Aashish Yadavally
Fall 2019

Homework V

Question 1

1.

S	**A**	**B**	**C**	**D**	**E**	**G1**	**G2**
Gray	Gray	Black	Gray	Gray	Black	Black	Black

2.

S	**A**	**B**	**C**	**D**	**E**	**G1**	**G2**
(1, 16)	(2, 15)	(3, 10)	(11, 14)	(12, 13)	(4, 9)	(5, 8)	(6, 7)

Question 2

1. We start the depth first traversal from the vertex $v \in V$, for which we need to check if is contained in a directed cycle. Through the traversal, if we visit a vertex which has already been marked visited, this means that the path represents a back edge, and the vertex v is contained in a directed cycle.

2.
```
INPUT : Adjacency list of graph, 'adj'; vertex v in V
OUTPUT : True, if graph contains a directed cycle.
         False, if graph doesn't contain a directed cycle

DETECT-CYCLES (G, adj, v)
    root <- v
    for each s in G.V
        s.visit = False
        s.pi = NULL
    return RECURSIVE-DFS(adj, v, root)
end function

RECURSIVE-DFS (adj, v, root)
    v.visit = True
    for each u in adj[v]
        if not u.visit
            u.pi = v
            if RECURSIVE-DFS(adj, u, root) is True
                return True
        elif u is equal to root and u.pi not equal to root
            return True
    return False
end function
```

Question 3

Proof:
Let T be a spanning tree found by Prim's algorithm for a given graph G, and let the minimum spanning tree be T'.
Let us assume that T is not the same as T', i.e, $T - T' \neq \phi$.
Let an edge from u to v be any edge in $T - T'$ crossing some cut $(S, V - S)$. This edge was added to T based on the Prim's algorithm, greedily, such that it is the least-cost edge crossing some cut $(S, V - S)$.
Since T' is an MST, there should be a path from u to v in T' such that it begins in S, ends in $V - S$. So, there is a path in T' from p to q along the path from u to v. Since (u, v) is the least-cost edge, cost of (u, v) is less than the cost of (x, y).
Let $T'' = T' \cup (u, v) - (x, y)$. Since (x, y) is on the cycle formed by adding (u, v), this means that T'' is also a spanning tree. However, cost of T'', $C(T'') = C(T') + C(u, v) - C(x, y)$.
This means that $C(T'') < C(T')$, contradicting that T' is an MST.
Thus, our assumption should have been wrong, and by proof of contradiction, $T = T$". Therefore, T is also a minimum spanning tree. Thus, it is proved that the spanning tree formed by the Prim's algorithm, based on the greedy choice of choosing the least-cost edge, forms a minimum spanning tree.

Question 4

1. On running the Prim's algorithm, the contents of set A assuming that we begin from vertex F, when it contains exactly 6 vertices are:
   {(F, I), (I, G), (I, E), (E, D), (D, C)}
2. On running the Kruskal's algorithm, the contents of set A, when it contains exactly 6 vertices are:
   {(D, E), (I, G), (I, E), (D, C), (C, H)}

Question 5

Given a directed graph $G = (V, E)$.
The All-Pairs Reachability problem can be solved using the Warshall's Algorithm, wherein, the recursive solution can be built from the adjacency matrix of the graph, which indicates the pair of vertices connected by an edge and is represented by r^0. By iterating over the matrix n number of times (where n is the number of vertices), according to the objective function defined below.

This can be built such that $r^1(i, j)$ contains paths from i to j going through the vertex

1; $r^2(i,j)$ contains paths going through the vertices 1 and/or 2; $r^k(i,j)$ contains paths going through any of the other vertices.

The objective function can be thus defined, where 'k' refers to the k^{th} iteration:

$$r^k(i,j) = \begin{cases} r^{k-1}(i,j) & \text{(path using just 1, ..., k-1)} \\ \text{r}^{k-1}(i,k) \text{ AND } r^{k-1}(k,j) & \text{(path from i to k and from k to j using just 1, ..., k-1)} \\ \end{cases}$$

The base case for $r(u,v)$ would be the adjacency matrix for the graph.

The worst case time complexity based on this dynamic programming idea would be $O(n^3)$, where 'n' is the number of vertices of the graph.