Create recommendation feature called “Games Your Friends Play”. The recommendation logic is based on the following rules:

- A customer should only be recommended games that their friends own but they don’t.
- The recommendations priority is driven by how many friends own a game – if multiple friends own a particular game, it should be higher in the recommendations than a game that only one friend owns.

You are provided two library functions to help you:

- `getFriendsListForUser` – returns a list of customer IDs (strings that uniquely identify an Amazon user) representing the friends of an Amazon user.
- `getLibraryForUser` – returns a list of product IDs (strings that uniquely identify a game) for an Amazon user

We supply an example of the following:

1. A function that provides a ranked (high to low) list of recommendations (product IDs) for a provided user.
2. Key unit tests.
3. The space and time complexity of our solution.

For a space-time complexity analysis applied to customer and friends, we use the following notations:

- L: average number of products in a customer library
- P: fraction of games unique to each customer library
- F: average number of friends of a customer
- N: total number of products in catalog
- R: expected number-of-recommendations

The expected number of recommendations is $R = F * (P * L)$; constrained to be less than N. The performance analysis is:

1. Count - $O(F * L)$
2. Remove - $O(L)$
3. Extract - $O(\log(R))$

The dominant performance from the steps above is $O(F * L)$.