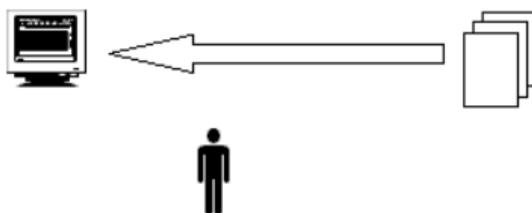# 2421   Mainframe

Mr. Ronald is responsible for the administration of the mainframe in ACM (Agent on Computing of Mathematics). The agent undertakes the mathematical computing jobs from some companies, and gain the rewards after has fulfilled the jobs on the mainframe. So the mainframe is very valuable to ACM. Mr. Ronald is required to arrange the order of those jobs running on mainframe. Once a job is to run, he examines the free resources for the job. If the resources meet the job's requirement, he assigns those resources to the job. Otherwise, the job will be suspended until there are enough resources.

Because of unfamiliar with the task at first, he turned everything upside down. As time went by, he became competent on it. Moreover, he had concluded a set of byelaw as following:

1. The mainframe has M CPUs and N memories can be assigned.

2. There exists a queue for the jobs waiting to be executed. You may assume the queue is large enough to hold all the waiting jobs.

3. A job J_i which need A_i CPUs and B_i memories, reaches the queue on time T_i. The job is required to be accomplished before time U_i. After successfully completed, ACM may get $V_i(\$)$ as the reward. If it finishes before the timeline, the extra bonus is $W_i(\$)$ per hour. If the job is late, the punishment is $X_i(\$)$ per hour. For example, we may assume that a job's value is 10\$, its timeline is 8, and the punishment is 2\$ per hour. If the job is completed at time 10, ACM will get $10 - (10 - 8) * 2 = 6\$$.

4. When the job start executing, the required CPUs and memories are seized by this job, and couldn't be assigned again for the other job to be executed simultaneously. After completing the job, those resources will be released. If the resources are enough, more jobs could be executed simultaneously.

5. For the sake of the share in the mainframe's computing capability, each job will be finished just in an hour from the start of executing. You may assume each job costs exactly one hour.

6. When there are no jobs to be executed, the mainframe will be idle until a job arrives at the job queue.

7. If there are more than one jobs arrive at the queue, the more valuable job will be executed first. You may assume the values of the jobs are always unequal ($V_i \neq V_j$).

8. If the free CPUs or memories couldn't satisfy the requirement of the job, the job will be suspended for an hour without occupying any resources. An hour later, the resources will be examined again for this job, regardless the other jobs in the queue. If the requirement unsatisfied again, it remains suspended for the next hour, and other jobs in the queue will try to be assigned the resources. Otherwise the job will seize the required CPUs and memories and start executing.

9. When more than one jobs are suspended, the earlier arrived will try to be assigned first.

Using the byelaw, Mr. Ronald may deal with the routines very well. But now, besides the routines, ACM ask him to compute the income according to the job list. Given the timeline F, he has to calculate the jobs that had been executed or should be executed. Of course, according to job J_i, if $U_i > F$ and the job hadn't been executed, it shouldn't been taken into account; but those which had been executed or $U_i \leq F$ should been counted. If the job hadn't been executed, it will not bring ACM any value, which means only punishment to the timeline should be calculated

Indeed, his programming ability is not good enough, and he does not like to compute manually. So he is uneasy about it. Could you help him to solve this problem?

Input

The input contains several test cases, each of which describes the mainframe's resources and the job list. Each test case begins with a line containing a single integer F, $0 \leq F \leq 10000$, the time line. The following line consists of three integers M, N and L ($M, N, L \geq 0$). M is the number of CPU in the mainframe, and N is the memory size. L represents the number of jobs in the job list. There will be 10000 jobs at most.

The subsequent L lines in the test case describe the information of the jobs. The data which describing job J_i consist of 7 integers A_i, B_i, T_i, U_i, V_i, W_i, X_i. A_i and B_i indicate the requirements on CPU and memory ($A_i, B_i \geq 0$). T_i and U_i indicate the job's arriving time and the timeline ($0 \leq T_i < U_i$). V_i, W_i, X_i are the reward, bonus and punishment of the job ($V_i, W_i, X_i \geq 0$).

The input file ends with an empty test case ($F = 0$). And this case should not be processed.

Output

Your program must compute the total income of the mainframe according to the job list. For each test case, print the case number, a colon, and a white space, then the income.

Print a blank line after each test case.

Note: Don't count the jobs which hadn't been executed, and their timelines are later than F.

Sample Input

```
10
4 256 3
1 16 2 3 10 5 6
2 128 2 4 30 10 5
2 128 2 4 20 10 5
0
```

Sample Output

```
Case 1: 74
```