

문제 C. 메모리 카드 게임

시간 제한 1 초 메모리 제한 1024 MB

이 문제는 인터랙티브 문제이다.

카드 $2N$ 장으로 메모리 카드 게임을 한다.

각 카드의 뒷면에는 1 이상 N 이하의 정수가 적혀 있고, 각 정수는 **정확히 두 장**의 카드에 적혀 있다. 카드에는 1 번부터 $2N$ 번까지 번호가 붙어 있으며, 처음에는 모든 카드의 뒷면이 가려져 있어 어떤 카드에 어떤 수가 적혀 있는지 알 수 없다.

한 턴은 다음 두 단계로 이루어진다.

1. 카드 하나를 골라 뒷면을 확인한다. (**첫 번째 카드**)
2. 카드 하나를 골라 뒷면을 확인한다. (**두 번째 카드**)

두 번째 카드까지 확인했을 때, 두 카드의 뒷면에 적힌 수가 **같으면** 두 카드는 짝이 맞은 것으로 게임판에서 사라진다. **다르면** 두 카드는 그대로 남아 이후 턴에서 다시 고를 수 있다.

모든 카드가 사라지면 게임이 끝난다. 이때 **카드의 뒷면을 $4N - 2$ 번 이하로 확인해** 게임을 끝내야 한다.

이 문제의 채점기는 **적응적(adaptive)**이다. 즉 카드의 배치는 미리 고정되어 있지 않으며, 지금까지의 질문과 답에 모순되지 않는 한 언제든지 다르게 정해질 수 있다.

입력

첫 번째 줄에 정수 N 이 주어진다.

이후 채점기와 인터랙션이 시작된다.

인터랙션

이 문제의 채점기는 적응적이다.

카드의 뒷면을 확인하려면 다음 두 형식 중 하나를 출력한다.

- ? first x — 이번 턴에서 첫 번째로 고른 x 번 카드의 뒷면을 확인한다.
- ? second x — 이번 턴에서 두 번째로 고른 x 번 카드의 뒷면을 확인한다.

각 질문에 대해 x 번 카드에 적힌 수가 한 줄에 주어진다.

다음 조건을 반드시 지켜야 하며, 하나라도 어기면 오답으로 처리된다.

- 질문은 first, second, first, second, ... 순서로 번갈아 출력해야 하며, first부터 시작한다.
- $1 \leq x \leq 2N$ 이어야 한다.
- 한 턴에서 고르는 두 카드는 **서로 다른 카드**여야 한다.
- 이미 사라진 카드는 고를 수 없다. (아직 남아 있다면, 값을 이미 아는 카드를 다시 골라도 된다.)
- 뒷면을 확인한 총 횟수가 $4N - 2$ 를 넘으면 안 된다.

모든 카드가 사라지면 게임은 **자동으로 끝난다**. 별도의 출력 없이 프로그램을 종료하면 된다. 카드가 남아 있는데 프로그램을 종료해도 오답으로 처리된다.

출력을 한 뒤에는 표준 출력 버퍼를 반드시 비워야 한다.

언어별로 표준 출력 버퍼를 비우는 방법은 다음과 같다.

- C: `fflush(stdout)`
- C++: `std::cout << std::flush`
- Java: `System.out.flush()`
- Python: `sys.stdout.flush()`

잘못된 질문을 하면 채점기가 즉시 종료하며, 이때 당신의 프로그램이 받게 되는 값은 정의되지 않는다.

제한

- $2 \leq N \leq 5000$

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
2	

노트

$N = 2$ 인 경우의 상호작용 예시이다.

참가자의 출력	채점기의 출력
	2
? first 1	
	2
? second 2	
	1
? first 3	
	1
? second 2	
	1
? first 1	
	2
? second 4	
	2

- 1번 턴: 1번 카드는 2, 2번 카드는 1이다. 서로 다르므로 두 장 모두 남는다.
- 2번 턴: 3번 카드는 1이고 2번 카드도 1이므로 두 카드가 사라진다.
- 3번 턴: 1번 카드는 2, 4번 카드도 2이므로 두 카드가 사라진다.

모든 카드가 사라졌고 뒷면을 확인한 횟수는 6번이다. $N = 2$ 이므로 확인 횟수 제한은 $4N - 2 = 6$ 이고, 이 예시는 제한을 정확히 채워 게임을 끝냈다.

채점기가 적응적이므로, 같은 질문을 해도 위와 다른 값이 돌아올 수 있다.