Round 1C 2012

A. Diamond Inheritance

B. Out of Gas

C. Box Factory

Contest Analysis

Questions asked

Submissions

Diamond Inheritance

14pt	Not attempted **3062/4215 users** correct (73%)
14pt	Not attempted **2374/3030 users** correct (78%)

Out of Gas

10pt	Not attempted **467/762 users** correct (61%)
27pt	Not attempted **73/250 users** correct (29%)

Box Factory

12pt	Not attempted **1064/1800 users** correct (59%)
23pt	Not attempted **308/786 users** correct (39%)

Top Scores

mystic	100
sourspinach	100
meret	100
FloppyCat	100
Yao	100
AS1	100
fuseidenamida	100
Tan909090909090	100
AdrianRoos	100
MaxBuzz	100

Problem C. **Box Factory**

This contest is open for practice. You can try every problem as many times as you like, though we won't keep track of which problems you solve. Read the Quick-Start Guide to get started.

Small input 12 points	Solve C-small
Large input 23 points	Solve C-large

Problem

You own a factory with two assembly lines. The first assembly line makes boxes, and the second assembly line makes toys to put in those boxes. Each type of box goes with one type of toy and vice-versa.

At the beginning, you pick up a box from the first assembly line and a toy from the second assembly line. You then have a few options.

- You can always throw out the box and pick up the next one.
- You can always throw out the toy and pick up the next one.
- If the box and toy are the same type, you can put the toy in the box, and send it out to customers.

You always pick boxes up in the order in which they are made, and similarly for toys. You know the order in which boxes and toys are made, and you want to plan out a strategy that will allow you to send as many boxed toys as possible to customers.

Warning: The two assembly lines make a *lot* of boxes and toys. However, they tend to make one kind of thing for a long period of time before switching.

Input

The first line of the input gives the number of test cases, **T**. **T** test cases follow.

Each test case begins with a line contains two integers **N** and **M**. It is followed by a line containing 2 * **N** integers $\mathbf{a_1}$, $\mathbf{A_1}$, $\mathbf{a_2}$, $\mathbf{A_2}$, ..., $\mathbf{a_N}$, $\mathbf{A_N}$, and another line containing 2 * **M** integers $\mathbf{b_1}$, $\mathbf{B_1}$, $\mathbf{b_2}$, $\mathbf{B_2}$, ..., $\mathbf{b_M}$, $\mathbf{B_M}$.

This means that the first assembly line will make $\mathbf{a_1}$ boxes of type $\mathbf{A_1}$, then $\mathbf{a_2}$ boxes of type $\mathbf{A_2}$, etc., until it finishes with $\mathbf{a_N}$ boxes of type $\mathbf{A_N}$. Similarly, the second assembly will make $\mathbf{b_1}$ toys of type $\mathbf{B_1}$, followed by $\mathbf{b_2}$ toys of type $\mathbf{B_2}$, etc., until it finishes with $\mathbf{b_M}$ toys of type $\mathbf{B_M}$.

A toy can be matched with a box if and only if they have the same type number.

Output

For each test case, output one line containing "Case #x: y", where x is the case number (starting from 1), and y is the largest number of boxed toys that you can send out to customers.

Limits

$1 \le \mathbf{T} \le 100$.
$1 \le \mathbf{a_i}, \mathbf{b_i} \le 10^{16}$.
$1 \le \mathbf{A_i}, \mathbf{B_i} \le 100$.

Small dataset

$1 \le \mathbf{N} \le 3$.
$1 \le \mathbf{M} \le 100$.

Large dataset

$1 \le \mathbf{N}, \mathbf{M} \le 100$.

Sample

```
Input                  Output

4                      Case #1: 35
3 3                    Case #2: 20
10 1 20 2 25 3         Case #3: 21
10 2 30 3 20 1         Case #4: 0
3 5

10 1 6 2 10 1
5 1 3 2 10 1 3 2 5 1
```

```
3 5
10 1 6 2 10 1
5 1 6 2 10 1 6 2 5 1
1 1
5000000 10
5000000 100
```