



```
System.out.println("hello, world!");
```

Practice Mode

AmrKeleg94@gmail.com | [Contest scoreboard](#) | [Sign out](#)

Round 1B 2008

[A. Crop Triangles](#)[B. Number Sets](#)**C. Mousetrap**[Contest Analysis](#)[Questions asked](#) 3

- Submissions

Crop Triangles

5pt	Not attempted 1445/2197 users correct (66%)
10pt	Not attempted 457/1287 users correct (36%)

Number Sets

10pt	Not attempted 777/1351 users correct (58%)
25pt	Not attempted 100/448 users correct (22%)

Mousetrap

15pt	Not attempted 610/862 users correct (71%)
35pt	Not attempted 95/387 users correct (25%)

- Top Scores

mystic	100
nika	100
bmerry	100
dgozman	100
ilyaraz	100
misof	100
tourist	100
vlad89	100
lordmonsoon	100
falagar	100

Problem C. Mousetrap

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
15 points

Solve C-small

Large input
35 points

Solve C-large

Problem

Mousetrap is a simple card game for one player. It is played with a shuffled deck of cards numbered 1 through K , face down. You play by revealing the top card of the deck and then putting it on the bottom of the deck, keeping count of how many cards you have revealed. If you reveal a card whose number matches the current count, remove it from the deck and reset the count. If the count ever reaches $K+1$, you have lost. If the deck runs out of cards, you win.

Suppose you have a deck of 5 cards, in the order 2, 5, 3, 1, 4. You will reveal the 2 on count 1, the 5 on count 2, then the 3 on count 3. Since the value matches the count, you remove the 3 from the deck, and reset the count. You now have 4 cards left in the order 1, 4, 2, 5. You then reveal the 1 on count 1, and remove it as well (you're doing great so far!). Continuing in this way you will remove the 2, then the 4, and then finally the 5 for victory.

You would like to set up a deck of cards in such a way that you will win the game and remove the cards in increasing order. We'll call a deck organized in this way "perfect." For example, with 4 cards you can organize the deck as 1, 4, 2, 3, and you will win by removing the cards in the order 1, 2, 3, 4.

Input

The first line of input gives the number of cases, T . Each test case starts with a line containing K , the number of cards in a deck. The next line starts with an integer n , which is followed by n integers (d_1, d_2, \dots), indices into the deck.

Output

For each test case, output one line containing "Case #x: " followed by n integers (k_1, k_2, \dots), where k_i is the value of the card at index d_i of a perfect deck of size K . The numbers in the output should be separated by spaces, and there must be at least one space following the colon in each "Case #x:" line.

Limits**Small dataset**
 $T = 100, 1 \leq K \leq 5000, 1 \leq n \leq 100, 1 \leq d_i \leq K.$
Large dataset
 $T = 10, 1 \leq K \leq 1000000, 1 \leq n \leq 100, 1 \leq d_i \leq K.$
Sample

Input	Output
2	Case #1: 1 3 2 5 4
5	Case #2: 2 8 13 4
5 1 2 3 4 5	
15	
4 3 4 7 10	

All problem statements, input data and contest analyses are licensed under the [Creative Commons Attribution License](#).

© 2008-2015 Google [Google Home](#) - [Terms and Conditions](#) - [Privacy Policies and Principles](#)

