**Grid**

**Time Limit: 6000/3000 MS (Java/Others) Memory Limit: 32768/32768 K (Java/Others)**

**Problem Description**

There are n boxes in one line numbered 1 to n, at the beginning, all boxes are black. Two kinds of operations are provided to you:

1 ai xi ：You can choose any xi black boxes in interval [1,ai], and color them white;

2 ai xi ：You can choose any xi black boxes in interval [ai,n], and color them white;

lcq wants to know if she use these operations in optimal strategy, the maximum number of white boxes she can get, and if she get maximum white boxes, the minimum number of operations she should use.

Tips: 1. It is obvious that sometimes you can choose not to use some operations.

2. If the interval of one operation didn’t have enough black boxes, you can’t use this operation.

**Input**

The first line contains one integer T, indicating the number of test case.

The first line of each test case contains two integers N (1 <= N <= 1000) and M (1<=M<=1000), indicating that there are N grids and M operations in total. Then M lines followed, each of which contains three integers si(1<=si<=2) , ai and xi (0 <= xi <= N,1<=ai<=N), si indicating the type of this operation, ai and xi indicating that the interval is [1,ai] or [ai,n](depending on si), and you can choose xi black boxes and color them white.

**Output**

For each test case, output case number first. Then output two integers, the first one is the maximum boxes she can get, the second one is the minimum operations she should use.

**Sample Input**

1

5 2

2 3 3

1 3 3

**Sample Output**

Case 1: 3 1