Round D APAC Test 2016

A. Dynamic Grid

B. gBalloon

C. IP Address Summarization

D. Virtual Rabbit

Questions asked 1

Submissions

Dynamic Grid

6pt | Not attempted **1392/1881 users** correct (74%)

8pt | Not attempted **1288/1368 users** correct (94%)

gBalloon

9pt | Not attempted **353/666 users** correct (53%)

17pt | Not attempted **266/338 users** correct (79%)

IP Address Summarization

10pt | Not attempted **123/236 users** correct (52%)

19pt | Not attempted **73/118 users** correct (62%)

Virtual Rabbit

11pt | Not attempted **18/166 users** correct (11%)

20pt | Not attempted **3/8 users** correct (38%)

Top Scores

nhho	100
sundar95	80
Shaon	80
ajkrish95	80
ojas.deshpande	80
NAFIS	69
JunoYu	69
wcwswsws	69
karanaggarwal	69
VotBear	69

Problem C. IP Address Summarization

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 10 points	Solve C-small
Large input 19 points	Solve C-large

Problem

An IP (Internet Protocol) address is a number that is assigned to each device on the Internet. At the time being, most devices use version four of this protocol (IPv4). An IPv4 address is a 32-bit string. IPv4 addresses are normally represented in dot-decimal notation, which consists of four decimal numbers called octets, each ranging from 0 to 255 (inclusive), separated by dots, e.g., 172.16.254.1. Each octet represents a group of 8 bits (one byte) of the address. The first 8 bits of the string (when interpreted as an unsigned integer, with the most significant bit first) form the first octet, the next 8 bits form the second octet, and so on.

An IP subnet addresses is used to represent a group of devices that belong to the same network. IP subnet addresses are expressed in the format of an IP address, followed by a slash and then a prefix length ranging from 0 to 32. A subnet address stands for all IP addresses that have the same first P bits of the given address, where P is the prefix length. For example 10.8.0.0/9 represents 2^{23} addresses that all have `000010100` (the first nine bits of 10.8.0.0) as their first 9 bits, that is, 10.0.0.0 through 10.127.255.255. Note that 10.8.0.0/9 and, for example, 10.0.0.0/9 (or any other address within the subnet) would be equivalent ways to refer to the same subnet, because those addresses start with the same nine bits.

A subnet is *normalized* if the bits of the address other than the prefix are all zeroes. For example, 10.8.1.0/24 and 10.8.1.2/24 represent the same subnet, but 10.8.1.0/24 is normalized. The normalization of 255.255.255.255/13 is 255.248.0.0/13.

You will be given a list of subnet addresses, and you must output the shortest ordered list of subnets such that all the addresses are normalized and an address belongs to some subnet in the input if and only if it belongs to some subnet in the output.

Input

The first line of the input gives the number of test cases, **T**. **T** test cases follow. Each begins with one line with an integer **N**, the number of subnets, and is followed by **N** more lines, each of which has a subnet addresses. Each subnet address is of the form `A.B.C.D/P`, where A, B, C, and D are integers from 0 to 255, inclusive, and P is an integer from 0 to 32, inclusive. No integer (apart from 0) has leading zeroes.

Output

For each test case, output one line containing "Case #x:", where x is the test case number (starting from 1). Then output a list of subnet addresses, one per line, meeting the conditions described above. These addresses must be normalized and must be ordered. An address X comes before another address Y if X's first integer is smaller than Y's first integer, or if X and Y have the same first integer but X's second integer is smaller than Y's second integer, and so on.

Note that the requirements of the problem guarantee that there is a single unique answer for each test case.

Limits

$1 \le \mathbf{T} \le 50$.

Small dataset

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

Large dataset

$1 \le \mathbf{N} \le 10000$.

Sample

Input	Output
3	Case #1:

```
2                       10.0.0.0/8
10.1.2.3/8              Case #2:
10.2.3.4/17             10.0.0.0/8
2                       Case #3:
10.2.3.4/9              52.22.138.0/23
10.128.2.3/9            52.56.134.128/26
10                      52.227.80.0/22
224.147.224.186/18      58.32.0.0/12
58.45.85.53/14          83.248.0.0/13
52.56.134.139/26        224.147.192.0/18
52.227.82.227/22        238.223.58.128/27
83.250.251.44/13
83.250.12.64/16
58.40.52.11/14
52.22.138.56/23
238.223.58.151/27
58.32.52.11/13
```