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

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

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

$$T(n) = \begin{cases} 
4T(n/2) + n, & n > 1 \\
1, & n = 1.
\end{cases}$$
2. Given an array of \( n \) elements with arbitrary values. Describe how to find the smallest 100 elements in sorted order, using each of the following methods. For each case, outline the algorithm in terms of the major steps involved, and analyze the worst-case time complexity.

(a) Use bubble sort.

(b) Use a heap.

(c) Use the selection algorithm with the “median-of-medians” pivot, as described in class.

(d) Which of the above methods would you expect to run the fastest when \( n = 2^{20} \)? Explain.
Given an undirected weighted graph, the Traveling-Salesperson-Problem (TSP) is to find a least-cost tour of the graph. (A tour is a simple cycle which goes through all vertices exactly once.) The TSP problem is NP-complete, but it can be solved easily for small graphs.

(a) Find a least-cost tour, and its cost, for the following graph by inspection. You can start with vertex 1, though the starting vertex does not matter.

(b) Now consider the following GREEDY algorithm for finding a good tour:
   Start with vertex 1 and go to the “NEAREST” vertex. Continue to build a tour by going from the last visited vertex to its nearest unvisited vertex, until you get back to vertex 1.
   Find a tour found by this greedy algorithm, and its cost. Is this greedy solution optimal?
4. (a) Consider the following directed, weighted graph. Show the working of Floyd’s all-pairs-shortest-path algorithm on this graph. Show the adjacency matrix at the beginning and the resulting matrix at the end of each iteration.

(b) What is the time complexity of Floyd’s all-pairs shortest-path algorithm?

(c) Explain how Dijkstra’s single-source-shortest-path algorithm may be used to find all-pairs-shortest-path. How does the resulting time complexity compare with Floyd’s?
5. Consider the following connected, undirected weighted graph, represented by its adjacency lists.

(a) Find a minimum-cost-spanning tree (MST) using PRIM’s algorithm, starting with vertex 1 as the root. After each iteration, show the resulting tree and the corresponding HEAP. Draw the heap as a binary tree, with entries \((i, j, C_{i,j})\). Show the final MST and its cost.

(b) What is the time complexity of PRIM’s algorithm, when the graph is represented by its adjacency lists? Explain.