Problem C
EDF Schedule

Input File: *testdata.in*
Time Limit: 1 seconds

Problem Description

There are n periodic tasks in a real-time system. Each task T_i is characterized by (ϕ_i, p_i, e_i, D_i), where $i = 1, 2, \ldots, n$. ϕ_i is the *phase* of task T_i, which indicates the release time of the first job in T_i. After the first job of a task is released, its subsequent jobs will be released with a *period* of p_i. In other words, if the first job of task T_i, denoted by $J_{i,1}$, is released at ϕ_i, its second job $J_{i,2}$ will be released at $\phi_i + p_i$, the third job $J_{i,3}$ will be released at $\phi_i + 2p_i$, and so on. The amount of time that each job in T_i will request to execute is e_i. D_i is the *relative deadline* of T_i; each job from task T_i is required to complete its execution within the time interval D_i since it is released. The absolute deadline of a job can be obtained by adding the relative deadline to its release time. To be more specific, the absolute deadline of the job $J_{i,j}$ is $\phi_i + (j-1)p_i + D_i$, where $\phi_i + (j-1)p_i$ is its release time and D_i is its relative deadline.

Earliest Deadline First (EDF) algorithm is an optimal priority-driven scheduling algorithm that assigns a priority to each job based on each job's absolute deadline. If jobs have the same absolute deadline, the scheduler always assigns a higher priority to the job with the shorter period. When a job is released, the scheduler will check whether its absolute deadline is earlier than the absolute deadline of the current executing job. If the job has an earlier absolute deadline, it will preempt the current job and execute. When the current job completes its execution, the scheduler will examine all the ready jobs, i.e., those jobs that have been released but not yet completed their executions, and pick the job with the earliest absolute deadline to execute. If there is no ready job, the system will be idle until a new job is released.

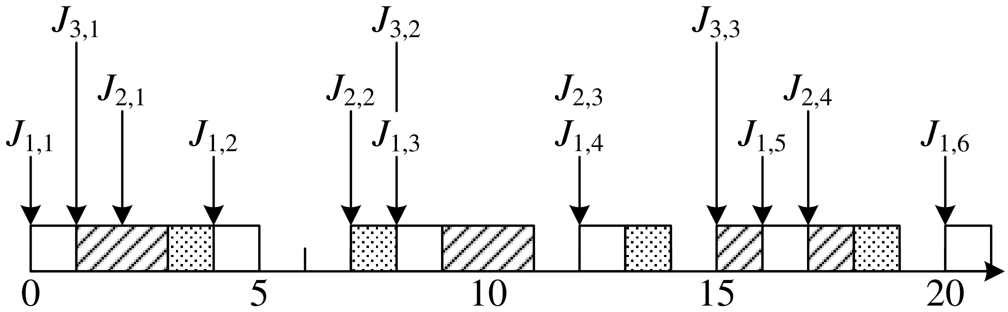

Figure 1: EDF Schedule.

To illustrate, suppose the system has three tasks: $T_1 = (0, 4, 1, 2)$, $T_2 = (2, 5, 1, 5)$, and $T_3 = (1, 7, 2, 4)$. Figure **??** shows the EDF schedule between the time interval $[0, 20]$. At time 0, $J_{1,1}$ is released. Since $J_{1,1}$ is the only ready job in the system, it executes. At time 1, $J_{3,1}$ is released. Since $J_{1,1}$ has completed its execution, $J_{3,1}$ is the only ready job and executes. At time 2, $J_{2,1}$ is released, and there are two ready jobs, $J_{3,1}$ and $J_{2,1}$, in the system. Since the absolute deadline of $J_{3,1}$ is 5, and the absolute deadline of $J_{2,1}$ is 7, $J_{3,1}$ continues its execution. At time 3, $J_{3,1}$ completes its execution, and $J_{2,1}$ is scheduled to execute. $J_{2,1}$ completes its execution at time 4. At the moment, $J_{1,2}$ is released and executes. In the time interval $[5, 7]$, since there is no ready job, the system is idle. We omit the trivial explanation of the schedule between the time interval $[7, 15]$. At time 15, $J_{3,3}$ is released and executes. At time 16, $J_{1,5}$ is released. Since the absolute deadline of $J_{1,5}$ is 18 and the absolute deadline of $J_{3,3}$ is 19, $J_{1,5}$ preempts $J_{3,3}$ and executes. At time 17, $J_{1,5}$ completes its execution and $J_{2,4}$ is released. Since $J_{3,3}$ has an earlier absolute deadline than $J_{2,4}$, $J_{3,3}$ is scheduled to execute. After $J_{3,3}$ completes its execution at time 18, $J_{2,4}$ executes until time 19. Then the system is idle again until time 20.

Given a task set $\mathbf{T}$ with n periodic tasks and a time interval $[a, b]$, can you determine the EDF schedule between the time interval $[a, b]$?

Technical Specifications

1. The phase ϕ_i of each task T_i is a non-negative integer; the period p_i, execution time e_i, and relative deadline D_i are all positive integers.

2. $e_i < p_i$ and $e_i < D_i$, for $1 \le i \le n$.
3. $\sum_{i=1}^{n} e_i/p_i \le 1$.
4. The examined time interval $[a, b]$ will not exceed the hyperperiod of the task set $\mathbf{T}$, i.e., the least common multiple of periods of all tasks.

Input Format

The first line of the input file specifies the time interval $[a, b]$; it contains two non-negative integers a and b, separated by space. The following n lines are a sequence of tasks; each line represents a task and contains five integers (separated by space): i, ϕ_i, p_i, e_i, and D_i, where i is the task index starting from 1 to n.

Output Format

Output the EDF schedule in the time interval $[a, b]$.
Each line specifies an executing job and its executing interval in the format of "`J(TaskNo,JobNo) BeginTime - EndTime`", where `TaskNo` is the task index and `JobNo` specifies the ordering of jobs generated by the task.

Sample Input

```
2 17
1 0 4 1 2
2 2 5 1 5
3 1 7 2 4
```

Sample Output

```
J(3,1) 2 - 3
J(2,1) 3 - 4
J(1,2) 4 - 5
J(2,2) 7 - 8
J(1,3) 8 - 9
J(3,2) 9 - 11
J(1,4) 12 - 13
J(2,3) 13 - 14
```

```
J(3,3) 15 - 16
J(1,5) 16 - 17
```