



H. The Most Reckless Defense

time limit per test: 3 seconds
 memory limit per test: 256 megabytes
 input: standard input
 output: standard output

You are playing a very popular Tower Defense game called "Runnerfield 2". In this game, the player sets up defensive towers that attack enemies moving from a certain starting point to the player's base.

You are given a grid of size $n \times m$, on which k towers are already placed and a path is laid out through which enemies will move. The cell at the intersection of the x -th row and the y -th column is denoted as (x, y) .

Each second, a tower deals p_i units of damage to all enemies within its range. For example, if an enemy is located at cell (x, y) and a tower is at (x_i, y_i) with a range of r , then the enemy will take damage of p_i if $(x - x_i)^2 + (y - y_i)^2 \leq r^2$.

Enemies move from cell $(1, 1)$ to cell (n, m) , visiting each cell of the path exactly once. An enemy instantly moves to an adjacent cell horizontally or vertically, but before doing so, it spends one second in the current cell. If its health becomes zero or less during this second, the enemy can no longer move. The player loses if an enemy reaches cell (n, m) and can make one more move.

By default, all towers have a zero range, but the player can set a tower's range to an integer r ($r > 0$), in which case the health of all enemies will increase by 3^r . However, each r can only be used for **at most one** tower.

Suppose an enemy has a base health of h units. If the tower ranges are 2, 4, and 5, then the enemy's health at the start of the path will be $h + 3^2 + 3^4 + 3^5 = h + 9 + 81 + 243 = h + 333$. The choice of ranges is made once before the appearance of enemies and cannot be changed after the game starts.

Find the maximum amount of base health h for which it is possible to set the ranges so that the player does not lose when an enemy with health h passes through (without considering the additions for tower ranges).

Input

The first line contains an integer t ($1 \leq t \leq 100$) — the number of test cases.

The first line of each test case contains three integers n, m , and k ($2 \leq n, m \leq 50, 1 \leq k < n \cdot m$) — the dimensions of the field and the number of towers on it.

The next n lines each contain m characters — the description of each row of the field, where the character "." denotes an empty cell, and the character "#" denotes a path cell that the enemies will pass through.

Then follow k lines — the description of the towers. Each line of description contains three integers x_i, y_i , and p_i ($1 \leq x_i \leq n, 1 \leq y_i \leq m, 1 \leq p_i \leq 500$) — the coordinates of the tower and its attack parameter. All coordinates correspond to empty cells on the game field, and all pairs (x_i, y_i) are pairwise distinct.

It is guaranteed that the sum of $n \cdot m$ does not exceed 2500 for all test cases.

Output

For each test case, output the maximum amount of base health h on a separate line, for which it is possible to set the ranges so that the player does not lose when an enemy with health h passes through (without considering the additions for tower ranges).

If it is impossible to choose ranges even for an enemy with 1 unit of base health, output "0".

Codeforces Round 938 (Div. 3)

Finished

→ Practice?

Want to solve the contest problems after the official contest ends? Just register for practice and you will be able to submit solutions.

[Register for practice](#)

→ Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

[Start virtual contest](#)

→ Problem tags

bitmasks brute force
 constructive algorithms dp flows
 graph matchings shortest paths

No tag edit access

→ Contest materials

- Announcement

Example

input	Copy
<pre> 6 2 2 1 #. ## 1 2 1 2 2 1 #. ## 1 2 2 2 2 1 #. ## 1 2 500 3 3 2 #. ##. .## 1 2 4 3 1 3 3 5 2 #.### #.##. ###.# 2 2 2 2 4 2 5 5 4 #. #. #. #. ##### 3 2 142 4 5 9 2 5 79 1 3 50 </pre>	
output	Copy
<pre> 0 1 1491 11 8 1797 </pre>	

Note

In the first example, there is no point in increasing the tower range, as it will not be able to deal enough damage to the monster even with 1 unit of health.

In the second example, the tower has a range of 1, and it deals damage to the monster in cells (1, 1) and (2, 2).

In the third example, the tower has a range of 2, and it deals damage to the monster in all path cells. If the enemy's base health is 1491, then after the addition for the tower range, its health will be $1491 + 3^2 = 1500$, which exactly equals the damage the tower will deal to it in three seconds.

In the fourth example, the tower at (1, 2) has a range of 1, and the tower at (3, 1) has a range of 2.

[Codeforces](#) (c) Copyright 2010-2024 Mike Mirzayanov
The only programming contests Web 2.0 platform
Server time: Apr/09/2024 13:56:35^{UTC+5.5} (h2).
Desktop version, switch to [mobile version](#).
[Privacy Policy](#)

Supported by



ITMO UNIVERSITY