# 826. Most Profit Assigning Work

You have n jobs and m workers. You are given three arrays: difficulty, profit, and worker where:

difficulty[i] and profit[i] are the difficulty and the profit of the ith job, and

worker[j] is the ability of jth worker (i.e., the jth worker can only complete a job with difficulty at most worker[j]).

Every worker can be assigned at most one job, but one job can be completed multiple times.

For example, if three workers attempt the same job that pays $1, then the total profit will be $3. If a worker cannot complete any job, their profit is $0.

Return the maximum profit we can achieve after assigning the workers to the jobs.

## SOLUTION IN JAVA

class Solution {

public int maxProfitAssignment(int[] difficulty, int[] profit, int[] worker) {

Arrays.sort(worker);

int n = profit.length;

int[][] jobs = new int[n][0];

for (int i = 0; i < n; ++i) {

jobs[i] = new int[] {difficulty[i], profit[i]};

}

Arrays.sort(jobs, (a, b) -> a[0] - b[0]);

int ans = 0, mx = 0, i = 0;

for (int w : worker) {

while (i < n && jobs[i][0] <= w) {

mx = Math.max(mx, jobs[i++][1]);

}

ans += mx;

}

return ans;

}

}