TDD, Unit Testing, & Smart Unit Tests

Jeremy Clark
www.jeremybytes.com

TDD Description

Red

- Failing unit test

Green

- Just enough code to pass the test

Refactor

- Adjust implementation

Conway's Game of Life

Live Cell -> Dead

- Any live cell with fewer than two live neighbors dies, as if caused by under-population.

Live Cell -> Alive

- Any live cell with two or three live neighbors lives on to the next generation.

Live Cell -> Dead

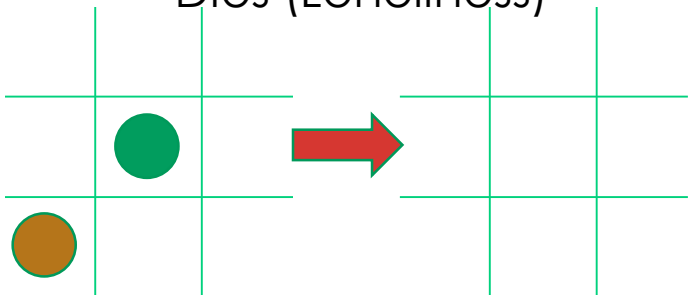
- Any live cell with more than three live neighbors dies, as if by overcrowding.

Dead Cell -> Alive

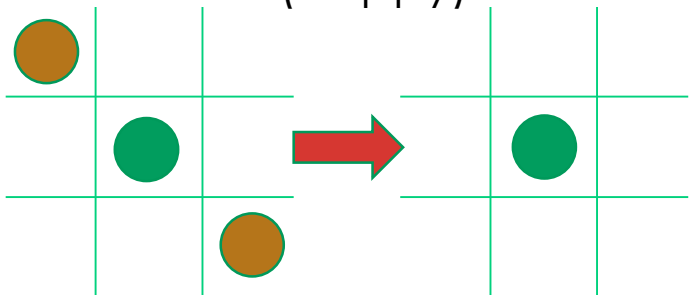
- Any dead cell with exactly three live neighbors becomes a live cell, as if by reproduction.

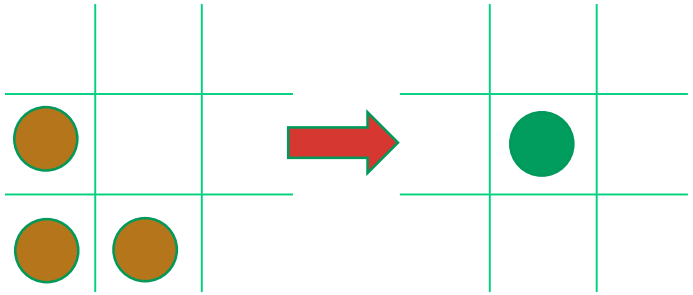
Examples

Dies (Loneliness)

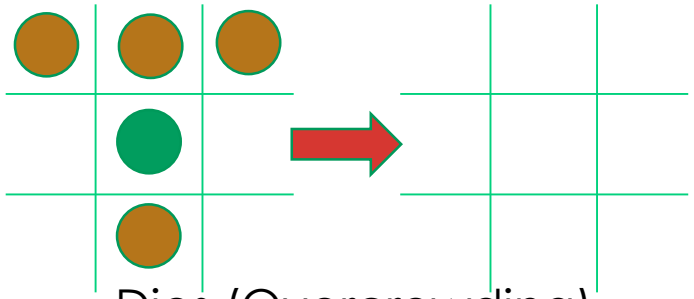


Lives (Happy)





Lives (Reproduction)



Dies (Overcrowding)

Smart Unit Tests

What is Smart Unit Tests?

- Available in Visual Studio 2015 (Preview)
- Generates tests based on existing code
- Follows all code paths to create 100% coverage
- Shows what the code *actually* does not what it is *expected* to do

Caveats

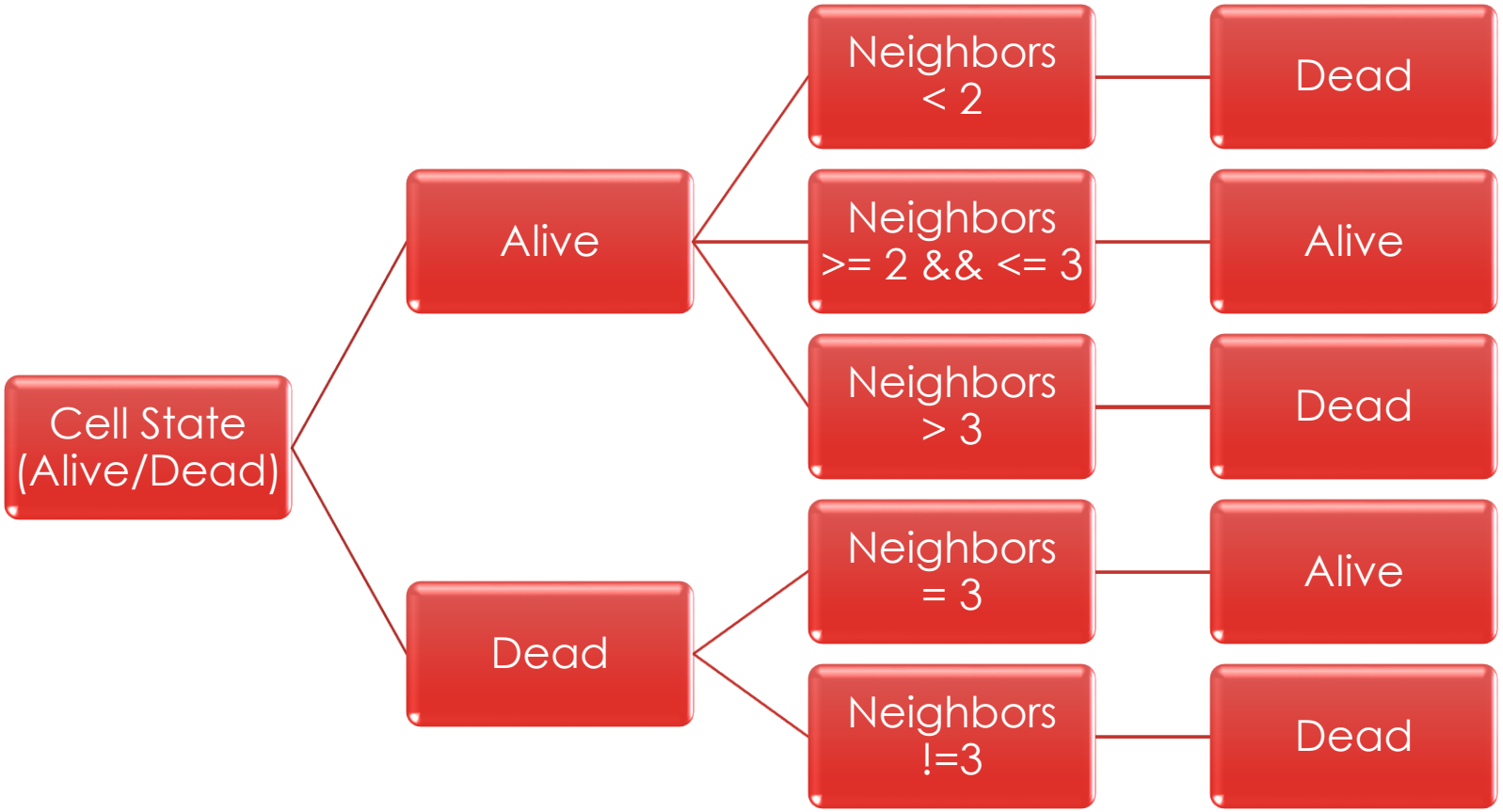
- Still in Preview
- Has some limitations

Sample Code

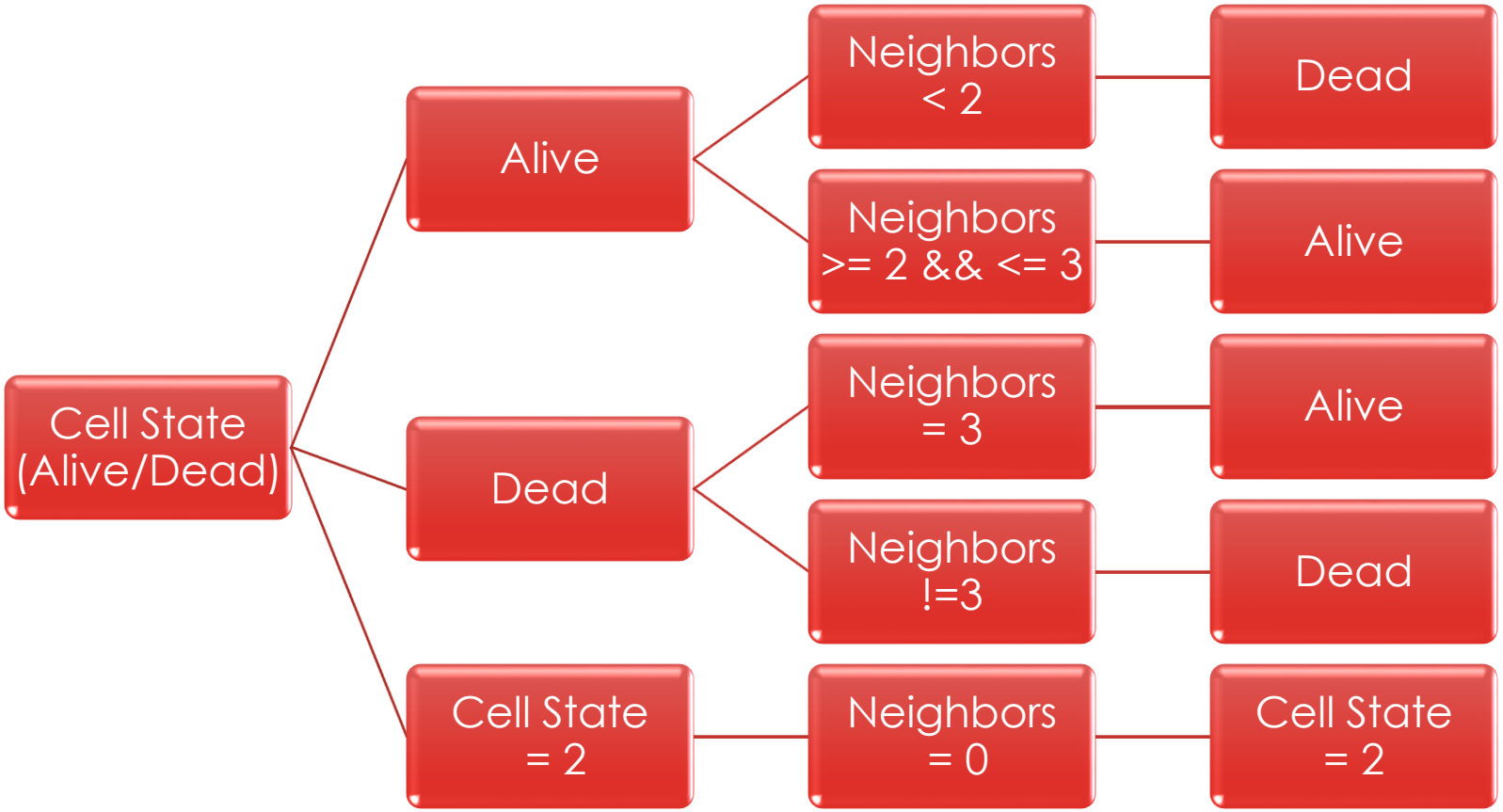
```
public static CellState GetNewState(
    CellState currentState, int liveNeighbors)
{
    switch (currentState)
    {
        case CellState.Alive:
            if (liveNeighbors < 2 || liveNeighbors > 3)
                return CellState.Dead;
            break;
        case CellState.Dead:
            if (liveNeighbors == 3)
                return CellState.Alive;
            break;
    }

    return currentState;
}
```

Code Path



Smart Unit Tests Code Path



Thank You

Jeremy Clark

- jeremy@jeremybytes.com
- www.jeremybytes.com