This homework contains four problems. **Read the instructions for submitting homework on the course webpage.**

**Collaboration Policy:** For this homework, Problems 2–4 can be worked in groups of up to three students.

**Problem 1 should be answered in Compass as part of the assessment HW6-Online and should be done individually.**

1. **HW6-Online.** (20 pts.)

2. **MST Stuff.** (30 pts.)
   You are given an undirected weighted graph $G = (V, E)$ with $n$ vertices and $m$ edges. Assume the weights of the edges are all distinct.
   
   (A) (10 pts.) Let $E' \subseteq E$ be the set of those edges of $G$ with weight less than or equal to the median of edge weights. Give an $O(n + m)$ time algorithm to determine whether the MST of $G$ uses only edges in $E'$ or not.
   
   (B) (20 pts.) Give an $O(n + m)$ time algorithm that outputs the heaviest edge in the (unique) MST of $G$.

3. **Palindrome II.** (30 pts.)
   A sequence is **palindromic** if it is the same whether read left to right or right to left. An example is $m, a, l, a, y, a, l, a, m$ (**Malyalam** is a Southern Indian language). Given a sequence $a_1, a_2, \ldots, a_n$ describe an algorithm to compute a **shortest** palindromic **supersequence** of the given sequence. For example, the sequence below

   ![blab](image)

   has the palindrome `blab` as the shortest supersequence.

   What is the running time of your algorithm?

4. **Serving time.** (20 pts.)
   A server has $n$ customers waiting to be served. The service time required by each customer is known in advance: it is $t_i$ minutes for customer $i$. So if, for example, the customers are served in order of increasing $i$, then the $i$th customer has to wait $\sum_{j=1}^{i} t_j$ minutes. We wish to minimize the total waiting time

   $$T = \sum_{i=1}^{n} (\text{time spent waiting by customer } i).$$
Give an efficient algorithm for computing the optimal order in which to process the customers. How fast is your algorithm?