Solution

The problem asks us to calculate the number of rooms on the map, in other words, to calculate the number of groups consisting of connected dots. One way to solve this problem is to consider the given grid as a graph where the floor characters represent the nodes and the vertical/horizontal adjacencies represent the edges. In this way, we transform this problem into a classical "connected component counting" problem that can be solved with some kind of graph search algorithm, such as depth-first search , breadth-first search or even floodfill.

The following code shows a recursive depth-first search solution that uses a boolean two-dimensional array to keep track of the visited rooms. When the algorithm finds an unvisited room cell, it increments the answer by one and it propagates recursively through its neighbors and marks them as visited to avoid recounting the same room several times.

This algorithm runs in a time and space complexity of O(NxM) allowing it to be a viable solution given the inicial constraints . The following animation shows how the algorithm works in the sample input.