Mirobot Lab

Instructor Guide

Overview

In this lab, students learn the basics of programming by using Scratch and programming the Mirobot to draw pictures and respond to collisions.

Learning Goals

- Understand the basics of coding: giving commands to a computer
- Understand the basics of loops: repeat this body of code

Personal Growth Goals

- Teamwork: for multiple parts of this lab, students will have to work with other students to draw figures that require multiple Mirobots.
- Patience: unfortunately, the Mirobots are not particularly fast robots. However, this will help the students practice their patience and get to know their mentors better :)

Skills Required

- A basic knowledge of geometric shapes

Resources Required

- One assembled Mirobot with 4 AA batteries per one or two students
- One computer with internet access per one or two students
- One Sharpie per Mirobot, and multiple extra multi-colored Sharpies for drawing
- One mentor per one or two students
- Multiple large (20x30 at a minimum) sheets of white paper per Mirobot. This is so each student group has space to do multiple drawings. Also bring multiple regular size sheets of white paper so the students can make art and easily take it home. If possible, also get one really large sheets that multiple Mirobots can use at once, so students can collaborate on their final drawings.
- Obstacles for Mirobot to collide into (at least one per Mirobot). These could be pieces of cardboard, books, boxes, etc. They must have planar surfaces.

Instructor Preparation

Turn on each Mirobot. Check that a computer can connect to the Mirobot's wireless network. While connected to the Mirobot's network, check that the browser has the necesary plugins to run ScratchX by going to the following link:
http://scratchx.org/?url=https://mirobot.github.io/mirobot-scratch/main.js#scratch

In Depth Description of Lab Activities

In this lab, students will work with Mirobots to learn Scratch and the basics of computational thinking. The bulk of the lab will have student use Mirobot to draw shapes, but at the end students will get to experiment with Mirobot's other capabilities. It build on the Introductory Lab, where students learned what a Mirobot was and experimented with moving and rotating it.

Part 1: Basic Drawing

In this part, students will use Scratch to draw basic shapes. First, have students use Scratch to draw a square. (Mentors should work with them to determine how to convert the "draw square" command to Scratch code made up of moveForwards and turns).

Then, have students draw an equilateral pentagon. If students are unsure of the number of angles in each corner of a pentagon, the mentors should work with them to figure it out.

Ask students what they noticed. If you ask them to draw an equilateral hexagon, what would they do? What about an equilateral dodecagon (a figure with 12 sides)?

Part 2: Looping

Once students realize that all they are doing when drawing planar figures is repeating the same blocks multiple times (move forward and turn left), it is time to introduce looping! Tell students there is a way to tell the program to keep repeating itself a certain number of times. Now have them write the same code, to make a square, except this time using a repeat loop.

Once they have finished, have the students modify their code to draw a hexagon. Is it easier?

Explain to the students that now that they consolidated their code into just a few lines, it is much easier to change the side length, angle, and number of times to repeat. What if, instead of drawing equilateral shapes, we changed the angle so it is slightly different from an equilateral shape, and we have it repeat 15 times? (Mentors should guide their students to picking an angle that is not divisible by 360.)

Once students are done, discuss why they got the shapes they got, and what factors caused different shapes to look different.

Part 3: Collision Detection

Explain that sometimes we don't know how much we want the robot to move, but rather we want the robot to move and we'll tell it when to stop. That is where Mirobot collision detection comes in.

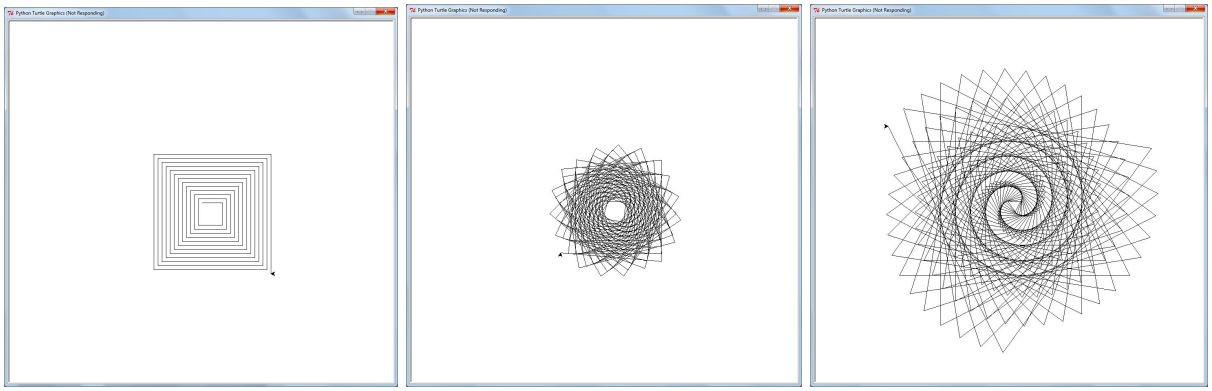
Have the students write a program where the Mirobot solely goes forward, except when it collides with something, in which case it turns left 90 degrees. Have students experiment with that program.

Explain to students that although it may be fun to have robots collide with something and turn, one of the big reasons to do this is so we have the freedom to decide how far we want them to go. For example, have the students try to make a series of concentric squares (like the ones below), where each Mirobot draws one square and they all start in the top-left corner. The students will each have to eyeball when it is time for their Mirobot to turn and insert the obstacle there, as opposed to programming their Mirobot with a preset distance.

Part 4: Free Drawing

In this part, students essentially get to work with their mentors and other students to draw whatever they want. Here are some ideas:

1. They can have Mirobot write out their name, by figuring out the lines and angles in all letters of their name
2. They can draw cool iterative shapes, such as the ones below:



All of these were made with the same iterative technique as the equilateral shapes (i.e. keep moving forward x and rotating by d), except their side length (x) got shorter each time. Images copied from: http://blog.rabidgremlin.com/2014/04/14/l-is-logo-and-wee-little-turtles/

3. They can experiment with drawing curves. Unfortunately, currently the only way of drawing curves is by drawing really small straight lines and then rotating by a small amount. However, once students are able to do that they hugely widen the range of possible things they can draw (stick figures, trees, etc.) If students are really interested, mentors can teach them how to wrap code inside a function, so they can repeatedly call it (for example, a drawCircle function).
4. Students can also work together! They can choose one long word to write and each take a letter. Or they can choose a landscape, and each take one component (one person draws a house, one person draws a stick figure, one person draws a tree, etc.)
5. Anything else! Students should be creative, and mentors should support them in whatever they want to do!

Nearing the end of the class, debrief with the students. What did they think of the Mirobots? What do they think about using robots to make art? What else could they imagine using the Mirobots for?

Lesson Plan

(:10) means that this part should be done by the tenth minute of the lesson

1. Part 1: Basic Drawing (:15)
2. Part 2: Looping (:35)
3. Part 3: Collision Detection (:60)
4. Part 4: Free Drawing (:90)