



# Virtual Friends

**Link submit:**

[https://uva.onlinejudge.org/index.php?option=com\\_onlinejudge&Itemid=8&page=show\\_problem&problem=2498](https://uva.onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&page=show_problem&problem=2498)

**Solution:**

C++	<a href="http://ideone.com/WRmJVd">http://ideone.com/WRmJVd</a>
Java	<a href="https://ideone.com/OdNY3x">https://ideone.com/OdNY3x</a>
Python	<a href="https://ideone.com/G7ECFD">https://ideone.com/G7ECFD</a>

**Tóm tắt đề:**

Bạn muốn thống kê số lượng người hiện tại trong danh sách các nhóm trên mạng xã hội. Cho bạn danh sách các cặp hai người kết nối bạn bè với nhau. Nhiệm vụ của bạn là thống kê xem hiện tại sau khi hai người đó kết bạn thì số người trong nhóm đó sẽ là bao nhiêu.

**Input:**

Dòng đầu chứa một số nguyên dương T là số lượng bộ test. Mỗi bộ test có định dạng như sau:

- Dòng đầu chứa số nguyên F không quá 100.000 là số cặp bạn bè kết nối với nhau.
- F dòng sau, mỗi dòng chứa tên (chuỗi ký tự không dài quá 20) hai bạn kết bạn nhau.

**Output:**

Với mỗi bộ test, in ra F dòng, dòng thứ i là số lượng người trong group lớn nhất sau khi kết nối hai người ở cặp thứ i của bộ test đó.

**Ví dụ:**

1 3 Fred Barney Barney Betty Betty Wilma	2 3 4
--	-------------

**Giải thích ví dụ:**

Ví dụ gồm một bộ test và trong test có danh sách 3 cặp kết bạn:

- Fred Barney kết bạn với nhau → nhóm có 2 người.

- Barney Betty kết bạn với nhau → nhóm có 3 người.
- Betty Wilma kết bạn với nhau → nhóm có 4 người.

### Hướng dẫn giải:

Bạn cần tạo thêm một mảng chứa số lượng người hiện tại mà người đó đang thuộc nhóm đó là bao nhiêu (mảng group).

Ban đầu khởi tạo giá trị cho toàn bộ mảng group là 1, nghĩa là mỗi người đều có 1 nhóm riêng cho mình.

Khi kết hợp hai người kết bạn với nhau, bạn sẽ tìm người chủ của nhóm này (nút gốc) và xem hiện tại nhóm đó có bao nhiêu người. So sánh nếu nhóm nào đông hơn thì gán số lượng qua cho nhóm kia.

Bạn sẽ chỉnh sửa lại một số cách xử lý trong hàm unionSet để phù hợp với ý tưởng trên.

Ngoài ra, vì dữ liệu đầu vào đều là tên, nên ta cần ánh xạ từ tên (chuỗi) sang dạng số để áp dụng DSU, để làm điều này ta cần sử dụng một map (đối với C++, hoặc TreeMap/HashMap với Java và dict với Python) để ánh xạ từ chuỗi sang số.

**Độ phức tạp:**  $O(T \cdot F \cdot (F + \log F))$  hay  $O(T \cdot F^2)$  trong trường hợp sử dụng DSU cơ bản, nếu sử dụng DSU nâng cao (update by rank, path compression, by size) thì sẽ là  $O(T \cdot F \cdot \log F)$