Documentation

The **Game of Life** is a cellular automaton created by John Horton Conway in 1970. It simulates the evolution of a grid of cells based on simple rules that dictate whether a cell lives, dies, or is born in the next generation. Each cell interacts with its eight neighbors (adjacent cells in all directions), and its state in the next step depends on the number of live neighbors. This problem requires updating the grid to its next state based on these rules while ensuring all updates co-occur, making it a computationally intriguing task.

The algorithm provided efficiently handles this challenge by iterating through each cell in the grid and calculating the count of live and dead neighbors. A dictionary stores each cell's neighbor counts, allowing for systematic updates without prematurely modifying the grid. For each cell, its neighbors are determined based on their positions relative to the current cell. Boundary conditions are handled carefully to avoid accessing indices outside the grid, ensuring the algorithm operates reliably even at the edges of the grid.

The state transitions follow the defined rules of Conway's Game of Life. For live cells (state `1`), the algorithm ensures survival only if the cell has two or three live neighbors; otherwise, the cell dies due to underpopulation or overpopulation. For dead cells (state `0`), the algorithm checks if there are exactly three live neighbors; if so, the cell is revived, simulating reproduction. These updates are applied directly to the board using the precomputed neighbor counts, making the solution in-place and memory-efficient.

Handling boundary conditions is a crucial aspect of this implementation. Each neighbor's existence is verified before it is counted, ensuring that the algorithm does not encounter errors while processing cells at the edges of the grid. The use of an in-place update method eliminates the need for an additional grid, which minimizes memory usage and adheres to the constraints of the problem. This approach ensures that the next state of all cells is computed simultaneously, as required.

This implementation is well-suited for finite grids, but it can also be adapted for infinite grids by dynamically expanding the grid size as live cells approach the boundaries. Applications of Conway's Game of Life are vast and range from modeling biological systems and population dynamics to studying emergent behaviors in complex systems. It is also a popular educational tool to demonstrate the power of simple rules in producing complex patterns and behaviors, making it a valuable example of computational simulation.