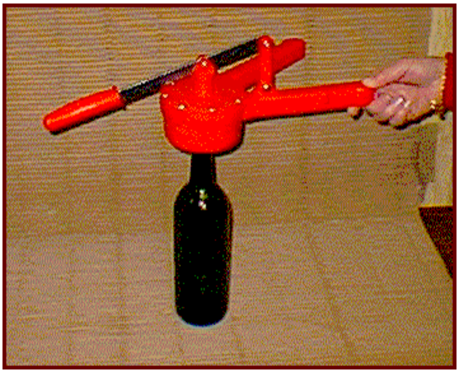

Wine bottles are never completely filled: a small amount of air must be left in the neck to allow for thermal expansion and contraction. If too little air is left in the bottle, the wine may expand and expel the cork; if too much air is left in the bottle, the wine may spoil. Thus each bottle has a minimum and maximum capacity.

Given a certain amount of wine and a selection of bottles of various sizes, determine which bottles to use so that each is filled to between its minimum and maximum capacity and so that as much wine as possible is bottled.

Input

The input begins with a single positive integer on a line by itself indicating the number of the cases following, each of them as described below. This line is followed by a blank line, and there is also a blank line between two consecutive inputs.

The first line of input contains two integers: the amount of wine to be bottled (in litres, between 0 and 1,000,000) and the number of sizes of bottles (between 1 and 100). For each size of bottle, one line of input follows giving the minimum and maximum capacity of each bottle in millilitres. The maximum capacity is not less than 325 ml and does not exceed 4500 ml. The minimum capacity is not less than 95% and not greater than 99% of the maximum capacity. You may assume that an unlimited number of each bottle is available.

Output

For each test case, the output must follow the description below. The outputs of two consecutive cases will be separated by a blank line.

Your output should consist of a single integer: the amount of wine, in *ml*, that cannot be bottled.

Sample Input

```
2

10 2
4450 4500
725 750

10000 2
4450 4500
725 750
```

Sample Output

```
250

0
```