5010 Go Deeper

Here is a procedure's pseudocode:

```
go(int dep, int n, int m)
begin
    output the value of dep.
    if dep < m and x[a[dep]] + x[b[dep]] != c[dep] then go(dep + 1, n, m)
end
```

In this code n is an integer. a, b, c and x are 4 arrays of integers. The index of array always starts from 0. Array a and b consist of non-negative integers smaller than n. Array x consists of only 0 and 1. Array c consists of only 0, 1 and 2. The lengths of array a, b and c are m while the length of array x is n.

Given the elements of array a, b, and c, when we call the procedure go$(0, n, m)$ what is the maximal possible value does the procedure output?

Input

There are multiple test cases. The first line of input is an integer T $(0 < T \leq 100)$, indicating the number of test cases. Then T test cases follow. Each case starts with a line of 2 integers n and m $(0 < n \leq 200,\ 0 < m \leq 10000)$. Then m lines of 3 integers follow. The i-th $(1 \leq i \leq m)$ line of them are a_{i-1}, b_{i-1} and c_{i-1} $(0 \leq a_{i-1}, b_{i-1} < n,\ 0 \leq c_{i-1} \leq 2)$.

Output

For each test case, output the result in a single line.

Sample Input

```
3
2 1
0 1 0
2 1
0 0 0
2 2
0 1 0
1 1 2
```

Sample Output

```
1
1
2
```