ASSIGNMENT - 4
Analysis and Design of Algorithms - CSE222
Each question carries 10 points.

1. Given a database of n documents, each of Yahoo (Y), Google (G) and Bing (B) produces an ordered list depending on their relevance to the searched term. You suspect that these search engines promote some documents unethically (spam). However, there's no way to know about it. To address this, you planned to design a meta-search engine that combines the wisdom of all three search engines, while punishing the spams. Mathematically this can be formulated as below.

   $$R_{meta} = min \sum_{\forall i \in \{G, Y, B\}} d(R_{meta}, R_i)$$

   Where R_{meta} is your meta-ranking, whereas d is a distance function measuring overall disagreement between two rankings. Distance d between two rankings can be measured by Kindall's Tau Distance (or Bubble Sort Distance). Unfortunately, the problem is NP-hard (http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.28.8702&rep=rep1&type=pdf). Implement a greedy algorithm for constructing R_{meta}. For small n, show its performance with respect to the exact solution.

   Can you provide any performance guarantee for your proposed algorithm? (Bonus Points - 10)

2. Given 1 ≤ C ≤ 5 chambers which can store 0, 1, or 2 specimens, 1 ≤ S ≤ 2C specimens, and M: a list of the mass of the S specimens, determine in which chamber we should store each specimen in order to minimize IMBALANCE. Which strategy did you use?

   A=($\sum_{j=1}^{S} M_j$)/C, i.e., A is the average of all mass over C chambers.

   IMBALANCE = $\sum_{i=1}^{C} |X_i - A|$, i.e., sum of differences between the mass in each chamber w.r.t A. where X_i is the total mass of specimens in chamber *i*.

3. **Answer the following questions-**
   3.1 Given two arrays A, B of n numbers, each sorted in increasing order, give an algorithm to compute the median of A ∪ B in O(log n) time.
   3.2 Can we do the same with 3 sorted arrays A, B, C? What is the running time?

4. A $m \times n$ matrix A is called a Monge matrix if for all $1 \le i \le k \le m$ and $1 \le j \le l \le n$, we have $A_{ij} + A_{kl} \le A_{il} + A_{kj}$
   4.1 Prove that a matrix is Monge if and only if for all $i = 1, \ldots, m-1$ and $j = 1, \ldots, n-1$ we have: $A_{ij} + A_{i+1,j+1} \le A_{i,j+1} + A_{i+1,j}$.

   4.2 Let $f(i)$ be the leftmost minimum element in row i. Show that $f(1) \le f(2) \le \ldots \le f(m)$ for any Monge array.

   4.3 In order to compute the leftmost minimum element, we can do the following:

- Create an array A' consisting of the even rows.
- Recursively determine the left-most minimum in each row of A'.
- Then, use this information to compute the leftmost minimum in each odd row.

  Show that if we knew the leftmost minimum element in each even row, then the leftmost minimum in each odd row can be computed in $O(m + n)$ time.

  4.4 Write a recurrence for the algorithm above. Show that the running time is O(m + n log m).

5. The randomized quick sort algorithm worked because of the following reason: If we choose an element at random, with probability at least 1/2, the element is chosen lies between the 1st and 3rd quartile.

   5.1 Prove this statement.

   5.2 Use this observation to obtain a randomized algorithm to compute the median, that runs in O(n) time in expectation.

6. After graduation, you are hired at a fancy new e-commerce start-up with a rather clever idea and a billion dollar valuation. Unlike other e-commerce start-ups that try to stock all things under the sun, your start-up focuses on selling just one item, namely diapers that a customer can order via a snazzy app. The diapers are shipped from one of two warehouses the company owns: Warehouse X in Okhla and Warehouse Y in Gurgaon. Warehouse X has n_X diapers, and Warehouse Y has n_Y diapers in stock.

There are n customers, where customer i has a demand of d_i diapers. It costs v_i rupees to ship a diaper from Warehouse X to customer i, and w_i rupees to ship a diaper from Warehouse Y to customer i.

Let us assume that we have sufficiently many diapers in the two warehouses to meet all demand, i.e., $\sum_{i=1}^{n} d_i \le n_X + n_Y$. We want to determine, for each customer i, the values x_i, y_i, namely the number of diapers from warehouses X, Y respectively, such that:

- The demand of each customer is satisfied, i.e., for each customer i, $x_i + y_i = d_i$
- The choice of x_i , y_i , $i = 1, . . . , n$ ensures that the total number of diapers shipped from either warehouse is at most the number available, i.e., $\sum_{i=1}^{n} x_i \le n_X$ and $\sum_{i=1}^{n} y_i \le n_Y$.

The objective is to find values x_i, y_i satisfying the two conditions above, such that the total shipping cost is minimized, i.e., min $\sum_{i=1}^{n} (x_i v_i + y_i w_i)$. Devise a dynamic programming algorithm that returns a minimum cost solution, and saves your company lots of money.