1. (a) (10 pts) Prove by induction that the solution of the following recurrence is $T(n) \leq An$, $n \geq 1$, and find the constant $A$.

$$T(n) \leq \begin{cases} \frac{T(n/5) + T(3n/4) + n}{n^2/2}, & n > 20 \\ n^2/2, & n \leq 20 \end{cases}$$

(b) (10 pts) Find the exact solution of the following recurrence, assuming $n$ is a power of 2.

$$T(n) = \begin{cases} 8T(n/2) + n^2, & n > 1 \\ 1, & n = 1. \end{cases}$$

(c) (This part has no credit, and answering it is optional!) On NJ Turnpike South (near exit 16) a mysterious billboard says: "The Algorithm is From Jersey." Any idea what it means?
2. Given an array of \( n \) elements with arbitrary random values. We want to obtain the smallest \( \log n \) elements in sorted order. Outline how each of the following methods may be adapted to solve this problem (if appropriate), and provide a very brief explanation. If the method is not applicable, state “NA” and provide a one-sentence explanation. Analyze the worst-case time complexity for each of the appropriate cases. (State the time complexity for each major step in your outline.)

(a) Radix-Sort.

(b) An adaptation of Bubble-Sort.

(c) Use a HEAP.

(d) Use the linear-time SELECTION algorithm (which used the median-of-medians for pivot).

(e) Which of the above methods is best-suited and most-efficient for this problem? Explain.
3. Given an undirected graph with \( n \) vertices and \( e \) edges, represented by its adjacency-matrix \( A \). We want an efficient algorithm to find the Connected-Components (CC) of the graph in the form of a Boolean matrix \( P \) such that \( P[i, j] = 1 \) if and only if vertices \( i \) and \( j \) are in the same components.

Outline an algorithm for this problem. Be very specific about how matrix \( P \) is computed. Analyze the worst-case time complexity.
4. Given the following weighted directed graph. Recall Floyd’s algorithm for all-pairs-shortest-paths (least-cost-paths). The algorithm produces two matrices $A$ (cost) and $P$ (path), such that at the end, $A[i, j]$ is the cost of the shortest $(i, j)$ path, and $P[i, j]$ is the first vertex immediately after vertex $i$ on the shortest-path.

(a) Show the working of Floyd’s algorithm for the following graph, by showing the resulting matrix $A$ and $P$ after each iteration. (You may show the entries $A[i, j]/P[i, j]$ as a pair, rather than two physically separate matrices.)

(b) What is the time complexity of Floyd’s algorithm for a directed weighted graph? (Let $n$ denote the number of vertices, and $e$ the number of edges.) Does the time complexity change if the graph is undirected and initially represented by its adjacency-lists? Explain.
5. Consider the following connected, undirected weighted graph, represented by its adjacency lists.

(a) Find a minimum-cost-spanning tree (MST) using Kruskal’s algorithm. Show the result after each iteration.

(b) Find a minimum-cost-spanning tree (MST) using Prim’s algorithm. Show the result after each iteration.

(c) What is the time complexity of Prim’s algorithm assuming the graph is represented by its adjacency lists. Use the space on the back of this page to provide a short analysis.