PRIME LESSONS

By the Makers of EV3Lessons

SQUARING ON LINES

BY SANJAY AND ARVIND SESHAN

This lesson uses SPIKE 3 software

LESSON OBJECTIVES

Learn how to get your robot to square up (straighten out) when it comes to a line

Learn how squaring (also known as aligning on a line) can help the robot navigate

Learn how to improve initial code for aligning by repeating a technique

Practice creating a useful async function with parameters

Practice having the runloop run multiple async functions concurrently

REVIEW

Move Steering lets you control both motors at the same time

What if you want to move or stop one motor at a time?

- Use the Single Motor Commands
- See the Knowledge base for the available commands

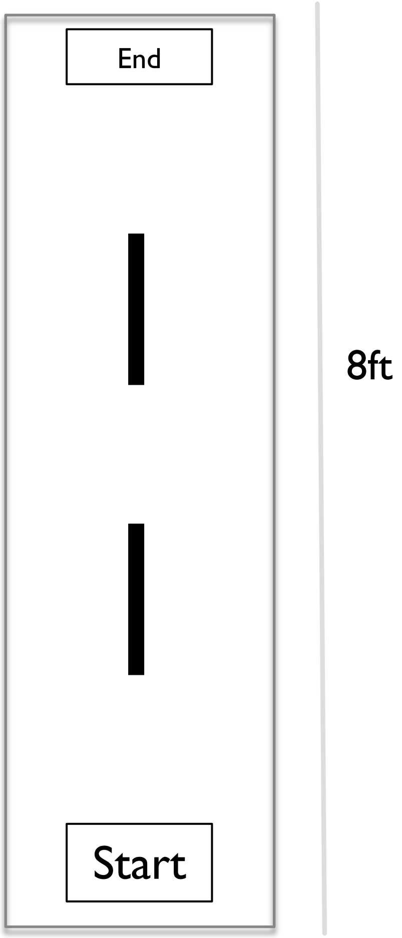
WHY ALIGN/SQUARE ON A LINE?

Aligning on a line helps the robot navigate

- Robots get angled as they travel farther or turn (the error accumulates)
- Aligning on a line can straighten out a robot.
- Aligning can tell a robot where it is when it has to travel far

Example Goal: Your robot must deliver an object only inside a small END area. The distance between start and end is 8 feet (~2.5m)

- Do you think your robot can travel 8 feet and continue to be straight?



THREE EASY STEPS TO ALIGN

Challenge: Make the robot straighten out (align/square up)

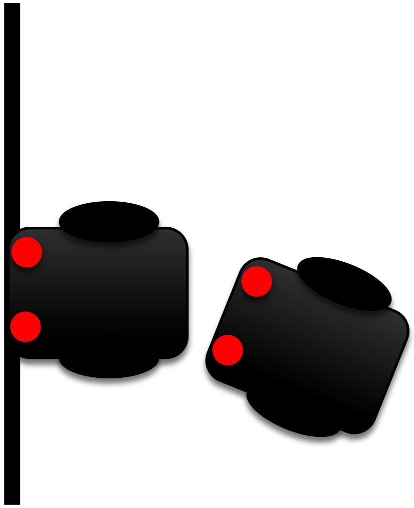
STEP 1: Start both motors and a loop

STEP 2: Stop one motor when the sensor on the corresponding side sees the line

STEP 3: Stop moving the second motor when the sensor on that side sees the line

STEP 4: Exit the loop

Hints: Use an async function with parameters, Single Motors and a loop



NOTES ON THE SOLUTION

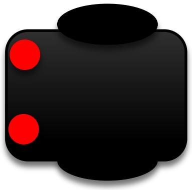
The robot used is Drive Base 1 with 2 color sensors

Our solution uses 2 Color Sensors (connected in ports A and B).

Our solution assumes that the color sensor on port A is next to the wheel on motor port C and color sensor on port B is next to the wheel on motor port D.

You should adjust the ports as needed

Your color sensors should NOT be placed right next to each other



BASIC SOLUTION: MOVE UNTIL LINE

```
from hub import port
import motor, color_sensor, runloop, sys

async def main():
    # Start the motors
    motor.run(port.C, -200)
    motor.run(port.D, 200)
    foundC = False
    foundD = False
    # wait for color sensors to detect black and stop motors
    while (not foundC or not foundD):
        if(color_sensor.reflection(port.A) < 50):
            motor.stop(port.C)
            foundC = True
        if(color_sensor.reflection(port.B) < 50):
            motor.stop(port.D)
            foundD = True
    sys.exit("Stopping")

runloop.run(main())
```

ALTERNATE SOLUTION: USING CONCURRENT FUNCTIONS

```
from hub import port
import motor, color_sensor, runloop, sys

async def all_done():
    return (motor.velocity(port.C) is 0 and motor.velocity(port.D) is 0)

# Function to move motor until the sensor in front of it senses black
# Parameters:
# motor_port: The port of the motor
# sensor_port: port of the color sensor in front of the motor
# direction: 1 for clockwise, -1 for counterclockwise
async def move_until_black(motor_port, color_port, direction):
    motor.run(motor_port, 200 * direction)
    while color_sensor.reflection(color_port) > 50:
        await runloop.sleep_ms(50)
    motor.stop(motor_port)

async def main():
    # create two async functions to send to the runloop
    a = move_until_black(port.C, port.A, -1)
    b = move_until_black(port.D, port.B, 1)
    # run both the functions together
    runloop.run(*[a,b])
    # wait until both motors have stopped
    await runloop.until(all_done)
    sys.exit("Stopping")

runloop.run(main())
```

IMPROVING YOUR CODE

What do you notice about the solution we just presented?

The robot is not always perfectly straight (aligned) at the end of it.

Both color sensors are on the line, but the robot stops at an angle if you started at a sharp angle

Challenge Continued: Think about how you can improve this code so that the robot ends straighter

Solution: repeat the align except look for white this time

CREDITS

This lesson was created by Sanjay Seshan and Arvind Seshan for Prime Lessons

Additional contributions by FLL Share & Learn community members.

More lessons are available at www.primelessons.org