



# Grid Walking

by HackerRank

Problem

Submissions

Leaderboard

Discussions

Editorial

You are situated in an  $N$  dimensional grid at position  $(x_1, x_2, \dots, x_N)$ . The dimensions of the grid are  $(D_1, D_2, \dots, D_N)$ . In one step, you can walk one step ahead or behind in any one of the  $N$  dimensions. (So there are always  $2 \times N$  possible different moves). In how many ways can you take  $M$  steps such that you do not leave the grid at any point? You leave the grid if at any point  $x_i$ , either  $x_i \leq 0$  or  $x_i > D_i$ .

## Input Format

The first line contains the number of test cases  $T$ .  $T$  test cases follow. For each test case, the first line contains  $N$  and  $M$ , the second line contains  $x_1, x_2, \dots, x_N$  and the 3rd line contains  $D_1, D_2, \dots, D_N$ .

## Constraints

- $1 \leq T \leq 10$
- $1 \leq N \leq 10$
- $1 \leq M \leq 300$
- $1 \leq D_i \leq 100$
- $1 \leq x_i \leq D_i$

## Output Format

Output  $T$  lines, one corresponding to each test case. Since the answer can be really huge, output it modulo **1000000007**.

## Sample Input

```
1
2 3
1 1
2 3
```

## Sample Output

```
12
```

## Explanation

Starting from (1, 1) in a  $2 \times 3$  2-D grid, and need to count the number of possible paths with length equal to 3. Here are the 12 paths:

```
(1, 1) -> (1, 2) -> (1, 1) -> (1, 2)
(1, 1) -> (1, 2) -> (1, 1) -> (2, 1)
(1, 1) -> (1, 2) -> (1, 3) -> (1, 2)
(1, 1) -> (1, 2) -> (1, 3) -> (2, 3)
(1, 1) -> (1, 2) -> (2, 2) -> (1, 2)
(1, 1) -> (1, 2) -> (2, 2) -> (2, 1)
(1, 1) -> (1, 2) -> (2, 2) -> (2, 3)
(1, 1) -> (2, 1) -> (1, 1) -> (1, 2)
(1, 1) -> (2, 1) -> (1, 1) -> (2, 1)
(1, 1) -> (2, 1) -> (2, 2) -> (2, 1)
```

(1, 1) -> (2, 1) -> (2, 2) -> (1, 2)  
(1, 1) -> (2, 1) -> (2, 2) -> (2, 3)

[f](#) [t](#) [in](#)Submissions: [2903](#)

Max Score: 55

Difficulty: Medium

Rate This Challenge:

☆☆☆☆☆

[More](#)

Current Buffer (saved locally, editable)  

Java 7  

```
1 import java.io.*;
2 import java.util.*;
3 import java.text.*;
4 import java.math.*;
5 import java.util.regex.*;
6
7 public class Solution {
8
9     public static void main(String[] args) {
10         /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
11     }
12 }
```

Line: 1 Col: 1

 [Upload Code as File](#) ☐ Test against custom input

Run Code

Submit Code

Join us on IRC at [#hackerrank](#) on freenode for hugs or bugs.[Contest Calendar](#) | [Blog](#) | [Scoring](#) | [Environment](#) | [FAQ](#) | [About Us](#) | [Support](#) | [Careers](#) | [Terms Of Service](#) | [Privacy Policy](#) | [Request a Feature](#)