



Problem I: Popisowa samoobrona

Przewidywanie ruchów przeciwnika to bezcenna umiejętność w rywalizacji każdego rodzaju. Kluczowym elementem jest możliwość odpowiedniej reakcji na ruch przeciwnika. Każdy ruch stawia nas jednak w innej sytuacji i zmienia możliwości dalszych ruchów.

Przyjmijmy, że analizujemy rywalizację, w której można znaleźć się w jednej z n sytuacji. Załóżmy ponadto, że w każdej z nich rozważamy dostępność dwóch typów ruchów oznaczonych jako A i B . Atakujący pierwszy wchodzi do rozgrywki i wybiera swoją sytuację początkową, w której się znajduje. Następnie Broniący również wybiera sytuację początkową różną od sytuacji początkowej Atakującego. Potem następuje faza rywalizacji. Atakujący musi wykonać ruch typu A lub B , o ile jest on dostępny w sytuacji w której się znajduje. Gdy nie może wykonać ruchu to przegrywa. Broniący musi odpowiedzieć ruchem tego samego typu ze swojej sytuacji. Jeśli nie jest to możliwe — przegrywa.

Twoim zadaniem jest napisanie programu, który analizuje sytuacje i ruchy w nich dostępne i powie, czy Broniący może zawsze odpowiadać na ruchy przeciwnika.

Wejście

Pierwsza linia wejścia zawiera małą liczbę całkowitą z – liczbę zestawów danych występujących kolejno po sobie. Opis jednego zestawu jest następujący:

W pierwszej linii występuje liczba całkowita n określająca liczbę sytuacji ($2 \leq n \leq 2000$). W kolejnych n liniach znajdują się opisy ruchów możliwych do wykonania w każdej sytuacji. Każda linia zawiera dwie liczby całkowite a, b ($0 \leq a, b \leq n$) opisujące odpowiednio sytuacje do których przejdziemy wybierając ruch typu A lub B . Liczba 0 oznacza brak możliwości wykonania ruchu danego typu.

Wyjście

Dla każdego zestawu danych wypisz słowo TAK jeśli Broniący może zawsze odpowiadać na ruchy Atakującego. Gdy nie jest to możliwe wypisz NIE.

Przykład

Dla danych wejściowych:

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

Poprawną odpowiedzią jest:

```
TAK
NIE
```