Job Sequencing with Deadlines

Problem Statement :-

We have set of jobs and along with each job we have profit & deadline given.

Assumptions:-

1. we are working on uniprocessor system. i.e. we have a single processor for scheduling all the jobs

2. no-preemption
   If we have started with one job then it will complete its execution then only new job can be scheduling. In between the execution of job can not be paused.

3. Every job will take 1 unit of time for execution.

Now based of these assumption we need to schedule the jobs to get maximum profit.

To get optimal solution we need to draw gantt-chart.
To draw gantt chart first we need to identify the maximum deadline

Example

	J_1	J_2	J_3	J_4
Profit	50	15	10	25
Deadline	2	1	2	1

Approach 1 :-

Step1 :- Draw grantt chart

To draw grantt chart first we need to identify the ~~deadline of~~ maximum deadline.

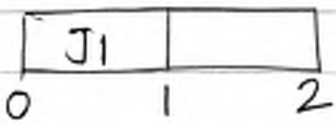
The maximum deadline ~~for~~ given example is 2

Job J1 has deadline 2 and as per assumption it requires 1 unit time. So deadline 2 ~~has~~ means J1 can be scheduled in 1st half or 2nd half.

Where J2 has deadline of 1 ~~let's say it month~~ then it should be scheduled in first half only.

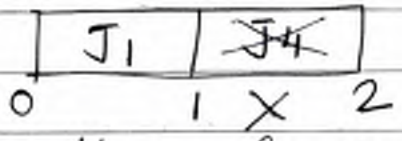
Step2 :- Now we need to schedule jobs to get maximum profit

So highest profit job is J1 with profit 50. & deadline 2



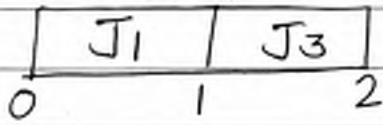
Profit = 50

Now next highest profit job i is J4 with deadline 1 & profit 25



As deadline for J4 is 1 so it can not be schedwed in second half.

So the only option available with us is J3 with deadline 2 & profit 10



Profit = 50 + 10
= 60

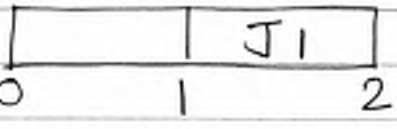
But this is not the optimum solution.

Approach 2 :-

Step 1 :- Draw gantt chart

Select the job with maximum profit.
So the highest profit job J1 with profit 50 & deadline is 2.

Now in gantt chart we need to schedule J1 to the maximum value or the deadline value

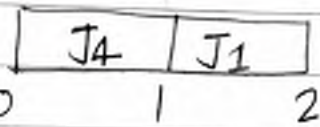


Profit = 50

Step 2:-

Now the next highest profit job is J4 with profit 25 & deadline 1

As deadline is 1 the maximum level to which the job can be schedwled is 1



Profit = 50+25
= 75

This the optimal solution.

So we need to schedwle the job with reference to the deadline. for last value ~~w~~ corresponding to deadline value.

Algorithm :-

1) Arrange all the jobs in decreasing order profit

2) for each job (mi)
do
linear search for finding particular slot in array of size(n)

where n= maximum deadline
m = total jobs.

Time Complexity

1) for sorting the elements in decreasing order we require $n \log n$ time.

2) To apply linear search we have

n = as array size
m = no of jobs

So linear search is performed $n \times m$ times

$O(m \times n)$
$O(n \times n)$
$O(n^2)$

3) out of $O(n \cdot \log n)$ & $O(n^2)$ the largest term is $O(n^2)$

So the time complexity of job scheduling with deadline using greedy algorithm is $O(n^2)$.