



# 北京大学

## PEKING UNIVERSITY

### JUDGE ONLINE FOR ACM/ICPC



Online Judge	Problem Set	Authors	Online Contests	User
<a href="#">Web Board</a> <a href="#">Home Page</a> <a href="#">F.A.Qs</a> <a href="#">Statistical Charts</a>	<a href="#">Problems</a> <a href="#">Submit Problem</a> <a href="#">Online Status</a> Prob.ID:	<a href="#">Register</a> <a href="#">Update your info</a> <a href="#">Authors ranklist</a>	<a href="#">Current Contest</a> <a href="#">Past Contests</a> <a href="#">Scheduled Contests</a> <a href="#">Award Contest</a>	User ID: Password: <a href="#">Register</a>

## STAMPS

Time Limit: 1000MS

Memory Limit: 10000K

Total Submissions: 20484

Accepted: 6060

Language: Default

## Description

Have you done any Philately lately?

You have been hired by the Ruritanian Postal Service (RPS) to design their new postage software. The software allocates stamps to customers based on customer needs and the denominations that are currently in stock.

Ruritania is filled with people who correspond with stamp collectors. As a service to these people, the RPS asks that all stamp allocations have the maximum number of different types of stamps in it. In fact, the RPS has been known to issue several stamps of the same denomination in order to please customers (these count as different types, even though they are the same denomination). The maximum number of different types of stamps issued at any time is twenty-five.

To save money, the RPS would like to issue as few duplicate stamps as possible (given the constraint that they want to issue as many different types). Further, the RPS won't sell more than four stamps at a time.

## Input

The input for your program will be pairs of positive integer sequences, consisting of two lines, alternating until end-of-file. The first sequence are the available values of stamps, while the second sequence is a series of customer requests. For example:

```
1 2 3 0 ; three different stamp types
7 4 0 ; two customers
1 1 0 ; a new set of stamps (two of the same type)
6 2 3 0 ; three customers
```

Note: the comments in this example are *\*not\** part of the data file; data files contain only integers.

## Output

For each customer, you should print the "best" combination that is exactly equal to the customer's needs, with a maximum of four stamps. If no such combination exists, print "none".

The "best" combination is defined as the maximum number of different stamp types. In case of a tie, the combination with the fewest total stamps is best. If still tied, the set with the highest single-value stamp is best. If there is still a tie, print "tie".

For the sample input file, the output should be:

```
7 (3): 1 1 2 3
4 (2): 1 3
6 ---- none
2 (2): 1 1
3 (2): tie
```

That is, you should print the customer request, the number of types sold and the actual stamps. In case of no legal allocation, the line should look like it does in the example, with four hyphens after a space. In the case of a tie, still print the number of types but do not print the allocation (again, as in the example). Don't print extra blank at the end of each line.

## Sample Input

```
1 2 3 0 ; three different stamp types
7 4 0 ; two customers
1 1 0 ; a new set of stamps (two of the same type)
6 2 3 0 ; three customers
```

## Sample Output

```
7 (3): 1 1 2 3
4 (2): 1 3
6 ---- none
2 (2): 1 1
3 (2): tie
```

## Source

Pacific Northwest 1998

[\[Submit\]](#) [\[Go Back\]](#) [\[Status\]](#) [\[Discuss\]](#)

 [Home Page](#)  [Go Back](#)  [To top](#)

All Rights Reserved 2003-2013 Ying Fuchen,Xu Pengcheng,Xie Di  
Any problem, Please [Contact Administrator](#)