NCM IST, Mathematics for Computer Science
Problems on graphs

19 June, 2018

1. (a) In IPL (Indian premier cricket league) 2018, at the end of May 18, 2018, each of the eight teams has completed 13 games each. Suppose there were totally 9 teams, is it possible to design fixtures so that each team has completed 13 games each at the end of some day?

   On May 20th, the last two matches of the playoff were played, and at the end of the day, each of the 8 teams has played against each other team twice. My son says that the matches played on 20th were the 53rd and 54th matches of the tournament? Is he correct? Explain.

   (b) Can you design a fixture so that each of the 8 teams has played different number of games at the end of some day? I.e. no two teams have completed the same number of matches. Recall that each team plays against every other team twice and at the beginning of day 1, each team has played 0 matches. (Hint: start with, say 4 teams first).

   (c) Can you design a fixture so that each of the 8 teams has played different number of games at the end of some day, when each team is to play against every other team exactly once?

2. A bridge in a connected undirected graph is an edge whose removal disconnects the graph.

   (a) Prove than an edge is a bridge if and only if there is no cycle containing that edge.

   (b) Give an $O(n+m)$ algorithm to find all bridges in a connected undirected graph using depth first search.

3. A directed graph G has an Euler tour if there is a directed cycle that visits every edge exactly once (though vertices can be repeated several times).

   (a) Show that a directed graph has an Euler tour if and only if for every vertex, its indegree and outdegree are the same.

   (b) Give an $O(m + n)$ algorithm to find an Euler tour in a directed graph where every vertex has indegree and outdegree the same.

4. A cut vertex x in an undirected graph G is a vertex such that $G \setminus x$ is disconnected.

   (a) Given an undirected graph on n vertices and m edges, how will you find all the cut vertices? How much time does your algorithm take?

   (b) Suppose that the graph has no cut vertex, and consider the DFS tree of the graph. Show that the root can not have more than one child in the tree.

5. Either prove or give a counterexample: if (u, v) is an edge in an undirected graph, and during DFS search $post(u) < post(v)$, then v is an ancestor of u in the DFS tree.

6. You are given a binary tree $T = (V, E)$ (in adjacency list format), along with a designated root vertex $r \in V$. A vertex u is called an anscestor of v in a rooted tree, if the path from r to v in T passes through u.

   We wish to preprocess the tree so that queries of the form "is u an anscestor of v?" can be answered in constant time. The preprocessing itself should take linear time. Suggest an algorithm for the same.

7. Give an efficient algorithm which takes as input a directed graph $G = (V, E)$ and determines whether or not there is a vertex $s \in V$ from which all other vertices are reachable.

8. (From Kleinberg and Tardos) Theres a natural intuition that two nodes that are far apart in a communication network—separated by many hops—have a more tenuous connection that two nodes that are close to each other. There are a number of algorithmic results which are based to some extent on different ways to make this notion precise. Heres one.

   Suppose that an n-node undirected graph G contains two nodes s, t such that the distance between two nodes s and t is strictly greater than $n/2$. Show that there must exist some node v, not equal to s or t such that deleting v from G destroys all $s - t$ paths. Give an $O(m + n)$-times algorithm to find such a v.

9. (From Kleinberg and Tardos) Suppose we are given an undirected graph G and u and v are two nodes in G. Give an algorithm to compute the number of shortest $u - v$ paths in G. The running time should be $O(n + m)$.

10. Suppose a CS curriculum consists of n courses, all of which are mandatory. The prerequisite graph G has a node for each course and an edge from course u to course v if and only if u is a prerequisite for v. Find an algorithm which works with the graph representation and computes the minimum number of semesters necessary to complete the curriculum (assume that a student can credit any number of courses in a semester). Your algorithm must run in linear time.

11. We are given an undirected graph G where every edge has a weight – however, the weight of any edge can be either 1 or 2. We are given a source s and we wish to compute the shortest path from s to every other vertex in G. Design a linear time algorithm for the problem.