

Validating & Parsing Email Addresses

You are given N names and email addresses. Your task is to print the names and email addresses if they are valid.

A valid email address follows the rules below:

- Email must have three basic components: *username* @ *website name* . *extension*.
- The *username* can contain: alphanumeric characters, -, . and _.
- The *username* must start with an English alphabet character.
- The *website name* contains only English alphabet characters.
- The *extension* contains only English alphabet characters, and its length can be 1, 2, or 3.

Input Format

name <example@email.com>

The first line contains an integer N .

The next N lines contains a *name* and an *email address* separated by a space.

Constraints

$$0 < N < 100$$

Output Format

Print the valid email addresses only. Print the space separated *name* and *email address* on separate lines. Output the valid results in order of their occurrence.

Sample Input

```
2
DEXTER <dexter@hotmail.com>
VIRUS <virus!@variable.:p>
```

Sample Output

```
DEXTER <dexter@hotmail.com>
```

Explanation

dexter@hotmail.com
This is a valid email address.

virus!@variable.:p
This is invalid because it contains a ! in the username and a : in the extension.

Bonus

Email.utils()

```
>>> import email.utils
>>> print email.utils.parseaddr('DOSHI <DOSHI@hackerrank.com>')
```

```
('DOSHI', 'DOSHI@hackerrank.com')  
>>> print email.utils.formataddr(('DOSHI', 'DOSHI@hackerrank.com'))  
DOSHI <DOSHI@hackerrank.com>
```