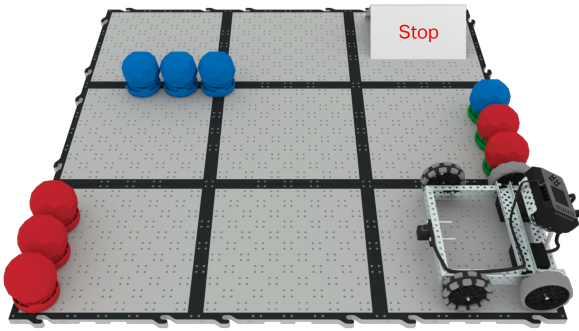




Sensor Maze

Use the Optical Sensor to solve a maze!

Step by Step

1. Build the [BaseBot](#). Attach a 1x2x1x16 C-Channel to the front and attach an Optical Sensor facing forward.
2. Open VEXcode EXP Python. Then, open the BaseBot (Drivetrain 2-motor) template, and configure the Optical Sensor.
3. Set up your maze using the EXP Field as shown above, or create a maze with classroom materials.
4. Create the starter project on the right. Download and run the project on your robot.
5. Add to the project to have the robot complete the sensor maze.

```
# Begin project code
optical_1.set_light_power(50, PERCENT)
while not optical_1.is_near_object():
    drivetrain.drive(FORWARD)
drivetrain.drive_for(REVERSE, 25, MM)
drivetrain.turn_for(RIGHT, 90, DEGREES)
while not optical_1.is_near_object():
    drivetrain.drive(FORWARD)
drivetrain.drive_for(REVERSE, 25, MM)
drivetrain.turn_for(RIGHT, 90, DEGREES)
```

'LEVEL UP'

- **Timed Trial** - How fast can your robot complete the maze? Experiment with drive and turn velocity to have the robot complete the maze in the quickest time possible!

```
drivetrain.set_drive_velocity(50, PERCENT)
```

```
drivetrain.set_turn_velocity(50, PERCENT)
```

- **Labyrinth** - Make the maze more complex, adding more walls or dead ends. Can you still solve the maze?

Pro Tips

- **Set the LED** - The Optical Sensor readings can be affected by ambient light. Experiment with LED light settings and see how this affects your robot's ability to detect the "walls" of the maze.
- **Adding Loops** - Want to repeat the same snippets of code? Use a **while True** command to repeat the code indented forever, or use a **for** loop to repeat code a specific number of times.