A social network platform wants to recommend some users who are not yet friends to become friends, as a campaign to boost community connection and activity level. Research on social networks has yielded the following general observations:

1. It is found that a user is more likely to accept a friend recommendation if he and the recommended friend have a large number of common friends.
2. In general, a user is more likely to accept a friend recommendation if the recommended friend is more selective in making friends. A user is deemed more selective if he has fewer friends.
3. In a social network, some users are indiscriminate when befriending others and therefore have an excessively large number of friends. These are considered non-selective users.

We have come up with a model which incorporates the above observations into a quantitative measure called the S score (S stands for "strength"). S is a quantitative measure of the strength of a potential link between two users in a social network.

**S Equation**

S(u,v)=1logf1+1logf2+1logf3+...

where:

* S(u,v) = strength of 2 users u and v who are currently not friends
* fx is the number of friends that user x has.
* Users 1, 2 and 3 are common friends of u and v. We are showing only 3 common friends here. If u and v have n common friends, there will be n terms in the equation.

This measure takes into consideration of the number of common friends of u and v -- the more common friends they have, the greater the measure (Observation 1) -- and the selectiveness of each common friend w -- the log() scales down the number of friends of w and the reciprocal for the log number of friends of w makes it such that the less selective w is in making friends, the less w would weigh in the counting of the common friends between u and v (Observations 2 and 3).

For this project, a social network is modelled as an undirected graph: each vertex is a user, and each edge represents a mutual friendship in this social network. Friendships must be mutual (i.e. is u is a friend of v, v will be a friend of u). Users who are not friends will not be directly connected via an edge.

A diagram of a network

Description automatically generated with low confidence

Example 1: K and G are not friends, and we can recommend that K and G become friends.

* K and G have got 4 common friends: F, I, J, H
* F, I, J and H have got 5, 4, 4, 5 friends respectively.
* This is how the strength of this recommendation is calculated:

S(K,G)=1/log5+1/log4+1/log4+1/log5=6.18

You are given a social network G with N users and E existing friend pairs. Your task is to design an algorithm to recommend a new friend pair in the network. (The optimal solution is the new friend pair with the highest S score in G, but your algorithm does not need to produce the optimal solution.)

Your algorithm will be evaluated in two ways:

* Running time. The algorithm execution time must not exceed 30 seconds. You will get a higher score for a shorter running time.
* Quality score. You will get a higher score for a higher quality score. Your quality score is S.

In addition to the small 11-user network given above, you are given 6 more sample networks for testing purposes. Each network is basically a text file (adjacency table) that represents an undirected graph.

* network\_example (11 users, 24 edges). Based on example above.
* network\_g1 (50 users, 605 edges).
* network\_g2 (100 users, 2513 edges)
* network\_g3 (200 users, 9986 edges)
* network\_t1 (50 users, 253 edges)
* network\_t2 (100 users, 1034 edges)
* network\_t3 (200 users, 3972 edges)