CS 5200 Final - Practice F23 Section 102, 103

Instructor: Avah Banerjee
Duration: 120 mins

Problem 1 ($5 \times 4 = 20$ pts): Determine whether the following statements are **true** or **false**. (no need to justify your answer)

1. We an always sort 6 numbers using at most 9 comparisons.

2. The number comparisons for the median of median algorithm is still $O(n)$ if we allow the blocks to be of size 4.

3. If we assume that the sorted ordering is chosen uniformly at randomly (from all $n!$ possibly orderings) then the expected complexity of deterministic `quick-sort` is $O(n \log n)$.

4. A DAG has no strongly connected components.

5. Heaviest edge of any cycle of a graph must not be part of any minimum spanning forest of the graph.

Problem 2 (20 pts) Imagine you have a collection of n books and n uniquely-sized bookshelves in a library. Each book is designed to fit perfectly on one specific bookshelf, and each bookshelf is meant to hold one specific book. However, the books and shelves are currently unsorted and not matched with their corresponding pairs. You can test whether a book fits a shelf by attempting to place the book on the shelf. From this, you can determine if the book is too large, too small, or the perfect fit for the shelf. However, you cannot directly compare two books or two shelves to each other.

Using these rules, describe an efficient randomized method to match each book with its corresponding bookshelf which only makes $O(n \log n)$ checks. Prove your procedure is correct.

Problem 3 (20 pts) Given an connected edge weighted graph $G = (V, E, w)$ and a minimum spanning tree of T of G , Develop a deterministic algorithm that, in $O(n)$ time, updates the minimum spanning tree T to reflect the reduced weight of the edge in G, if such an update is necessary. The updated tree should remain a valid minimum spanning tree for the modified graph.

Problem 4 (20 pts) Find the maximum weight independent set (using a dynamic program) in a cycle graph $C_n = ([n-1], \{(i, i+1) \mod n \mid i \in [n]\})$ with vertex weights which are positive real numbers. Determine the DP DAG, the DP recurrence and the running time.

Problem 5 (20 pts) Give a deterministic $O(n^2)$ time algorithm to detect a square (a length 4 cycles) in an undirected graph G with n vertices. [Hint: in any square there are two distinct paths between pairs of diagonal vertices. As an example, for the square $(uvxy)$, we can go from u to x using either v or y.]