

28 May -

Problem 1: Cyber Heist - Maximum loot in Minimum Time.

g/p : N=4

vaults =  $\{[4, 20], [1, 10], [2, 40], [2, 30]\}$   
output: 90

→ Approaches : Sort the value by loot in descending order

- Schedule each job using greedy.  
latest available minute  $\leq$  its deadline
- Track available time slot using array.

pseudo code - { Arrays.sort(vaults, (a, b) →  
 $b[1] - a[1]$ );

```
int maxD = 0;  
for (int [] vault : vaults) {  
    maxDeadline = Math.max(maxD, vault[0]);  
    boolean [] slots = new boolean [maxD + 1];  
    int totalLoot = 0;  
    for (int [] vault : vaults) {  
        int deadline = vault[0];  
        int loot = vault[1];  
        for (int t = deadline; t >= 1; t--) {  
            if (!slot[t]) {  
                slot[t] = true;  
                totalLoot += loot;  
                break;  
            }  
        }  
    }  
    return totalLoot;
```

Sug-Run :-  $N = 4$   
Vaults =  $(4, 20), (1, 10), (2, 40), (2, 30)$

$\Rightarrow$  Sort :  $(2, 40) (2, 30) (4, 20) (1, 10)$

$\cdot (2, 40) \rightarrow$  Assign to time 2

$\cdot (2, 30) \rightarrow$  time 2 is occupied so assign to time 1

$\cdot (4, 20) \rightarrow$  time 4

$\cdot (1, 10) \rightarrow$  can't be assigned, 1 is already occupied

$$\text{Total Loot} = 40 + 30 + 20 + 10 = 90$$

Problem 2: Festival Gold Coins - Reward the Most Deserving.

I/P:  $N = 5$        $C = 10$

coins =  $[2, 3, 5, 7, 1]$

O/P: 4

Approach:- we will sort in ascending order  
• assign to the performer  
• count no. of coins

pseudo-code:

```
Array .sort (coins);
int total=0, count=0;
for (int c: coins) {
    if (total+c <= C) {
        total+=c;
        count++;
    } else break;
}
```

return count;

Sug-Ran's

sort: [1, 2, 3, 5, 7]

• coin 1 =

total = 1 (count = 1)

• coin 2 =

total = 3 (count = 2)

• coin 3 =

total = 6 (count = 3)

• coin 5 =

total = 11  $\geq 10$  (not possible)

so, Max coins can be distributed = 3

$$\begin{aligned} \text{total} + c &\leq C \\ 0 + 1 &\leq 10 \end{aligned}$$