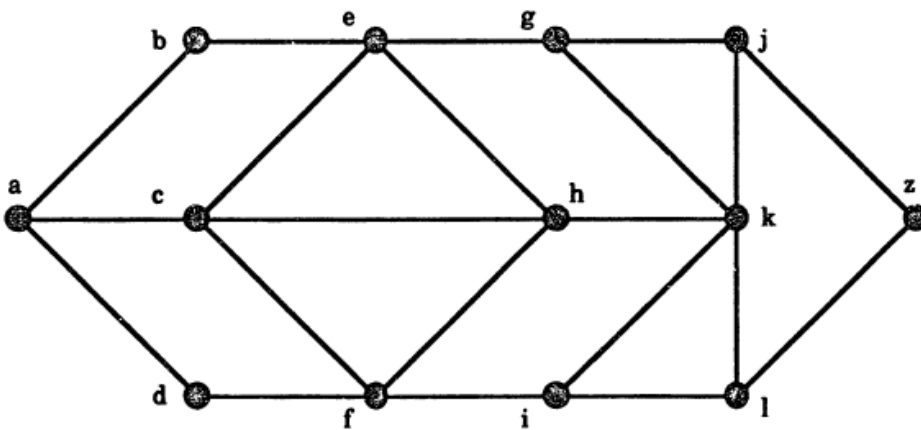
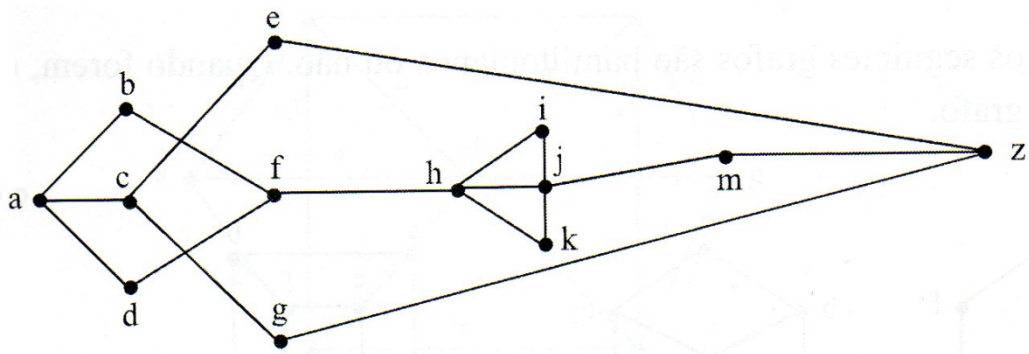
b) Mostre que o algoritmo BFS produz uma árvore de profundidade mínima, ou seja, que ele termina com $\lambda(v) = d(s, v)$, onde $d(s, v)$ é a menor distância da raiz s ao vértice v .

c) Tendo como base o pseudo-código a seguir, explique o funcionamento do algoritmo de Busca em Profundidade (DFS).

<pre> DFS(G) 1 for each vertex $u \in G.V$ 2 $u.color = WHITE$ 3 $u.\pi = NIL$ 4 $time = 0$ 5 for each vertex $u \in G.V$ 6 if $u.color == WHITE$ 7 DFS-VISIT(G, u) </pre>	<pre> DFS-VISIT(G, u) 1 $time = time + 1$ 2 $u.d = time$ 3 $u.color = GRAY$ 4 for each $v \in G.Adj[u]$ 5 if $v.color == WHITE$ 6 $v.\pi = u$ 7 DFS-VISIT(G, v) 8 $u.color = BLACK$ 9 $time = time + 1$ 10 $u.f = time$ </pre>
---	--

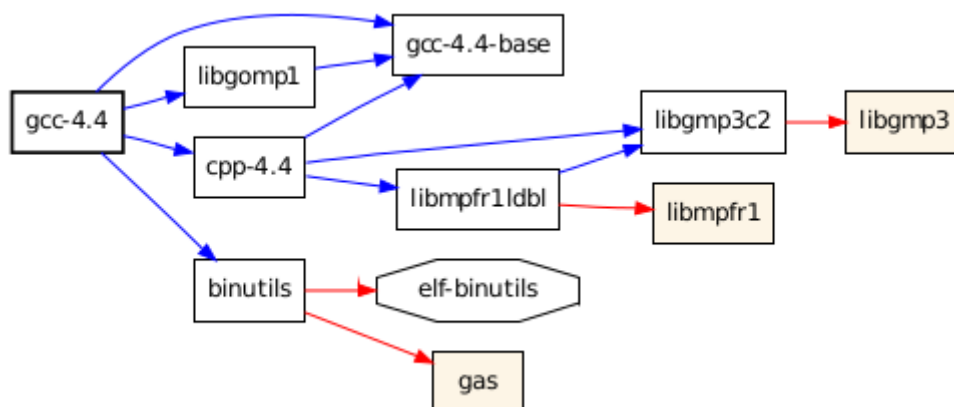
2) Execute os algoritmos de busca em largura e busca em profundidade nos grafos a seguir. Após a execução especifique árvores BFS (Breadth-First Search) e DFS (Depth-First Search) com s vértices rotulados (distância geodésica ou tempos de entrada e saída. Encontre também o número de caminhos mais curtos do vértice inicial (**a**, **v0**) ao vértice final (**z**, **v7**). Construa as árvores resultantes e veja a relação entre as árvores BFS e DFS. Qual a principal diferença entre elas? Faça o trace completo dos algoritmos.





3) Utilizando o algoritmo DFS, realize a ordenação topológica dos DAG's (direct acyclic graphs) a seguir:

a)



b)

