Job Sequencing Problem

```
#include <stdio.h>
#include <vector>
#include <bits/stdc++.h>

int count = 0, maxProfit = 0;
void sortJobs(std::vector<std::pair<int, std::pair<int, int>>> &jobs) {
    // find max dealine
    int maxDeadline = jobs[0].second.first;
    for (int i = 1; i < jobs.size(); ++i) {
        maxDeadline = std::max(maxDeadline, jobs[i].second.first);
    }

    // 根据profit降序
    std::sort(jobs.begin(), jobs.end(),
              [](const std::pair<int, std::pair<int, int>> &a, const std::pair<int, std::pair<int, int>> &b) {
                  return a.second.second > b.second.second;
              });

    // 记录对应的deadline是否已经发生过了
    std::vector<bool> occupiedJob(maxDeadline);

    for (int i = 0; i < jobs.size(); ++i) {
        for (int j = jobs[i].second.first - 1; j >= 0; j--) {
            if (occupiedJob[j] == false) {
                maxProfit += jobs[i].second.second;
                count += 1;
                occupiedJob[j] = true;
                break;
            }
        }
    }
}
int main() {
    int T;

    scanf("%d", &T);
    while (T--) {
        int N;
        scanf("%d", &N);
        std::vector<std::pair<int, std::pair<int, int>>> jobs;
        for (int i = 0; i < N; ++i) {
            int jobId, deadline, profit;
            scanf("%d %d %d", &jobId, &deadline, &profit);
            jobs.push_back(std::make_pair(jobId, std::make_pair(deadline, profit)));
        }
        count = 0, maxProfit = 0;
        sortJobs(jobs);
        printf("%d %d\n", count, maxProfit);
    }
    return 0;
}
```

Job sequencing problem

https://www.includehelp.com/icp/job-sequencing-problem.aspx

TODO

该题目还有很多解法