Greedy.

→ feasible : it has to satisfy the problem constraints.

→ locally optimal : it has to be the best local choice among all feasible choices available.

→ irrecoverable : once made it cannot be changed on subsequent steps of the algorithm.

eg : Chessboard.

Ex : Prim's algorithm, kruskal's algorithm.

→ Dijkstra's algorithm.

→ Huffman trees & codes.

→ Task scheduling problem. etc.

Comparison between divide and Conquer and Greedy.

Divide & conquer	Greedy
→ not aim for optimal solution	→ obtain optimal solution.
→ Slow & less efficient.	→ Efficient & faster.
→ recursive	→ iterative
→ generate duplicate solution.	→ avoid recomputation
→ Top down approach	→ bottom up approach.

Task scheduling.

Job	J_1	J_2	J_3	J_4	J_5	J_6	J_7	J_8	J_9	J_{10}	J_{11}	J_{12}.
dead line	6	5	6	3	2	3	2	1	6	5	7	8
Profit	10	5	20	25	20	25	15	5	10	5	6	18.

Greedy strategy.

- Used to solve optimizatiai Problem.

Optimizatiai Problem will have.

* objective functiai. Value.

* Predicdite (condition). : P.

* Solutiai space U (set of feasible solutions).

* optimal solutiai

Task scheduling algorithm.

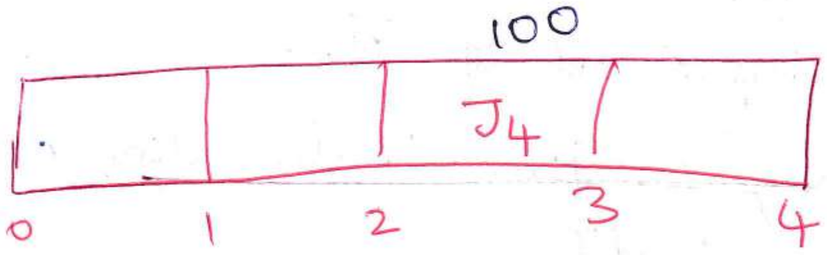
- Find the maximum deadline Value from the input set of jobs.
- Arrange the jobs in descending order of their Profit.
- Select the job with highest profit, their time period not exceed the maximum deadline.
- Selected jobs are the Output.

Example : A.

Jobs.	J_1	J_2	J_3	J_4	J_5
Deadline	2	2	1	3	4
Profit	20	60	40	100	80

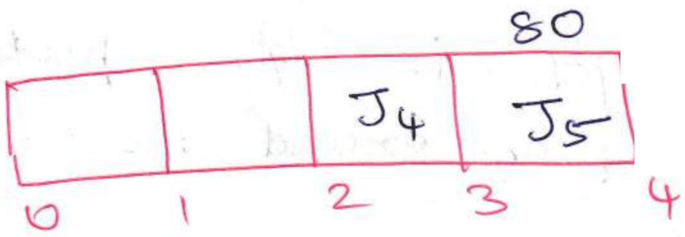
Job	deadline	Profit.
J_4	3	100
J_5	4	80.
J_2	2	60.
J_3	1	40
J_1	2	20

J_1 J_2 J_3 J_4 J_5



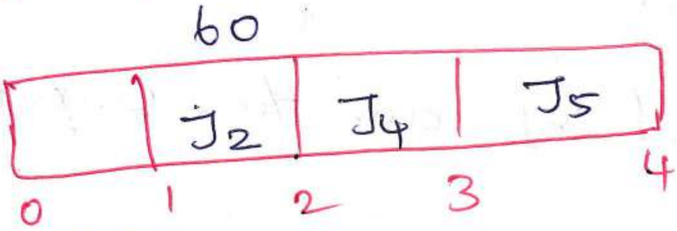
J_4

100.



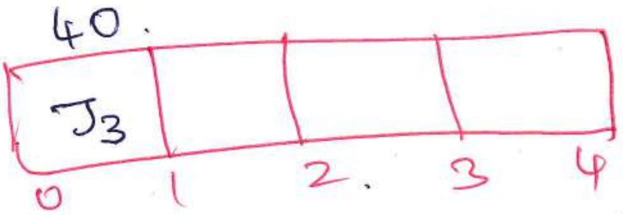
J_4 J_5

100 + 80.



J_4 J_5 J_2

100 + 80 + 60.



J_4 J_5 J_2 J_3.

100 + 80 + 60 + 40.

Profit = 280.

J_1 is not scheduled because it was already occupied by J_2

Job/ scheduling using Greedy.
Task

②

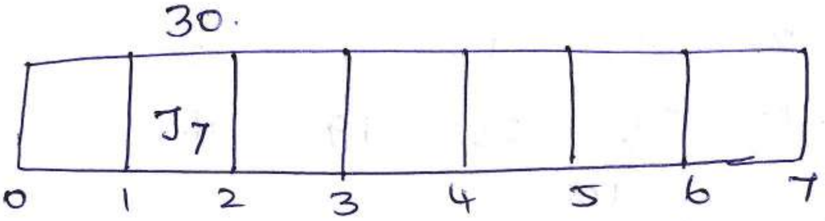
Ex : B

Jobs	J_1	J_2	J_3	J_4	J_5	J_6	J_7
Deadline	1	3	4	3	2	1	2
Profit	3	5	20	18	1	6	30

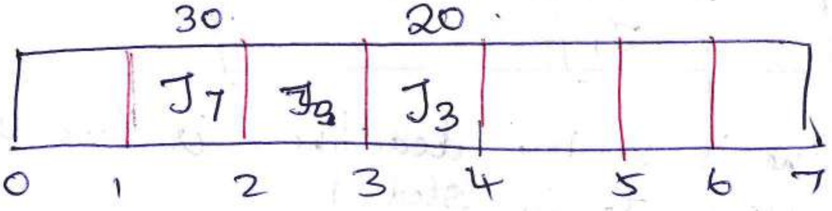
Sort all jobs in descending order of profit.

Jobs	J_7	J_3	J_4	J_6	J_2	J_1	J_5
Deadline	2	4	3	1	3	1	2
Profit	30	20	18	6	5	3	1

J_7

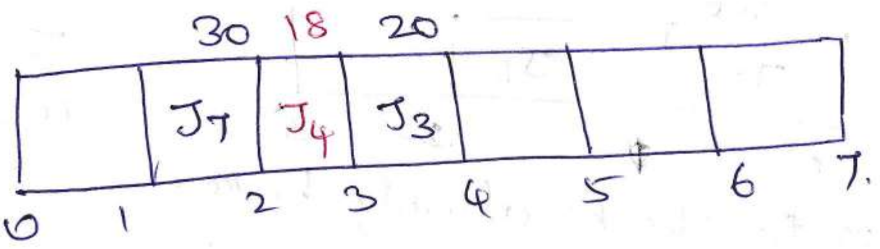


J_3



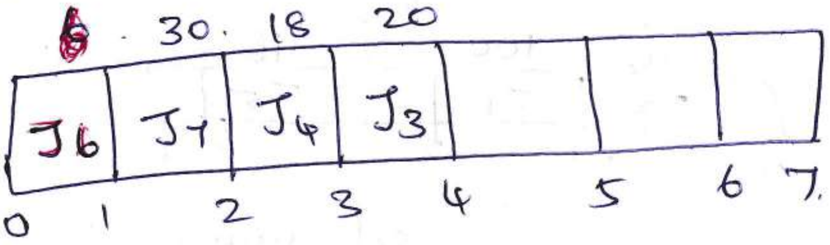
J_7, $\boxed{J_3}$
30 + 20.

J_4.



J_7 J_3 $\boxed{J_4}$
30+20+18

J_6



J_7 J_3 J_4 $\boxed{J_6}$
30+20+18+6.

J_2 — slot already booked
J_1 — slot " "
J_5 — slot " "
} so J_2 J_1 J_5 ~~are~~ rejected

P.T.O.