Algorytmiczne pola

Algorytmiczne pola to niezwykle ciekawa gra. Plansza, na której jest ona rozgrywana składa się z n wierszy i m kolumn wypełnionych liczbami całkowitymi. W każdej komórce planszy umieszczona jest jedna liczba całkowita. Gra polega na przejściu od pola, w którym znajduje się najmniejsza liczba do pola zawierającego największą liczbę. Wartości znajdujące się w tych polach są unikalne. Wędrując po planszy dodajemy do siebie liczby umieszczone w odwiedzanych komórkach. Ruchy możliwe do wykonania z danej komórki zależą od tego jaka liczba się w niej znajduje:

1. Z pola, w którym znajduje się liczba pierwsza możemy przejść na pola bezpośrednio sąsiadujące z nim po skosach. Np. z pola 4 4 możemy przejść na pole: 3 5, 5 5, 5 3 albo 3 3.
2. Z pola, w którym nie znajduje się liczba pierwsza możemy przejść na pola bezpośrednio sąsiadujące z nim w pionie i w poziomie. Np. z pola 4 4 możemy przejść na pole: 4 5, 5 4, 4 3 albo 3 4.

Dodatkowym celem gry jest takie przejście od pola początkowego do końcowego (o ile w ogóle jest to możliwe), aby suma wartości odwiedzonych pól była jak najmniejsza. Twoim zadaniem jest obliczenie i wypisanie dla wczytanej planszy kosztu jej optymalnego przejścia albo wypisanie słowa "NIE" jeżeli przejście nie jest możliwe.

Wejście

W pierwszej linii wejścia znajduje się jedna liczba naturalna Z ($1 \leq Z \leq 100$) określająca ilość zestawów danych. W kolejnych liniach znajdują się zestawy danych.

W pierwszej linii każdego zestawu danych znajdują się dwie liczby całkowite n oraz m ($2 \leq n, m \leq 20$) określające odpowiednio ilość wierszy i kolumn na planszy. W kolejnych n liniach znajduje się po m liczb całkowitych. Liczba j-ta w i-tym wierszu a_{ij} ($0 \leq a_{ij} \leq 10^6$) określa wartość pola w i-tym wierszu i j-tej kolumnie.

Wyjście

Dla każdego zestawu danych należy w osobnej linii wypisać jedną liczbę całkowitą określającą koszt optymalnego przejścia planszy zgodnie z zasadami gry albo słowo "NIE" jeżeli takie przejście nie jest możliwe.

Przykład

Wejście:

```
3
2 2
1 2
4 3
3 3
2 4 3
4 3 4
3 4 5
```

3 3
2 6 3
3 3 3
3 3 3

Wyjście:

5
10
NIE