TOPIC 5.3 : JOB ASSIGNMENT TO MINIMIZE MAXIMUM WORKING TIME

Problem Statement  
You are given an integer array jobs, where jobs[i] is the amount of time it takes to complete the ith job.  
There are k workers that you can assign jobs to. Each job should be assigned to exactly one worker.  
The working time of a worker is the sum of the time it takes to complete all jobs assigned to them.  
Your goal is to devise an optimal assignment such that the maximum working time of any worker is minimized.  
Return the minimum possible maximum working time of any assignment.

Example 1  
Input: jobs = [3,2,3], k = 3  
Output: 3  
Explanation: By assigning each person one job, the maximum time is 3.

Example 2  
Input: jobs = [1,2,4,7,8], k = 2  
Output: 11  
Explanation: Assign the jobs the following way:  
Worker 1: 1, 2, 8 (working time = 1 + 2 + 8 = 11)  
Worker 2: 4, 7 (working time = 4 + 7 = 11)  
The maximum working time is 11.

Aim  
To write a program that assigns jobs to k workers such that the maximum working time among all workers is minimized.

Algorithm

1. Start
2. Sort the jobs array in descending order
3. Initialize an array workers[k] to track total time for each worker
4. Use backtracking:
   * Assign each job to a worker
   * Recurse to assign next job
   * Track the maximum time of any worker in each assignment
   * Prune search if current max ≥ best answer found so far
5. Return the minimum maximum working time found
6. Stop

Input and Output  
A screenshot of a computer

AI-generated content may be incorrect.

Result  
The program successfully assigns jobs to workers in an optimal way such that the maximum working time is minimized.

Performance Analysis  
Time Complexity:  
Exponential in worst case (backtracking), but pruning and sorting reduce unnecessary states  
Efficient enough for small to medium input sizes

Space Complexity:  
O(k) for storing workers' current loads