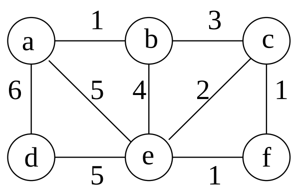
Analysis of Algorithms: Solutions 7

```
                    X
        X           X
        X           X
        X           X   X
        X           X   X
    X   X           X   X
    X   X   X       X   X
X   X   X   X       X   X
X   X   X   X   X   X   X
--------------------------
4   5   6   7   8   9   10
         grades
```

Problem 1



(a) Illustrate the steps of Kruskal's algorithm on the above undirected graph.

Kruskal's algorithm may add edges in any of the following six ways:

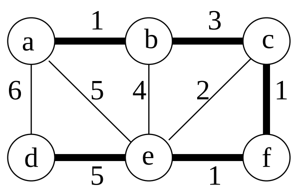
(1) (a, b), (c, f), (e, f), (b, c), (d, e)
(2) (a, b), (e, f), (c, f), (b, c), (d, e)
(3) (c, f), (a, b), (e, f), (b, c), (d, e)
(4) (c, f), (e, f), (a, b), (b, c), (d, e)
(5) (e, f), (a, b), (c, f), (b, c), (d, e)
(6) (e, f), (c, f), (a, b), (b, c), (d, e)

(b) Show the steps of Prim's algorithm on the same graph, with vertex d as the source.

Prim's algorithm adds edges in the following order:

$$(d, e),\ (e, f),\ (c, f),\ (b, c),\ (a, b).$$

Both algorithms construct the same spanning tree:



Problem 2
Give a BFS algorithm for the adjacency matrix, and determine its running time (O-notation).

```
MATRIX-BFS(G, s)
for each vertex u ∈ V[G]
    do color[u] ← WHITE
       d[u] ← ∞
       parent[u] ← NIL
color[s] ← GRAY
d[s] ← 0
Q ← {s}
while Q is not empty
    do u ← DEQUEUE(Q)
       for each vertex v ∈ V[G]
           do if Adj[u, v] = 1 and color[v] = WHITE
                 then color[v] ← GRAY
                      d[v] ← d[u] + 1
                      parent[v] ← u
                      ENQUEUE(Q, v)
       color[u] ← BLACK
```

The running time is $O(V^2)$.

Problem 3
Give a nonrecursive version of depth-first search.

```
ITERATIVE-DFS(G)
for each vertex u ∈ V[G]
    do color[u] ← WHITE
       parent[u] ← NIL
initialize an empty stack S
for each vertex u ∈ V[G]
    do if color[u] = WHITE
          then PUSH(S, u)
               while S is not empty
                   do v ← POP(S)
                      for each w ∈ Adj[v]
                          do if color[w] = WHITE
                                then color[w] ← GRAY
                                     parent[w] ← v
                                     PUSH(S, w)
                      color[v] ← BLACK
```

The running time is the same as the time of the recursive algorithm, $\Theta(V + E)$.