ACM-ICPC Live Archive

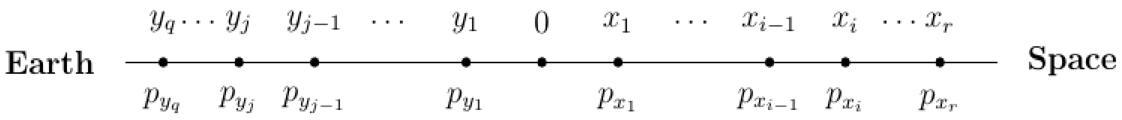
4266 Space Elevator

Seen as an engineering undertaking for the opening decades of the 21st century, the space elevator proposal was highlighted during the 2002 Space and Robotics Conferences. In 2108, after a sequence of technology breakthroughs, the dream of traveling from Earth to outer space with an elevator finally comes to reality. A space elevator made of a *carbon nanotubes composite ribbon* anchored to an offshore sea platform stretches to a small counterweight approximately 62,000 miles (100,000 km) into space. Mechanical lifters attached to the ribbon could then climb the ribbon, carrying cargo and humans into space. Carbon nanotubes have the potential to be 100 times stronger than steel and are as flexible as plastic. The strength of carbon nanotubes comes from their unique structure, which resembles soccer balls.

To better understand the concept of a space elevator, think of the game tetherball in which a rope is attached at one end to a pole and at the other to a ball. In this analogy, the rope is the carbon nanotubes composite ribbon, the pole is the Earth and the ball is the counterweight. Now, imagine the ball is placed in perpetual spin around the pole, so fast that it keeps the rope taut. This is the general idea of the space elevator. The counterweight spins around the Earth, keeping the cable straight and allowing the robotic lifters to ride up and down the ribbon.

Air Climber Motor (ACM), a space-infrastucture company, has been among those who support construction of a space elevator. ACM provides various of services to maintain the functionality of the space elevator. A crew of ACM routinely travels along the ribbon up and down at unit speed to serve the tourists and scientists at various stops, which are set up for spectacular sightseeing or scientific activities. ACM makes profits out of serving the requests at the stops. If the crew serves stop x at time t, then the revenue collected equals $p_x - t$, where we assume the service can be done instantly and ignore the service time.

We use the following diagram to illustrate the space elevator with a horizontal line, where x_i's and y_j's are positive integers (except $x_0 = y_0 = 0$ as the origin) to indicate the locations of stops and p_{x_i}'s, p_{y_j}'s are the profits for the corresponding services at the stops. Initially, the crew starts at the origin. We refer to the stops right of the origin as $x_1, \ldots, x_r$, and those to the left as $y_1, \ldots, y_q$. Both sequences are in increasing order.



The goal is to select stops to serve (i.e., not every client needs to be served) and to find a route for the crew such that the total profits of the served stops minus the latency of the corresponding route is maximized. Since the length of the space elevator is so long and the number of stops is large, you are asked to write a program to help the crew select the stops to serve and maximize the revenue. (You see. This is yet another example that no matter how technology advances, the quest for programmer's help never ends!)

Technical Specification

1. All the numbers are non-negative integers.
2. n: the number of test cases. $n \geq 10$.
3. r, s: the number of requests to the right and left of the origin, respectively. $1 \leq r, s \leq 100$.

4. $1 \leq x_i, y_j, p_{x_i}, p_{y_j} \leq 1000$, for all $1 \leq i \leq r$ and $1 \leq j \leq s$.

5. $x_0 = y_0 = 0$, $p_{x_0} = p_{y_0} = 0$.

Input

The first line of the input file contains an integer n indicating the number of test cases to follow. Each test case starts with two positive integers r and q, which are at most 100. In each of the following $r + q$ lines, there are two positive integers, where the first one is the position of a request and the second one indicates the corresponding profit. The first r lines are for x_i's and the following q lines are for y_j's. The range of x_i's, y_j's and the profits is from 1 to 1000. Note that x_i's may not be sorted in the corresponding r lines and likewise for y_j's.

Each test case ends with a '0' in a separate line.

Output

For each test case, output the maximum profit in a separate line.

Sample Input

```
2
2  1
1  1
2  5
1  4
0
3  3
3  1
1  2
2  3
2  2
3  3
1  1
0
```

Sample Output

```
Case 1: 4
Case 2: 2
```