D) Implement job Scheduling with deadlines using a greedy method.

Aarti Ambekar

Rollno:20121056

import java.util.Arrays;

class Job {

    char id;

    int deadline;

    int profit;

    public Job(char id, int deadline, int profit) {

        this.id = id;

        this.deadline = deadline;

        this.profit = profit;

    }

}

public class JobSchedulingDeadlines {

    public static void scheduleJobs(Job[] jobs)

    {

        int n=jobs.length;

        Arrays.sort(jobs,(a, b) -> b.profit-a.profit);

        boolean[] slot=new boolean[n];

        char[] result=new char[n];

        int totalProfit=0;

        for (int i = 0; i < n; i++) {

            for (int j = Math.min(n, jobs[i].deadline) - 1; j >= 0; j--) {

                if (!slot[j]) {

                    slot[j] = true;

                    result[j] = jobs[i].id;

                    totalProfit += jobs[i].profit;

                    break;

                }

            }

        }

        System.out.println("Scheduled Jobs:");

        for (int i = 0; i < n; i++) {

            if (slot[i]) {

                System.out.println(result[i] + " (Profit: " + jobs[i].profit + ")");

            }

        }

        System.out.println("Total Profit: " + totalProfit);

    }

    public static void main(String[] args) {

        Job[] jobs = {

                new Job('A', 2, 100),

                new Job('B', 3, 200),

                new Job('C', 4, 400),

                new Job('D', 5, 500),

                new Job('E', 6, 600),

                new Job('F', 7, 700),

        };

        scheduleJobs(jobs);

    }

}

Output:

