QNO1

dues psets = N
repire day → d_i
cost penalty → c_i

Penalty = $i \cdot c_i$

Problemsets

→ 3 days has 12 points/day penalty
→ 4 days has penalty 20 points/day
→ 2 days has penalty 4 points/day

Best order

2nd, 1st, 3rd

if we sort by increasing d_i/c_i
$d_i = [3, 4, 2]$ $c_i = [12, 20, 4]$

$d_i/c_i = 3/12, 4/20, 2/4$

$\Rightarrow 0.25, 0.2, 0.5$

The complexity taken by this will be $O(N \log N)$

But in case of unsorted problem set

we can more improve this by simply swapping

$d_i/c_i > d_j/c_j = c_j d_i + c_i d_i + c_j d_j >$
$c_i d_j + c_i d_i + c_j d_j$

$\Rightarrow c_j(d_i + d_j) + c_i d_i > c_i(d_j + d_i) + c_j d_j$

Q No 2

Here we are trying to calculate $SL_{i,j}$.
~~He~~ It can be obvious for us that maximum can be at (i,j) index.
if it is not case then it must lie either in the rectangle form $(1,1)$ to $(i,j-1)$ or $(1,1)$ to $(i-1,j)$. so these are actually our overlapping cases which we need for dynamic programming.

so it will be

$$SL_{ij} = \max\{S_{i,j}, SL_{i-1,j}, SL_{i,j-1}\}$$

For all valid values of i, and j it can be

$$SL_{0,j} = SL_{i,0}.$$

For each computing value in Dynamic Programming it will take $O(1)$ time. As there (xy) states so this will be $O(xy)$.

Q No 3

Suppose we have 7 books
{20, 20, 20, 50, 20, 20, 50} and shelves have length 100.
shelf 1 = {20, 20, 20} - 40}
shelf 2 = {50, 20, 20} - 10}

shelf 3 = {501-50} using 3 shelves with 100 units of free space.

<u>By contradiction</u>

Suppose a given arrangement P uses n shelves, but the greedy algorithm would use more than n shelves.

For simplicity, there is no empty shelf space b/w books. On every shelf the books are pushed all the way to the left, and any free space occurs to their right.

There must have occured a time when a book was placed on the next shelf, because the current arrangement P is not greedy.

In the current arrangement there must be a book i at the beginning of a shelf that can be moved to the end of the previous shelf. This will reduce the amount of space on the earlier shelf and will increase the amount of free space on the later shelf. If only the move book was the ? book on its shelf. then we have just free up a shelf. Any other move will ignore that shelf. After repeating this move many time we have greedy arrangement.

Because of each move could only decrease the amount of shelves moved and never can increase, the greedy solution contains the same number of shelves or less than the original "optimal" arrangement.

So therefore the greedy arrangement is optimal.

(b)

for i: = n-1
  current shelf : = C[i];
  Append start to left end list

for j:= i+1 to n
  currentshelf : = currentshelf + C[j];
  if currentshelf ≤ L and ~~min~~ [H[i] + cost[j+1] ≤ cost[i]
  then
    cost[i] = min { H[i] + cost [j+1] .

It will take $O(n^2)$.

Q.no 4

(a)

A wheel graph of n vertices contains a cycle graph of order n-1 and all the vertices of the cycle are connected to single vertex.

number of edges in a wheel graph =
$W_n = 2n-2$

(b)

Given a directed graph where every edge has weight as either 1 or 2. we can find the shortest path from a given source vertex 's' to a given destination vertex 't'. Expected time Complexity is $O(V+E)$. But By using dijistra Algorithm it will take $O(V+E\log V)$.

QNo 5

By contradiction

Let us suppose that (a_i, a_{i+1}) belongs to minimum spanning Tree T. if we remove (a_i, a_{i+1}) from T, then it will divide the connected components into two ~~connected~~ components (ie A and B)

Some of the nodes of the cycle will be in A and some will be B.

Here For any cycle there should be atleast two edges that must cross this cut. so there will be other edge on the cycle. named as $(a_j + a_{j+1})$

Now when we add these edge it will connect A and B again and new spanning tree T_1 will be made.

so here will be contradiction as weight of (a_j, a_{j+1}) is less than (a_i, a_{i+1}) or weight of T_1 is less than T so it cannot be a minimum spanning Tree.

$\longleftrightarrow$