Test assignment
for the Junior Python Developer position

Imagine that a telecommunications company is working on designing an efficient 7G-network layout for a new city. The city can be represented as a grid, where some blocks are obstructed and cannot have towers, while others can. The goal is to provide the maximum coverage with the minimum number of towers.

Task 1: Grid Representation
Create a class CityGrid that can represent the city as an N x M grid. During the initialization of the class, obstructed blocks are randomly placed with coverage >30% (we can change this parameter).

Task 2: Tower Coverage
Each tower has a fixed range R (in blocks) within which it provides coverage. This coverage is a square, with the tower in the center.
Implement a method in the CityGrid class to place a tower and visualize its coverage.

Task 3: Optimization Problem
Design an algorithm to place the minimum number of towers such that all of non-obstructed blocks are within the coverage of at least one tower. The algorithm cannot place towers on obstructed blocks.
Implement a method in the CityGrid class to display the placement of towers.

Task 4: Path Reliability
Imagine that data is transmitted between towers. For simplicity, assume that each tower can directly communicate with any other tower within its range.
Design an algorithm to find the most reliable path between two towers. The reliability of a path decreases with the number of hops (tower-to-tower links). So, a path with fewer hops is more reliable.

Task 5: Visualization
Implement functions to visualize the CityGrid, including obstructed blocks, towers, coverage areas, and data paths.
Use any Python plotting library of your choice, such as matplotlib or seaborn.

Bonus tasks (optional):

1. Extend the optimization problem: Now towers have a cost, and you have a limited budget. Modify your algorithm to maximize coverage while staying within the budget.

2. Consider different types of towers with different ranges and costs. How would this change your optimization approach?