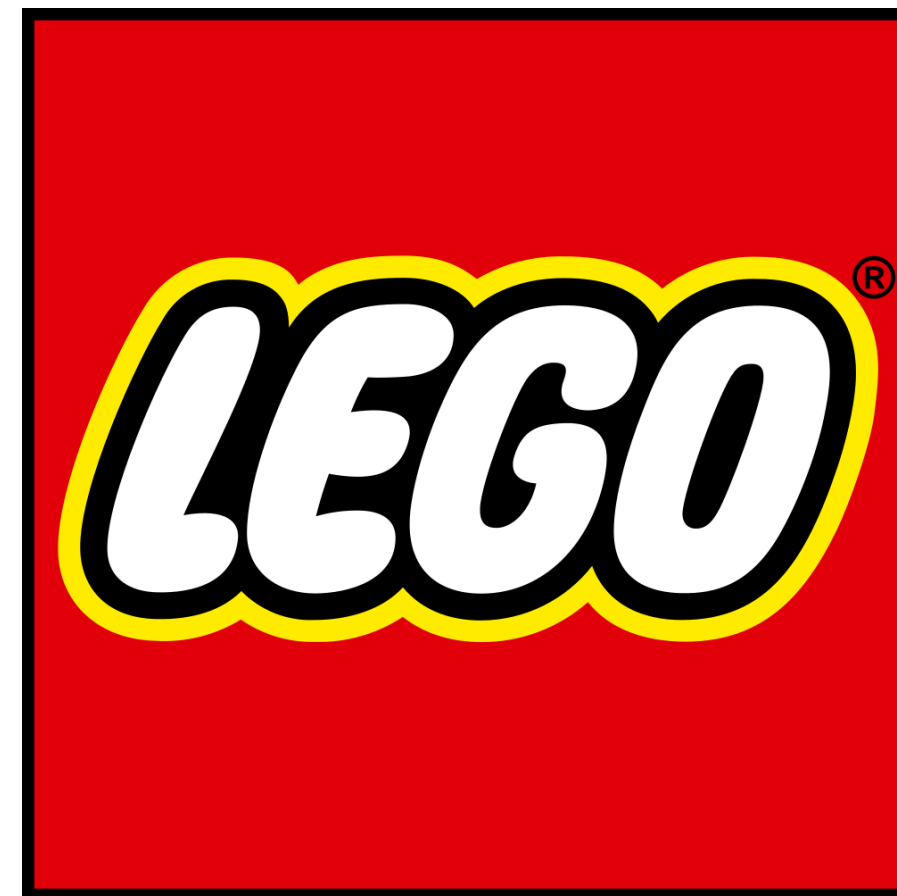


Parkside Montessori



May 16, 2024

Review

```
from hub import port
import runloop
import motor
import motor_pair

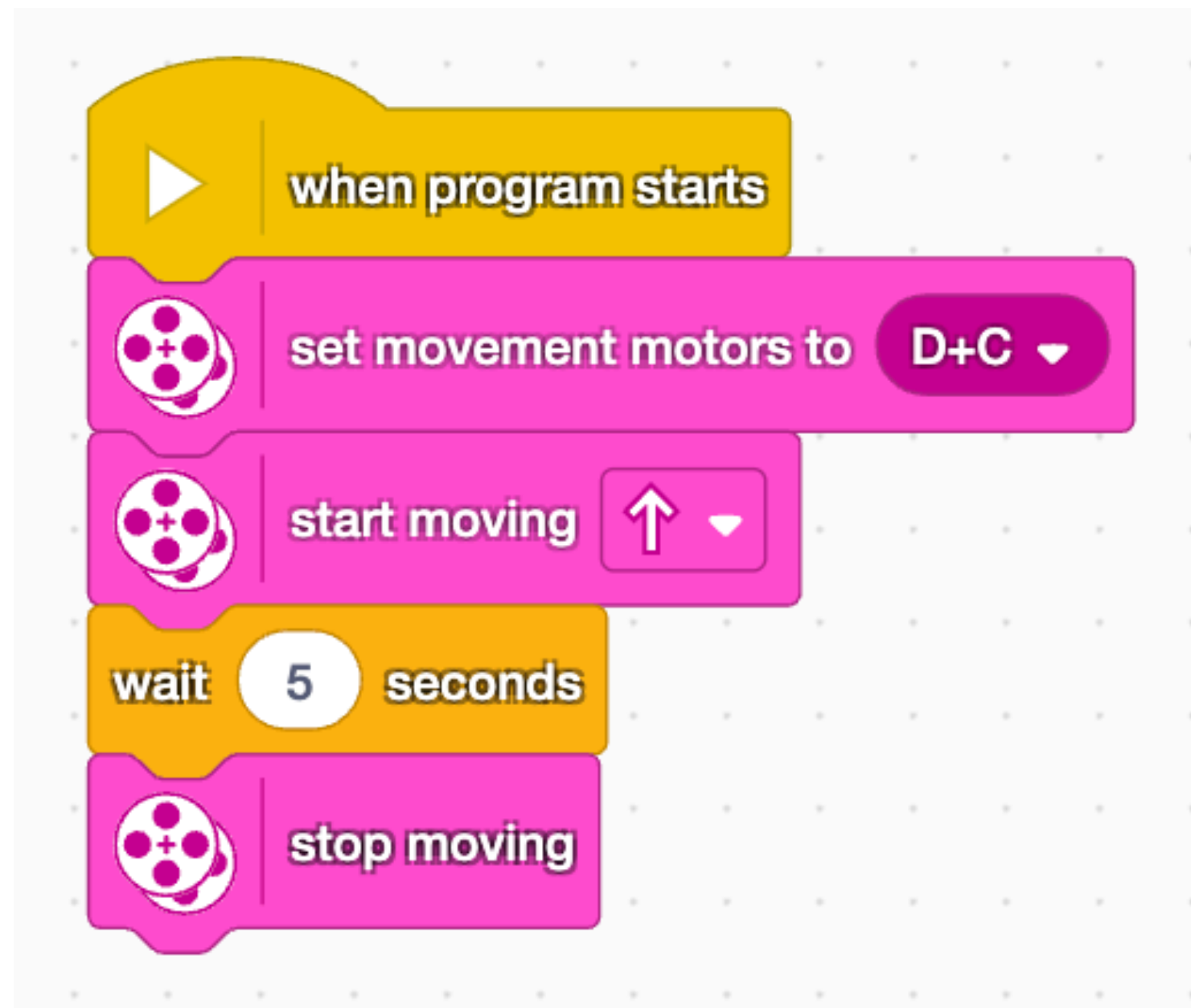
async def raise_arm():
    await motor.run_to_absolute_position(port.A, 45, 300)
    print("Arm raised")

async def main():
    motor_pair.pair(motor_pair.PAIR_1, port.D, port.C)

    await raise_arm()
    await runloop.sleep_ms(1000)
    await motor_pair.move_tank_for_time(motor_pair.PAIR_1, 1000, 550, -550)

runloop.run(main())
```

Continuous Movement



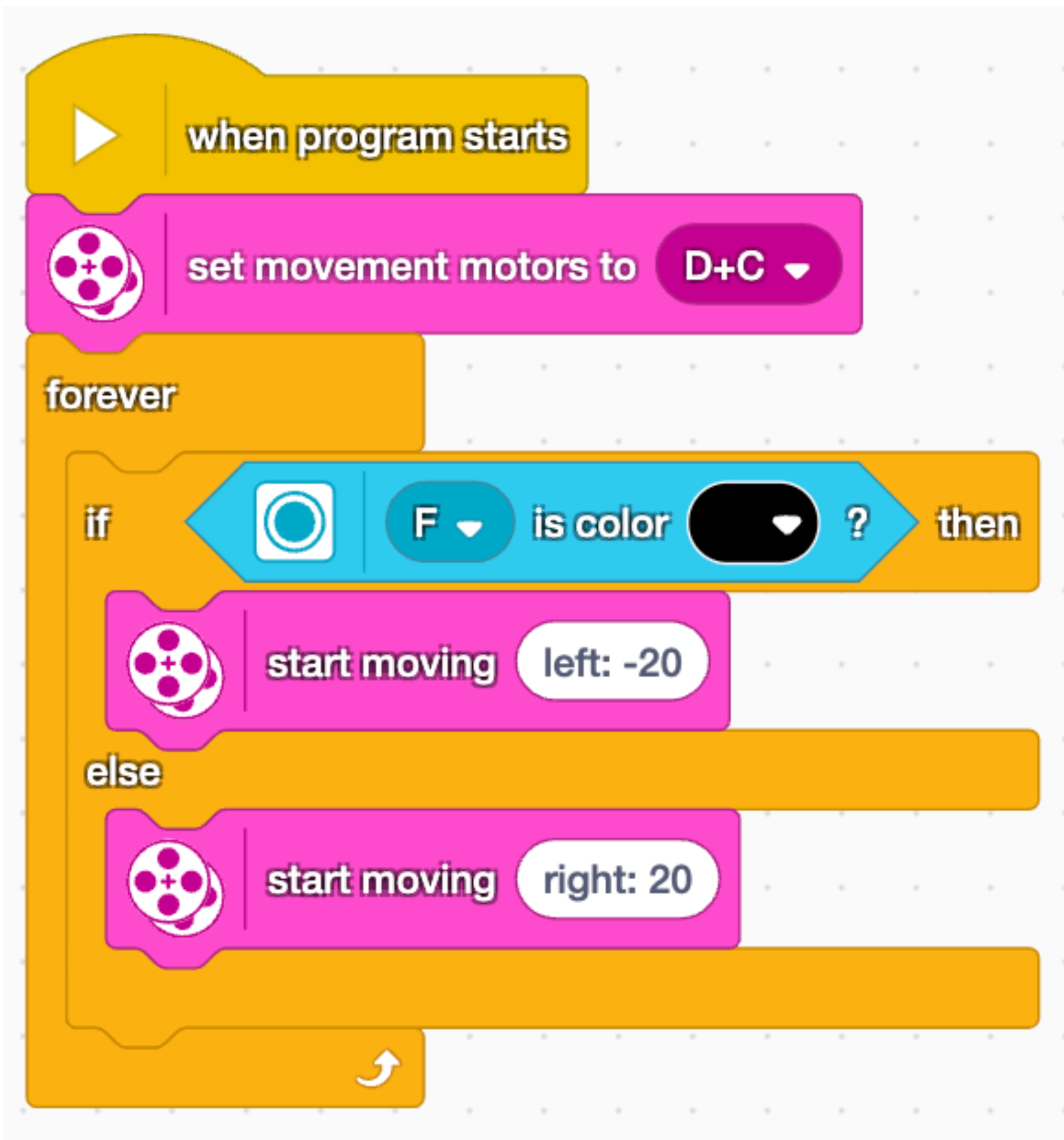
```
from hub import port
import runloop
import motor_pair
```

```
async def main():
    motor_pair.pair(motor_pair.PAIR_1, port.D, port.C)
    motor_pair.move(motor_pair.PAIR_1, 0)
    await runloop.sleep_ms(5000)
    motor_pair.stop(motor_pair.PAIR_1)
```

```
runloop.run(main())
```

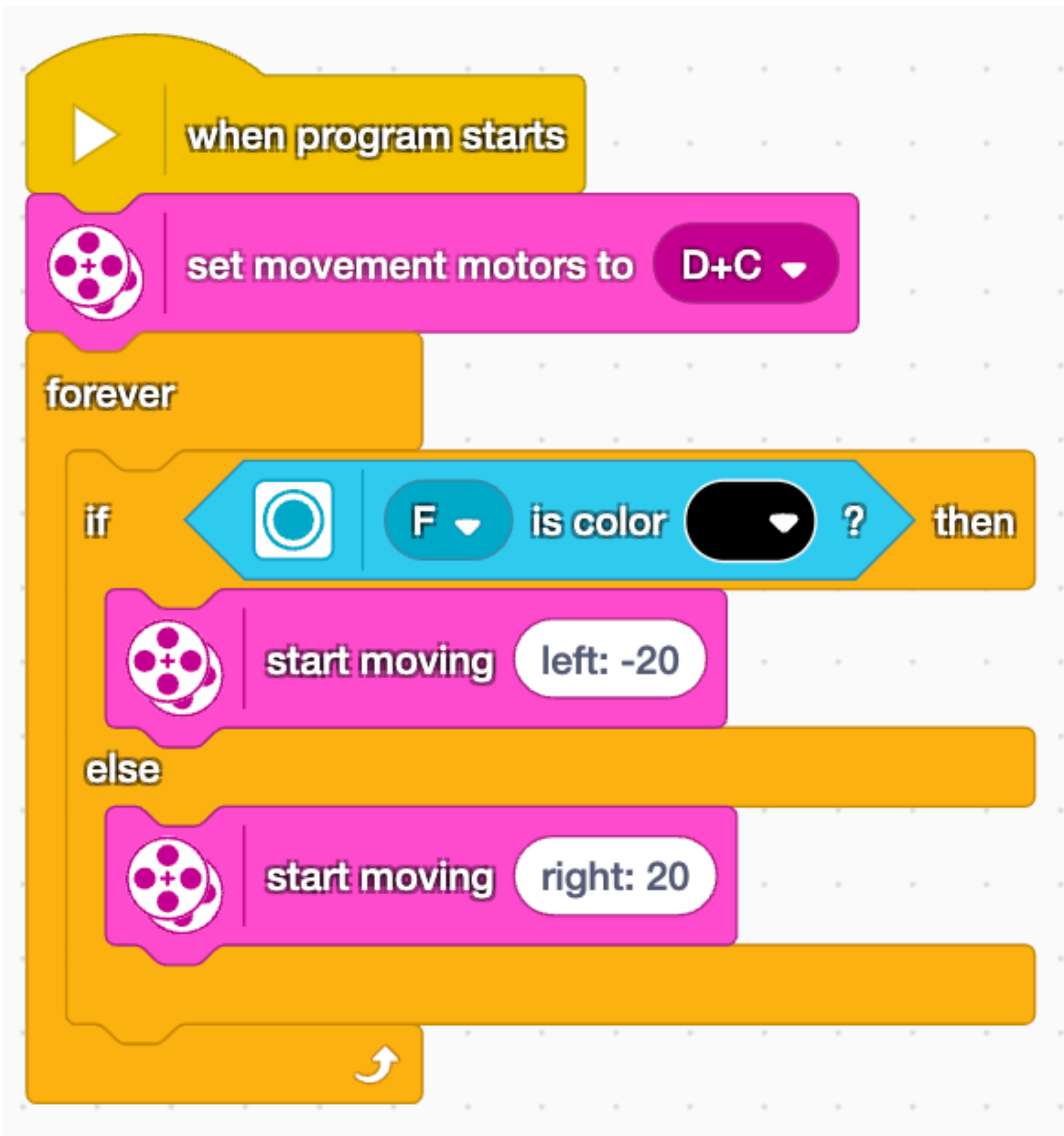
Steering
(-100 to 100)

Line Following



- What direction will TACObot move when program is first run?
- Does it matter whether TACObot starts to the left or right of the black line?

Line Following

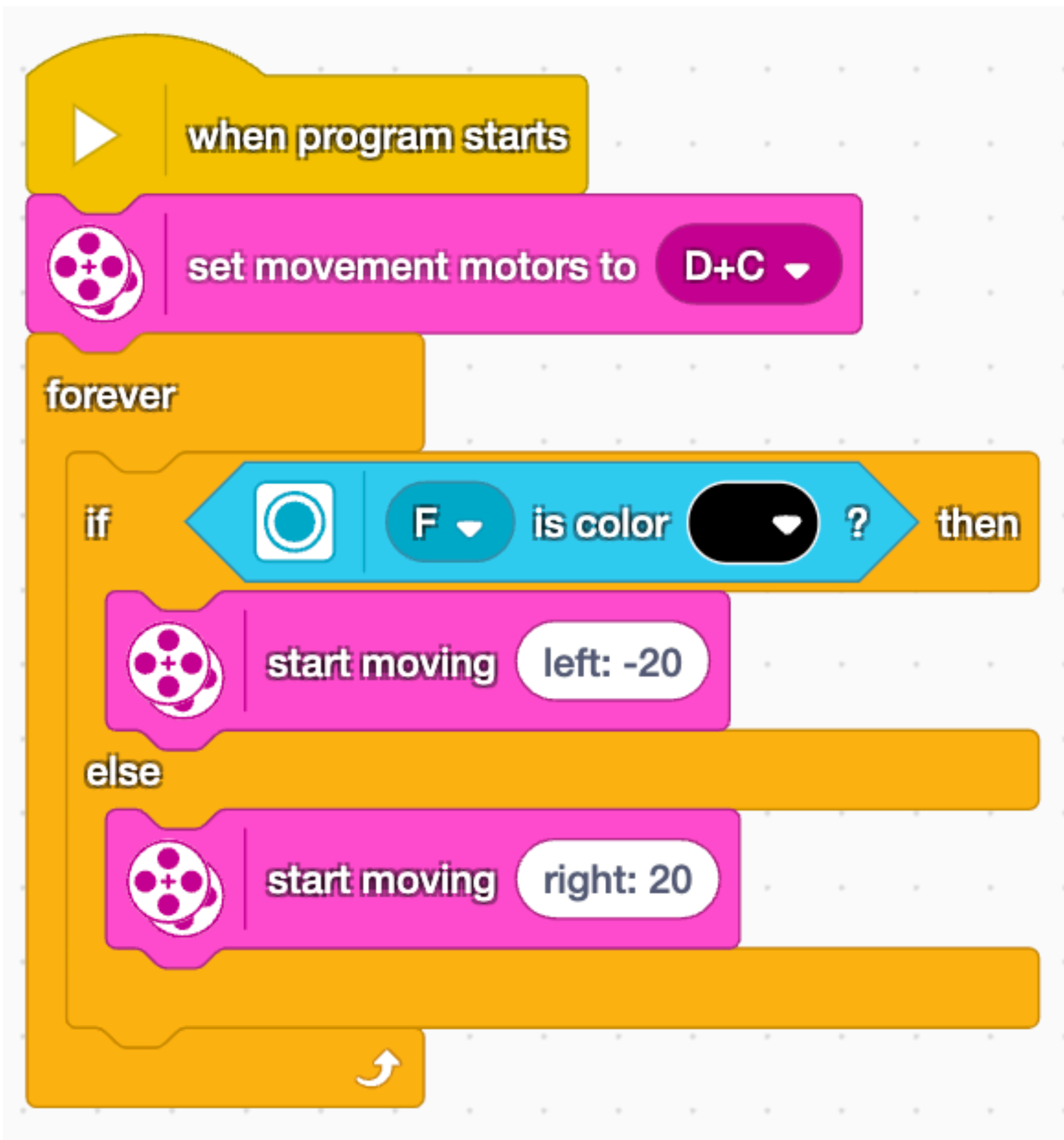


```
from hub import port
import runloop
import color
import color_sensor
import motor_pair
```

```
async def main():
    motor_pair.pair(motor_pair.PAIR_1, port.D, port.C)
    while True:
        if color_sensor.color(port.F) == color.BLACK:
            motor_pair.move(motor_pair.PAIR_1, -20)
        else:
            motor_pair.move(motor_pair.PAIR_1, 20)
```

```
runloop.run(main())
```

Line Following

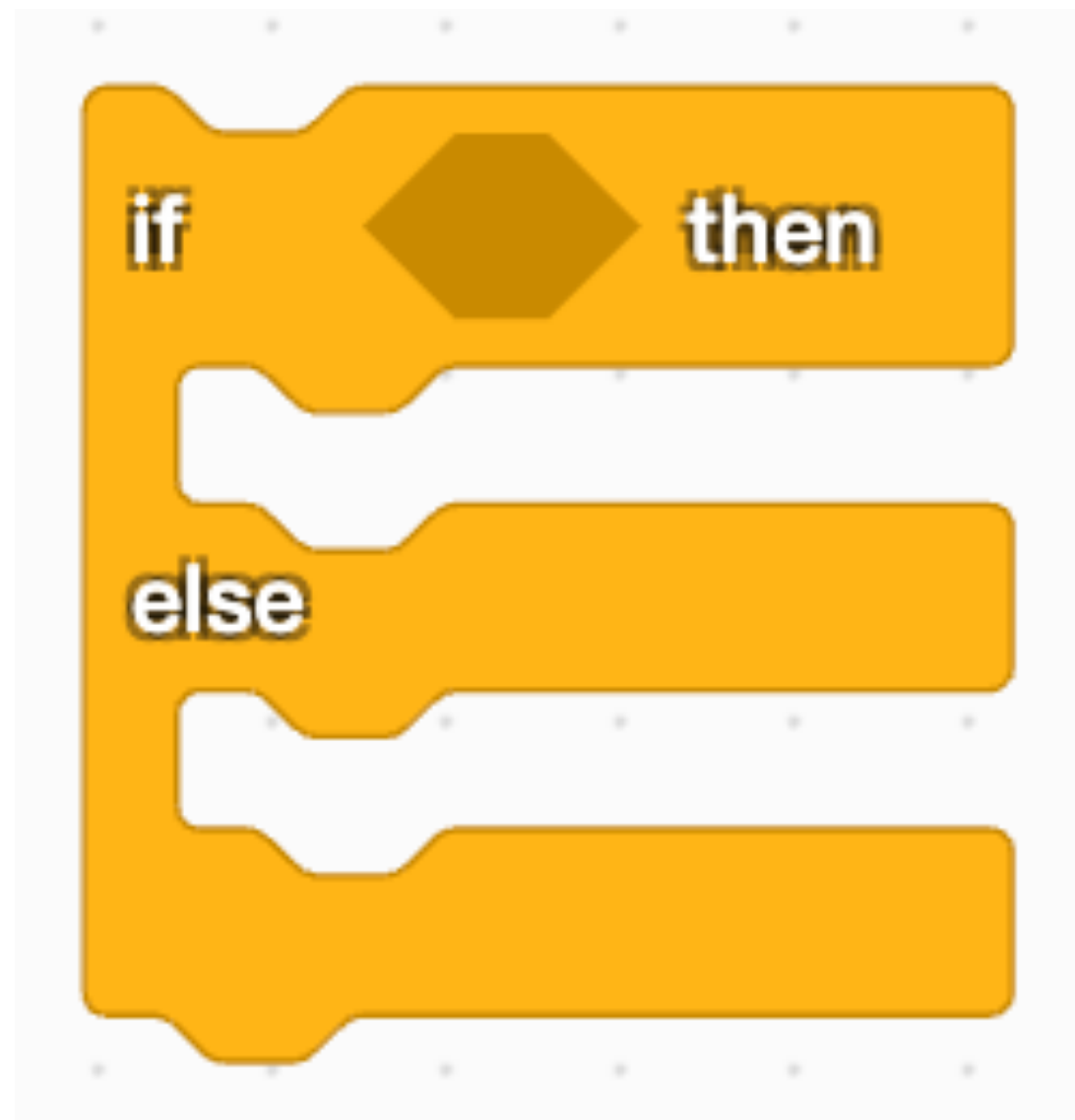


```
from hub import port
import runloop
import color
import color_sensor
import motor_pair
```

```
async def main():
    motor_pair.pair(motor_pair.PAIR_1, port.D, port.C)
    while True:
        if color_sensor.color(port.F) == color.BLACK:
            motor_pair.move(motor_pair.PAIR_1, -20)
        else:
            motor_pair.move(motor_pair.PAIR_1, 20)
```

```
runloop.run(main())
```

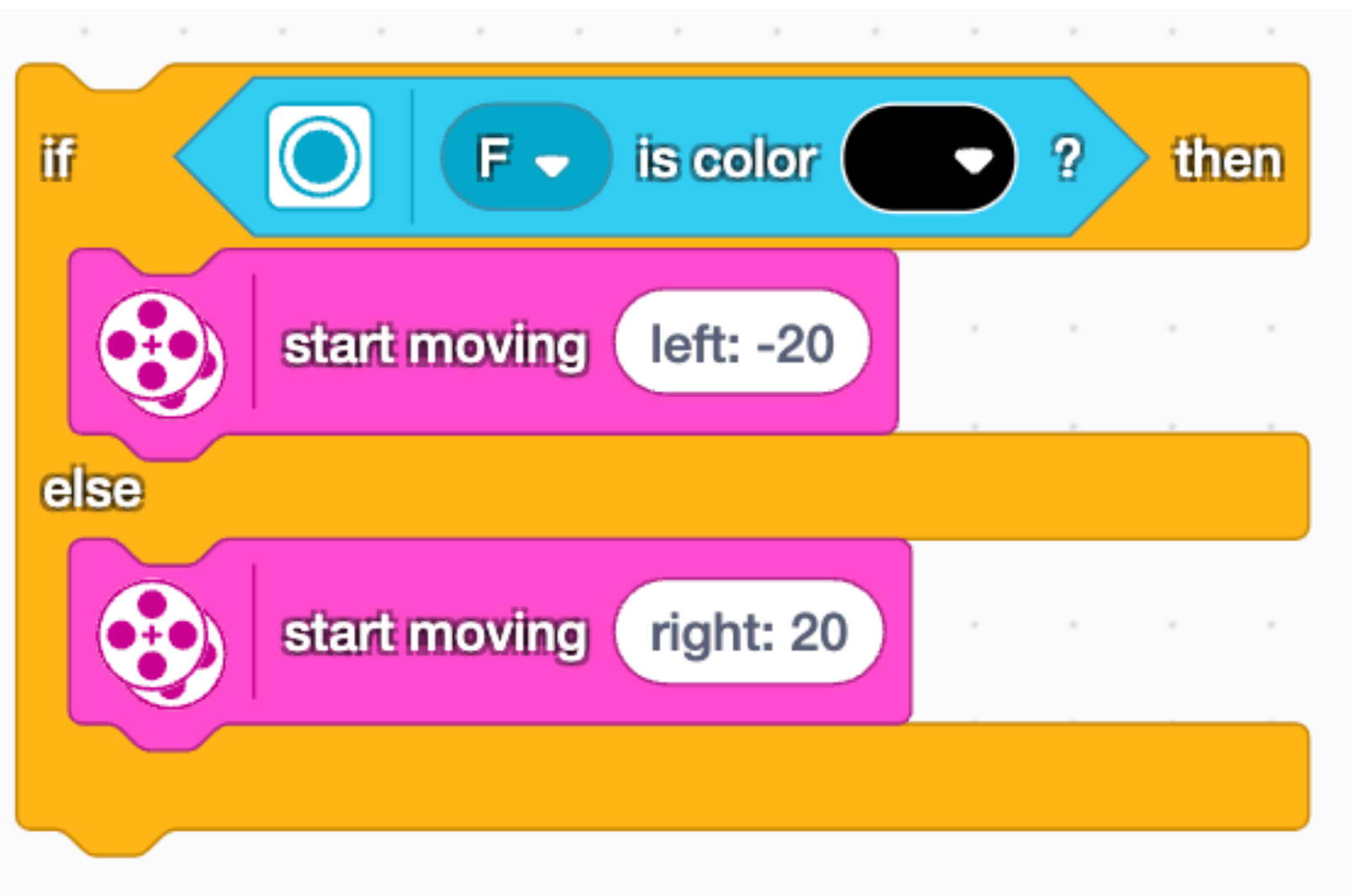
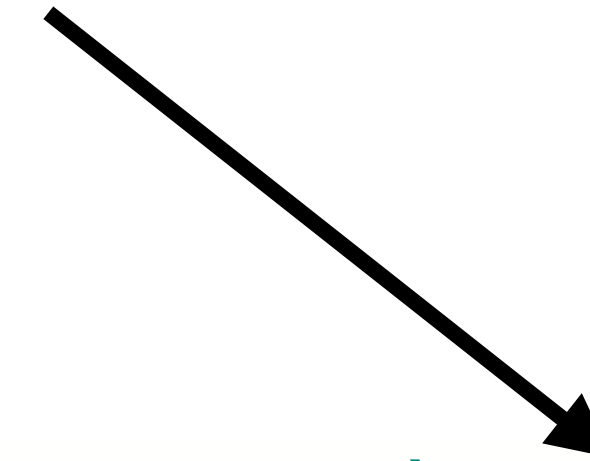

Conditionals: if-else



```
if <expression>:  
    # Do something if expression is True  
else:  
    # Do something if expression is False
```

Conditionals: if-else

Note the double (==) equals to check if the left and right sides are the same



```
if color_sensor.color(port.F) == color.BLACK:  
    motor_pair.move(motor_pair.PAIR_1, -20)  
else:  
    motor_pair.move(motor_pair.PAIR_1, 20)
```

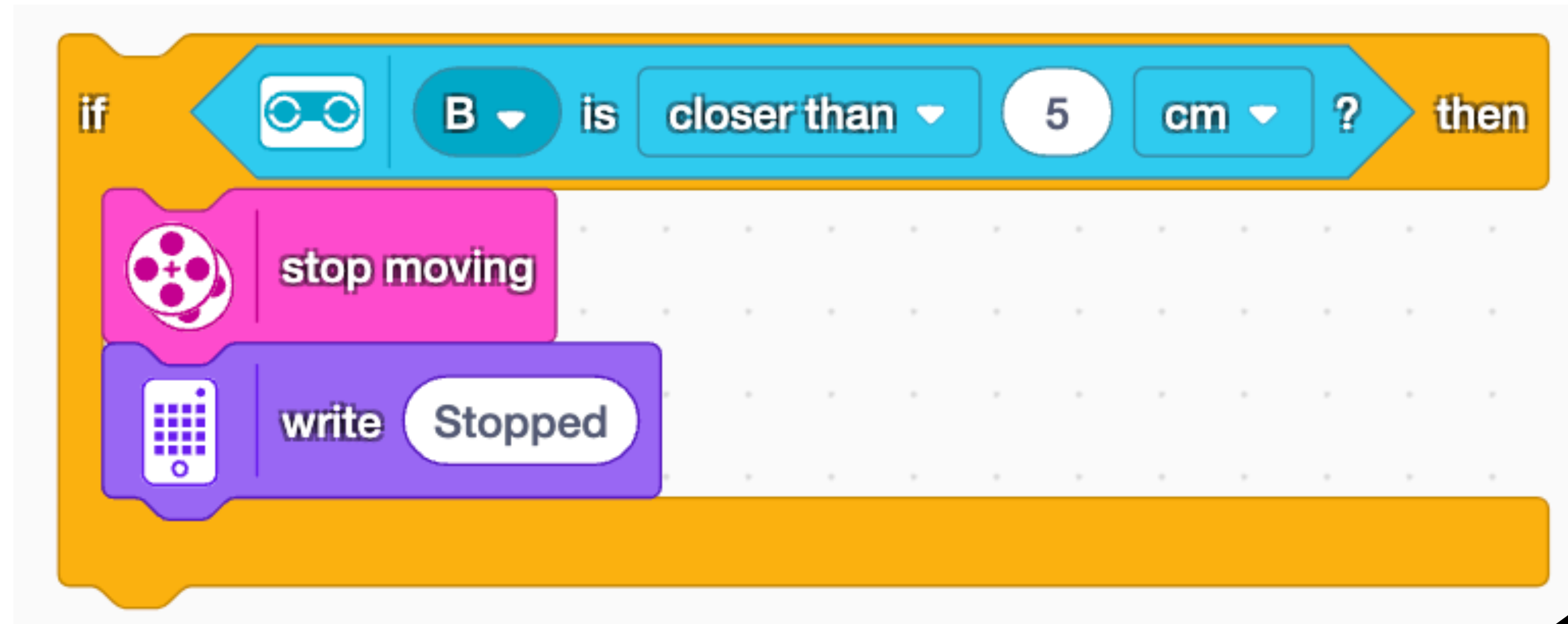

Conditionals: if



You can also use != to check if the left side doesn't equal the right side

```
if color_sensor.color(port.F) == color.BLACK:  
    motor_pair.stop(motor_pair.PAIR_1)  
    light_matrix.write("Stopped")
```

Conditionals: if

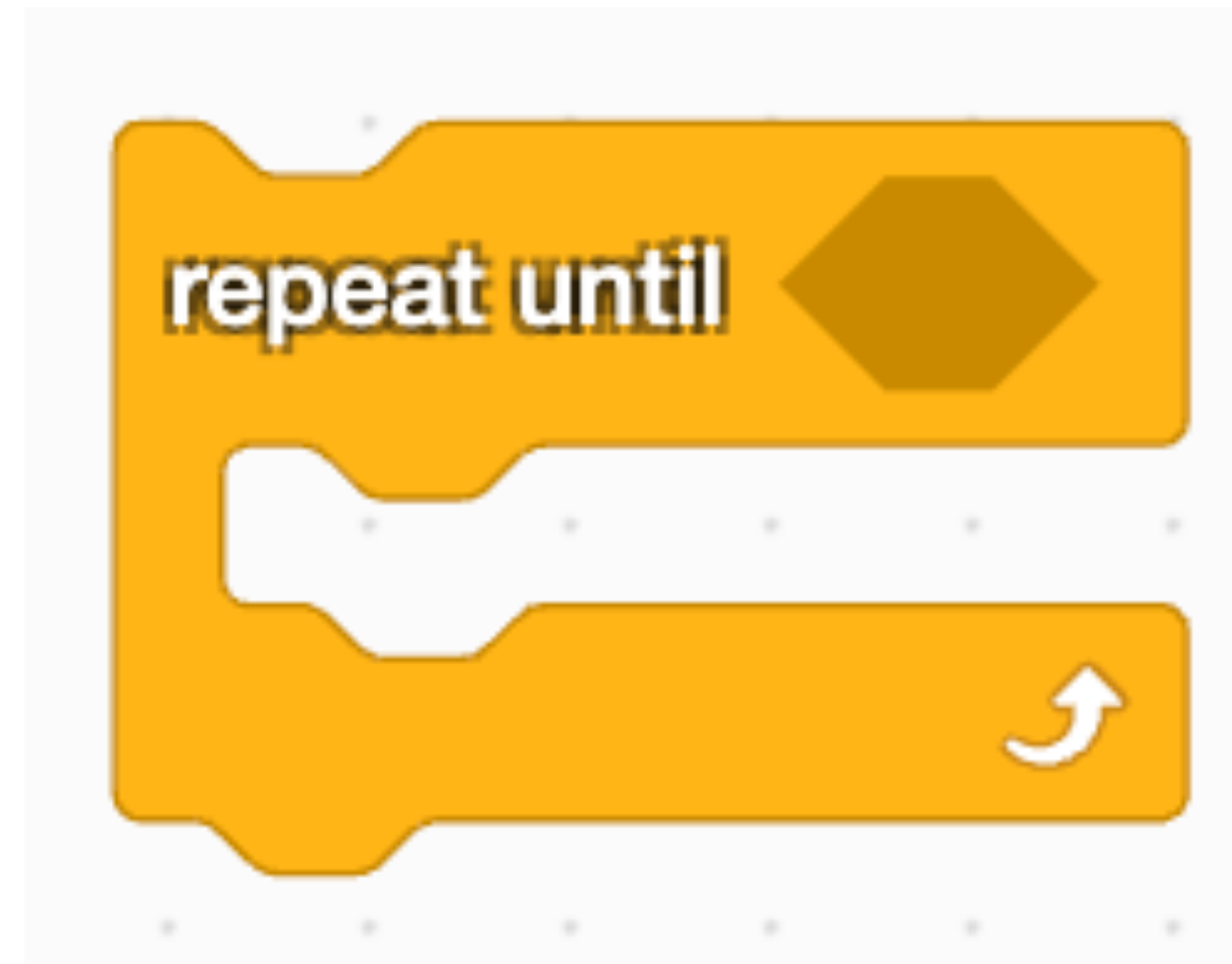


You can use < for less-than and > for greater-than

```
if distance_sensor.distance(port.B) < 50:  
    motor_pair.stop(motor_pair.PAIR_1)  
    light_matrix.write("Stopped")
```

Distance sensor returns millimeters (mm)

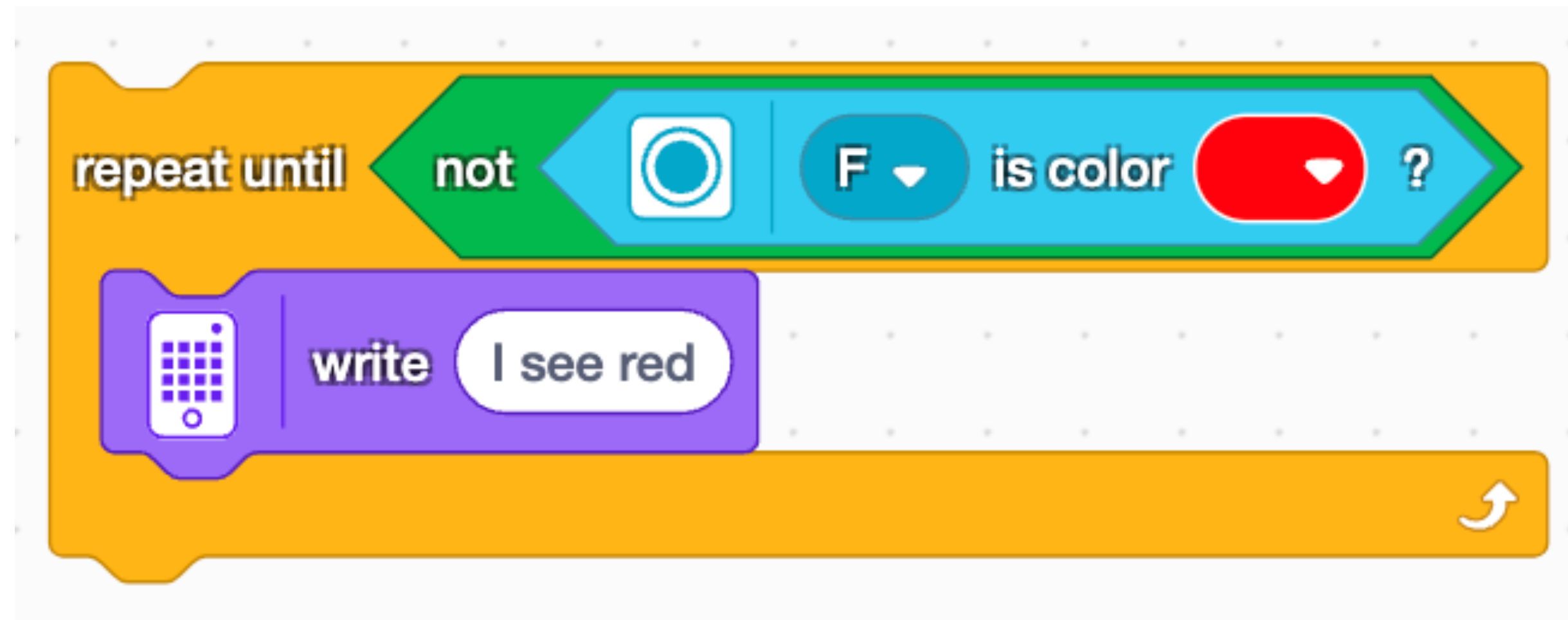
Conditionals: while



← Repeat until expression is True

```
while <expression>:  
    # Repeat until expression is False
```

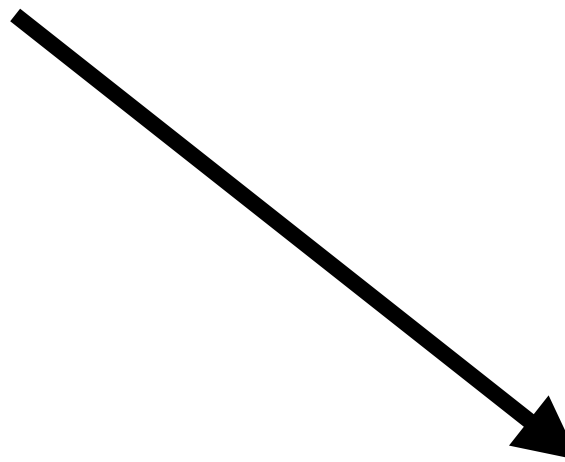
Conditionals: while



```
while color_sensor.color(port.F) == color.RED:  
    light_matrix.write("I see red")
```

Conditionals: while

Pause for a really short time = do nothing



```
while distance_sensor.distance(port.B) > 50:  
    runloop.sleep_ms(1)  
    motor_pair.stop(motor_pair.PAIR_1)
```

Conditionals: while

What does this do?



```
while True:  
    light_matrix.write("Hi")
```


Conditionals: while



```
while True:  
    light_matrix.write("Hi")
```

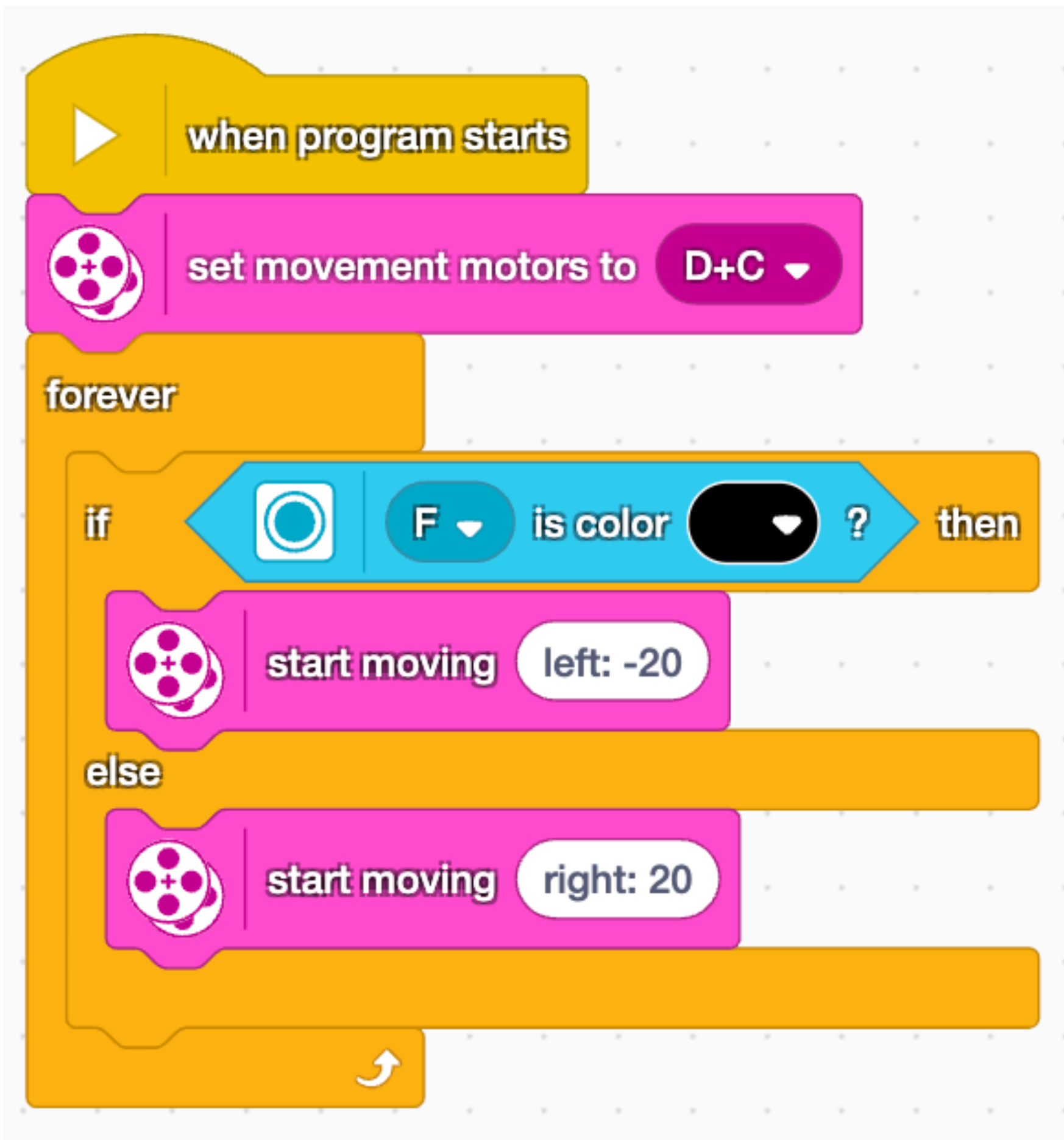
Conditionals: while

What about this?



```
while False:  
    light_matrix.write("Hi")
```

Line Following



```
from hub import port
import runloop
import color
import color_sensor
import motor_pair
```

```
async def main():
    motor_pair.pair(motor_pair.PAIR_1, port.D, port.C)
    while True:
        if color_sensor.color(port.F) == color.BLACK:
            motor_pair.move(motor_pair.PAIR_1, -20)
        else:
            motor_pair.move(motor_pair.PAIR_1, 20)
```

```
runloop.run(main())
```

Distance Sensor

- Distance sensor returns -1 if it doesn't detect obstacle
- Otherwise it returns the distance in millimeters
- Must check for both using “and” or “or”

```
if distance_sensor.distance(port.B) > -1 and distance_sensor.distance(port.B) < 80:  
    motor_pair.stop(motor_pair.PAIR_1)
```

```
while distance_sensor.distance(port.B) == -1 or distance_sensor.distance(port.B) > 80:  
    runloop.sleep_ms(1)  
motor_pair.stop(motor_pair.PAIR_1)
```

Today's Goals

- 1.) Write a line following program that stops at walls
- 2.) Write a program to complete the challenge course