1335. Minimum Difficulty of a Job Schedule <Hard>

***Dynamic Programming***

class Solution {

public:

int minDifficulty(vector<int>& jobDifficulty, int d) {

const int n = jobDifficulty.size();

if (n < d)

return -1;

// dp[i][k] := min difficulty to schedule the first i jobs in k days

vector<vector<int>> dp(n + 1, vector<int>(d + 1, INT\_MAX / 2));

dp[0][0] = 0;

for (int i = 1; i <= n; ++i)

for (int k = 1; k <= d; ++k) {

int maxDifficulty = 0; // max(job[j + 1..i])

for (int j = i - 1; j >= k - 1; --j) { // 1-based

maxDifficulty = max(maxDifficulty, jobDifficulty[j]); // 0-based

dp[i][k] = min(dp[i][k], dp[j][k - 1] + maxDifficulty);

}

}

return dp[n][d];

}

};