



# Bishu and his Girlfriend

Link: <https://www.hackerearth.com/practice/algorithms/graphs/depth-first-search/practice-problems/algorithm/bishu-and-his-girlfriend/>

Solution:

C++	<a href="https://ideone.com/LVqKVQ">https://ideone.com/LVqKVQ</a>
Java	<a href="https://ideone.com/Plpk91">https://ideone.com/Plpk91</a>
Python	<a href="https://ideone.com/fHHFvS">https://ideone.com/fHHFvS</a>

**Tóm tắt đề:** Cho  $N$  vùng đất và nó  $N-1$  đoạn đường (theo dạng cây). Bishu sống ở thành vùng đất số 1. Có  $Q$  cô gái sống ở các vùng đất khác (không phải vùng đất 1). Hãy tìm cô gái có thể đi từ vùng đất của cô ấy đến vùng đất của Bishu ngắn nhất. Nếu có 2 đường đi ngắn nhất thì bạn tìm cô gái sống ở vùng đất có chỉ số nhỏ hơn.

**Input:**

Dòng đầu tiên chứa số  $N$  là số lượng vùng đất.  $2 \leq N \leq 1000$ .

$N-1$  dòng tiếp theo lần lượt là các cặp vùng đất có đường đi với nhau.  $1 \leq u, v \leq N$ .

Dòng tiếp theo chứa số  $Q$  là số lượng các cô gái.  $2 \leq Q \leq (N-1)$ .

$Q$  dòng tiếp theo chứa các vùng đất là nơi các cô gái đang ở đó.

**Output**

In ra vùng đất cần tìm là nơi cô gái gần nhất với vùng đất số 1.

6 1 2 1 3 1 4 2 5 2 6 4 5 6 3 4	3
---	---

**Giải thích:** Có 6 vùng đất. Và vùng đất đi đến gần vùng đất số 1 có 2 vùng là vùng số 3 và số 4. Bạn in ra số 3 do ID nhỏ hơn vùng đất kia.

**Hướng dẫn giải:** Chạy thuật toán DFS, sau đó lấy lần lượt từng vùng đất của các cô gái rồi tìm đường đi tới vùng đất số 1. Lấy vùng đất có lượng đường đi qua các đỉnh là ít nhất. Để chọn ID nhỏ thì chỉ khi vùng đất nào thật sự lớn hơn vùng đất hiện tại bạn mới cập nhật lại. Nếu bằng thì cũng không cập nhật.

Bạn cũng có thể cải tiến một chút bằng cách chạy thuật toán từ vùng đất số 1 mà Bishu đang đứng để lấy kết quả đường đi đến các vùng đất mà các cô gái đang đứng. Cách này sẽ đỡ phải mất thời gian chạy đi chạy lại DFS nhiều lần.

**Độ phức tạp:**  $O(V + E + Q)$  với  $V$  là số lượng đỉnh của cây,  $E = V - 1$  là số lượng cạnh của cây và  $Q$  là số lượng vùng đất cô gái đang ở.