Zadanie: TUR
Turniej

Etap II, dzień 2, plik źródłowy `tur.*`, dostępna pamięć 32 MB *12–02–2004*

Światowa Federacja Gry X organizuje turniej programów grających w tę grę. Startuje w nim n programów ponumerowanych od 1 do n. Zasady rozgrywania turnieju są następujące: dopóki w turnieju pozostaje więcej niż jeden program, losowane są dwa różne spośród nich, które rozgrywają partię gry X. Przegrany (w grze X nie ma remisów) odpada z turnieju, po czym cała procedura jest powtarzana. Program, który pozostanie w turnieju jako jedyny po rozegraniu wszystkich $n-1$ gier, zostaje zwycięzcą.

Federacja dysponuje tabelą wyników poprzednich turniejów. Wiadomo, że programy grają deterministycznie (tzn. w powtarzalny sposób) i tak samo jak w poprzednich turniejach. Zatem jeżeli pewne dwa programy już kiedyś ze sobą grały, to na pewno ich kolejna gra zakończy się tak samo. Jeśli jednak dwa programy jeszcze nigdy nie walczyły ze sobą, rezultatu rozgrywki nie da się przewidzieć — oba mają szansę na wygraną. Federacja chciałaby poznać listę wszystkich tych programów, które mają szansę na wygranie turnieju.

Zadanie

Napisz program, który:

- wczyta ze standardowego wejścia liczbę uczestniczących programów oraz tabelę wyników ich wcześniejszych gier,
- wyznaczy wszystkie programy, które mają szansę wygrać turniej,
- wypisze wynik na standardowe wyjście.

Wejście

Pierwszy wiersz wejścia zawiera liczbę całkowitą n, $1 \leq n \leq 100\,000$. Kolejnych n wierszy zawiera tabelę wyników wcześniejszych gier: $i+1$-szy wiersz zawiera liczbę całkowitą k_i, $0 \leq k_i < n$, a następnie k_i numerów programów, różnych od i, podanych w kolejności rosnącej — są to numery programów, z którymi program nr i już kiedyś wygrał. Liczby w wierszach są pooddzielane pojedynczymi odstępami. Liczba wszystkich znanych wyników wcześniejszych gier nie przekracza $1\,000\,000$.

Wyjście

Pierwszy i jedyny wiersz standardowego wyjścia powinien zawierać liczbę w wszystkich programów, które mają szansę wygrać turniej, a następnie w liczb będących numerami tych programów, podanymi w kolejności rosnącej. Liczby w wierszu powinny być pooddzielane pojedynczymi odstępami.

Przykład

Dla danych wejściowych:

```
4
2 2 3
0
1 2
1 2
```

poprawnym wynikiem jest:

```
3 1 3 4
```