

```

myVariable = 0
Holding = 0

def Start_pressed_callback_0():
    global myVariable, Holding
    while True:
        # Sets the optical light on, and the claws speed.
        ClawRotate.set_velocity(10, PERCENT)
        optical_4.set_light_power(15, PERCENT)
        if not distance_5.object_distance(MM) <= 85:
            servo_h.set_position(180 - 50.0, DEGREES)
        # Code lowering claw when an object is detected more than less than 85mm
        away.
        if distance_5.object_distance(MM) <= 85:
            Rightmotor.stop()
            LeftMotor.stop()
            ClawRotate.spin_for(FORWARD, 185, DEGREES)
            wait(2, SECONDS)
            servo_h.set_position(5 - 50.0, DEGREES)
            wait(2, SECONDS)
            ClawRotate.spin_to_position(30, DEGREES)
            LeftMotor.stop()
        else:
            # Colorpathing code deciding when the robot needs to turn or drive
            forward.
            Rightmotor.spin(FORWARD)
            LeftMotor.spin(FORWARD)
            if optical_4.color() == Color.BLUE:
                Rightmotor.set_velocity(20, PERCENT)
                LeftMotor.set_velocity(20, PERCENT)
            if optical_4.color() == Color.RED:
                Rightmotor.set_velocity(-7.5, PERCENT)
                LeftMotor.set_velocity(7.5, PERCENT)
            if optical_4.color() == Color.YELLOW:
                Rightmotor.set_velocity(7.5, PERCENT)
                LeftMotor.set_velocity(-7.5, PERCENT)
            wait(5, MSEC)

def Stop_pressed_callback_0():
    global myVariable, Holding
    # When the stop button is pressed the project stops.
    brain.program_stop()

# system event handlers
Start.pressed(Start_pressed_callback_0)
Stop.pressed(Stop_pressed_callback_0)
# add 15ms delay to make sure events are registered correctly.

```

```
wait(15, MSEC)
```