

G. Moindre effort

On dispose d'une matrice carrée de taille $n \times n$ contenant des nombres strictement positifs. On veut se rendre du coin en haut à gauche au coin en bas à droite *en dépensant le moins d'énergie possible*, sachant que :

- on ne peut se déplacer que d'une case vers la droite ou d'une case vers le bas à chaque coup,
- le passage d'une case à une autre a un coût d'énergie égal à la valeur absolue de la différence des valeurs des deux cases.

Par exemple, soit la matrice 4×4 suivante :

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

Il existe plusieurs chemins possibles du coin haut-gauche au coin bas-droit avec des coûts différents :

```
1 - 2   2   3
      |
2   3   1   3
      |
3   3   2   2
      |
2   3 - 4 - 3
```

Ce chemin a un coût total égal à 4.

```
1 - 2 - 2 - 3
              |
2   3   1   3
              |
3   3   2   2
              |
2   3   4   3
```

Ce chemin a également un coût total égal à 4.

```
1 - 2   2   3
      |
2   3 - 1 - 3
              |
3   3   2   2
              |
2   3   4   3
```

Ce chemin a un coût total égal à 8.

Dans tous les cas de cet exemple, on ne peut pas trouver un chemin ayant un coût inférieur à 4. La réponse dans ce cas-ci est donc 4, qui est l'énergie minimale à dépenser pour se rendre du coin haut-gauche au coin bas-droit.

En entrée, vous trouverez sur la première ligne en entier N correspondant à la taille de la matrice. Puis sur chacune des N lignes suivantes, N nombres positifs séparés par des espaces.

En sortie, votre programme doit afficher une seule valeur, l'énergie minimale à dépenser pour se rendre du coin haut-gauche au coin bas-droit.

Exemple d'entrée :

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

Sortie :

```
4
```

Exemple d'entrée :

```
10
8 8 3 9 8 6 1 3 5 6
7 5 9 4 8 3 6 9 8 1
3 5 6 4 2 6 0 7 4 2
3 8 5 5 0 1 3 6 7 0
8 0 9 6 2 4 5 3 1 6
3 6 6 8 0 4 6 5 8 5
6 9 3 2 8 2 8 3 5 0
8 1 1 3 6 5 0 1 3 2
9 0 7 2 7 9 9 3 2 3
3 7 7 1 3 2 9 2 1 2
```

Sortie :

```
24
```