code jam
printf("hello, africa!");

Practice Mode

Contest scoreboard | Sign in

Online Competition Africa 2010

A. Odd Man Out

B. Get to Work

C. Qualification Round

D. Polygraph

Contest Analysis

Questions asked 1

Submissions

Odd Man Out

7pt	Not attempted **209/214 users** correct (98%)
7pt	Not attempted **206/209 users** correct (99%)

Get to Work

9pt	Not attempted **127/149 users** correct (85%)
9pt	Not attempted **124/127 users** correct (98%)

Qualification Round

11pt	Not attempted **47/87 users** correct (54%)
22pt	Not attempted **4/32 users** correct (13%)

Polygraph

12pt	Not attempted **14/30 users** correct (47%)
23pt	Not attempted **0/2 users** correct (0%)

Top Scores

RalfKistner	77
mohamedafattah	65
Ahmed.Kamel	65
gwylim	65
Nooodles	55
amrSamir	55
Blazerfrost	55
naguib	55
Kosie	55
mRefaat88	55

Problem A. Odd Man Out

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 7 points	Solve A-small
Large input 7 points	Solve A-large

Problem

You are hosting a party with G guests and notice that there is an odd number of guests! When planning the party you deliberately invited only couples and gave each couple a unique number C on their invitation. You would like to single out whoever came alone by asking all of the guests for their invitation numbers.

Input

The first line of input gives the number of cases, **N**.
N test cases follow. For each test case there will be:

- One line containing the value **G** the number of guests.
- One line containing a space-separated list of **G** integers. Each integer **C** indicates the invitation code of a guest.

Output

For each test case, output one line containing "Case #**x**: " followed by the number C of the guest who is alone.

Limits

$1 \leq N \leq 50$
$0 < C \leq 2147483647$

Small dataset

$3 \leq G < 100$

Large dataset

$3 \leq G < 1000$

Sample

```
Input                           Output
3                               Case #1: 1
3                               Case #2: 7
1 2147483647 2147483647         Case #3: 5
5
3 4 7 4 3
5
2 10 2 10 5
```

Online Competition Africa 2010

A. Odd Man Out

B. Get to Work

C. Qualification Round

D. Polygraph

Contest Analysis

Questions asked 1

Problem B. Get to Work

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
9 points

Solve B-small

Large input
9 points

Solve B-large

Problem

You work for a company that has **E** employees working in town **T**. There are **N** towns in the area where the employees live. You want to ensure that everyone will be able to make it to work. Some of the employees are drivers and can drive **P** passengers. A capacity of P == 1 indicates that the driver can only transport themselves to work. You want to ensure that everyone will be able to make it to work and you would like to minimize the number of cars on the road.

You want to calculate the number of cars on the road, with these requirements:

- Every employee can get to town **T**.
- The only way an employee may travel between towns is in a car belonging to an employee.
- Employees can only take rides from other employees that live in the same town.
- The minimum number of cars is used.

Find whether it is possible for everyone to make it to work, and if it is, how many cars will end up driving to the office.

Input

One line containing an integer **C**, the number of test cases in the input file.

For each test case there will be:

- One line containing the integer **N**, the number of towns in your area and the integer **T**, the town where the office is located.
- One line containing the integer **E**, the number of employees.
- E lines, one for each employee, each containing:
  - An integer H >= 1, the home town of the employee, followed by
  - An integer P >= 0, the number of passengers they can drive. If the employee is not licensed to drive the number will be 0.

Output

- **C** lines, one for each test case in the order they occur in the input file, each containing the string "Case #**X**: " where **X** is the number of the test case, starting from 1, followed by:
  - The string **IMPOSSIBLE**, if there are not enough drivers for everyone to commute; **OR**
  - **N** space-separated integers, one for each town from **1** to **N**, which indicate the number of vehicles commuting from the town.

Limits

1 ≤ T ≤ N
1 ≤ H ≤ N
0 ≤ P ≤ 6

Small dataset

C = 50
1 ≤ N ≤ 10
1 ≤ E ≤ 100

Large dataset

C = 100
1 ≤ N ≤ 100
1 ≤ E ≤ 500

Sample

```
Input     Output
3         Case #1: 0 0 0 0 0
5 1       Case #2: IMPOSSIBLE
3         Case #3: 1 0 0 1 0
1 0
1 0
1 0
```

Submissions

Odd Man Out
7pt Not attempted **209/214 users** correct (98%)
7pt Not attempted **206/209 users** correct (99%)

Get to Work
9pt Not attempted **127/149 users** correct (85%)
9pt Not attempted **124/127 users** correct (98%)

Qualification Round
11pt Not attempted **47/87 users** correct (54%)
22pt Not attempted **4/32 users** correct (13%)

Polygraph
12pt Not attempted **14/30 users** correct (47%)
23pt Not attempted **0/2 users** correct (0%)

Top Scores

RalfKistner	77
mohamedafattah	65
Ahmed.Kamel	65
gwylim	65
Nooodles	55
amrSamir	55
Blazerfrost	55
naguib	55
Kosie	55
mRefaat88	55

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

Online Competition Africa 2010

A. Odd Man Out

B. Get to Work

C. Qualification Round

D. Polygraph

Contest Analysis

Questions asked 1

Submissions

Odd Man Out

7pt	Not attempted **209/214 users** correct (98%)
7pt	Not attempted **206/209 users** correct (99%)

Get to Work

9pt	Not attempted **127/149 users** correct (85%)
9pt	Not attempted **124/127 users** correct (98%)

Qualification Round

11pt	Not attempted **47/87 users** correct (54%)
22pt	Not attempted **4/32 users** correct (13%)

Polygraph

12pt	Not attempted **14/30 users** correct (47%)
23pt	Not attempted **0/2 users** correct (0%)

Top Scores

RalfKistner	77
mohamedafattah	65
Ahmed.Kamel	65
gwylim	65
Nooodles	55
amrSamir	55
Blazerfrost	55
naguib	55
Kosie	55
mRefaat88	55

Problem C. Qualification Round

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 11 points	Solve C-small
Large input 22 points	Solve C-large

Problem

You've just advanced from the Qualification Round of Google Code Jam Africa 2010, and you want to know how many of your fellow contestants advanced with you. To give yourself a challenge, you've decided only to look at how many people solved each problem.

The Qualification Round consisted of **P** problems; the i^{th} problem was fully solved by $\mathbf{S_i}$ contestants. Contestants had to solve **C** problems in order to advance to the next round. Your job is to figure out, using only that information, the maximum number of contestants who could have advanced.

Input

The first line of the input gives the number of test cases, **T**. T lines follow. Each will consist only of space-separated integers: first **P**, then **C**, then P integers $\mathbf{S_0 ... S_{P-1}}$.

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 maximum number of contestants who could have advanced (in other words, the maximum number of contestants who could have solved at least **C** problems).

Limits

$1 \le T \le 100$
$1 \le C \le P$

Small dataset

$1 \le P \le 6$
$0 \le S_i \le 1000$

Large dataset

$1 \le P \le 60$
$0 \le S_i \le 10^{17}$

Sample

Input	Output

`2`
`2 2 73 100`
`3 2 245 272 238`
	`Case #1: 73`
	`Case #2: 377`

Powered by

Google Cloud Platform

code jam
printf("hello, africa!");

Practice Mode

Contest scoreboard | Sign in

Online Competition Africa 2010

A. Odd Man Out

B. Get to Work

C. Qualification Round

D. Polygraph

Contest Analysis

Questions asked 1

− Submissions

Odd Man Out

7pt	Not attempted **209/214 users** correct (98%)
7pt	Not attempted **206/209 users** correct (99%)

Get to Work

9pt	Not attempted **127/149 users** correct (85%)
9pt	Not attempted **124/127 users** correct (98%)

Qualification Round

11pt	Not attempted **47/87 users** correct (54%)
22pt	Not attempted **4/32 users** correct (13%)

Polygraph

12pt	Not attempted **14/30 users** correct (47%)
23pt	Not attempted **0/2 users** correct (0%)

− Top Scores

RalfKistner	77
mohamedafattah	65
Ahmed.Kamel	65
gwylim	65
Nooodles	55
amrSamir	55
Blazerfrost	55
naguib	55
Kosie	55
mRefaat88	55

Problem D. **Polygraph**

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 D-small
Large input 23 points	Solve D-large

Polygraph

On the distant isle of Googlia, there are two cities, Truthtown and Liarville. People from Truthtown always tell the truth and people from Liarville always lie. While exploring Googlia, you have run across a group of N inhabitants, and you want to figure out which city each one came from.

To make life simpler, you begin by numbering these people 1 through N. You then question each person, and record their M statements in the short-hand described below.

Short-hand	Meaning
`i T j`	Person #i says, "Person #j is from Truthtown."
`i L j`	Person #i says, "Person #j is from Liarville."
`i S j k`	Person #i says, "Persons #j and #k are from the same city."
`i D j k`	Person #i says, "Persons #j and #k are from different cities."

Your task is to deduce which city each person came from. It is guaranteed that there will always be at least one solution.

For example, suppose you were given the following statements:

`1 D 2 3`, `1 D 2 4`, `1 D 3 4`, and `2 L 1`.

Then, you could reason as follows:

- There are only two cities, so persons #2, #3, and #4 could not all have come from different cities.
- Therefore, at least one of person #1's claims must have been a lie.
- Therefore, person #1 is from Liartown, and all of his claims must have been lies.
- Therefore, persons #2, #3, and #4 must all be from the same city.
- Person #2's claim is true, so he must be from Truthtown.
- Therefore, persons #3 and #4 are also from Truthtown.

Input

The first line of the input gives the number of test cases, **T**. **T** test cases follow. Each case begins with a line containing the integers **N** and **M**. The following **M** lines each contain a single statement from one inhabitant, formatted as described above.

Output

For each test case, output one line containing "Case #x: y_1 y_2 ... y_N", where x is the case number (starting from 1) and y_i is a single letter indicating which city person #i is from:

- If the statements you have been given imply person #i must be from Truthtown, then y_i should be 'T'.
- If the statements you have been given imply person #i must be from Liarville, then y_i should be 'L'.
- If the statements you have been given are not enough information to determine where person #i is from, then y_i should be '-'.

Limits

$1 \leq T \leq 100$
$1 \leq i, j, k \leq N$
j and k are distinct

Small dataset

$1 \leq N \leq 10$
$1 \leq M \leq 500$

Large dataset

1 ≤ N ≤ 500
1 ≤ M ≤ 500

Sample

```
Input       Output

2           Case #1: L T T T
4 4         Case #2: - T -
1 D 2 3
1 D 2 4
1 D 3 4
2 L 1
3 1
1 S 1 2
```