COMP 251 - Fall 2017 - Assignment 3

Due: 11:59pm Nov 3rd

General rules: In solving these questions you may consult your book; You can discuss high level ideas with each other, but each student must find and write his/her own solution. You should upload the pdf file (either typed, or a clear scan) of your solution to mycourses.

1. (15 points) We are given a tree T (not necessarily binary). Design a greedy algorithm that finds the largest number of nodes in T such that no two of them are adjacent. (You do not need to optimize the running time of your algorithm, as long as your running time is polynomial).

2. (15 points) We have a single processor, and we are given a sequence of jobs with processing times $t_1, \ldots, t_n$. Each job has to be processed in a continious time interval, and these intervals cannot overlap. We want to minimize the average finishing times of these jobs. Prove that the optimal strategy is to process them in increasing order according to their processing times.

3. (15 points) Consider the same setting as the previous questionm but now we have 3 processors instead of 1. Each job has to be processed in a continious time interval on one of the processors, and the intervals on each one of the processors cannot overlap. Again we want to minimize the average finishing times. Is the greedy algorithm still optimal? In the greedy algorithm, every time a processor becomes available, among the unprocessed jobs we assign the one that has the smallest processing time to that processor. Either prove that this algorithm is optimal or give a counter-example showing that it does not always minimize the average of the finishing times.

4. (15 points) We are given an undirected graph $G = (V, E)$ as an input, and our goal is to find the minimum of vertices whose deletion will remove all the edges in G (when we delete a vertex, the edges incident to it will be removed). Is the following greedy algorithm optimal?

   - While there are still edges left:
   - pick a vertex with current maximum degree and remove it.
   - endwhile

5. (20 points) We are given the coordinates of n points on the plane: $(x_1, y_1), \ldots, (x_n, y_n)$. Give a polynomial time algorithm that finds the smallest circle that is centered at one of these points and contains at least half of the points.

6. (20 points) Let $G = (V, E)$ be an undirected weighted graph, and let $F \subset E$ be a collection of edges in G that contain no cycles (i.e. they form a forest). Design an efficient algorithm to find a spanning tree in G that contains all the edges in F and has minimum cost among all such spanning trees.