CSE 3101: Design and Analysis of Algorithms

Assignment on graph algorithms

1. Do problem 22-1 (page 558 in Edition 2), (pg 621 in Edition 3).
2. Do problem 23.1-11 (page 567 in Edition 2), (page 630 in Edition 3). Prove that your algorithm is correct and compare its running time with that of running Prim's or Kruskal's algorithm from scratch.
3. Solve problem 22.4-5, page 552 in Edition 2, page 615 in Edition 3.
4. Solve problem 23.2-8, page 574 in Edition 2, page 637 in Edition 3.
5. Solve problem 23-4, page 577 in the 2nd Edition, page 641 in Edition 3.
6. Dijkstra's algorithm fails when there are negative weights on the edges.
   (a) Give a counter-example showing the failure.
   (b) Examine the proof of Dijkstra's algorithm. Identify the part of the proof where the reasoning is faulty when negative weights are present, and explain why it is faulty.
7. Provide examples of connected graphs for situations specified below. If an example cannot exist for the situation, provide reasons.
   (a) A graph in which a maximum cost edge is a part of every MST in the graph.
   (b) A graph in which a maximum cost edge is never a part of any MST.
   (c) A graph in which a least cost edge is not a part of any MST.
8. In 1976 the "Four Color Map Theorem" was proven with the assistance of a computer. This theorem states that every map can be colored using only four colors, in such a way that no region is colored using the same color as a neighbor region.

   Here you are asked to solve a simpler problem. You have to decide whether a given connected, undirected graph can be bicolored. That is, if one can assign colors (from a palette of two) to the nodes in such a way that no two adjacent nodes have the same color. To simplify the problem you can assume that the graph has no self-loops (i.e., no node will have an edge to itself).

   You must prove that your algorithm is correct. Also, analyze its time and space complexity.