

Plansza

 0.3 sek.  256 MB

Simona marzy o niezmierzonym bogactwie. Zaproponowano jej grę o wielką stawkę.

Simona będzie umieszczona w komórce $(0, 0)$ planszy A o rozmiarze $N \times M$ wypełnioną dodatnimi liczbami całkowitymi. Musi dotrzeć do komórki $(N - 1, M - 1)$. Żeby tego dokonać, zawsze może się ruszyć ze swojej komórki (x, y) do dowolnej innej komórki $(x + d, y)$ lub $(x, y + d)$, takiej że $d > 0$. Za każdy taki ruch, Simona otrzyma $|A_{x,y} - A_{x',y'}| - C$ monet nagrody, gdzie x', y' to jej nowe koordynaty, a C to stała kosztu ustalona przed początkiem podróży. Zauważ, że jeśli wyrażenie $|A_{x,y} - A_{x',y'}| - C$ jest ujemne, Simona straci monety. Zauważ również na to, że można skończyć grę z ujemną liczbą monet.

Pomóż Simonie określić największą liczbę monet, z którą może skończyć grę.

Zauważ, że $|a| = a$ jeśli $a \geq 0$ i $|a| = -a$, w przeciwnym razie.

Szczegóły implementacji

Należy zaimplementować funkcję `max_profit`:

```
long long max_profit(int N, int M, int C,  
                    std::vector<std::vector<int>> A)
```

- N, M : rozmiary planszy;
- C : stała kosztu dla testu;
- A : vector vectorów intów rozmiaru $N \times M$, reprezentujący planszę (indeksowaną najpierw wierszami, potem kolumnami).

Ta funkcja będzie wywołana raz dla każdego testu i powinna zwrócić największą liczbę monet, z którą Simona może skończyć grę.

Ograniczenia

- $1 \leq N, M$
- $NM \leq 500\,000$
- $1 \leq A_{i,j} \leq 1\,000\,000$ dla $0 \leq i < N$ i $0 \leq j < M$
- $0 \leq C \leq 1\,000\,000$



Podzadania

Podzadanie	Punkty	Wymagane podzadania	Dodatkowe ograniczenia
0	0	—	Przykład.
1	9	—	$N = 1, M \leq 200$
2	5	—	$N = 1, A_{i,j} \leq A_{i,j+1}$
3	8	—	$N = 1, C = 0$
4	10	1	$N = 1, M \leq 50\,000$
5	7	1 – 4	$N = 1$
6	15	1	$N, M \leq 200$
7	9	2	$A_{i,j} \leq A_{i+1,j}, A_{i,j+1}$
8	12	3	$C = 0$
9	12	0 – 1, 4, 6	$NM \leq 50\,000$
10	13	0 – 9	—



Przykład

Rozważ następujące wywołanie:

```
max_profit(5, 6, 4, {{20, 24, 31, 33, 36, 40},
                     {25, 23, 25, 31, 32, 39},
                     {31, 26, 21, 24, 31, 35},
                     {32, 28, 25, 21, 26, 28},
                     {36, 35, 28, 24, 21, 27}})
```

W tym przykładzie optymalna ścieżka to: $(0, 0) \xrightarrow{7} (0, 2) \xrightarrow{2} (1, 2) \xrightarrow{10} (1, 5) \xrightarrow{8} (4, 5)$ i liczba monet wygranych za podążanie nią wynosi $7 + 2 + 10 + 8 = 27$. Twoja funkcja powinna zwrócić 27.

```
max_profit(2, 2, 100, {{1, 2}, {3, 4}})
```

W tym przykładzie Twoja funkcja powinna zwrócić: -197 . Zauważ, że odpowiedź może być ujemna.



Przykładowa biblioteczka

Format wejścia jest następujący:

- wiersz 1: trzy liczby całkowite - wartości N , M i C .
- wiersze 2 – $(N + 1)$: M liczb całkowitych - wartości $A_{i,j}$.

Format wyjścia jest następujący:

- wiersz 1: jedna liczba całkowita - wartość zwrócona przez Twoją funkcję.