**Schedule**

**Time Limit: 4000/2000 MS (Java/Others)    Memory Limit: 153428/153428 K (Java/Others)**

**Problem Description**

There are N schedules, the i-th schedule has start time*si* and end time *ei* (1 <= i <= N). There are some machines. Each two overlapping schedules cannot be performed in the same machine. For each machine the working time is defined as the difference between *timeend* and *timestart*, where *timeend* is time to turn off the machine and *timestart* is time to turn on the machine. We assume that the machine cannot be turned off between the *timestart* and the *timeend*.   
Print the minimum number K of the machines for performing all schedules, and when only uses K machines, print the minimum sum of all working times.

**Input**

The first line contains an integer T (1 <= T <= 100), the number of test cases. Each case begins with a line containing one integer N (0 < N <= 100000). Each of the next N lines contains two integers *si* and *ei* (0<= *si* < *ei* <=1e9).

**Output**

For each test case, print the minimum possible number of machines and the minimum sum of all working times.

**Sample Input**

1

3

1 3

4 6

2 5

**Sample Output**

2 8