



## Prostokąty

Na początku dziewiętnastego wieku władca Hoseyngulu Khan Sardar nakazał wybudowanie pałacu na płaskowyżu nad piękną rzeką. Płaskowyż jest reprezentowany przez siatkę złożoną z  $n \times m$  kwadratowych pól. Wiersze siatki są numerowane od 0 do  $n - 1$ , a kolumny od 0 do  $m - 1$ . Pole, które znajduje się w wierszu  $i$  oraz kolumnie  $j$  ( $0 \leq i \leq n - 1, 0 \leq j \leq m - 1$ ) nazywamy polem  $(i, j)$ . Każde pole  $(i, j)$  ma określoną wysokość oznaczoną przez  $a[i][j]$ .

Hoseyngulu Khan Sardar rozkazał swoim architektom wybranie prostokątnego **obszaru**, na którym zostanie zbudowany pałac. Ten obszar nie może zawierać żadnego pola leżącego na brzegu siatki (czyli w wierszu 0, wierszu  $n - 1$ , kolumnie 0 lub kolumnie  $m - 1$ ). W związku z tym architekci powinni wybrać cztery liczby całkowite  $r_1, r_2, c_1$  oraz  $c_2$  ( $1 \leq r_1 \leq r_2 \leq n - 2$  oraz  $1 \leq c_1 \leq c_2 \leq m - 2$ ), które definiują obszar złożony ze wszystkich pól  $(i, j)$  takich, że  $r_1 \leq i \leq r_2$  oraz  $c_1 \leq j \leq c_2$ .

Dodatkowo obszar nazywamy **prawidłowym** wtedy i tylko wtedy, gdy dla każdego należącego do niego pola  $(i, j)$  spełniony jest następujący warunek:

- Rozważmy dwa pola przyległe do obszaru w wierszu  $i$  (czyli pola  $(i, c_1 - 1)$  oraz  $(i, c_2 + 1)$ ) i dwa pola przyległe do obszaru w kolumnie  $j$  (pola  $(r_1 - 1, j)$  oraz  $(r_2 + 1, j)$ ). Wysokość pola  $(i, j)$  powinna być ściśle mniejsza niż wysokości tych czterech pól.

Twoim zadaniem jest pomóc architektom w wyznaczeniu liczby prawidłowych obszarów, na których można zbudować pałac (czyli liczby sposobów, na które można wybrać  $r_1, r_2, c_1$  oraz  $c_2$  definiujące prawidłowy obszar).

## Szczegóły implementacyjne

Twoim zadaniem jest zaimplementowanie następującej funkcji:

```
int64 count_rectangles(int[][] a)
```

- $a$ : dwuwymiarowa tablica o wymiarach  $n$  na  $m$  zawierająca liczby całkowite reprezentujące wysokości pól.
- Funkcja powinna wyznaczać liczbę prawidłowych obszarów, na których można zbudować pałac.

# Przykłady

## Przykład 1

Rozważmy następujące wywołanie.

```
count_rectangles([[4, 8, 7, 5, 6],  
                 [7, 4, 10, 3, 5],  
                 [9, 7, 20, 14, 2],  
                 [9, 14, 7, 3, 6],  
                 [5, 7, 5, 2, 7],  
                 [4, 5, 13, 5, 6]])
```

4	8	7	5	6
7	4	10	3	5
9	7	20	14	2
9	14	7	3	6
5	7	5	2	7
4	5	13	5	6

Mamy 6 prawidłowych obszarów, które są wypisane poniżej:

- $r_1 = r_2 = c_1 = c_2 = 1$
- $r_1 = 1, r_2 = 2, c_1 = c_2 = 1$
- $r_1 = r_2 = 1, c_1 = c_2 = 3$
- $r_1 = r_2 = 4, c_1 = 2, c_2 = 3$
- $r_1 = r_2 = 4, c_1 = c_2 = 3$
- $r_1 = 3, r_2 = 4, c_1 = c_2 = 3$

Na przykład,  $r_1 = 1, r_2 = 2, c_1 = c_2 = 1$  jest prawidłowym obszarem, ponieważ spełnione są następujące dwa warunki:

- $a[1][1] = 4$  jest ściśle mniejsze niż  $a[0][1] = 8$ ,  $a[3][1] = 14$ ,  $a[1][0] = 7$  oraz  $a[1][2] = 10$ .
- $a[2][1] = 7$  jest ściśle mniejsze niż  $a[0][1] = 8$ ,  $a[3][1] = 14$ ,  $a[2][0] = 9$  oraz  $a[2][2] = 20$ .

## Ograniczenia

- $1 \leq n, m \leq 2500$
- $0 \leq a[i][j] \leq 7\,000\,000$  (dla każdego  $0 \leq i \leq n - 1, 0 \leq j \leq m - 1$ )

## Podzadania

1. (8 punktów)  $n, m \leq 30$
2. (7 punktów)  $n, m \leq 80$
3. (12 punktów)  $n, m \leq 200$
4. (22 punkty)  $n, m \leq 700$
5. (10 punktów)  $n \leq 3$
6. (13 punktów)  $0 \leq a[i][j] \leq 1$  (dla każdego  $0 \leq i \leq n - 1, 0 \leq j \leq m - 1$ )
7. (28 punktów) Brak dodatkowych założeń.

## Przykładowa sprawdzaczka

Przykładowa sprawdzaczka wczytuje wejście w następującym formacie:

- wiersz 1:  $n m$
- wiersz  $2 + i$  (dla  $0 \leq i \leq n - 1$ ):  $a[i][0] a[i][1] \dots a[i][m - 1]$

Przykładowa sprawdzaczka wypisuje jeden wiersz, który zawiera wynik wywołania funkcji `count_rectangles`.