Specification of 2.3(4)

In this homework, you need to design the data structure and implement the four actions described in `hw2.pdf`. Please be aware that the data set is super-huge—about 7.7G. Thus, if you are not careful with your implementations, your program can easily crash. Also, **DO NOT copy the data set to your own directory**. Your own directory is on the NFS system and copying it there would only slow down your program (and other users' programs).

The input/output/data formats are in the following sections. There is no sample output this time and you'll need to validate your code by yourself.

Data Format

- UserId, ItemId, Rating and Date are integers.
- Date is the number of days from a unknown starting date.
- Time is a string in the form of "Hour:Min:Sec", in the 24-hour format, for example, 15:05:00.

Input Format

The first line is the number of testing actions. Each testing action contains two lines, the first one is the action name, *retrieve, items, users, club*. And the second line contains the parameters:

- retrieve(u,i,d,t): 3 integers and a time string separated by a space, u i d t.
- items(u1,u2): 2 integers separated by a space, $u1$ $u2$.
- users(i1,i2,d1,t1,d2,t2): 4 integers and 2 time strings separated by a space, $i1$ $i2$ $d1$ $t1$ $d2$ $t2$, with $(d1, t1) \leq (d2, t2)$.
- club(r1,r2,Is): 2 integers followed by the members of *Is* separated by a space, with *Is* being non-empty.

The TAs will only test your program with valid (UserId), (ItemId), (Rating), (Date), (Time) so there is no serious need of error handling in this part.

Output Format

For each action, follow the directions below for outputs.

- retrieve(u,i,d,t): outputs one line that includes the Rating, or -1 for a non-existent record.
- items(u1,u2): outputs the sorted (ItemId) line by line in ascending order.
- users(i1,i2,d1,t1,d2,t2): outputs the sorted (UserId) line by line in ascending order.
- club(r1,r2,Is): outputs the sorted (UserId) line by line in ascending order.

For actions *items*, *users* and *club*, if the output list is empty, please print a newline with string EMPTY.

Sample Input

```
4
retrieve
770 536681 5038 15:31:00
items
70 1311
users
401347 216505 4145 18:04:00 5869 20:49:00
club
40 80 303423 611087 161573 317156 504547
```