1915 가장 큰 정사각형

1915번 제출 맞은 사람 숏코딩 재채점/수정 채점 현황 내 소스 난이도 기여 강의 질문 검색

가장 큰 정사각형 성공 분류

시간 제한	메모리 제한	제출	정답	맞은 사람	정답 비율
2 초	128 MB	20844	6370	4564	29.133%

문제

n×m의 0, 1로 된 배열이 있다. 이 배열에서 1로 된 가장 큰 정사각형의 크기를 구하는 프로그램을 작성하시오.

0	1	0	0
0	1	1	1
1	1	1	0
0	0	1	0

위와 같은 예제에서는 가운데의 2×2 배열이 가장 큰 정사각형이다.

입력

첫째 줄에 n, m(1 ≤ n, m ≤ 1,000)이 주어진다. 다음 n개의 줄에는 m개의 숫자로 배열이 주어진다.

출력

첫째 줄에 가장 큰 정사각형의 넓이를 출력한다.

1. 접근 방식

1. 1인 지점에서 아래로 퍼져나가면서 모든 면이 1이면 크기를 늘려보자.
   => 숫자가 커지면 일반화가 안될뿐 아니라 연산이 너무 많아짐 (Fail)

2. 1인 지점에서 상 , 좌, 좌상 을 확인하며 이미 계산된 크기를 갱신하며
   아래 지점까지 나아간다. (DP)

0	0	0	0	0
0	0	1	0	0
0	0	1	1	1
0	1	1	1	0
0	0	0	1	0

0	0	0	0	0
0	0	1	0	0
0	0	1	1	1
0	1	1	2	0
0	0	0	1	0

좌 , 좌상 , 상의 값이 의미하는 것은 해당 좌표를 기준으로 생성된 정사각형의 한변의 길이

그 중에 만약 0이 있다면, 해당 칸은 1 x 1 이 되고,

모두 0 이 아니라면 가장 작은 크기를 가진 좌표 보다 1 큰 값의 정사각형이 만들어진다.

해당 수식을 일반화 하여 점화식을 세우게 되면

```
dp[i][j] = Math.min(Math.min(dp[i-1][j], dp[i][j-1]), dp[i-1][j-1]) + 1

최종 답을 계산하기 위해 최대 값을 갱신하면서 전체를 순회한다.

답으로 나온 숫자는 넓이가 아닌 한 변의 길이가 되므로

최종 답으로는 max 값의 제곱이 답으로 산출 된다.
```

코드

```
package Algo_Study_BOJ;

import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.util.StringTokenizer;

public class Main_1915_가장큰정사각형 {
  public static void main(String[] args) throws Exception {

    BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
    StringTokenizer str = new StringTokenizer(br.readLine());

    int N = Integer.parseInt(str.nextToken()); // N
    int M = Integer.parseInt(str.nextToken()); // M

    int[][] map = new int[N + 1][M + 1];
    int[][] dp = new int[N + 1][M + 1];
    for(int i = 1; i <= N; i++) {
      String tmp = br.readLine();
      for(int j = 1; j <= M; j++) {
        map[i][j] = tmp.charAt(j - 1) - '0';
      }
    }
    int max = 0;
    for(int i = 1; i <= N; i++) {
      for(int j = 1; j <= M; j++) {
        if(map[i][j] == 1) {
          dp[i][j] = Math.min(Math.min(dp[i-1][j], dp[i][j-1]), dp[i-1][j-1]) + 1;

          max = Math.max(dp[i][j], max);
        }
      }
    }
    System.out.println(max * max);
  }

}
```