

In this task your goal is to implement a simple phone book manager. It should be able to process the following types of user's queries:

- **add *number name*** - It means that the user adds a person with name and number to the phone book. If there exists a user with such number already, then your manager has to overwrite the corresponding name.
- **del *number*** - It means that the manager should erase a person with number from the phone book. If there is no such person, then it should just ignore the query.
- **find *number*** - It means that the user looks for a person with phone number. The manager should reply with the appropriate name, or with string "not found" (without quotes) if there is no such person in the book.

Input

There is a single integer **N** ($1 \leq N \leq 10^5$) in the first line | the number of queries. **N** lines follow it; each of them contains one query in the format described above.

All phone numbers consist of decimal digits, they don't have leading zeros, and each of them has no more than 7 digits. All names are non-empty strings of Latin letters, and each of them has length at most 15. It's guaranteed that there is no person with name "not found".

Output

Print the results of each find query – the name corresponding to the phone number or "not found" (without quotes) if there is no person in the phone book with such phone number. Output one result per line in the same order as the find queries are given in the input.

Sample Input

```
12
add 911 police
add 76213 Mom
add 17239 Bob
find 76213
find 910
find 911
del 910
del 911
```

```
find 911
find 76213
add 76213 daddy
find 76213
```

Sample Output

```
Mom
not found
police
not found
Mom
daddy
```