Exercise 3

(ii) The first map step produce pairs of users who are have friends in common or either friends with each other. If the users have became friends, it outputs a tuple ((userID,friend), 0), and if users have the same friends, then it prints a tuple ((userID1,userID2), 1).

The first reduce step counts the number of mutual friends for each pair of users. The keys mean the pairs of users, and the values only be 0 and 1, and they generated in the map step. The reducer is going to find the sum of the values and the product of the values. If the product of the values is 0, then the users are already friends, then we stop recommend those users to each other. If not, we print the tuple

((userID1,userID2), numCommonFriends).

The second map step maps the tuple

((userID1,userID2), numCommonFriends) to the tuple

(userID1, (userID2,numCommonFriends)).

The second reduce step takes those values and pick up the 10 values of userID2 that have the most mutual friends with userID1.

Final Result with recommendations users IDS:

```
924   439,2409,6995,11860,15416,43748,45881

8941  8943,8944,8940

8942  8939,8940,8943,8944

9019  9022,317,9023

9020  9021,9016,9017,9022,317,9023

9021  9020,9016,9017,9022,317,9023
```

9022	9019,9020,9021,317,9016,9017,9023
9990	13134,13478,13877,34299,34485,34642,37941
9992	9987,9989,35667,9991
9993	9991,13134,13478,13877,34299,34485,34642,37941