Save Yourself

	1 sec	**512000KB**	**100**
Difficulty	Time Limit	Memory	Score

80/80 XP ℹ️　　⚠ 30/30 ℹ️

Description

You and some monsters are in a matrix. When taking a step to some direction in the matrix, each monster may simultaneously take one as well. Your goal is to reach one of the boundary squares without ever sharing a square with a monster.

Your task is to find out if your goal is possible, and if it is, print the *shortest length of the path* that you can follow. Your plan has to work in any situation; even if the monsters know your path beforehand.

Input Format

The first input line has two integers *n* and *m*: the height and width of the matrix.

After this, there are *n* lines of *m* characters describing the matrix. Each character is '.' (floor), '#' (wall), 'A' (start), or 'M' (monster). There is exactly one 'A' in the input.
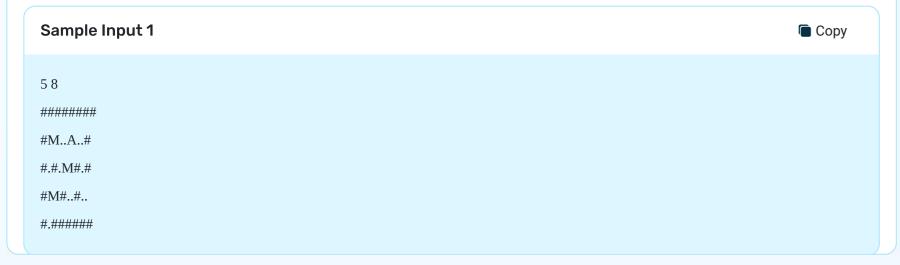
Output Format

First, print "YES" if your goal is possible, and "NO" otherwise.

If your goal is possible, also print the length of the shortest path that you'll follow.

Constraints

1 ≤ *n, m* ≤ 1000

Sample Input 1　　📋 Copy

```
5 8
########
#M..A..#
#.#.M#.#
#M#..#..
#.######
```

```cpp
68    {
69        queue<pair<int, int>> q;
70
71        for (auto i : monstors)
72        {
73            q.push(i);
74            vis[i.first][i.second] = 1;
75            closest_monstor_distance[i.first][i.second] = 0;
76        }
77        while (!q.empty())
78        {
79            pair<int, int> v = q.front();
80            q.pop();
81            for (auto i : neighbour(v.first, v.second))
82            {
83                if (!vis[i.first][i.second])
84                {
85                    vis[i.first][i.second] = 1;
86                    q.push(i);
87                    closest_monstor_distance[i.first][i.
                       second] = closest_monstor_distance[v.
                       first][v.second] + 1;
88                }
89            }
90        }
91    }
92
93    int main()
94    {
95        ios_base::sync_with_stdio(0);
96        cin.tie(0);
```

Sample Tests　　Manual Tests

Console　　　　　　　　　　Run on Sample