Cpt S 515 Homework #2

No late homework!

1. In Lesson 3, we talked about the Tarjan algorithm (SCC algorithm). Now, you are required to find an efficient algorithm to solve the following problem. Let G be a directed graph where every node is labeled with a color. Many nodes can share the same color. Let v_1, v_2, v_3 be three distinct nodes of the graph (while the graph may have many other nodes besides the three). I want to know whether the following items are all true: there is a walk α from v_1 to v_2 and a walk β from v_1 to v_3 such that

- α is longer than β;
- α contains only red nodes (excluding the two end nodes);
- β contains only green nodes (excluding the two end nodes).

2. In Lesson 4, we learned network flow. In the problem, capacities on a graph are given constants (which are the algorithm's input, along with the graph itself). Now, suppose that we are interested in two edges e_1 and e_2 whose capacities c_1 and c_2 are not given but we only know these two variables are nonnegative and satisfying $c_1+c_2 < K$ where K is a given positive number (so the K is part of the algorithm's input). Under this setting, can you think of an effcicient algorithm to solve network flow problem? This is a difficult problem.

3. There are a lot of interesting problems concerning graph traversal — noticing that a program in an abstract form can be understool as a directed graph. Let G be a SCC, where v_0 is a designated initial node. In particular, each node in G is labeled with a color. I have the following property that I would like to know whether the graph satisfies:

> For each inifintely long path α starting from v_0, α passes a red node from which, there is an infinitely long path that passes a green node and after this green node, does not pass a yellow node.

Please design an algorithm to check whether G satisfies the property.

4. Path counting forms a class of graph problems. Let G be a DAG where v and v' be two designated nodes. Again, each node is labeled with a color.

(1). Design an algorithm to obtain the number of paths from v to v' in G.

(2). A good path is one where the number of green nodes is greater than the number of yellow nodes. Design an algorithm to obtain the number of good paths from v to v' in G.