


6761 Super Ants

Here is another game, this game is called the ants game. In this game, you are given a grid, with N rows and M columns. The rows are numbered from 1 to N from top to bottom, and the columns are numbered from 1 to M from left to right. Each cell in the grid has unlimited store of a certain type of sugar, each unit of sugar gives the player a specific score. The game is just 1 step, given 1 super ant, position it in any cell in the grid, and the ant will do its magical work to determine your score.

Once a super ant is placed in one of the cells, it does a systematic behavior to gather sugar units from the stores. If there is no remaining time, the ant gets 1 unit of sugar from the store of its cell and stops working. If there is some remaining time, it starts a magical operation, which is the ants cloning operation. The super ant starts to clone itself to all the possible cells according to the current remaining time (all of its cloning operations start at the same time). It takes D seconds to clone itself to a cell that is far with D units. An ant at position (R_1, C_1) is far from the position (R_2, C_2) by $\max(|R_1 - R_2|, |C_1 - C_2|)$ units (where $|X|$ means the absolute value of X). Once a new super ant is cloned, it behaves as a super ant by utilizing the remaining time.

At the end, when the remaining time is not enough for any ant to clone itself to any other cell, each ant will collect one unit of sugar from its cell and your score is the sum of the values of all the collected sugar units. Note that an ant can not clone itself to another cell with distance more than the current remaining time, and an ant can not clone itself to the same cell more than once, and it can not clone itself to its cell, and any cell can contain any number of ants at any time.

Okay, which positions an ant can clone itself to them? First of all, an ant can not clone itself to a cell outside of the grid. Second, it can only clone to any cell in one of its 8 directions' vectors. Direction vector means all consecutive cells on the same direction, for example from position (A, B) , east vector will generate $(A, B + 1)$, $(A, B + 2)$, $(A, B + 3)$ and so on. In the below image, a super ant with 2 seconds remaining, can clone itself to 16 positions, 8 of them with distance 1 and 8 with distance 2.

	2		2		2	
		1	1	1		
	2	1		1	2	
		1	1	1		
	2		2		2	

Given the score value of each sugar unit in the grid, the total allowed time you have (the time starts once the first ant is placed) and a cell which the first super ant will start from it, can you write a program which calculates the score you will get if you used that cell?

Input

Your program will be tested on one or more test cases. The first line of the input will be a single integer T , the number of test cases ($1 \leq T \leq 100$). Followed by the test cases, each test case starts with a

line containing 3 integers separated by a single space $N M S$ ($1 \leq N, M \leq 50$) and ($0 \leq S \leq 500$) representing the number of rows in the grid, the number of columns and the remaining time, respectively. Followed by a line containing 2 integers separated by a single space $I J$ ($1 \leq I \leq N$) and ($1 \leq J \leq M$) representing the row number and the column number of the cell which the first ant will start from it, respectively. Followed by N lines each line contains M integers separated by a single space, representing the value of each sugar unit in each cell in that row. Each integer in the grid will not be less than 0 and will not be greater than 9.

Output

For each test case, print a single line which contains a single integer representing the score you will get at the end of the game, since the result may be very large, print it modulo 1,000,000,007 ($10^9 + 7$).

Note

In the first example, the first ant will collect one sugar unit from its cell with score 5. In addition, it will clone 8 ants in the 8 directions, each one of them can only get 1 sugar unit from its cell. The total score is $1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 = 45$.

In the second example, the first ant will clone itself to 8 cells at distance 2, with remaining time 0, and to 8 cells at distance 1 with remaining time 1 (thus each of the latter clones will further clone itself).

Sample Input

```
2
5 6 1
3 3
0 2 4 6 8 1
1 1 2 3 1 2
1 4 5 6 3 4
2 7 8 9 5 8
3 5 8 0 7 0
5 6 2
3 3
0 2 4 6 8 1
1 1 2 3 1 2
1 4 5 6 3 4
2 7 8 9 5 8
3 5 8 0 7 0
```

Sample Output

```
45
349
```