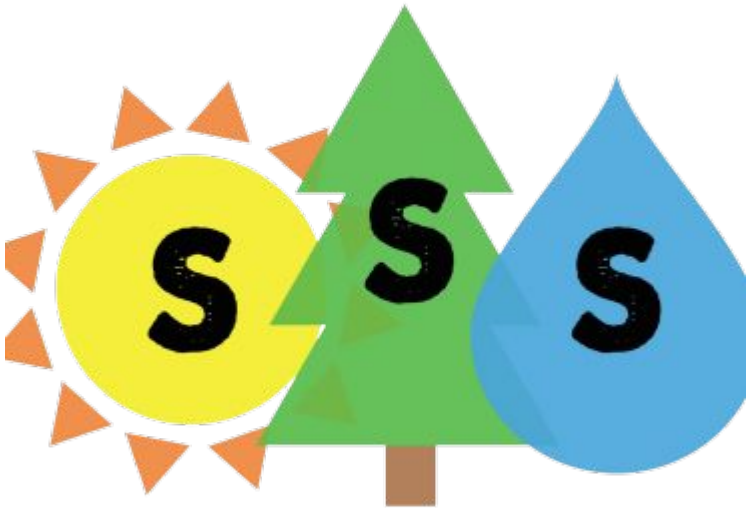


CS106S: Week 7

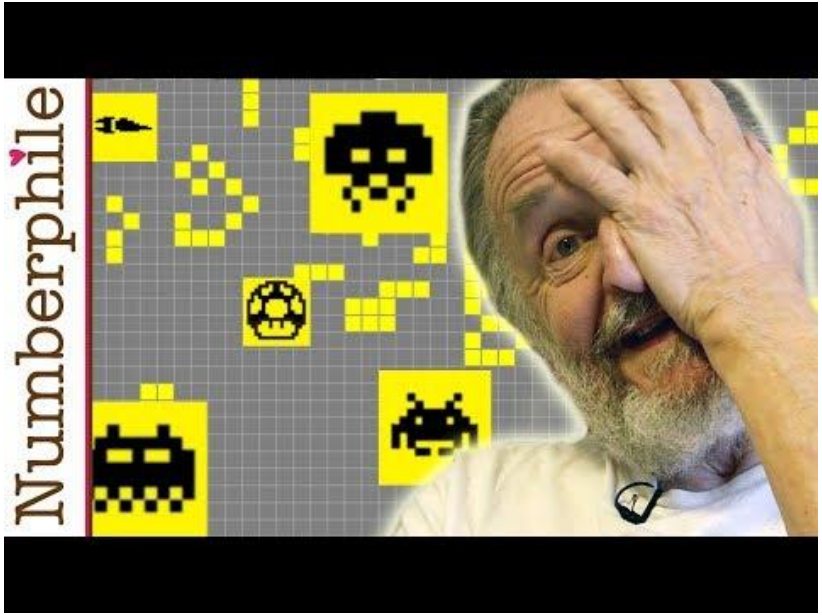
Automaton & Population Modeling

Download the starter code:
bit.ly/cs106sss

Plan For Today

1) Game of Life
2) What is Population Modeling?
3) Predator vs Prey Automaton

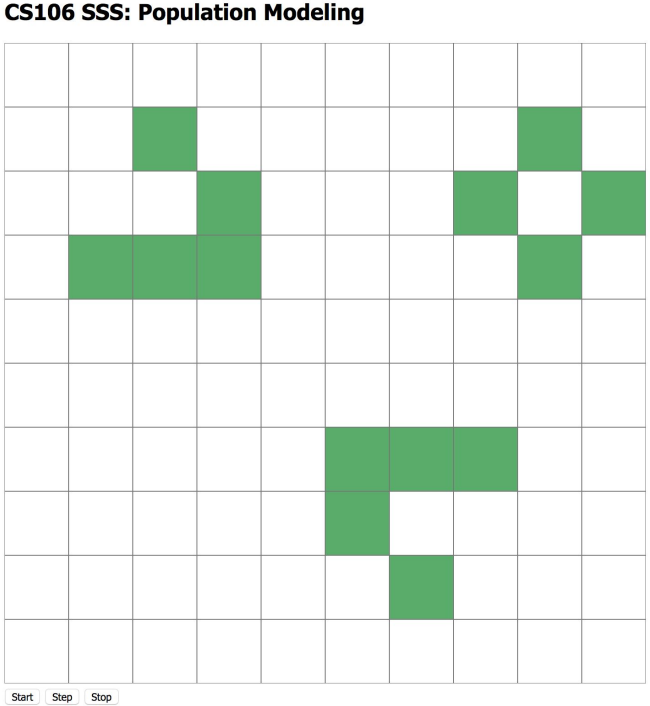
What is the Game of Life?



What are the rules?

1. If the cell is empty, and has 3 neighbors, it is born.
2. If it is alive, and has more than 3 neighbors, it dies from overpopulation.
3. If it is alive, and has less than 2 neighbors, it dies from isolation.
4. Otherwise, the cell remains the same.

How can we represent this on a computer?



Open the Starter Code, let's see it!

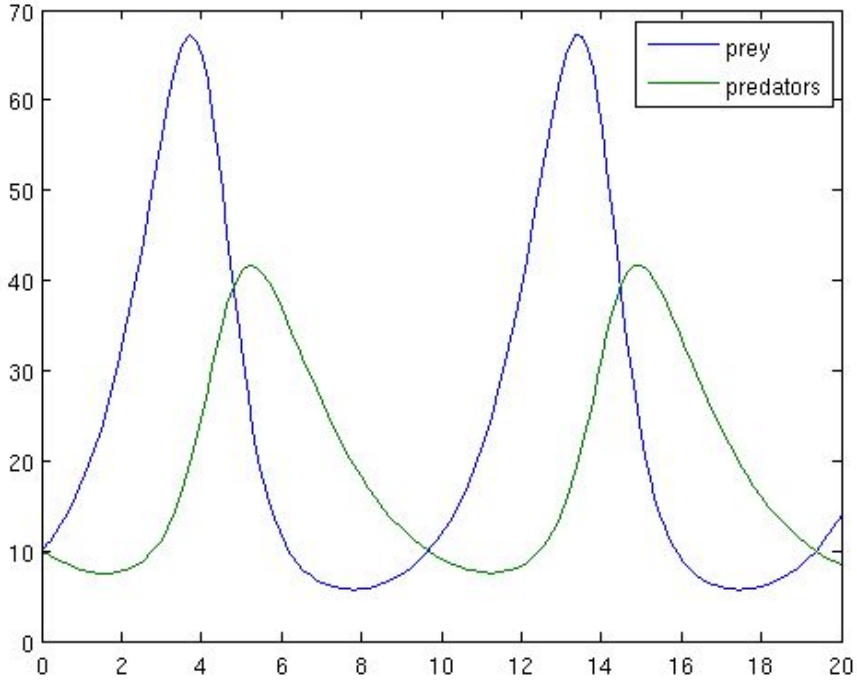
bit.ly/cs106sss

Subtle detail: why do we setNext, and then update?

Fill in the blanks, let's get it working!

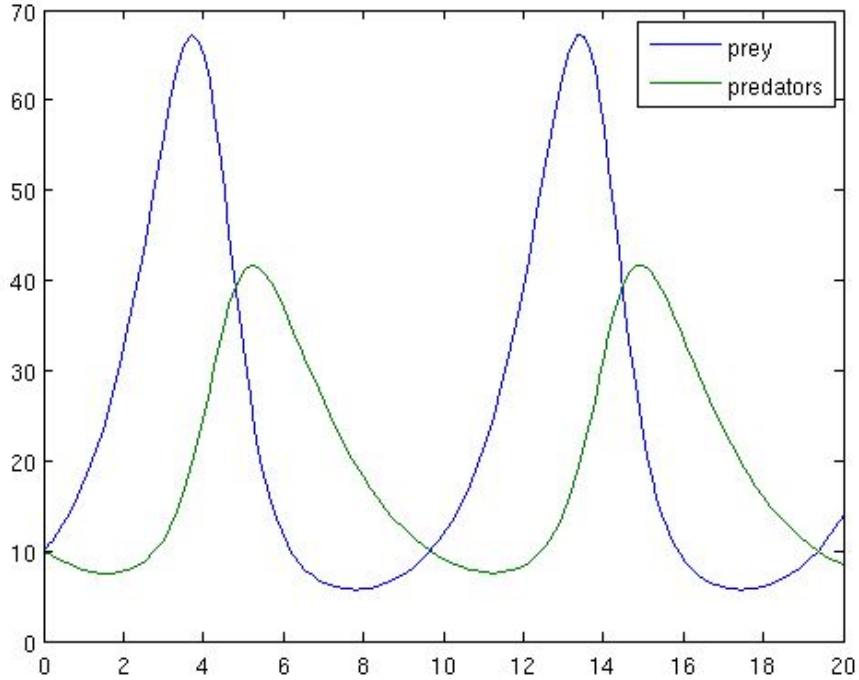
bit.ly/cs106sss

Predator v. Prey Modeling



1. The prey population finds ample food at all times.
2. The food supply of the predator population depends entirely on the size of the prey population.
3. The rate of change of population is proportional to its size.
4. During the process, the environment does not change in favour of one species, and genetic adaptation is inconsequential.
5. Predators have limitless appetite.

Lotka Volterra Equations



$$\frac{dx}{dt} = \alpha x - \beta xy,$$

$$\frac{dy}{dt} = \delta xy - \gamma y,$$

$$\frac{dx}{dt} = \alpha x - \beta xy,$$
$$\frac{dy}{dt} = \delta xy - \gamma y,$$

x is the number of prey (for example, rabbits);

y is the number of some predator (for example, foxes);

$\frac{dy}{dt}$ and $\frac{dx}{dt}$ represent the instantaneous growth rates of the two populations;

Can we adapt the Game of Life to model this, and take into consideration space, with roughly the same results?

http://www.vuuren.co.za/Theses/Project2.pdf

First, switch the logic files!

```
<!-- Logic for Game: choose one of the below -->
<!-- <script type="text/javascript" src="life.js"></script> -->
<script type="text/javascript" src="prey.js"></script>
```

Remainder of Class: Prey v. Pred

Your Final Goal: Look into possible extensions and implement/try to implement a feature that would augment our current app.

Check-Off

Announced in Class