The 30th Annual ACM International Collegiate Programming Contest ASIA Regional - Seoul

Problem F
Seminar Room

My department, Department of Computer Science, has a seminar room equipped with luxurious conveniences. Many research groups want to utilize the seminar room among others. To make out a weekly schedule of the seminar room, each research group was requested to submit a candidate time interval for its own seminar. Having realized that some candidate time intervals were overlapped, the department chairman Prof. Yang fell in a difficult situation. In order to assign the seminar room to the groups as fair as possible, he decided to request the seminar groups to submit two candidate time intervals. The two time intervals submitted by a group can be overlapped; however, one should not be contained in the other. Now, Prof. Yang's problem of preparing a weekly schedule is to assign one of the candidate time intervals submitted by each group to the group so that the time intervals assigned to groups do not overlap each other. If the finish time of one time interval coincides with the start time of the other, the two time intervals are not considered to be overlapped.

A candidate time interval is specified by a pair of the start time and finish time. The time is written in format $ddd : hh : mm$, where ddd are three capital letters representing the day of a week, hh are two digits representing full hours $(00 \le hh \le 23)$, and mm are two digits representing minutes $(00 \le mm \le 59)$. On Sundays, a reservation will not be made. Thus, ddd are one of MON, TUE, WED, THU, FRI, SAT.

For example, suppose you are given three seminar groups and its candidate time intervals as shown in the table below.

seminar group 1	MON:09:00 MON:11:00
	MON:10:00 MON:12:00
seminar group 2	MON:09:00 MON:11:30
	TUE:13:25 TUE:14:27
seminar group 3	MON:09:30 MON:11:00
	MON:23:00 TUE:01:00

If three time intervals [MON 09:00 - MON 11:00], [TUE 13:25 - TUE 14:27], and [MON 23:00 - TUE 01:00] are assigned to seminar groups 1, 2, and 3, respectively, the time intervals do not overlap each other, and thus all the seminar groups will be happy.

Now, Prof. Yang wants to find an efficient method to determine, for given two candidate time intervals per each seminar group, whether or not one of the candidate time intervals can be assigned to each group so that all the time intervals assigned to groups do not overlap each other. Write a program that can help Prof. Yang.

Input

Your program is to read from standard input. The input consists of T test cases. The number of test cases T is given in the first line of the input. Each test case starts with a line containing a positive integer n, $n \le 1{,}000$, representing the number of seminar groups. In the next n lines, each line contains two candidate time intervals for a seminar group in the form $d_1d_1d_1 : h_1h_1 : m_1m_1$, $d_2d_2d_2 : h_2h_2 : m_2m_2$, $d_3d_3d_3 : h_3h_3 : m_3m_3$, and $d_4d_4d_4 : h_4h_4 : m_4m_4$. The two candidate time intervals are [$d_1d_1d_1 : h_1h_1 : m_1m_1$ - $d_2d_2d_2 : h_2h_2 : m_2m_2$] and [$d_3d_3d_3 : h_3h_3 : m_3m_3$ - $d_4d_4d_4 : h_4h_4 : m_4m_4$]. There is a single space between the start time and finish time of a time interval and between the two time intervals.

Output

Your program is to write to standard output. Print exactly one line for each test case. Print `YES` if one of the candidate time intervals can be assigned to each group so that all the time intervals assigned to groups do not overlap each other. Otherwise, print `NO`.

The following shows sample input and output for three test cases.

Sample Input

```
3
3
MON:09:00 MON:11:00 MON:10:00 MON:12:00
MON:09:00 MON:11:30 TUE:13:25 TUE:14:27
MON:09:30 MON:11:00 MON:23:00 TUE:01:00
2
SAT:08:00 SAT:09:00 FRI:13:13 FRI:14:14
SAT:08:00 SAT:09:00 FRI:13:13 FRI:14:14
4
FRI:13:13 FRI:14:14 SAT:08:00 SAT:09:00
FRI:13:13 FRI:14:14 SAT:08:00 SAT:09:00
FRI:13:13 FRI:14:14 SAT:08:00 SAT:09:00
FRI:13:13 FRI:14:14 SAT:08:00 SAT:09:00
```

Output for the Sample Input

```
YES
YES
NO
```