CSCB63 Assignment 2

Due: 11.59pm, Saturday March 6, 2021 ⇐ Note New Deadline

1. (not for marks) Make sure you are capable of applying BFS, DFS, Prim's algorithm, Kruskal's algorithm and Dijkstra's algorithm to an appropriate type of graph (directed, not directed or weighted).

2. In class, we have (or will soon) seen Dijkstra's algorithm to find all the shortest paths from a single source in a weighted graph. In this question, you will construct a graph that has many shortest paths, in fact, your graph will have 3^n shortest paths between a source vertex s and a sink vertex t, where the number of vertices is a function of the form $cn + k$ for some $c, k \in \mathbb{N}$. More precisely, prove:

   > *For every natural number n, there is an undirected graph of $cn + k$ vertices such that for some pair of vertices s and t in the graph, there are 3^n shortest paths from s to t.*

   You select the constants k and c to make it work.

3. Determine whether the following claim is true and either prove the claim or it's negation.

   > *Given a graph G with n vertices such that for every $v \in V$, **deg**$(v) \geq \frac{n}{2}$ then G is one connected component.*

4. We can model the build process of an object as a directed graph. For example, suppose we are building a house. The walls can't be painted until the drywall is installed which cannot happen until after the studs are built. Suppose we model the process with a vertex representing the components of the object and a directed edge from a to b if a must be completed before b can be completed. Notice that if this directed graph has a cycle, then there is no way to construct the object.

   (a) Give an algorithm to determine whether a directed graph has a cycle. What should your complexity be?

   (b) Using DFS, construct an algorithm that either returns a valid ordering of the vertices to build the object or a cycle confirming no such ordering exists. Again, what should your complexity be?

5. Consider an undirected graph $G = (V, E)$ with non-distinct, non-negative edge weights. If the edge weights are not distinct, it is possible to have more than one MST. Suppose we have a spanning tree $T \subset E$ with the guarantee that for every $e \in T$, e belongs to some minimum-cost spanning tree in G. Can we conclude that T itself must be a minimum-cost spanning tree in G? Give a proof or a counter example with explanation.

6. The programming assignment for A2 and A3 is a combined assignments that will be posted on Markus. It will not be due until A3 is due as it requires a bit more thought and problem solving that A1. As with any assignment, make sure that the work is your own - do not copy for another student or a website.