**ABM – Week 4 – Seminar – LVL2**

**Purpose**

This task will allow you to code simple interactions between turtles and patches. It will also prepare you for Coursework 2, in which you will use a similar model.

**Model**

Open the model tutorial\_4\_starting\_model.nlogo. This model provides a basic framework as a starting point for this task.

**Task**

Create a model where…

* Patches contain food that turtles can eat, with each patch having a maximum amount of food that it can contain (between 0 and 100). Food starts at its maximum value and grows back at a rate of 1 unit per tick if it is depleted.
* Each turtle (initial population 100) has an energy level (initially 100), which reduces by a fixed amount (between 2 and 10, varying between turtles) at every tick. Turtles can consume the food on their patch at a rate of 5 units per tick to replenish their energy. Turtles with zero energy will die.
* If a neighbouring patch contains more food than their current patch, turtles move to the patch with more available food.

Create graphs to monitor how key quantities change as the simulation runs.

**Extensions**

1. There are several fixed values in the model (e.g. turtles can eat a maximum of 5 units of food per tick; food grows back at a rate of 1 unit per tick; etc.). Replace some of these fixed values with variables and create sliders in the Interface for these variables. Investigate how the behaviour of the model changes as these variables are changed.
2. Rather than distribute food randomly, create “hills” of food, like the geography used in the butterflying hilltopping model. How does this affect the behaviour of the model?
3. **MORE CHALLENGING** Allow turtles to give birth to offspring if their energy level is sufficiently high. Perhaps these offspring will have the same energy usage as their parent, or perhaps it will be randomised. How does this affect the behaviour of the model? *[Useful primitive: hatch]*

**Step-By-Step Guide**

Creating the environment

Add patch variables to record the *maximum* food that a patch can contain and the *current* amount of food at a patch. For each patch, set the values of these variables as described in the task description. Colour the patches by their current amount of food.

*Useful primitives: ask, patches, pcolor, random, scale-color, set*

Creating the agents

Add turtle variables to record the current energy level of a turtle and the amount of energy that the turtle uses at each tick. For each turtle, set the values of these variables as described in the task description. Distribute the turtles randomly across the patches and adjust their size and colour so that you can see them clearly.

*Useful primitives: color, crt, random-pxcor, random-pycor, setxy, size*

Adding a behaviour: Consuming food

Create a new procedure to model turtles consuming food from their patch. For each turtle, create a temporary variable called meal\_size equal to the minimum of (i) 5 and (ii) the current food on the turtle’s patch. Increase the turtle’s energy by meal\_size and decrease the food available on the turtle’s current patch by the same amount. Recolour the patches based on their current food available.

*Useful primitives: let, list, min*

Adding a behaviour: Food growback

Create a new procedure to model the food on the patches growing back. For each patch, increase its available food by 1, unless this would surpass the maximum food that the patch can contain.

*Hint: You can do this in a very similar way to that described above, when defining the temporary variable meal\_size. Alternatively, you could use an if statement.*

Adding a behaviour: Using energy

Create a new procedure to model the energy use of turtles. For each turtle, its current energy should reduce by the appropriate amount. If this takes the turtle’s energy below zero, it should die.

*Useful primitives: die, if*

Adding a behaviour: Moving to the most abundant food source

Create a new procedure to model the movement of turtles to find the most abundant food source in their neighbourhood (the eight surrounding patches, plus their current patch).

*Useful primitive: uphill*

Creating graphs

Create two plots to monitor (i) the number of turtles and (ii) the mean energy use per tick of the surviving turtles. Give the plots appropriate titles and axis labels and set the axes to appropriate scales.

*Useful primitives: count, mean, of, plot*