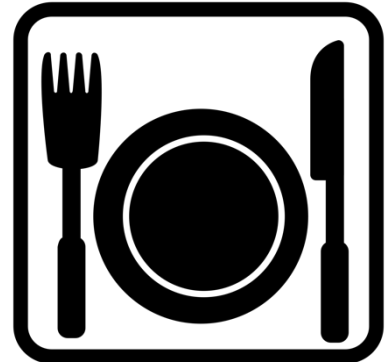


Restaurant

Andica is the owner of a famous restaurant called “The Crunchy Chicken”. His restaurant serves “drunken chicken” as its best-selling menu, attracting many customers because of its peculiar name and taste. In his restaurant, he wants everyone to have a good time, maybe for a drink or two after their meals. In short, his restaurant is very successful and is well-known throughout all parts of Singapore.



Andica is very old-fashioned. His restaurant does not take in any reservations and serves only walk-in customers. However, to cope with technological advancements, Andica wants to build a simple electronic restaurant program (ERP) which can help him allocate tables to customers. He thinks that everyone will love ERP (at least those who work in the restaurant will appreciate its existence) and wants it to be deployed in his restaurant as soon as possible.

Too busy with his restaurant work, Andica approaches you, the best programmer in the world (at least according to him), to build this simple ERP. As a start, he sent you an email with the specification of his desired ERP. You open the email and see a detailed explanation of the desired system. In the restaurant, there are many tables, each with its own unique name and groups of people come (and go) to the restaurant. All groups have a specific number of people and each group has a unique group name. Each table can only be occupied by one group at a time and all group members of a particular group must be placed on the same table.

The ERP system takes in 5 (five) types of operations:

1. Allocate a group consisting of a number of people to a specific table. If the table is not available or the table is not suitable for the group (i.e. too small), the group will simply leave the restaurant.
2. Allocate a group of people to any (available) table. If there are no available (and suitable) tables, the group will simply leave the restaurant.
3. Clear a table that is occupied by a certain group, i.e. that group leaves the restaurant.
4. Prints the table name of a specific group.
5. Prints the name of a group that occupies a specific table.

Without wasting too much time, you start working immediately to finish this ERP. This ERP is crucial for Andica to maximise his profits, so it is urgent that the system must be deployed as soon as possible. Furthermore, the system must be robust, i.e. it will never break down. You are feeling great and confident about getting this done quickly and you start coding immediately. Good luck!

Input

The first line of input consists of a single integer **N** ($1 \leq N \leq 100$). **N** rows follow. In each of the **N** rows, there are two values separated by a single space. It contains a table name followed by its capacity. All table names are guaranteed to be single words and consist of lowercase letters with capacities guaranteed to fit in a 32-bit integer. It is guaranteed that the group size and table capacity do not exceed 10000.

The following line contains a single integer **Q** ($1 \leq Q \leq 100$). **Q** rows follow. Each of the **Q** rows contains queries that are acceptable by the ERP. The queries will follow the following specification:

Query Type Input Format: <QUERY_TYPE> <APPROPRIATE_PARAMETERS>

1. **1 GROUP_NAME GROUP_SIZE TABLE_NAME**
A group with the name **GROUP_NAME** of size **GROUP_SIZE** arrives at the restaurant and they request to be seated at the table **TABLE_NAME**. If that group can be seated at the table **TABLE_NAME**, print **TABLE_NAME**. Otherwise, print “not possible”.
2. **2 GROUP_NAME GROUP_SIZE**
A group with the name **GROUP_NAME** of size **GROUP_SIZE** arrives at the restaurant. If that group can be seated at any suitable table, print **TABLE_NAME**. Otherwise, print “not possible”. The table that you allocate should be the lexicographically smallest table that is available and suitable for the group.
3. **3 GROUP_NAME**
A group with the name **GROUP_NAME** leaves the restaurant. If there are no groups with the name **GROUP_NAME** inside the restaurant, the system does not do anything.
4. **4 GROUP_NAME**
Print the name of the table that the group **GROUP_NAME** is currently assigned to. If there are no groups with the name **GROUP_NAME** inside the restaurant, print “invalid”.
5. **5 TABLE_NAME**
Print the name of the group that currently occupies the table **TABLE_NAME**. If the table **TABLE_NAME** is currently empty or there are no tables with the name **TABLE_NAME**, print “invalid”.

All group and table names are guaranteed to be at most 20 characters.

Output

Print according to the specification of the queries given above.

Sample Input

```
4
a 5
b 4
c 1
d 2
9
1 ann 3 a
2 bob 2
2 charles 2
1 donny 2 c
4 ann
5 a
3 ann
1 eames 5 a
4 random
```

Sample Output

```
a
b
d
not possible
a
ann
a
invalid
```