apac test
powered by Google Code Jam

Practice Mode

Contest scoreboard | Sign in

Round A APAC Test 2016

**A. Googol String**

B. gCube

C. gCampus

D. gSnake

Questions asked

Submissions

Googol String

| | |
|---|---|
| 7pt | Not attempted<br>**2083/5209 users** correct (40%) |
| 12pt | Not attempted<br>**957/1730 users** correct (55%) |

gCube

| | |
|---|---|
| 8pt | Not attempted<br>**1557/2234 users** correct (70%) |
| 16pt | Not attempted<br>**855/1488 users** correct (57%) |

gCampus

| | |
|---|---|
| 10pt | Not attempted<br>**493/1232 users** correct (40%) |
| 15pt | Not attempted<br>**227/482 users** correct (47%) |

gSnake

| | |
|---|---|
| 13pt | Not attempted<br>**121/629 users** correct (19%) |
| 19pt | Not attempted<br>**41/88 users** correct (47%) |

Top Scores

| | |
|---|---|
| cebrusfs | 100 |
| sgtlaugh | 100 |
| usaxena95 | 100 |
| akovski | 100 |
| NAFIS | 100 |
| liuyibo1994 | 100 |
| dtyfc | 100 |
| Legendks | 100 |
| Shaon | 100 |
| jki14 | 100 |

# Problem A. **Googol String**

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<br>7 points | Solve A-small |
| Large input<br>12 points | Solve A-large |

## Problem

A "0/1 string" is a string in which every character is either `0` or `1`. There are two operations that can be performed on a 0/1 string:

- **switch**: Every `0` becomes `1` and every `1` becomes `0`. For example, "100" becomes "011".
- **reverse**: The string is reversed. For example, "100" becomes "001".

Consider this infinite sequence of 0/1 strings:

$S_0$ = ""

$S_1$ = "0"

$S_2$ = "001"

$S_3$ = "0010011"

$S_4$ = "001001100011011"

...

$S_N = S_{N-1}$ + "0" + **switch**(**reverse**($S_{N-1}$)).

You need to figure out the Kth character of $S_{googol}$, where googol = $10^{100}$.

## Input

The first line of the input gives the number of test cases, **T**. Each of the next **T** lines contains a number **K**.

## Output

For each test case, output one line containing "Case #x: y", where x is the test case number (starting from 1) and y is the **K**th character of $S_{googol}$.

## Limits

$1 \le \mathbf{T} \le 100$.

### Small dataset

$1 \le \mathbf{K} \le 10^5$.

### Large dataset

$1 \le \mathbf{K} \le 10^{18}$.

## Sample

```
Input      Output

4          Case #1: 0
1          Case #2: 0
2          Case #3: 1
3          Case #4: 0
10
```

apac test
powered by Google Code Jam

Practice Mode

Contest scoreboard | Sign in

Round A APAC Test 2016

- A. Googol String
- **B. gCube**
- C. gCampus
- D. gSnake

Questions asked

Submissions

Googol String

| | |
|---|---|
| 7pt | Not attempted **2083/5209 users** correct (40%) |
| 12pt | Not attempted **957/1730 users** correct (55%) |

gCube

| | |
|---|---|
| 8pt | Not attempted **1557/2234 users** correct (70%) |
| 16pt | Not attempted **855/1488 users** correct (57%) |

gCampus

| | |
|---|---|
| 10pt | Not attempted **493/1232 users** correct (40%) |
| 15pt | Not attempted **227/482 users** correct (47%) |

gSnake

| | |
|---|---|
| 13pt | Not attempted **121/629 users** correct (19%) |
| 19pt | Not attempted **41/88 users** correct (47%) |

Top Scores

| | |
|---|---|
| cebrusfs | 100 |
| sgtlaugh | 100 |
| usaxena95 | 100 |
| akovski | 100 |
| NAFIS | 100 |
| liuyibo1994 | 100 |
| dtyfc | 100 |
| Legendks | 100 |
| Shaon | 100 |
| jki14 | 100 |

## Problem B. gCube

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<br>8 points | Solve B-small |
| Large input<br>16 points | Solve B-large |

### Problem

Googlers are very interested in cubes, but they are bored with normal three-dimensional cubes and also want to think about other kinds of cubes! A "D-dimensional cube" has D dimensions, all of equal length. (D may be any positive integer; for example, a 1-dimensional cube is a line segment, and a 2-dimensional cube is a square, and a 4-dimensional cube is a hypercube.) A "D-dimensional cuboid" has D dimensions, but they might not all have the same lengths.

Suppose we have an **N**-dimensional cuboid. The **N** dimensions are numbered in order (0, 1, 2, ..., N - 1), and each dimension has a certain length. We want to solve many subproblems of this type:

1. Take all consecutive dimensions between the $\mathbf{L_i}$-th dimension and $\mathbf{R_i}$-th dimension, inclusive.

2. Use those dimensions to form a D-dimensional cuboid, where D = $\mathbf{R_i}$ - $\mathbf{L_i}$ + 1. (For example, if $\mathbf{L_i}$ = 3 and $\mathbf{R_i}$ = 6, we would form a 4-dimensional cuboid using the 3rd, 4th, 5th, and 6th dimensions of our **N**-dimensional cuboid.)

3. Reshape it into a D-dimensional cube **that has exactly the same volume as that D-dimensional cuboid**, and find the edge length of that cube.

Each test case will have **M** subproblems like this, all of which use the same original **N**-dimensional cuboid.

### Input

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

Each test case begins with two integers **N** and **M**; **N** is the number of dimensions and **M** is the number of queries. Then there is one line with **N** positive integers $\mathbf{a_i}$, which are the lengths of the dimensions, in order. Then, **M** lines follow. In the ith line, there are two integers $\mathbf{L_i}$ and $\mathbf{R_i}$, which give the range of dimensions to use for the ith subproblem.

### Output

For each test case, output one line containing "Case #x:", where x is the test case number (starting from 1). After that, output **M** lines, where the ith line has the edge length for the ith subproblem. An edge length will be considered correct if it is within an absolute error of $10^{-6}$ of the correct answer. See the FAQ for an explanation of what that means, and what formats of real numbers we accept.

### Limits

$1 \le \mathbf{T} \le 100$.
$1 \le \mathbf{a_i} \le 10^9$.
$0 \le \mathbf{L_i} \le \mathbf{R_i} < \mathbf{N}$.

### Small dataset

$1 \le \mathbf{N} \le 10$.
$1 \le \mathbf{M} \le 10$.

### Large dataset

$1 \le \mathbf{N} \le 1000$.
$1 \le \mathbf{M} \le 100$.

### Sample

```
Input      Output

2          Case #1:
2 2        1.000000000
1 4        2.000000000
0 0        Case #2:
```

```
0 1    1.414213562
3 2    2.449489743
1 2 3
0 1
1 2
```

Round A APAC Test 2016

A. Googol String
B. gCube
**C. gCampus**
D. gSnake

Questions asked

Submissions

| | |
|---|---|
| Googol String | |
| 7pt | Not attempted **2083/5209 users** correct (40%) |
| 12pt | Not attempted **957/1730 users** correct (55%) |
| gCube | |
| 8pt | Not attempted **1557/2234 users** correct (70%) |
| 16pt | Not attempted **855/1488 users** correct (57%) |
| gCampus | |
| 10pt | Not attempted **493/1232 users** correct (40%) |
| 15pt | Not attempted **227/482 users** correct (47%) |
| gSnake | |
| 13pt | Not attempted **121/629 users** correct (19%) |
| 19pt | Not attempted **41/88 users** correct (47%) |

Top Scores

| | |
|---|---|
| cebrusfs | 100 |
| sgtlaugh | 100 |
| usaxena95 | 100 |
| akovski | 100 |
| NAFIS | 100 |
| liuyibo1994 | 100 |
| dtyfc | 100 |
| Legendks | 100 |
| Shaon | 100 |
| jki14 | 100 |

# Problem C. gCampus

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<br>10 points | Solve C-small |
| Large input<br>15 points | Solve C-large |

## Problem

Company G has a main campus with **N** offices (numbered from 0 to **N** - 1) and **M** bidirectional roads (numbered from 0 to **M** - 1). The ith road connects a pair of offices ($U_i$, $V_i$), and it takes $C_i$ minutes to travel on it (in either direction).

A path between two offices X and Y is a series of one or more roads that starts at X and ends at Y. The time taken to travel a path is the sum of the times needed to travel each of the roads that make up the path. (It's guaranteed that there is at least one path connecting any two offices.)

Company G specializes in efficient transport solutions, but the CEO has just realized that, embarrassingly enough, its own road network may be suboptimal! She wants to know which roads in the campus are *inefficient*. A road is inefficient if and only if it is not included in **any** shortest paths between **any** offices.

Given the graph of offices and roads, can you help the CEO find all of the inefficient roads?

## Input

The first line of the input gives the number of test cases, **T**. **T** test cases follow. Each case begins with one line with two integers **N** and **M**, indicating the number of offices and roads. This is followed by **M** lines containing three integers each: $\mathbf{U_i}$, $\mathbf{V_i}$ and $\mathbf{C_i}$, indicating the ith road is between office $\mathbf{U_i}$ and office $\mathbf{V_i}$, and it takes $\mathbf{C_i}$ minutes to travel on it.

## Output

For each test case, output one line containing "Case #x:", where x is the test case number (starting from 1). Then output the road numbers of all of the inefficient roads, in increasing order, each on its own line. (Note that road 0 refers to the first road listed in a test case, road 1 refers to the second road, etc.)

## Limits

$0 < \mathbf{C_i} \leq 1000000$.

### Small dataset

$1 \leq \mathbf{T} \leq 10$.
$1 \leq \mathbf{N} = \mathbf{M} \leq 100$.

### Large dataset

$1 \leq \mathbf{T} \leq 3$.
$1 \leq \mathbf{N} \leq 100$.
$1 \leq \mathbf{M} \leq 10000$.

## Sample

```
Input         Output

2             Case #1:
3 3           0
0 1 10        Case #2:
1 2 3
2 0 3
3 3
0 1 10
1 2 3
2 1 3
```

Round A APAC Test 2016

A. Googol String

B. gCube

C. gCampus

**D. gSnake**

Questions asked

Submissions

Googol String

| | |
|---|---|
| 7pt | Not attempted<br>**2083/5209 users** correct (40%) |
| 12pt | Not attempted<br>**957/1730 users** correct (55%) |

gCube

| | |
|---|---|
| 8pt | Not attempted<br>**1557/2234 users** correct (70%) |
| 16pt | Not attempted<br>**855/1488 users** correct (57%) |

gCampus

| | |
|---|---|
| 10pt | Not attempted<br>**493/1232 users** correct (40%) |
| 15pt | Not attempted<br>**227/482 users** correct (47%) |

gSnake

| | |
|---|---|
| 13pt | Not attempted<br>**121/629 users** correct (19%) |
| 19pt | Not attempted<br>**41/88 users** correct (47%) |

Top Scores

| | |
|---|---|
| cebrusfs | 100 |
| sgtlaugh | 100 |
| usaxena95 | 100 |
| akovski | 100 |
| NAFIS | 100 |
| liuyibo1994 | 100 |
| dtyfc | 100 |
| Legendks | 100 |
| Shaon | 100 |
| jki14 | 100 |

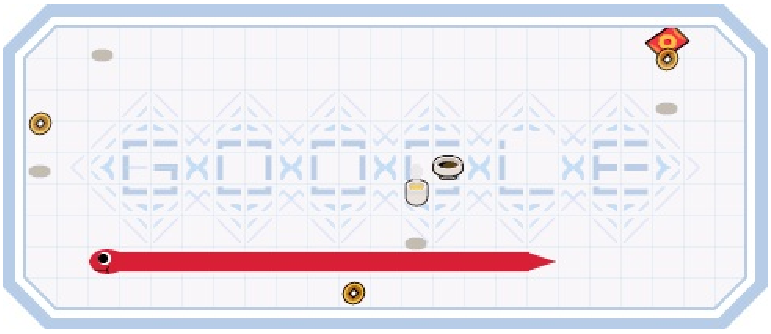
## Problem D. gSnake

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<br>13 points | Solve D-small |
| Large input<br>19 points | Solve D-large |

### Problem

Alex is a huge fan of the Snake game.



**Note:** This Google Doodle does not exactly match the rules of the Snake game we will describe below. It is only intended to give you a general idea of what the game looks like.

Alex just learned how to program and wants to develop his own version of Snake, with the following rules:

- The game board has **R** rows and **C** columns. The top left cell of the board has coordinates (1, 1), and the bottom right cell has coordinates (**R**, **C**).
- At the start of the game, in every cell with coordinates (r, c) such that r + c is odd, there is one piece of food. No other cells have food.
- The snake's body is always an ordered, connected sequence of one or more cells on the board. The first cell of the sequence is called the "head" of the snake. The second cell (if any) shares an edge (not just a corner) with the first cell, and so on. The last cell in the sequence is called the "tail" of the snake.
- The snake's head is always facing in one of four directions: left, up, right, or down.
- At the start of the game, the snake is at cell (1, 1) and has a length of one (that is, the snake consists of only a head), and the head faces right.
- At each integer time (1 second, 2 seconds, etc.), the head of the snake will move into the adjacent cell that its head is facing toward. The board is **cyclic**, i.e., trying to move off an edge will cause the head to appear on the opposite edge of the board. For example, if the snake is at (1, **C**) and its head is facing right, the head will next move to (1, 1). If the snake is at (1, **C**) and its head is facing up, the head will next move to (**R**, **C**).
- When the snake's head moves into a cell with no food, the snake does not grow. The snake's second cell (if any) moves to the place where the snake's head was, the snake's third cell (if any) moves to the place where the second cell was, and so on.
- When the snake's head moves into a cell with a piece of food, it eats the food (meaning that cell no longer has food), and grows its body. A new head is created in the cell where the food was. The cell that was the snake's head becomes the snake's second cell, the cell that was the snake's second cell (if any) becomes the snake's third cell, and so on.
- If, after a move is complete, the snake's head is in the same place as one of another of its cells, the snake dies and the game ends immediately. (Note that if the snake's head moves toward a cell where its tail was, the game will not end, because the tail will move out of the way before the move is complete.)
- In the game, the player can let the snake perform some turn actions. Each action $A_i$ will happen between the $T_i$th and $T_i$+1 th seconds. There are two possible actions: "L" and "R". An "L" action will turn the head 90 degrees to the left, so, for example, if the snake had been facing down before, it would face right after. An "R" action will turn the head 90 degrees to the right, so, for example, if the snake had been facing down before, it would face left after.
- The game has a time limit: it will end after the move on the $\mathbf{10^9}$th second is complete (if the game has even gone on that long!)

To test the game, Alex has written a series of TURN actions. Your task is to simulate that series of actions, and tell Alex the final length of the snake when the game is over. Remember that the game can end either because the snake's head and another cell of its body are in the same place after a move is complete, or because time runs out. In the former case, you should count both the head and the overlapping cell of its body as two separate cells, for the purpose of determining length.

### Input

The first line of the input gives the number of test cases, **T**. **T** test cases follow. Each test cases starts with three integers **S**, **R**, and **C**, where **S** gives the number of turn actions and **R** and **C** represent the number of rows and columns of the board. **S** lines follow; the ith of these lines has an integer $X_i$, then a character $A_i$ that is either L or R. Each of these lines corresponds to performing an action between $X_i$th and $X_i$+1 th seconds. It's guaranteed that the actions are given in time order and there will never be more than one action between the same two seconds. However, you should note that the game may end before the snake gets to execute all of these actions.

### Output

For each test case, output one line containing "Case #x: y", where x is the test case number (starting from 1) and y is the length of the snake when the game is over.

### Limits

1 ≤ **T** ≤ 10.

### Small dataset

1 ≤ **R, C** ≤ 100;
1 ≤ **S** ≤ 100;
1 ≤ $X_i$ ≤ 2000.

### Large dataset

1 ≤ **R, C** ≤ 100000;
1 ≤ **S** ≤ 100000;
1 ≤ $X_i$ ≤ 1000000.

### Sample

```
Input      Output

2          Case #1: 3
3 3 3      Case #2: 5
1 R
2 L
3 R
5 3 3
2 R
4 R
6 R
7 R
8 R
```