CSL 356 Analysis and Design of Algorithms

Minor I, Sem I 2013-14, Max 40, Time 1 hr

Name ______________________ **Entry No.** ______________________**Group**

Note (i) Write your answers in the blank sheets including the back of the sheet. Do all roughwork in separately provided paper.

(ii) Write your answers neatly and precisely. You won't get a second chance to explain what you have written.

(iii) Unless stated otherwise, every algorithm must be accompanied by proof of correctness and a formal analysis of running time and space bound. Feel free to quote any result from the lectures without proof - for any anything new, you must prove it first.

1. Write answers to the following problems (no proof or algorithm required) **(3×4)**

   (a) What is the time complexity (tight bound) to find a number *not* appearing in a set of n elements

   $O(n)$ by finding a number less than min or greater than max. Can't assume that numbers are integers in the range $[1, n]$ or that they are sorted.

   (b) What is the time complexity (tight bound) to merge d sorted sequences (not necessarily equal lengths) having a total of n elements (as a function of n, d) using only comparisons.

   $O(n \log d)$ by using a heap of size d and using n extract min and n insert operations.

   (c) What is the space complexity (tight bound) for the median-of-median algorithm on a set of n elements

   $O(n)$ by arguing tha the space is the space required at the bottom-most level of recursion and the size of the problem decreases by a constant factor at each level.

   (d) In a given set S of n real numbers, the *mode* is an element that has the maximum frequency of occurence - break ties arbirarily. What is the time complexity of finding a mode with (unknown) frequency M (in terms of n, M).

   $O(n \log(n/M))$ which is not the same as $O(n \log(n-M))$ by using repeated pivoting and arguing that the algorithm will pick up the mode by the level $\log_M n$.

2. To find out the strength of a newly manufactured helmet, we subject it to a specific force f to see if it shatters. If not, we subject it to $f + \Delta$ for some increment Δ and so on. If it shatters when subjected to f, then we try $f - \Delta$ on a *new* helmet. We want to measure the strength within some accuracy, say $[x, x+1]$ for some integer x. We have a known upperbound of F. If we had only one helmet for testing then (tick the right option). **(3)**

   (a) It can't be done

   (b) Can be done in $\log F$ trials always

   (c) Greater than $\log F$ but at most $F/2$ trials.

   (d) Can be done in $\leq F - 1$ trials. **Ans**

If we had two helmets for testing then it can be done in ________________________________ trials (tight bound). **(5)** Answer $O(\sqrt{F})$. Can be done using two applications of searching among $\sqrt{F}$ intervals - in the top level these are separated by $\sqrt{F}$ to pick the correct $i\sqrt{F}, (i+1)\sqrt{F}$ interval. Then within this interval, we refine the search using the other helmet.

If you are interested to argue taht one can't do better, first argue taht using one helmet you will require at least F trials in the worst case. For the two helmet case, you can argue that the interval that you need to search using the second helmet cannot be larger than $\sqrt{F}$ if you must have a $o(\sqrt{F})$ algorithm.

3. Mrs. White is planning to throw a big party and call as many friends as she can accommodate. She has a list of n friends, and she would like to ensure that each guest knows at least k other guests ($k < n$). Moreover there is a congeniality factor $C_{i,j} > 0$ between guests i, j who know each other - the higher the better. Mrs. White wants to invite a set of F friends, such that $\sum_{i,j \in F} C_{i,j}$ is maximized. **(3+3+14)**

   (a) Model this is a problem using graphs. What are the vertices, edges and weights ?

   The vertices are the n friends, we have an edge (i, j) if i knows j (either way) and the weight of an edge is the $C_{i,j}$. The objective function is to maximize the number of vertices such that every chosen vertex has degree $\geq k$ in the induced subgraph (among the other vertices chosen). This will also maximize $\sum C_{i,j}$ as these are non-negative, i.e., there is no need to state this as an objective function for this problem. We can't choose an edge without choosing the end-points in the solution set.

   Some students thought F is the maximum cardinality and others forgot to mention the objective function

   (b) Does this problem fall under the basic greedy framework ? Why ?

   This is not a subset system, i.e., the subset of a feasible (degree k subgraph) may not be feasible. Hence the framework of greedy doesn't apply.

   Trying to prove that exchange property doesn't hold is irrelevant for this part since it was not about the optimality of greedy but if we can at all invoke the framework

   (c) Can basic greedy solve this optimally ? If Yes, prove it - If not, suggest an efficient algorithm to solve it optimally.

   Since the framework doesn't apply greedy cannot be used. An alternate algorithm will be to eliminate vertices with degrees less than k and subsequently more vertices whose degrees fall below k because of earlier deletions. We stop when all surviving vertices have degrees $\geq k$ and output the result. **Note that the final answer may have several components, for instance union of k-cliques etc**

   Correctness: Any vertex that was deleted cannot be a part of the solution (you can do an inductive argument starting from the first vertex that was deleted). So finally we have the best solution - as noted in part (i) the sum of weights is redundant.

   Running Time: We look upon the problem as a graph $G = (V, E)$ where V is the set of n friends and E denotes the pairs who know eachother. Any reasonable data structure will allow us to implement the algorithm in time proportional to the number of edges plus vertices, $O(|V| + |E|)$. We can initially compute the degree of each vertex and the deletions can be done in time proportional to the degree of a vertex (for every edge that is deleted, we decrease the degree of the other end-point and it happens exactly once for every edge).

   Common mistakes were to show that exchange property holds
   Or that only connected sets can be output - so choose the best
   Or that all maximal subsets have same cardinality