CS 331: Algorithms and Complexity (Fall 2016)
Unique numbers: 51420, 51425, 51430, 51435

Assignment 2

Due on Wednesday, September 14, by 11.59pm

Problem 1

(15 points) The Department of Computer Science at the University of Texas at Austin (UTCS) has received several applications for teaching assistant positions from UTCS students. The responsibility of each applicant is to assist a faculty member in instructional responsibilities for a specific course. The applicants are provided with the list of available courses and asked to submit a list of their preferences. The corresponding committee at UTCS prepares a sorted list of the applicants (according to the applicants' background and experience) for each course and the total number of TAs needed for the course. Note that there are more students than the available TA spots. The committee wants to apply an algorithm to produce stable assignments such that each student is assigned to at most one course. The assignment of the TAs to courses is stable if none of the following situations arises.

1. If an applicant A and a course C are not matched, but A prefers C more than his assigned course, and C prefers A more than, at least, one of the applicants assigned to it.
2. If there is unmatched applicant A, and there is a course C with an empty spot or an applicant A' assigned to it such that C prefers A to A'.

Your task is to do the following:

i **(10 points)**Modify the Gale-Shapley algorithm to find a stable assignment for the TAs-Courses matching problem

Answer: While there is a course c who doesn't have all TA positions full and hasn't asked all students
Choose such a course c.
Let s be the highest-ranked student in c's preference list
to whom c has not yet proposed
If s is free then
(c, s) becomes a course-TA pair

```
Else s is currently a TA for another course c'
If s prefers c' to c then
c still has a free spot
Else s preferes c to c'
(c, s) becomes a course-TA pair
c' has a free spot
Endif
Endif
Endwhile
Return the set T of pairs.
```

ii **(5 points)** Prove that the modified algorithm produces stable TAs-Courses assignments.

Answer: Proof: Suppose for contradiction that applicant A and a course C are not matched, but A prefers C more than their current course, and C prefers A more than at least one of its TAs. However, since in the algorithm, s is always selected as the highest preference on C's list, we know C will ask A to be a TA before one of the current TAs. This is a contradiction because A preferring C would mean that A would switch to C.

Again for contradiction, suppose there is an unmatched applicant A and course C with empty spot or that prefers A to A'. If course C has an empty spot, they will continue asking students. If they get to A on their preference list, they will select A and be a match since A is free. Furthermore, because C goes down a preference list, the course could not select A' before A because A would be asked first.

Problem 2

(a) (5 points) Let T be a tree such that every node has either two or zero children, and the lengths of all the paths from the root to the leaves are equal. That is, for any two leaf nodes l_1, l_2, the distance from the root to l_1 is the same as the distance from the root to l_2. If T has height d, prove that the total number of nodes in T is $2^{d+1} - 1$. Hint: use proof by induction.

Answer: Proof: Let d = 0. Then $2^{d+1} - 1 = 1$. And since the distance from the root is 0, we know there can only be 1 node. When d = 1. We have $2^{d+1} - 1 = 3$. This is true because the single root node can only have 2 children at once. This proves the base case. For the inductive case, assume $2^{d+1} - 1$ is the number of nodes in T. We must show this continues to be the case for (d+1). If $2^{d+1} - 1$ is the number of nodes for height d, then since the distance from the root to a leaf is always equal, we know for height (d+1), every leaf node in the height d tree will have 2 children. The number of leaf nodes in the height

Problem 2 continued on next page...

d graph is $2^{d+1}-1-(2^d-1)$, so the number of nodes in the height (d+1) graph must be $2^{d+1}-1+2*(2^{d+1}-1-(2^d-1))$ (by adding the new nodes to the height d graph). This gives:

$$2^{d+1}-1+2^{d+2}-2^{d+1}$$

which simplifies to $2^{(d+1)+1}-1$. Thus, proving the theorem.

(b) (5 points) Given an undirected, connected graph $G=(V,E)$, an edge is a bridge if and only if removing it disconnects the graph (e.g., let $e \in E$ be a bridge such that $G=(V,E)$ is connected but $G'=(V,E-\{e\})$ is disconnected.). Show that if e is a bridge in G and T is a BFS tree of G, then e appears in T.

Answer: Proof: Suppose for contradiction that e does not appear in T. But this would mean T is missing an node of G because BFS requires following along edges and there is a disconnected node without e. Thus, e must appear in T to form the BFS of G.

Problem 3

(10 points) You are moving to a new city and need to rent an apartment that is close to your work. You are planning to choose the apartment complex that has largest number of shortest paths to your work. Assume that you are given a map for the city that shows the locations in the city. The distance between any two directly connected locations is one kilometer. Since you studied computer science at UT, you decide to formulate this as a graph problem. You turn the map into a graph, with nodes being the locations and edges being roads between them.

For a graph $G=(V,E)$, describe an algorithm that calculates the number of shortest paths between the apartment complex and your work. The time complexity of the algorithm should be $O(V+E)$. Prove the correctness of your algorithm.

Answer: At every step, choose the shortest path. If a step has equal paths, choose the path where the following node would have the shortest path.

Proof: because the shortest path is always being taken, it will be inductively equal to the optimal solution. The base case is the first shortest step. And the inductive case is that if you get to a node and take it's shortest path and then will take the shortest path of the following node you have furthered the shortness of the path.