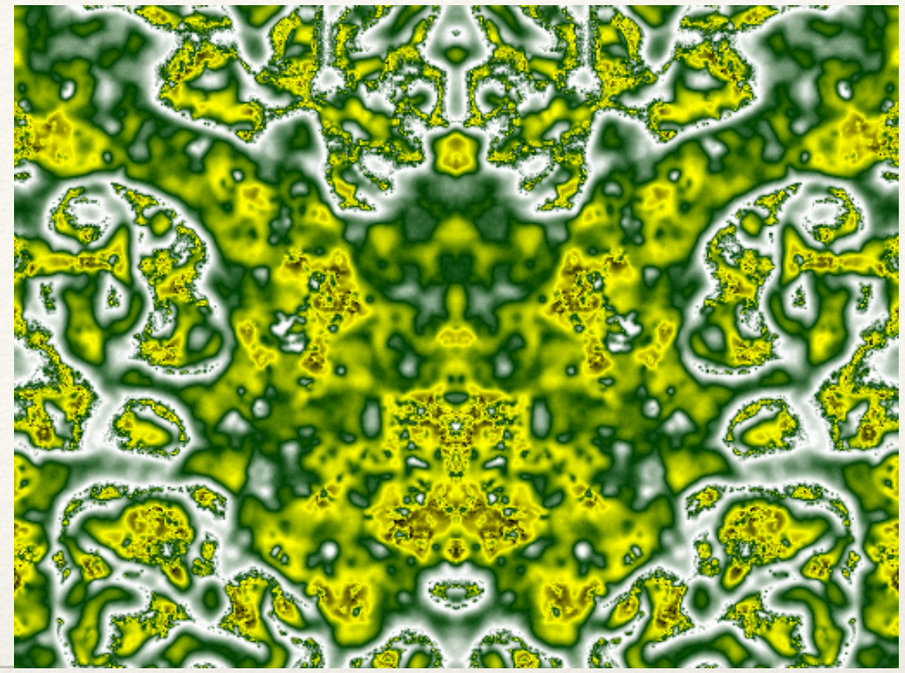
Dr. Sarah Abraham
University of Texas at Austin
Computer Science Department

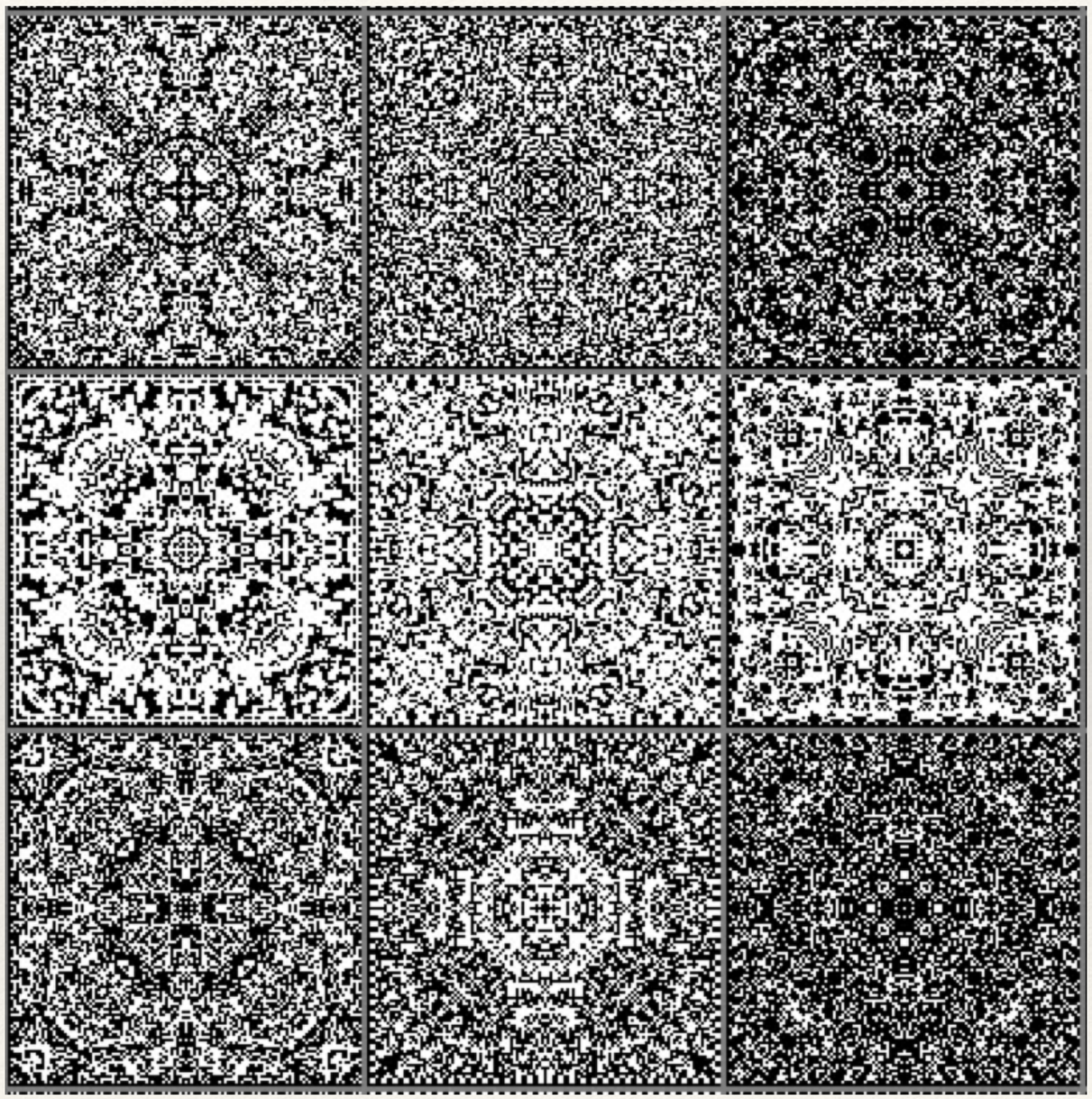


Cellular Automata

Elements of Graphics
CS324e

Cellular Automata (CA)

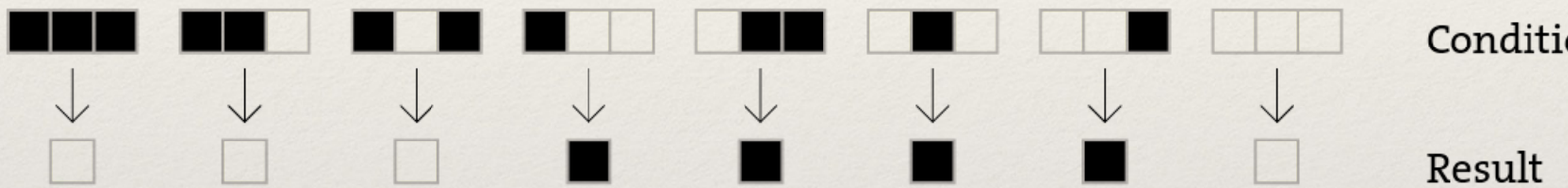
- Grid-based system
- Self-operating
- Rules-based
- Observation: simple rules can lead to complex patterns



(http://www.lactamme.polytechnique.fr/)

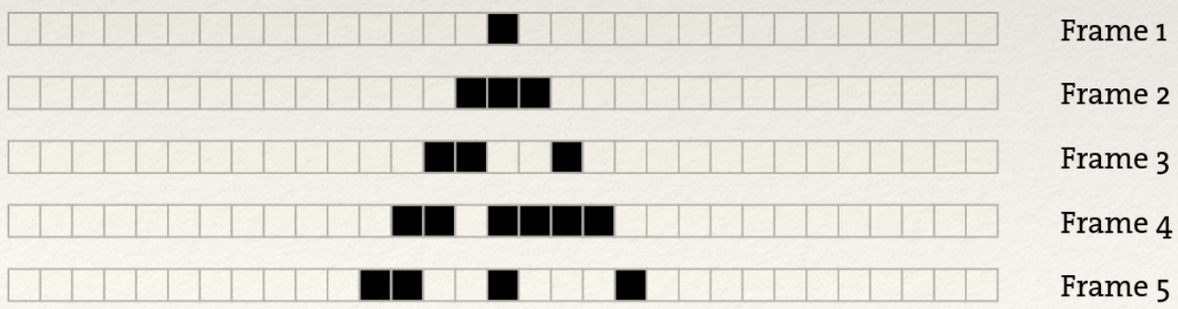
One-dimensional CAs

- Cells have two neighbors and two possible outcomes:



(Example rule set)

- Changes over time can be stacked to form 2D images:



Conway's Game of Life

- Two-dimensional CA
- Value of cell defined by values of neighboring cells
- Cells have two states: alive or dead
- At each time step, a cell will either:
  1. Be born
  2. Survive
  3. Die

Game of Life Rules

1. Underpopulation: Any cell with < 2 neighbors will die
2. Stable population: Any cell with 2-3 neighbors will survive
3. Overpopulation: Any cell with > 3 neighbors will die
4. Reproduction: Any “dead” cell with exactly 3 neighbors will be born

Simulating Game of Life

- Each cell represented by value 1 (alive) or 0 (dead)
- Current state of game represented by an array of integers
- Future state of game represents updated values of current array of integers
- Current and future states “swap” at each time step

Instapoll Question: Game of Life

- What needs to be initiated in the setup method for the Game of Life to run?

Autonomous Agents

- Autonomous agents are self-directed systems that sense and act upon the environment
- Agents have:
    1. Environmental input
    2. Rules for interacting with the environment
    3. Goals
- Structures and patterns can emerge even without global direction or understanding

Chris Langton's Ants

- Ant agents face a direction (north, south, east, west)
- Ant follows rules at each time step:

1. Move one frame forward
2. If on a white pixel, make the pixel black and turn right
3. If on a black pixel, make the pixel white and turn left

Question

- Assume ant `direction` maps to one of the directions / ints SOUTH (0), EAST (1), NORTH (2), and WEST (3).
- What does this line of code do to the ant?

```
direction = (direction-1) % 4;
```

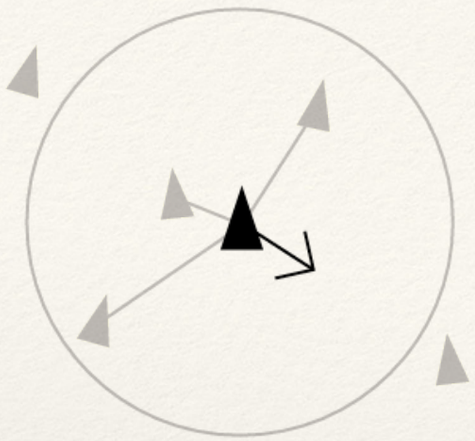
Gridless Autonomous Agents

- Autonomous agents can move freely in their environment
- Rules based on Cartesian position rather than grid cell position
- Same idea otherwise!

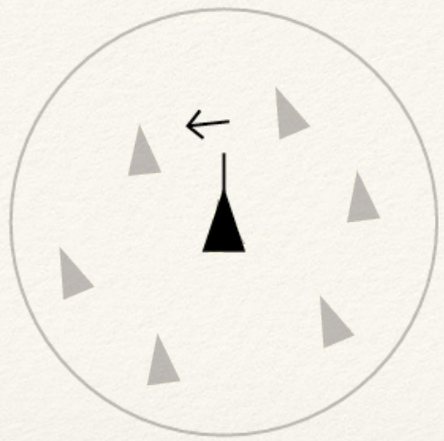
Flocking Behavior

- Boids system models bird flocking behavior
- No leadership or hierarchical structure to direct group
- Three rules for behavior:

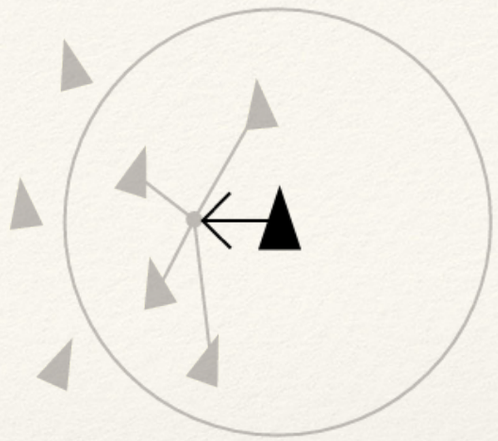
1. Separation: Move position to avoid crowding neighbors
2. Alignment: Move trajectory toward average heading of neighbors
3. Cohesion: Move to average position of neighbors



Separation:
Steer to avoid crowding local flockmates



Alignment:
Steer toward the average heading of local flockmates



Cohesion:
Steer to move toward the average position of local flockmates

1. Get position
2. Find all neighbors in radius
3. Calculate new heading and position based on neighbor’s data
4. Update position

Boids Example

https://www.youtube.com/watch?v=rN8DzlgMt3M

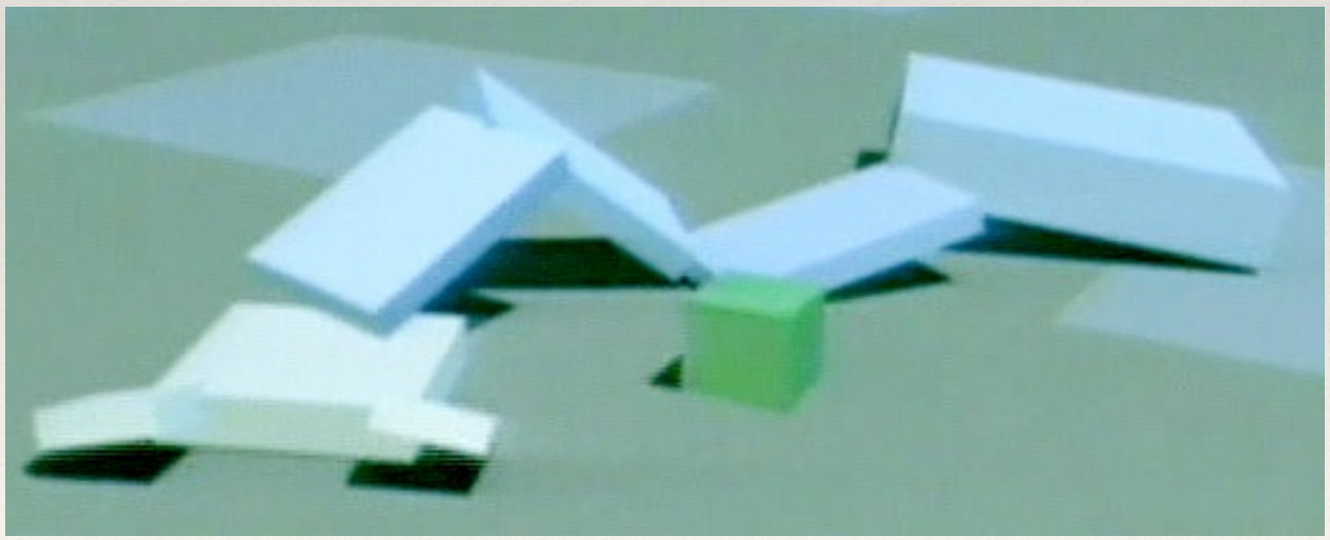
Note: Boids and A6

- This is true of all code for these projects, but remember that if you build on top of the existing Processing examples, **that code must be cited as not written by you** and **it will not count toward meeting the project requirements**
- That said, you can extend it or re-write it yourself to receive credit

Evolved Virtual Creatures

- In the extreme, we can further emulate life!
- Karl Sim's Evolved Virtual Creatures:

https://www.youtube.com/watch?v=JBgG_VSP7f8



Hands-on: Game of Life and Ants

- Today's activities:

1. Implement Conway's Game of Life. This may converge too quickly, so consider slowing the frame rate, or allowing interactivity to update the world

Or

2. Implement Langton's Ants. It may take a little while for the pattern to emerge