

## Bài toán C. ngfam làm Hoa Tiêu

Tệp vào:	Đầu vào chuẩn
Tệp ra:	Đầu ra chuẩn
Giới hạn thời gian:	4 giây
Giới hạn bộ nhớ:	256 megabytes

Đây là một bài toán tương tác (interactive)

Sau khi đã mua xong bàn và màn hình, *TrungNotChung* cần mang bàn và màn hình đến địa điểm thi VNOI Cup. Thành phố Hà Long – nơi VNOI Cup diễn ra, có  $n$  nút giao thông được đánh số từ 1 đến  $n$ , và có  $n - 1$  con đường. Từ một nút giao thông có thể đến được một nút giao thông bất kì sử dụng một số con đường trong thành phố. Như vậy đường đi ở thành phố Hà Long có cấu trúc *cây*.

Hiện tại *TrungNotChung* đang đứng tại nút giao thông thứ 1. Không may thay, điện thoại của *TrungNotChung* không bắt được tín hiệu mạng, do đó *TrungNotChung* không thể sử dụng tính năng định vị được. *TrungNotChung* không biết bản đồ của thành phố Hà Long ra sao, chỉ nhớ rằng VNOI Cup được tổ chức tại trường THPT chuyên Hà Long – nơi có vị trí đặc địa chính là một **trọng tâm** của thành phố.

Vì không biết đến địa điểm thi như thế nào, *TrungNotChung* liền gọi điện cho *ngfam* và nhờ *ngfam* chỉ đường. *ngfam* cũng không biết vị trí của *TrungNotChung* trông như thế nào, nên *ngfam* cũng không biết làm thế nào để chỉ đường thẳng đến trường THPT chuyên Hà Long được. Nhưng sau một hồi suy nghĩ, hai bạn đã thống nhất với nhau sẽ liên lạc với nhau theo mô tả sau.

Gọi vị trí mà *TrungNotChung* đang đứng là  $r$ , ban đầu  $r = 1$ . Coi  $r$  là gốc cây, *TrungNotChung* sẽ sử dụng các câu lệnh sau để hỏi *ngfam*.

Câu lệnh		Giải thích
adj	$v$	Tìm nút giao thông kề với nút giao thông $r$ trên đường đi từ $r$ đến $v$ .
subtree	$v$	Tìm số lượng đỉnh trong cây con có gốc là nút giao thông $v$ .
move	$v$	<i>TrungNotChung</i> di chuyển đến vị trí $v$ với điều kiện $v$ là <b>đỉnh kề với đỉnh <math>r</math></b> . Sau đó gán $r \leftarrow v$ .

Tất nhiên thời gian là vàng, do đó *TrungNotChung* cần di chuyển đến trường THPT chuyên Hà Long càng sớm càng tốt để lắp đặt các thiết bị. Chỉ với 15 555 câu lệnh, hãy giúp *TrungNotChung* tìm đến một **trọng tâm** của thành phố.

Thành phố có thể có nhiều trọng tâm, tuy nhiên *TrungNotChung* chỉ cần tìm đến một trọng tâm **bất kì**. Khi *TrungNotChung* đã đến một trọng tâm mà không phải là địa điểm thi, lúc đó *ngfam* cũng đã biết *TrungNotChung* ở đâu và sẽ đi đón bạn ấy.

Một nút giao thông được gọi là **trọng tâm** của thành phố có cấu trúc cây gồm  $n$  đỉnh, nếu như khi xóa nút giao thông này (và các cạnh nối đến nút giao thông này) này ra khỏi thành phố, các thành phần liên thông được tạo ra đều có số đỉnh không quá  $\frac{n}{2}$ .

### Giao thức tương tác

Trong bài toán này, chương trình của BTC sẽ đóng vai trò của *ngfam* – thực hiện các câu lệnh của *TrungNotChung*, và chương trình của bạn sẽ đóng vai trò của *TrungNotChung* – đưa ra những câu lệnh để tìm được trọng tâm của thành phố.

Đầu tiên bạn cần đọc vào một số  $n$  ( $1 \leq n \leq 10\,000$ ) – số lượng đỉnh trên cây.

Tiếp đó, bạn cần thực hiện tương tác với chương trình của BTC sử dụng các thao tác đã được đề cập ở đề bài như sau:

Câu lệnh	Tương tác	
adj	In ra	adj $v$ ( $1 \leq v \leq n$ )
	Đọc vào	<ul style="list-style-type: none"><li>Số nguyên <math>t</math> khi <math>v \neq r</math> — đỉnh kề đỉnh <math>r</math> trên đường đi từ <math>r</math> đến <math>v</math>.</li><li><math>-1</math> khi <math>v = r</math>. Bạn nên kết thúc chương trình ngay lập tức trong trường hợp này.</li></ul>
subtree	In ra	subtree $v$ ( $1 \leq v \leq n$ )
	Đọc vào	Số nguyên $s$ — số lượng đỉnh trên cây con gốc $v$ .
move	In ra	move $v$ ( $1 \leq v \leq n$ )
	Đọc vào	<ul style="list-style-type: none"><li>1 khi <math>v</math> là đỉnh kề đỉnh <math>r</math>. Sau đó <math>r</math> sẽ được gán bằng <math>v</math>.</li><li><math>-1</math> khi <math>v</math> không kề đỉnh <math>r</math>.</li></ul> <p>Bạn nên kết thúc chương trình ngay lập tức trong trường hợp này.</p>

Khi đã chắc chắn đỉnh  $r$  là một trọng tâm của thành phố, hãy in ra `found` và kết thúc chương trình. Câu lệnh này sẽ không tính vào giới hạn thao tác đã cho.

Nếu cây có nhiều trọng tâm, hãy di chuyển  $r$  đến trọng tâm bất kì.

Cây đã được **cố định** trước quá trình tương tác, và sẽ **không thay đổi** trong quá trình tương tác.

Sau khi in ra một câu lệnh, đừng quên xuống dòng và flush đầu ra chuẩn, nếu không bạn có thể nhận verdict `Time limit exceeded`. Để làm điều này, bạn có thể sử dụng:

- `fflush(stdout)` hoặc `cout.flush()` trong C++;
- `System.out.flush()` trong Java;
- `flush(output)` trong Pascal;
- `stdout.flush()` trong Python;
- xem tài liệu chuẩn đối với các ngôn ngữ khác.

## Điểm

Subtask	Điểm	Giới hạn
1	500	$n \leq 100$
2	750	$n \leq 1000$
3	1000	Không có giới hạn gì thêm
Total	2250	

## Ví dụ

Đầu vào chuẩn	Đầu ra chuẩn
2 1	move 2 found
3 1 3 1	subtree 2 adj 2 move 3 found

## Chú thích

Trong ví dụ đầu tiên, thành phố Hà Long gồm duy nhất 2 nút giao thông được nối với nhau. Hai nút giao thông này đều là trọng tâm của thành phố. Do đó, ngoài cách di chuyển đến nút giao thông 2, có thể ngay lập tức đáp lại BTC khi đang ở nút giao thông 1.

Trong ví dụ thứ hai, thành phố gồm 3 nút giao thông và 2 đường đi nối 1-3 và 2-3.

- Với truy vấn đầu tiên, ta có thể kết luận được 2 là nút giao thông lá.
- Với truy vấn thứ hai, ta có thể kết luận được rằng nút giao thông 1 không kề nút giao thông 2.

Từ đây có thể kết luận rằng 3 là trọng tâm của thành phố.