

Jenga (jenga)

Limit pamięci: 256 MB

Limit czasu: 3.00 s

To zadanie jest interaktywne.



W zadaniu będą obowiązywać następujące zasady gry w Jengę: Gra polega na naprzemiennym wyciągania klocków z wieży i budowaniu z nich kolejnych pięter. Każde piętro zbudowane jest z trzech podłużnych klocków o proporcji długości i szerokości 3:1 ułożonych w szeregu obok siebie w kierunku prostopadłym do kierunku klocków piętra poniżej. Gracz, po ruchu którego wieża się przewróci, przegrywa. Uznajemy, że wieża zawsze się przewraca, gdy na piętrze został tylko skrajny klocek, lub gdy zostanie podjęta próba wyjęcia środkowego klocka, gdy nie ma już innych klocków na piętrze. Wyciągnięte klocki są odkładane na najwyższe piętro, chyba że ma już ono trzy klocki, w tej sytuacji jest tworzone nowe piętro. Nie można wyciągać klocków nad którymi nie znajduje się żaden klocek. Rozgrywka nie musi się rozpoczynać z pełnej wieży, to znaczy, że na początku mogą występować piętra zawierające tylko dwa lub nawet jeden klocek (w przypadku jednego klocka może być to tylko środkowy).

Krzysztof i Igor ostatnio bez przerwy grają w Jengę. Wciągnęli się w nią tak bardzo, że opanowali tę grę do perfekcji. Jeżeli któryś z nich przegrywa, to tylko dlatego, że nie ma już ruchu, który nie przewróciłby wieży. Niestety Igor się rozchorował i teraz Krzysztof nie ma z kim grać, zaprosił więc Ciebie do wspólnej rozgrywki. Jako że Krzysztof wie, że nie masz takiego doświadczenia w tej grze jak on oraz Igor, postanowił, że da ci fory i pozwoli ci rozpocząć rozgrywkę z pozycji wygrywającej. Aby jeszcze mocniej zachęcić Ciebie do rozgrywki, Krzysztof pozwolił ci napisać program komputerowy, który będzie pomagał ci w rozgrywce. Możesz założyć, że już kiedyś grałeś w Jengę i też tak, jak Krzysztof i Igor potrafisz idealnie wyjmować klocki z wieży.

Twoim zadaniem jest więc napisanie programu, który poda ci ruchy pozwalające na wygranie z Krzysztofem.

Protokół interakcji

Na początku interakcji w pierwszym wierszu wejścia będzie znajdować liczba naturalna N opisująca liczbę pięter początkowej wieży. W następnych N wierszach pojawi się opis początkowego stanu wieży. W i -tym wierszu będzie znajdował się opis i -tego piętra S_i składający się ze trzech znaków # oraz ., oznaczających odpowiednio klocek oraz pustą przestrzeń. Dla przykładu $S_3 = \#\#.$ oznacza, że trzecie piętro zawiera wszystkie klocki oprócz prawego.

Po wczytaniu wieży należy rozpocząć rozgrywkę. Ty wykonujesz ruch jako pierwszy. Aby wyciągnąć j -ty klocek od lewej z i -tego piętra należy użyć zapytania poprzez wypisanie dwóch liczb i i j . Po każdym zapytaniu program sprawdzający wypisze na standardowe wejście dwie liczby całkowite i oraz j , które oznaczają, że Krzysztof wyciągnął j -ty klocek z i -tego piętra. W przypadku zakończonej rozgrywki lub wykonania ruchu niezgodnego z zasadami, program sprawdzający wypisze na wejście -1 -1. Należy wtedy natychmiast zakończyć działanie programu.

Należy pamiętać o opróżnianiu bufora wypisywania po każdym zapytaniu. Aby to uczynić należy wykonać `cout.flush()`; lub `cout << endl` jeżeli używamy `cin/cout` w C++, `fflush(stdout)` dla `printf/scanf` w C++, `sys.stdout.flush()` w Pythonie oraz `System.out.flush()` w Javie.

Ograniczenia

$2 \leq N \leq 50\,000$.

Podzadania

Podzadanie	Warunki	Punkty
1	$N \leq 5$	35
2	$N \leq 5\,000$	25
3	brak dodatkowych ograniczeń	40

Przykładowa interakcja

Wejście	Wyjście
2	
###	
#.#	
	1 1
1 3	
	2 2
-1 -1	

Wyjaśnienie przykładu: Pomimo niestandardowego ułożenia klocków, po wyciągnięciu klocka z pierwszego piętra w pierwszym ruchu jest on odłożony na puste miejsce na drugim piętrze. Po wykonaniu drugiego ruchu stało się możliwe wyciąganie klocków z drugiego piętra.