Greedy

Assignment Questions

PW SKILLS

Q1. Minimum Cost of ropes
There are given N ropes of different lengths, we need to connect these ropes into one rope. The cost to connect two ropes is equal to sum of their lengths. The task is to connect the ropes with minimum cost. Given N size array arr[] contains the lengths of the ropes.

Constraints:
1 ≤ N ≤ 200000
1 ≤ arr[i] ≤ 106

Input:
n = 5
arr[] = {4, 2, 7, 6, 9}
Output:
62
Explanation:
1. First, connect ropes 4 and 2, which makes the array {6,7,6,9}. Cost of this operation 4+2 = 6.
2. Next, add ropes 6 and 6, which results in {12,7,9}. Cost of this operation 6+6 = 12.
3. Then, add 7 and 9, which makes the array {12,16}. Cost of this operation 7+9 = 16.
4. And finally, add these two which gives {28}.
5. Hence, the total cost is 6 + 12 + 16 + 28 = 62.

Q2. Minimum Number of Platforms Required for a Railway/Bus Station
Given the arrival and departure times of all trains that reach a railway station, the task is to find the minimum number of platforms required for the railway station so that no train waits. We are given two arrays that represent the arrival and departure times of trains that stop.

Note: Consider that all the trains arrive on the same day and leave on the same day. Arrival and departure time can never be the same for a train but we can have arrival time of one train equal to the departure time of the other. At any given instance of time, the same platform can not be used for both departure of a train and the arrival of another train. In such cases, we need different platforms.

Constraints:
1 ≤ n ≤ 50000
0000 ≤ A[i] ≤ D[i] ≤ 2359

Input:
arr[] = {9:00, 9:40, 9:50, 11:00, 15:00, 18:00}
dep[] = {9:10, 12:00, 11:20, 11:30, 19:00, 20:00}
Output: 3
Explanation: There are at-most three trains at a time (time between 9:40 to 12:00)

Q3. Minimum Fibonacci terms with sum equal to K
Given a number k, find the required minimum number of Fibonacci terms whose sum is equal to k.
Constraints:
1 <= k <= 109

Input : k = 17
Output : 3
Explanation: terms are: 13 + 3 + 1 = 17

Input: k = 4

Output: 2
Explanation: Many alternatives are there:
2+2, 1+3, 1+1+2, 1+1+1+1
The minimum number of terms is 2 i.e. in 2+2 and 1+3. So, you can choose any of the two.

Input: k = 3
Output: 1
Explanation: It is possible as:
3, 1+2, 1+1+1
Here, 3 itself is a term in the Fibonacci sequence, hence minimum no. of terms = 1 i.e. 3 itself.

Q4. Divide 1 to n into two groups with minimum sum difference
Given a positive integer n such that n > 2. Divide numbers from 1 to n in two groups such that absolute difference of sum of each group is minimum. Print any two groups with their size in first line and in next line print elements of that group.

Input: 5
Output: 2
5 2
3
4 3 1
Explanation: Here sum of group 1 is 7 and sum of group 2 is 8.
Their absolute difference is 1 which is minimum.
We can have multiple correct answers. (1, 2, 5) and (3, 4) is another such group.

Input: 6
Output: 2
6 4
4
5 3 2 1
Explanation: There can be multiple correct answers. Like [6,3,2] and [5,4,1] is also correct.

Q5. Divide cuboid into cubes such that sum of volumes is maximum
Given the length, breadth, height of a cuboid. The task is to divide the given cuboid in minimum number of cubes such that size of all cubes is same and sum of volumes of cubes is maximum. This output should be the side of cube and number of cubes.

Input : l = 2, b = 4, h = 6
Output : 2 6
A cuboid of length 2, breadth 4 and height 6 can be divided into 6 cube of side equal to 2.
Volume of cubes = 6*(2*2*2) = 6*8 = 48.
Volume of cuboid = 2*4*6 = 48.

Input : 1 2 3
Output : 1 6

Q6. Find minimum number of currency notes and values that sum to given amount
Given an amount, find the minimum number of notes of different denominations that sum up to the given amount.
We may assume that we have infinite supply of notes of values {2000, 500, 200, 100, 50, 20, 10, 5, 1}

Input : 800
Output :

```
Currency   Count
    500    :   1
    200    :   1
    100    :   1
```

Input : 2456
Output :

```
Currency   Count
    2000   :    1
    200    :   2
    50     :   1
    5      :   1
    1      :   1
```

Q7. Buy Maximum Stocks if i stocks can be bought on ith day

In a stock market, there is a product with its infinite stocks. The stock prices are given for N days, where arr[i] denotes the price of the stock on the ith day. There is a rule that a customer can buy at most i stock on the ith day. If the customer has k amount of money initially, find out the maximum number of stocks a customer can buy.

Input: price[] = { 10, 7, 19 },
k = 45
Output: 4
Explanation:
A customer purchases 1 stock on day 1 for 10 rs, 2 stocks on day 2 for 7 rs each and 1 stock on day 3 for 19 rs.Therefore total of 10, 7 * 2 = 14 and 19 respectively. Hence, total amount is 10 + 14 + 19 = 43 and number of stocks purchased is 4.

Input: price[] = { 7, 10, 4 },
k = 100
Output: 6