

NAOI TST - Day 1

Problem 1 : Jump

تتكون لوحة لعبة بطول $n \times n$ من أعداد صحيحة، حيث تحتوي كل خانة على عدد صحيح غير سالب. الهدف هو القفز عبر أي مسار شرعي من الزاوية العلوية اليسرى إلى الزاوية السفلية اليمنى من اللوحة. العدد الموجود في أي خانة يحدد حجم القفزة التي يجب اتخاذها بعيداً عن ذلك الموقع. إذا كانت القفزة ستؤدي إلى الخروج من حدود اللوحة، فإن القفز في ذلك الاتجاه يكون ممنوعاً. جميع القفزات يجب أن تكون إما نحو اليمين أو نحو الأسفل.

لاحظ أن الرقم 0 يمثل طريقاً مسدوداً، مما يمنع أي تقدم.

على سبيل المثال، انظر إلى اللوحة بحجم الموضحة في الشكل 1، حيث الدائرة المملوءة تمثل نقطة البداية، والدائرة المنقطعة تمثل الهدف. يوضح الشكل 2 ثلاثة مسارات شرعية من نقطة البداية إلى الهدف، مع إزالة الأرقام غير المهمة من كل مسار.

2	3	3	1
1	2	1	3
1	2	3	1
3	1	1	0

الشكل 1 : مثال على لوحة بطول 4×4

2		3	
		1	0

2			
1	2		1
			0

2			
1			
3			0

شكل 2 : ثلاثة مسارات ممكنة إلى الهدف

مهمتك هي كتابة برنامج يحدد عدد المسارات الشرعية من الزاوية العلوية اليسرى إلى الزاوية السفلية اليمنى .

Constraints

- $4 \leq n \leq 100$
- في حالات الاختبار التي تساوي 70% من النقاط، ستكون الإجابة ضمن عدد صحيح مكون من 64 بت، وكل حالة اختبار صحيحة تساوي 5 نقاط
- من المضمون أن جميع المدخلات ستؤدي إلى إجابة لا تتجاوز 100 رقم

Input Specification

يحتوي السطر الأول من الإدخال على عدد صحيح n وهو عدد الصفوف في هذه اللوحة. يتبعه n صفًا من البيانات، حيث يحتوي كل صف على أعداد صحيحة، كل منها في النطاق $0 \dots 9$

Output Specification

قم بطباعة عدد صحيح واحد: عدد المسارات الشرعية من الزاوية العلوية اليسرى إلى الزاوية السفلية اليمنى.

Sample Input

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

Sample Output

```
3
```