

Der kleine Roboter

Nachdem Wolfgang bei der Informatikolympiade teilgenommen hat, wollte er sich neuen Herausforderungen stellen. Beim Autonomously Operated Instruments (AOI) Wettbewerb werden die ausgetüfteltsten Robotergehirne in einem Parkour auf die Probe gestellt. Es geht darum, einen intelligenten Algorithmus zu entwickeln, um einen Roboter durch ein $n \times m$ großes Feld (Zeilen, Spalten) zu lenken und dabei Hindernissen auszuweichen. Der Roboter startet in der links-unteren Ecke $(1, 1)$ und hat die rechts-obere Ecke (n, m) als Ziel. Es gibt insgesamt r *rechteckige* Hindernisse, die verschiedene Zellen des Feldes blockieren.



Sofort erkennt Wolfgang, dass es offensichtlich optimal ist, wenn man sowieso die Ecke oben-rechts erreichen will, den Roboter immer nur nach rechts oder oben zu bewegen. Wieso sollte man denn auch jemals einen Schritt in die verkehrte Richtung machen? Für Wolfgang gibt es nur eine Richtung, nämlich vorwärts zum Ziel! Nachdem er wenig Ahnung von Robotik hat, schaffte er es nicht, Augen in seinen Roboter einzubauen. Deshalb blieb ihm nichts anderes übrig als den Roboter in jedem Schritt zufällig nach rechts bzw. oben zu bewegen. Besorgt um seinen Roboter, damit die Zahnräder nicht einseitig abgenutzt werden, macht er allerdings nie mehr als k aufeinanderfolgende Schritt in dieselbe Richtung.

Schon bald sieht Wolfgang allerdings seinen Roboter traurig in der Ecke stehen, blockiert von Hindernissen rechts und oberhalb seines kleinen Gefährten. Scheinbar führen nicht alle Wege zum Ziel, wenn sich der Roboter nur nach oben und rechts bewegt. Auch die Zahnradschonung ist oft weniger hilfreich. Der Roboter kann aus zwei Gründen steckenbleiben:

1. Die Zellen rechts und oberhalb der aktuellen Position sind durch Hindernisse markiert.
2. Der Roboter ist bereits k Schritte nach rechts bzw. oben gefahren, kann aber an der aktuellen Position nicht nach oben bzw. rechts fahren und bleibt somit wegen der Zahnradschonung stecken.

Zudem ist zu beachten, dass der Roboter nicht aus dem Feld hinausfahren darf und somit auch dort steckenbleiben.

Mit dieser abenteuerlichen Konstruktion probiert Wolfgang sein Glück bei der AOI. Er fragt sich, ob sein Fehler beim Wettbewerb auffallen wird und möchte deshalb berechnen, auf wie vielen Wegen sein Roboter steckenbleiben würde. Zwei Wegen werden als unterschiedlich betrachtet, wenn sie sich in irgendeinem Schritt unterscheiden. Da die Anzahl an Wege sehr groß werden kann, gib sie stattdessen modulo $10^9 + 7 = 1000000007$ aus.

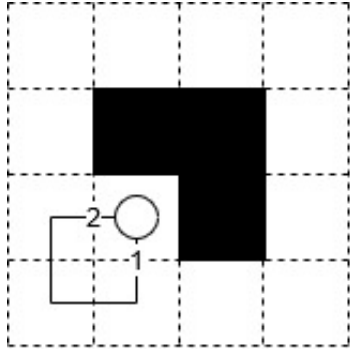
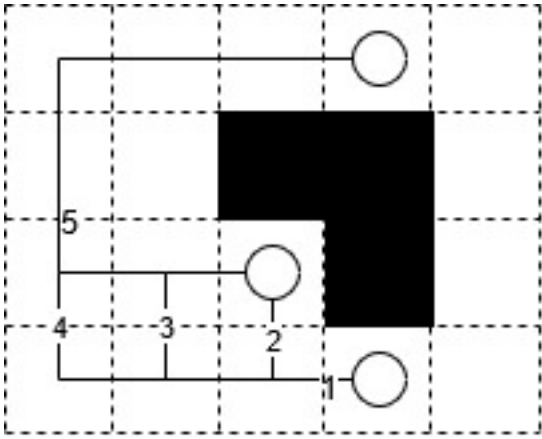
Eingabe

Die erste Zeile enthält n , m , k und r . Die folgenden r Zeilen beschreiben die Hindernisse im Format a_i, b_i, c_i, d_i . Dabei gibt (a_i, b_i) die links-untere Ecke des Rechtecks und (c_i, d_i) an.

Ausgabe

Die Anzahl an Wege, bei denen der Roboter steckenbleibt.

Beispiele

Eingabe	Ausgabe	Anmerkungen
<pre> 4 4 4 2 3 2 3 3 2 3 3 3 </pre>	2	 <p>Zwei Wege führen in die Ecke. Die Zahnradsteuerung kommt nie zum Einsatz.</p>
<pre> 4 5 3 2 3 3 3 4 2 4 3 4 </pre>	5	 <p>Bei Wegen 2, 3 und 4 bleibt der Roboter in der Ecke stecken. In Fall 1 und 5 wegen der Zahnradsteuerung.</p>

Subtasks

Allgemein gilt:

- $1 \leq n, m \leq 2000$
- $1 \leq k \leq \max(n, m)$
- $0 \leq r \leq 10^6$
- $1 \leq a_i \leq c_i \leq n$ und $1 \leq b_i \leq d_i \leq m$ (alle Hindernisse sind innerhalb des Feldes und sinnvolle Rechtecke)

- Zelle $(1, 1)$ ist nie blockiert.
- Es kann sein, dass es keinen Weg zum Ziel gibt oder Zelle (n, m) blockiert ist.

Subtask 1 (1 Punkt): $r = 0$, $k = \max(n, m)$

Subtask 2 (19 Punkte): $k = \max(n, m)$ und genau ein Rechteck mit Ecken $(n, 1)$ (n, m) .
Hier ist das Ziel also nicht erreichbar und jeder Weg führt zum Stopp.

Subtask 3 (15 Punkte): $k = \max(n, m)$ und die Rechtecke überlappen sich nicht.

Subtask 4 (15 Punkte): $k = \max(n, m)$, $n, m \leq 200$

Subtask 5 (20 Punkte): $k = \max(n, m)$

Subtask 6 (10 Punkte): $n, m \leq 200$

Subtask 7 (20 Punkte): Keine Einschränkungen

Hinweis: Da das Feld nur $n \times m$ groß ist, bedeutet $k = \max(n, m)$ insbesondere, dass die Anzahl an Schritten jede Richtung für alle Zwecke unbeschränkt sind.

Limits

Zeitlimit: 1 s

Speicherlimit: 256 MB