LAB 3 – Sprite Interaction through Broadcasting of Messages

Broadcasting and Responding to Messages

On the left are the fundamental instructions for controlling the interaction among a set of sprites through the broadcasting of messages. Any given sprite can broadcast a message. Each broadcast message is simultaneously sent to every other sprite in the program.

The `when I receive` … instruction is placed at the start of all sprites that are to react to a given broadcasted message.

The `broadcast` … `and wait` message broadcasts a message to the other sprites, and does not continue to the next instruction until all sprites have finished their actions in response to the message.

The plain `broadcast` message broadcasts a message an immediately continues with the execution.

Scratch Sensing Instructions
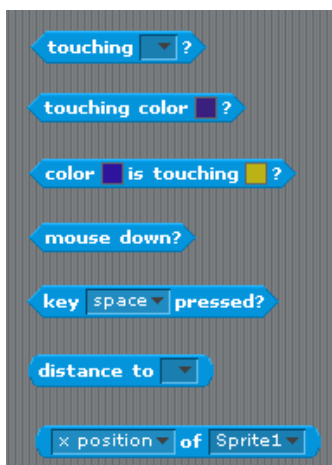
The `touching?` instruction can be used to determine if a given sprite is currently touching either an edge, the pointer, or a specific other sprite. The `touching color?` instruction can be used to determine if a given sprite is touching a specific color. To place the color in the instruction, use the eyedropper tool to select the color from a given sprite or background. The `color is touching color?` instruction similarly checks if one specific color is touching another.

The `mouse down?` and `key` … `pressed` instructions can be used to determine mouse clicks and key presses by the user. Note that the key presses that can be checked for includes the up and down arrow keys, the space bar, and letters and digits.

The `distance to` … allows the program to determine the distance from a given sprite to either another sprite or the pointer. The … `of Sprite` instruction allows the program to determine the following properties of another sprite: xloc, yloc, direction, costume # , size and sound volume.

Task

Create a scratch program that makes use of broadcasting and event handling. Use the Scratch sensing instructions for determining when something has occurred that other sprites should be notified of (i.e., by receiving a broadcasted message). You may add background(s), sounds, etc. to enhance the program. Be creative!

Some examples to give you the general idea. You can use these ideas, but you are encouraged to come up with your own idea.

<u>Balloons in flight</u>
Could have a number of balloons floating in the sky. They may pop whenever one balloon touches another, or then they get too high in the sky, or when they touch the edge of the screen, or when they are close to a certain object , etc.

<u>Ants</u>
Could have a number of ants randomly meandering around. (There is an ant costume you can use in the provided costumes. You will have to make it about 20% of its normal size.) There can be a piece of food that when a given ant finds it, it broadcasts to the other ants that it found food. The other ants can then move in that direction.

<u>Animals on the Prowl</u>
Animals that randomly meander around. When they touch, the predominant animal eats the other so that the consumed animal disappears. Maybe as a result the consuming animal gets bigger (fatter).

<u>Aquarium</u>
Have a lot of different fish randomly swimming around an aquarium, also containing a shark. When the shark gets near a given fish, it eats it. (Note there is a set of corresponding shark costumes, one with jaws closed, and the other with jaws open.)