```
function NewBoard = Evolve(OldBoard,Rule)

%Evolves board of 2d cellular automata according to chosen rule,
   assuming
%periodic boundary conditions

switch Rule
    case 'Life'
        %Number of neighboring cells required for a dead cell to turn
           alive
        Birth = 3;
        %Number of neighboring cells required for an alive cell to
           survive
        Survival = [2 3];
    case 'Fredkin' %Chaotic rule, fill percentage converges to 50%
        Birth = 1:2:7;
        Survival = 0:2:8;
end

%Last column will be counted as adjacent to the first column, etc.
PeriodicBoard = ...
    [OldBoard(end,end), OldBoard(end,:),OldBoard(end,1)
    OldBoard(:,end),OldBoard,OldBoard(:,1)
    OldBoard(1,end), OldBoard(1,:), OldBoard(1,1)];

%Neighbors are positions reachable by a King
MooreNeighborhood = [1 1 1; 1 0 1; 1 1 1];
%Gives number of alive neighbors of each cell
Neighbors = conv2(PeriodicBoard, MooreNeighborhood, 'same');
%Modifies a cell state according to how many alive neighbors it has
NewBoard = ...
    ismember(Neighbors,Birth).*(1-PeriodicBoard)+...
    ismember(Neighbors,Survival).*PeriodicBoard;

%Removes the boundary copies
NewBoard = NewBoard(2:end-1,2:end-1);

end
```