

Turmbau

Als Lea neulich ihren Onkel in Riesenhausen besuchte, wurde sie daran erinnert, dass manche Menschen einfach größer sind als andere. In der Wohnung ihres Onkels konnte sie nicht einmal die Gläser erreichen, die im obersten Küchenfach standen. Allerdings lagen einige Umzugskartons herum, die sie nun benutzen möchte, um einen Turm daraus zu bauen und so an die Gläser zu gelangen.

Ein solcher Turm ist selbstverständlich eine wacklige Angelegenheit. Damit er nicht umstürzen kann, darf Lea einen Umzugskarton a mit Breite x_a , Länge y_a und Höhe z_a lediglich dann auf einen anderen Karton b mit Abmessungen x_b, y_b, z_b stapeln, wenn sowohl $x_a < x_b$ als auch $y_a < y_b$ gilt; der obere Karton muss also vollständig auf dem unteren aufliegen und etwas zusätzlichen Platz haben. Kartons können beliebig gedreht werden, um diese Bedingung zu erfüllen.

Lea möchte nun herausfinden, ob es mithilfe eines Karton-Turms möglich ist, die beabsichtigte Höhe zu erreichen.

Eingabe

Die erste Zeile der Eingabe enthält eine Ganzzahl t . Darauf folgen, jeweils durch eine Leerzeile getrennt, t Testfälle.

Jeder Testfall beginnt mit der Höhe h , die der Turm erreichen soll, und der Anzahl verschiedener Kartons n . Darauf folgen n Zeilen, von denen die i -te die Dimensionen x_i, y_i, z_i der i -ten Karton-Sorte enthält. Von jeder Sorte stehen Lea jeweils genau 5 Kartons zur Verfügung.

Ausgabe

Gibt für jeden Testfall eine Zeile der Form „Case # i : x “ aus, wobei i bei 1 beginnend die Nummer des Testfalls ist und x entweder „yes“, falls Lea unter Beachtung des obigen Sicherheit-Kriteriums einen Turm mit Höhe mindestens h aus den verfügbaren Kartons bauen kann, oder andernfalls „no“, wenn dies nicht möglich ist.

Beschränkungen

- $1 \leq t \leq 20$
- $1 \leq n \leq 1000$
- $1 \leq x_i, y_i, z_i \leq 40000$
- $1 \leq h \leq 5 \cdot 10^6$

Sample Input 1

11
9 1
5 4 3

7 2
4 2 2
3 1 5

8 2
3 5 5
2 2 3

6 3
4 1 1
5 2 4
3 1 3

8 2
5 2 2
4 5 1

10 4
2 1 5
2 5 3
4 1 1
5 2 4

6 3
1 2 1
1 5 3
5 4 2

7 1
3 1 3

7 2
1 5 4
6 5 6

2 4
5 4 1
1 3 4
3 6 3
2 6 2

8 2
1 6 6
2 2 2

Sample Output 1

Case #1: no
Case #2: yes
Case #3: yes
Case #4: yes
Case #5: no
Case #6: yes
Case #7: yes
Case #8: no
Case #9: yes
Case #10: yes
Case #11: no

Packing Cases

Just recently, during Lea's visit at her uncle's house, she was reminded that while some people are quite tall, sadly she is not. She could not even reach the glasses that were stored in the topmost shelf in the kitchen. Luckily for her, there were a lot of packing cases lying around and she could use them to build a tower and then climb on it to reach the glasses.

Building such a tower is of course a very shaky endeavour, and Lea does not want to fall. So she imposed the following restriction on the tower: Given two packing cases a and b with dimensions x_a, y_a, z_a and x_b, y_b, z_b , case a may only be stacked onto case b if $x_a < x_b$ and $y_a < y_b$. Please remember that a case can be rotated to fit that restriction.

Lea now has to figure out whether it is possible to reach the desired height if she stacks the cases optimally, or not.

Input

The first line of the input contains an integer t . t test cases follow, each of them separated by a blank line.

Each test case starts with two integers, h and n , the height the tower should reach and the number of case types, n lines follow. The i -th line describes the i -th case layout and contains three integers x_i, y_i, z_i . Lea has exactly 5 Boxes of each type at her disposal.

Output

For each test case, output one line containing "Case # i : x " where i is its number, starting at 1, and x is either "yes" if Lea can build a tower of height at least h according to the constraints, or "no" if it is not possible to do so. Each line of the output should end with a line break.

Constraints

- $1 \leq t \leq 20$
- $1 \leq n \leq 1000$
- $1 \leq x_i, y_i, z_i \leq 40000$
- $1 \leq h \leq 5 \cdot 10^6$

Sample Input 1

11
9 1
5 4 3

7 2
4 2 2
3 1 5

8 2
3 5 5
2 2 3

6 3
4 1 1
5 2 4
3 1 3

8 2
5 2 2
4 5 1

10 4
2 1 5
2 5 3
4 1 1
5 2 4

6 3
1 2 1
1 5 3
5 4 2

7 1
3 1 3

7 2
1 5 4
6 5 6

2 4
5 4 1
1 3 4
3 6 3
2 6 2

8 2
1 6 6
2 2 2

Sample Output 1

Case #1: no
Case #2: yes
Case #3: yes
Case #4: yes
Case #5: no
Case #6: yes
Case #7: yes
Case #8: no
Case #9: yes
Case #10: yes
Case #11: no