B - Precision Agriculture

Context

Would you be able to solve the problem of the irrigation programs in a more precise way, second by second? Be careful, because the execution times can rise quickly...

The Problem

You have to solve the same problem of the irrigation programs, by with more programs and second by second. Given some irrigation programs, your task is to transform them into a series of open/close commands for the valves.

In this case, the opening times and closing times include also the seconds. So, a time is given by a day of the week (Monday, Tuesday...), hour (from 0 to 23), minute (from 0 to 59) and second (from 0 to 59). For example, the following irrigation program:

```
L 00:00:05 L 00:00:10
```

means: start irrigating every Monday at 00:00:05, and finish at 00:00:10. The same restrictions and assumptions as in the previous problem apply in this case. Thus, this program would be transformed into a sequence of close (0) / open (1) commands, from second to second, starting from second 00:00:00 of Monday, as:

```
000001111100000000000000000...
```

The Input

The input consists of a series of irrigation programs. The first line of the input contains two integer numbers: *N M*. *N* is the number of irrigation programs that we have to process; it can be from 1 to 100000, inclusive. And *M* is the number of seconds that we want to obtain in the sequence of commands, always staring from second 00:00:00 of Monday; it can be from 1 to 604800, inclusive (604800 is the total number of seconds of a week = 7*24*60*60).

Then, we have *N* lines, where each line contains a program. Each program has two times, and each time has: day of the week, hour (from 0 to 23), minute (from 0 to 59) and second (from 0 to 59). The day of the week is a letter: L, M, X, J, V, S, D, for Monday, Tuesday, Wednesday, Thursday, Friday, Saturday and Sunday, respectively. The hour:minute:second can be written with or without preceding zeros, that is, both 1:2:3 and 01:02:03 are valid.

The Output

In the output, you have to write a single line (finishing with a "\n"), with *M* characters. This line contains the close (0) / open (1) commands of the valves for the given programs, second by second, always starting from second 00:00:00 of Monday. You have to output these 0/1 characters without any separators between them.

Sample Input

```
7 30
M 13:22:12 M 17:00:00
L 0:0:23 L 00:0:24
```

```
L 0:00:05 L 0:00:10
L 00:0:10 L 00:0:15
L 0:00:15 L 0:00:20
L 0:0:11 L 0:0:18
L 16:55:30 D 23:59:59
```

Sample Output

```
00000111111111111111110001000000
```

OMP'22
Facultad de Informática
Universidad de Murcia