GirlsSolveIT Quest 3

Materials Required:

- Ping Pong Ball
- Paper Cup
- Straw
- Plastic Fork
- Tape
- Measuring tape
- Trolly tour map

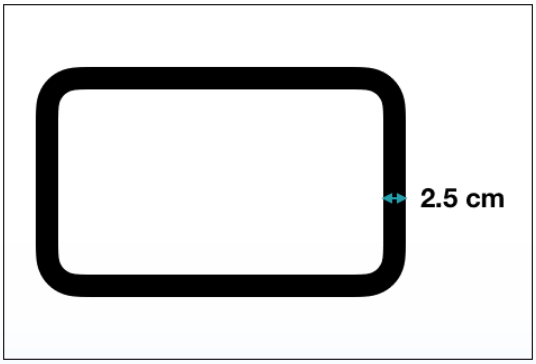
NOTE: Actions that are underlined are graded. All Challenges are mandatory for full credit. Bonus Challenges will earn your team extra points! Bonus challenges determine the winner in case of a tie! Extra points will be awarded to teams for their creativity.

Quest 3 Competition

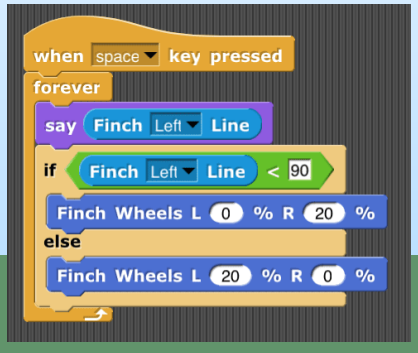
Concepts Covered:

- Moving & Turning
- **Rotation/Turns:** https://learn.birdbraintechnologies.com/finch/snap/program/3-3
  - Finch Turn Block
  - The Finch Turn block is used to control the angle and speed of your Finch's turn.
    - Use the drop down arrow to select "Right" or "Left."
    - Set the angle your Finch will turn (0-180) by typing a number into the __ degrees space.
    - Set the speed your Finch will move (0%-100%) in the __% space.

- **Finch Line Sensor:** The line-following sensors are located on the bottom of your Finch's body, on either side of the power button.
  - Infrared Light is the form of light that can't be seen using the naked eye but can be felt in the form of heat. For example, Infrared light waves are used by remote controls to change the channel on your TV.
  - The line sensors emit infrared light, and then detect how much bounces back. Some surfaces, like black tape, absorb infrared light. White paper and many other surfaces reflect infrared. You can use this to tell if your left or right line sensor is over a white or dark surface. The range of the line following sensors is 0 – 100, with 100 meaning that nearly all light is being reflected back to the sensor (white paper), and low numbers (0 to 50) indicating very little light is reflecting back to the sensor (black tape).
- **Line Tracking:** https://learn.birdbraintechnologies.com/finch/snap/program/16-3
  - **Basic Line Tracking**
  - To track a line, the Finch should turn right or left based on whether or not it detects the line.



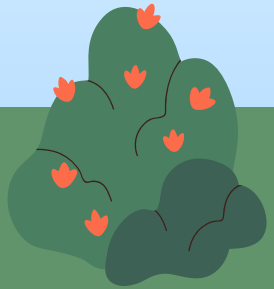
  - Let's think about tracking the left side of the line with the left line sensor as the Finch moves clockwise around the loop. When the left line sensor is over the black line, the Finch should turn left. When the left line sensor is over the white line, the Finch should turn right.
  - Notice that this sample code uses the Finch Wheels block to turn using only one wheel at a time. This means that both turns move the robot slightly forward. As the Finch repeats the if else block over and over, it will follow the line as it turns back and forth.



- Turning left or right based on a single line sensor is the simplest version of a line tracking algorithm, but there are many variations. An alternate line tracking algorithm described below
  - If the left line sensor is over white and the right is over black, turn right.
  - If the right line sensor is over white and the left is over black, turn left.
  - If both line sensors are over the same color, move forward in a straight line.

- **Distance Sensor:**
  - The distance sensor is located beneath your Finch's beak.
  - The distance sensor measures the distance between itself and an object placed in front of it. (FYI: This distance sensor is ultrasonic, meaning it detects sound waves, sort of like a bat or dolphin using sonar!)
  - The range of a sensor means the minimum and maximum numbers it can read. The range of the distance sensor is 0 to 275 cm. The reading will be more accurate for larger and closer objects.
- **Encoders:** https://learn.birdbraintechnologies.com/finch/snap/program/14-1
  - **The encoders are part of the Finch wheels, and each encoder measures how far its wheel has turned.** The diameter of a Finch wheel is 5 cm. To find what one rotation equals, you multiply the diameter by Pi. Pi = 3.14. This means that for one rotation, a Finch wheel moves about 15.7 cm. If we then want to know how many rotations it will take to move a given distance, you can divide that distance (in centimeters) by 15.7. For example, it will take about 1.9 rotations for the Finch to move 30 cm.
  - The Sensing menu contains two encoder blocks. The Finch Encoder block measures the number of wheel rotations that the wheel has moved since it was last reset. Before the Finch makes a movement that you want to measure, you need to use the Finch Reset Encoders block. This block sets both encoder values to zero.

- **Variables:** https://learn.birdbraintechnologies.com/finch/snap/program/18-1
  - As covered in Quest 2.
- **Loops:** https://learn.birdbraintechnologies.com/finch/snap/program/16-4
  - As covered in Quest 2.
- **Functions:**
  - What is a function? Imagine you have a magical box called a "function." This special box can do specific tasks for you. Let's break it down:
    1. Naming the Box: First, you give your box a name. Let's call it "MagicDrawer."
    2. What's Inside?: Inside the MagicDrawer, you put a set of instructions. These instructions tell the MagicDrawer what to do. For example:
       - "MagicDrawer, draw a smiley face!"
       - "MagicDrawer, count from 1 to 10!"
       - "MagicDrawer, add 5 and 3!"
    3. Reuse Magic: Now, here's the cool part! Once you've created the MagicDrawer, you can use it over and over again. Whenever you need a smiley face, counting, or adding, you just say, "MagicDrawer, do your thing!" And it does it instantly!
    4. Custom Magic: You can even customize your MagicDrawer. Maybe you want it to draw hearts instead of smiley faces. No problem! You just change the instructions inside.
    5. Sharing Magic: Guess what? You can share your MagicDrawer with your friends! They can use it too. Imagine if everyone had their own MagicDrawer for different tasks!
  - So, a function in programming is like your magical box. It holds instructions, can be reused, and makes coding easier.
  - To create a function in Snap:
    1. Navigate to any section you want to add the function to, if it's focused on moving Finch, use the Motion section
    2. Choose 'Make a block' at the bottom of the left-hand side
    3. Name the function
    4. Drag over the steps you want to repeat each time the function is called
    5. If you need more help with functions, here's a video showing how to create a function to draw a circle: https://www.youtube.com/watch?v=bvqR3WDVmOk



Challenge 1

- Finch is loving this road trip! Before getting back on the road, Finch checks her gas gauge and decides to setup a quick alert to let her know when she's running low on gas. Finch's road trip is 214 miles long, and the gas tank allows her to travel 107 miles.
- **Time to build a function!** Finch needs a counter that counts down from 107 and alerts her when she needs to fill up again.
    - **Calculate wheel rotation and translate that into measuring the distance traveled.**
    - **Once the distance traveled reached 20% of 107, the light on Finch's car changes to Yellow.**
    - **Once the distance traveled reached 80% of 107, the light on Finch's car changes to Red.**
- **When the light turns Red, Finch needs to stop at a gas station and fill up.**
    - **After Finch fills up gas, reset the counter to 107, and change the light on Finch's car to Green to show the full tank.**
    - **Note: the above checks should be within a function**

Challenge 2

"Ah great, now my tank is all filled up for some sightseeing!". Finch had read up on a lot of fun stuff to see in Atlanta and was really looking forward to this pitstop!

She decides to take a trolly tour around the city so that she can cover multiple locations in a short time. Her tour will cover:

- Centennial Olympic Park
- Martin Luther King Jr. National Historic Site
- The Woodruff Arts Center
- Georgia State Capitol

Use the Trolly Map shared in the kit.

Help Finch navigate through the tour along the designated path.

- **Program Finch to follow the black line path set for the trolly tour using the line tracking sensors**
- **Traverse the entire path of the tour.**
- **Add sounds / lights/ or anything else to convey Finch's excitement while sightseeing**

Challenge 3

Ah finally my tank is all filled up and it's time to head off straight to Florida!! Next stop- the amusement park! Finch hummed along with the crickets in the night sky, merrily continuing her journey. She halted to a stop as she noticed a humongous boulder blocking her path! It was too wide to go around it. Finch could of course fly over it, but she knew that humans coming this way would get stuck. Her second nature to help others prevented her from going her own way. She notices a big log a wood by the side of the road. She decides to use the log of wood as a jousting stick to knock the boulder.

Help Finch knock the boulder away!

To get ready for this challenge, you need to set up the target. The ping pong ball will represent the boulder.

Place the ping pong ball in the paper cup.

Next, let's create the jousting stick. Tape the plastic fork to the straw to make the jousting stick, then tape the jousting stick to Finch.

Now that the challenge is set up, Finch needs to take her place to start the match. Place Finch 60 cm (2 feet) back and 30 cm (1 feet) over from the paper cup.

To get a better idea of the task watch this video: https://www.youtube.com/watch?v=9AqfxIQ2vZQ

- **Place ping pong ball on the paper cup**
- **Tape the plastic fork to the straw to make a jousting stick.**
- **Tape the jousting stick to your Finch robot.**
- **Place Finch about 60 cm (2 feet) back and 30 cm (1 feet) over from the paper cup**
- **Program Finch to move forward 60 cm (2 feet) and 30 cm (1 feet) to the right, towards the paper cup**
- **Program Finch to spin around when near the cup to knock the ping pong ball of the cup using the jousting stick**

Bonus Challenge 1 (Optional):

Can you help Finch avoid obstacles along the path? There may be obstacles along the path (eg: construction work, debris, parades etc)

- **Paste the images of the obstacles onto the cups and place the cups along the path at the marked positions.**
- **Program Finch to follow the black line path set for the trolly tour using the line tracking sensors**
- **Program finch to use the distance sensors to look out for any obstacles along the path**
- **Program a function that gets called every time Finch is near an obstacle. This function should have Finch**
  - **Move around the obstacle**
  - **Come back to the designated black line path**
- **Finch should continue to traverse forward through the tour each time she returns to the path after overtaking the obstacle**
- **Traverse the entire tour without crashing into any obstacle**

Bonus Challenge 2 (Optional):

Can you help Finch calculate the distance she had to travel to finish the tour?

- **Use Finch Encoders block to calculate the total distance travelled along the entire tour**
- **Program Finch to display the calculated total distance**

"Wow, that was such a fun tour!" exclaims Finch.

To submit the Quest:

- Record the entire Quest in one continuous and complete video.
- Upload the video to YouTube – be sure the account is public.
- Upload pictures or videos to Instagram for extra points – be sure the account is public.
- Screenshot all the Snap code used to complete the quest – be sure the screenshots are clear.
- Submit the quest with a link to your public YouTube video, and upload screenshots of the code.
- If reviewers are unable to read your screenshots or access your team's YouTube Video because the url is not set to public, your team will receive zero points for those items.
- Remember, all underlined actions are being graded. Be sure to complete them all for an opportunity for full credit.