CS-510 Problem Set 3

Hassan Ali Khan Student ID: 15030044

May 14, 2017

Problem 1. Given a graph $G = (V, E)$, prove that for $s \in V$, $BFS(s)$ will visit only those vertices that are reachable from s.

Answer 1. In BFS 's' vertex is added in Q then the loop line of code is while (Q is not empty) - on every iteration
head of Q, 'V', is dequeued and all such vertices 'U' where E(V,U) are added which are not visited yet are enqueued in Q and so on.This means every vertex by any path connected to 's' is visited and every vertex which is not connected to 's' by any possible path is not visited in BFS.

Problem 2. Given a graph $G = (V, E)$. Let For $a, b \in V$, let $d(a, b)$ be the length of the shortest path between a and b. Suppose we run $BFS(s)$ for $s \in V$. Prove that a vertex $x \in V$ is in level i of the $BFS(s)$ tree if and only if $d(s, x) = i$.

Answer 2. The BFS holds true for sub-path optimality theorem, as every neighbour of specific node of level '1' is inserted in queue at level '2', this can be extended to level 'i' vertex as well. Which proves that rove that a vertex x is in level 'i' of the BFS tree if and only if distance from source vertex to x is =i.

Problem 3. Given an undirected graph G on n vertices, design an algorithm that determines whether G is a tree.

Answer 3. There definition of tree is an undirected fully connected graph with no cycles.
1) If an unexplored edge leads to a node that already exists in visited array, then the graph contains a cycle and in that case it is not a tree.
2) Run BFS and after its execution check the size of visited array if it is not equal to number of vertices in graph then it can not be a tree.

Problem 4. Given a connected, undirected graph G, design an algorithm that determines if G is bipartite or not. Also figure out how many steps this algorithm would take

Answer 4. We will need a mofified version of BFS to find bipartition, basic steps involved in algorithm are,
1) Assign RED color to the source vertex (putting into set U).
2) Color all the neighbors with BLUE color (putting into set V).
3) Color all neighborâĂŹs neighbor with RED color (putting into set U).
4) This way, assign color to all vertices such that it satisfies all the constraints of m way coloring problem where m = 2.
5) While assigning colors, if we find a neighbor which is colored with same color as current vertex, then the graph cannot be colored with 2 vertices (or graph is not Bipartite).

Problem 5. Prove the following statement. You're given a graph $G = (V, E)$. Suppose the shortest path from $s \in V$ to $t \in V$ is $s, v_1, v_2, \ldots, v_k, t$, then the length of the shortest path from s to v_i must be exactly i.