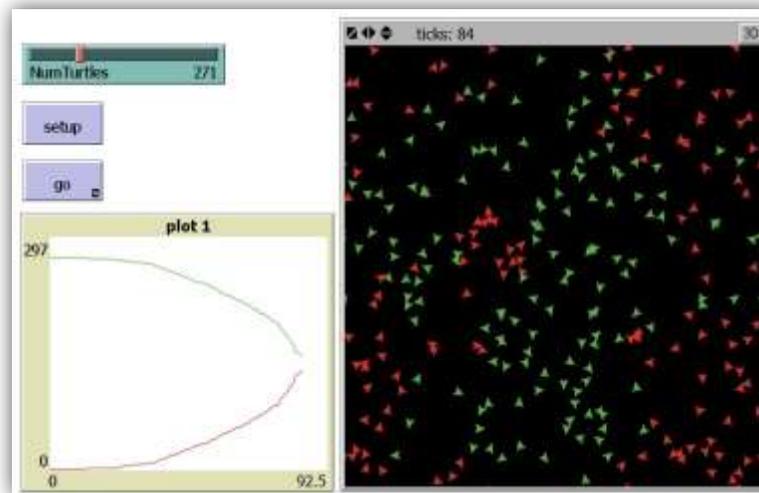


CS108L Computer Science for All

Module 9: Spread the Red: An Epidemiology Model



Model Overview:

In this lab, you will be writing model that represents a simple epidemic spreading through a group of agents. The epidemic starts with one sick agent and the disease spreads to the rest of the agents by contact (sharing the same patch). We will build the simple Epidemic Model first and then add different disease related things to the model (doctors, immunity, etc.). The more things you add, the more points you get!

The Basic Model:

The basic model has the characteristics:

- ❖ There are agents in the model.
- ❖ You should be able to change the number of agents using a slider!
- ❖ Each agent is either healthy or sick.
 - o You need to be able to visualize (be able to tell by looking at the agent) whether they are healthy or sick by their color (i.e. red agents are sick and green agents are healthy).
 - o You also need to set up Boolean Agent variable that contains that



information (i.e. turtles-own [sick?]) If this Boolean variable is true for an agent then that agent is sick. If it is false than that agent is healthy)

- ❖ Initially only one agent is sick.
- ❖ Sick agents get other agents sick. This happens when a sick agent collides with a sick agent (i.e. the sick agent shares the same patch as a healthy agent).
- ❖ Agents wiggle to move around.
- ❖ The world wraps around! Your agents live on a doughnut (a torus)!
- ❖ For the basic model - once an agent is sick, it stays sick forever: no recovery, no death, nothing. It is forced to wiggle around its world sick as a Well you get the idea.
- ❖ You are very anxious about your agents and like to keep track of their health. You graph the number of sick agents and the number of healthy agents giving each line a different color – there was a video that covered graphing so go back and watch it if you need help! Don't forget about ticks!

That's it for the basic model!

Coding Hints:

any? *agentset* returns true if agentset is not empty

turtles-here returns an agentset of all turtles on the local patch (including the turtle making the call)

other says to ignore the turtle making the call

So, putting it all together:

“if any? other turtles-here [....]” is a good way to run code that should only run when you are encountering another agent. This is standard NetLogo “collision” code. Of course you can further refine the agentset using the with command.

Example, if any? other turtles-here with [color = red]

Adding Features - The more you do, the more points you get – BUT YOU MUST DO AT LEAST ONE:

- ❖ You allow the poor sick agents to pass away peacefully after a period of time. To do this you will probably need a new agent (Turtles-own) variable that would contain a number that represents how long they have been sick (perhaps **SickTime**) and a new global variable that holds the maximum amount of time that the turtles can be sick (perhaps **MaxSickTime**).
- ❖ After wandering around aimlessly, the sick agents finally find a hospital and are



cured. The sad truth that they do not know is that they can get sick again! What does this mean in terms of your model? Will you set aside a part of your world for the hospital and make the patches a different color? How will you get the agents to notice that they have entered the hospital? What happens to the turtle's color and turtle's variable values?

- ❖ Good news, some of your agents are getting better on their own! They are actually able to fend off this dread disease and are cured without going to the doctor. Not only are they cured; they are IMMUNE to the disease afterwards. They are joyous and turn yellow – or whatever color means happiness to you. You probably will want to create a new agent variable that keeps track of whether or not they are immune. Remember not all the agents who are sick get better again and those that do will not be able to get sick again – the lucky.... Agents!