

문제 1. 점치기

입력 파일: standard input
출력 파일: standard output
시간 제한: 2초
메모리 제한: 64MB

K 이사장은 점을 매우 좋아해서, 언제나 이런저런 점을 치고 있다. 오늘은 카드를 사용해서 올해 IOI에서 일본선수단의 미래를 점치기로 했다.

점을 치는 방법은 다음과 같다.

- 처음 카드를 세로로 M 행, 가로 N 열의 직사각형 모양으로 모두 앞면이 보이도록 배치한다.
- $i = 1, \dots, K$ 에 대해, “위에서 A_i 번째 행부터 B_i 번째 행까지, 왼쪽에서 C_i 번째 행부터 D_i 번째 열까지에 있는 모든 카드의 앞뒷면을 뒤집는다.”라는 조작을 한다. 즉, 위에서부터 a 번째 행, 왼쪽에서부터 b 번째 열에 있는 카드를 (a, b) 번 카드라고 하면, 각 i 에 대해 $A_i \leq a \leq B_i$ 이면서 $C_i \leq b \leq D_i$ 를 만족하는 (a, b) 번 카드를 모두 뒤집는다.
- 조작이 끝난 경우, 뒷면과 앞면이 되어있는 카드가 몇 장인지에 따라 점의 결과가 결정된다.

K 이사장은 카드를 뒤집는 횟수가 생각보다 많다는 것을 깨달았기 때문에, 카드를 실제로 사용해서 점을 두는 대신 조작이 끝났을 때 앞면인 카드가 몇 장인지만 구하기로 했다.

행의 수 M , 열의 수 N , 조작의 횟수 K 와 K 번의 조작이 주어졌을 때, 조작 후에 앞면인 카드의 수를 구하는 프로그램을 작성하여야.

제한

$1 \leq M \leq 1\,000\,000\,000 (= 10^9)$ 행의 수
 $1 \leq N \leq 1\,000\,000\,000 (= 10^9)$ 열의 수
 $1 \leq K \leq 100\,000$ 조작의 회수

입력 형식

다음 정보가 표준 입력으로 주어진다.

- 첫째 줄에는 정수 M, N, K 가 공백으로 구분되어 주어지며, 카드가 M 행 N 열로 나열되어 있고, 조작을 K 번 한다는 것을 의미한다.
- $1 + i$ 번째 행 ($1 \leq i \leq K$) 에는 네 개의 정수 A_i, B_i, C_i, D_i ($1 \leq A_i \leq B_i \leq M, 1 \leq C_i \leq D_i \leq N$)가 주어지며, i 번째 조작이 위에서 A_i 번째 행부터 B_i 번째 행까지, 왼쪽에서 C_i 번째 행부터 D_i 번째 열까지에 있는 모든 카드의 앞뒷면을 뒤집는다는 것을 의미한다.

출력 형식

표준 출력으로, K 번의 조작 이후에 앞면이 되어 있는 카드의 장수를 첫째 줄에 출력하여야.

배점

채점 데이터 중, 배점의 30%에 대해 $K \leq 3\,000$ 을 만족한다.

예제

standard input	standard output
6 5 3 2 4 1 4 4 6 3 5 1 2 3 5	11

이 예에서, $K = 3$ 번의 조작은 다음과 같이 진행된다.

앞면인 카드를 □, 뒷면인 카드를 ■라고 나타내면

초기상태

□□□□□
□□□□□
□□□□□
□□□□□
□□□□□
□□□□□

↓

□□□□□
■ ■ ■ ■ □
■ ■ ■ ■ □
■ ■ ■ ■ □
□□□□□
□□□□□

↓

□□□□□
■ ■ ■ ■ □
■ ■ ■ ■ □
■ ■ □ □ ■
□ □ ■ ■ ■
□ □ ■ ■ ■

↓

□ □ ■ ■ ■
■ ■ □ □ ■
■ ■ ■ ■ □
■ ■ □ □ ■
□ □ ■ ■ ■
□ □ ■ ■ ■

최종상태

최종상태에서 앞면인 카드의 수인 11을 출력한다.

문제 2. 캥거루

입력 파일: standard input
출력 파일: standard output
시간 제한: 2초
메모리 제한: 64MB

K 이사장은 캥거루에 흥미를 느껴서, 캥거루의 행동을 관찰하기로 했다. K 이사장은 N 마리의 캥거루를 관찰하고 있다. 캥거루에는 주머니가 하나씩 달려있다. 캥거루에는 번호 $1, 2, \dots, N$ 이 붙어있다. i 번 캥거루의 몸 크기는 A_i 이며, i 번 캥거루의 주머니 크기는 B_i 이다. 주머니 크기는 몸 크기보다 작다. ($A_i > B_i$) 처음에는 어떤 캥거루의 주머니 안에도 캥거루가 들어있지 않다. 캥거루는 다음의 조작을 **조작을 더 할 수 없을 때까지** 반복한다.

$A_i < B_j$ 를 만족하는 i 번 캥거루와 j 번 캥거루가 있어서, i 번 캥거루가 다른 캥거루의 주머니에 들어있지 않고, j 번 캥거루의 주머니 안에 어떤 캥거루도 들어있지 않는 두 캥거루가 존재할 때, i 번 캥거루는 j 번 캥거루 안에 들어간다. 이때, i 번 캥거루에 주머니 안에 있는 다른 캥거루가 있어도, j 번 캥거루가 다른 캥거루의 주머니 안에 있어도 상관없다. 이런 (i, j) 쌍이 여러 개 있으면, 어떤 쌍이 선택될지는 알 수 없다. i 번 캥거루 안에 다른 캥거루가 들어있을 경우, 안에 들어 있는 캥거루는 i 번 캥거루와 같이 이동한다.

캥거루의 몸 크기와 주머니 크기가 주어졌을 때, 최종 상태로 가능한 경우가 몇 가지인지를 $1\,000\,000\,007 (= 10^9 + 7)$ 로 나눈 나머지를 구하여라.

제한

$1 \leq N \leq 300$ 캥거루의 수
 $1 \leq B_i < A_i \leq 1\,000\,000\,000$ i 번째 캥거루의 주머니 크기와 몸 크기

입력 형식

다음 정보가 표준 입력으로 주어진다.

- 첫째 줄에는 정수 N 이 주어진다. N 은 캥거루의 수를 의미한다.
- 다음 N 개의 줄에는 캥거루의 정보가 주어진다. $i+1$ 번째 ($1 \leq i \leq N$) 줄에는 두 개의 정수 A_i, B_i 가 공백으로 구분되어 주어진다. A_i 는 i 번 캥거루의 몸 크기를, B_i 는 i 번 캥거루의 주머니 크기를 각각 의미한다.

출력 형식

표준 출력으로, 최종 상태로 가능한 경우가 몇 가지인지를 $1\,000\,000\,007 (= 10^9 + 7)$ 로 나눈 나머지를 첫째 줄에 출력하여라.

배점

채점 데이터 중, 배점의 50%에 대해 $N \leq 30$ 을 만족한다.

채점 데이터 중, 배점의 70%에 대해 $N \leq 70$ 을 만족한다.

예제

standard input	standard output
5 4 3 3 1 6 5 2 1 4 2	4

1, 2, 5번 캥거루는 3번 캥거루의 주머니 안에 들어갈 수 있다. 또한, 4번 캥거루는 1번 캥거루 혹은 3번 캥거루의 들어갈 수가 있고, 3번 캥거루는 다른 어떤 캥거루의 주머니에도 들어갈 수 없다. 그러므로, 최종 상태로 가능한 것은 다음의 네 종류이다.

- 4번 캥거루가 3번 캥거루의 주머니 안에 들어있다.
- 4번 캥거루는 1번 캥거루의 주머니 안에, 1번 캥거루는 3번 캥거루의 주머니 안에 들어있다.
- 4번 캥거루는 1번 캥거루의 주머니 안에, 2번 캥거루는 3번 캥거루의 주머니 안에 들어있다.
- 4번 캥거루는 1번 캥거루의 주머니 안에, 5번 캥거루는 3번 캥거루의 주머니 안에 들어있다.

standard input	standard output
20 7 6 7 3 10 1 7 2 10 7 10 7 8 6 3 2 5 4 7 2 3 2 10 9 9 4 7 2 8 6 5 4 8 6 7 4 10 5 9 3	21060

문제 3. 소코반

입력 파일: standard input
출력 파일: standard output
시간 제한: 2초
메모리 제한: 512MB

“소코반”은 오랫동안 사람들에게 사랑받아온 퍼즐이다. “소코반”은 세로 M 행, 가로 N 열의 격자에서 진행되는 게임이다. 격자 위에 있는 상자를 플레이어를 조작해서 목적지까지 옮기는 것이 목적이다. 이 문제에서는 상자가 한 개 있는 경우만 생각한다. 격자의 일부 격자 칸에는 벽이 있어서 플레이어, 상자, 목적지는 각각 벽이 아닌 한 격자 칸에 있다(플레이어나 상자를 격자 칸 밖으로 움직일 수는 없다). 플레이어는 다음 중 하나의 조작을 할 수 있다.

- 플레이어가 있는 격자 칸에 인접한 격자 칸 중 벽이 아닌 상자가 없는 격자를 한 칸 골라서, 플레이어가 그 칸으로 이동한다.
- 플레이어가 있는 격자 칸과 상자가 있는 격자 칸이 인접하면서, 플레이어의 반대쪽으로 상자와 인접한 벽이 아닌 격자 칸이 있으면, 상자는 해당 격자 칸으로 이동하며, 플레이어는 상자가 있는 격자 칸으로 이동한다.

여기서 격자와 격자가 인접한다는 것은 두 격자가 한 변을 공유한다는 것을 의미한다.

아래는 “소코반” 문제의 예이다. #은 벽, @은 플레이어, 0은 상자, X는 목표지점, .은 그 외의 격자 칸을 의미한다.

```
..#@.  
.X.0.  
##...#
```

이 상태에서, 다음의 조작을 통해 상자를 목적지에 놓을 수 있다.

1. 플레이어를 오른쪽으로 움직인다.
2. 플레이어를 아래쪽으로 움직인다.
3. 상자와 플레이어를 왼쪽으로 움직인다.
4. 상자와 플레이어를 왼쪽으로 움직인다.

한편, 다음 상태라면 상자를 목적지점까지 움직일 수 없다.

```
..#..  
.X.0.  
##.@#
```

당신은 격자의 벽의 위치와 목적지 지점이 정해져 있을 때, 플레이어와 상자를 배치해서 풀 수 있는 “소코반” 문제가 몇 종류 있는지 알고 싶다. 여기서 풀 수 있는 “소코반” 문제는, 조작을 몇 번 반복해서 목적지점으로 상자를 이동할 수 있는 초기상태를 의미한다. 단, 플레이어와 상자는 각각 벽이 아니면서 목적지점이 아닌 격자에 배치되어야 하며, 플레이어와 상자를 서로 다른 격자에 배치해야 한다.

격자의 크기와 벽의 위치와 목적지의 위치가 주어졌을 때, 풀 수 있는 “소코반” 문제가 몇 종류 있는지 구하는 프로그램을 작성하여라.

제한

$1 \leq M \leq 1\,000$ 격자의 세로 길이
 $1 \leq N \leq 1\,000$ 격자의 가로 길이

입력 형식

다음 정보가 표준 입력으로 주어진다.

- 첫째 줄에는 정수 M, N 이 공백으로 구분되어 주어지며, 각각은 격자의 세로와 가로 길이이다.
- 다음 M 개의 줄에는 격자의 정보가 주어진다. 각 줄은 N 개의 문자로 되어있다. 각 문자는 $\#$ X $.$ 중 하나이며, $\#$ 은 벽, X 는 목적지, $.$ 은 그 외의 격자 칸(플레이어나 상자의 초기위치로 가능하다.)을 의미한다. 문자 X 는 정확히 한 번 주어진다.

출력 형식

표준 출력으로, 풀 수 있는 “소코반” 문제가 몇 종류 있는지 의미하는 정수를 첫째 줄에 출력하여라.

배점

채점 데이터 중, 배점의 20%에 대해 $M \leq 50, N \leq 50$ 을 만족한다.

예제

standard input	standard output
3 5 ..#.. .X.. ##..#	9

풀 수 있는 소코반 문제는 다음과 같이 9종류이다.

..#@. ..#.@ ..#.. ..#.. ..#.. ..#.. ..#@. ..#.@ ..#..
.X0.. .X0.. .X0@. .X0.@ .X0.. .X0.. .X.0. .X.0. .X.0@
##..# ##..# ##..# ##..# ##@.# ##.@# ##..# ##..# ##..#

standard input	standard output
2 3 .X. ...	0

이 예에서는, 풀 수 있는 “소코반” 문제를 만들 수 없다.

standard input	standard output
4 7 .#.### ##.##X.. ##.###	24