3349 - Real-Time Disk Scheduling

Asia - Taipei - 2005/2006

A magnetic disk is made up of the rotating disks, the heads to access disks, and an assembly arm that moves the heads into the specified cylinder. Generally, a disk task would specify the track location of data we want to retrieve. When serving the disk task, a seek-time is taken to move the disk arm to the appropriate cylinder. The seek-time to move the disk arm from track location x to track lo cation y is $b|x - y|$. Here b is a given constant, representing the speed of the disk, $|x - y|$ is the distance (absolute value) between x and y .

Conventionally, the goal of disk scheduling is to maximize the disk-throughput for serving a given set of tasks. That is, all tasks must be scheduled and executed as so on as possible to minimize the execution time. However, in a time-critical application, the first goal of disk scheduling is to meet the specified real-time constraints. That is, all tasks must be finished before its deadlines. Maximizing the disk throughput is also important, but the schedule is meaningless if the specified real-time constraints are not satisfied. Therefore, the formal definition of the real-time disk scheduling is as follows:

> Given a set of real-time disk tasks $T = \{T_1, T_2,..., T_n\}$, where n is the number of input disk tasks. Each task has a deadline and a track location. We say that a schedule is feasible if all tasks meet their deadlines in the schedule. Our goal is to find a feasible schedule with minimum execution time.

For example, assume that we have four tasks,

$$\{T_1, T_2, T_3, T_4\}$$

whose deadlines are

$$\{5, 10, 12, 24\}$$

and track lo cations at

$$\{100, 40, 600, 70\}.$$

Assume that the current position of disk head is 20 and $b = 0.01$. The optimal solution is $(2, 4, 1, 3)$. That is, the tasks are scheduled in the order:

$$T_2, T_4, T_1, T_3.$$

The minimum execution time for the schedule of this four tasks are

$$0.01(40 - 20) + 0.01(70 - 40) + 0.01(100 - 70) + 0.01(600 - 100) = 5.8.$$

In this schedule, all tasks are scheduled before their deadlines. Note that the disk head can be at any position before the schedule starts, and can be at any position when the schedule ends.

Input

The input file may contain several test cases. Each test case consists of a set of real-time disk tasks. The first line of test case is a single integer n , representing the number of tasks. The second line of the test case has

two numbers. The first number is a real number, representing the disk speed b . The second number is an integer p , representing the current position of disk head. Starting from the third line, each line specifies a task. There are two integers in each line. The first integer is the track lo cation of the task. The second integer is the deadline of the task. All integers have the maximum value 1000000.

The last test case is followed by a line containing `0'. This signals the end of the input file.

Output

The output is a number t , the minimum execution time of all tasks. If no feasible schedules exist, output `-1'.

Sample Input

```
4
0.01 20
100 5
40 10
600 12
70 24
0
```

Sanple Output

```
5.8
```

Taipei 2005-2006