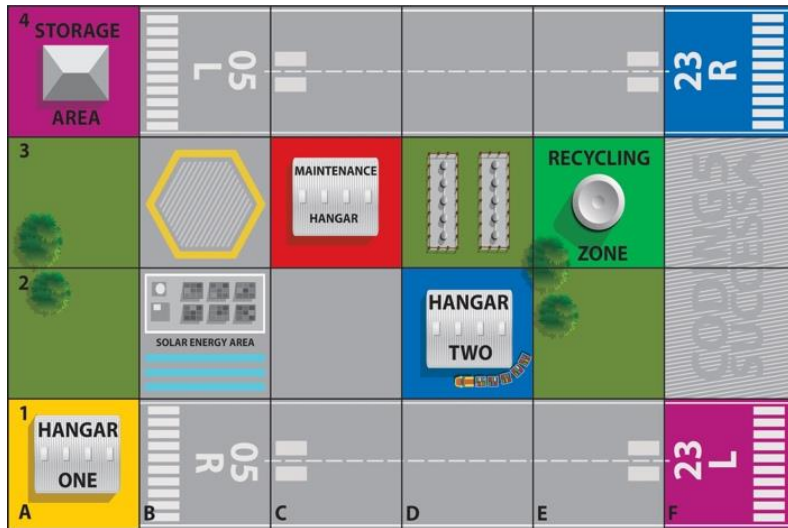


SMART SENSORS!

Detect, clear, collect and deliver!

It's time to take control of your Multi-Mover by Python coding the colour and distance sensors! Find your smart speed as you tackle these mat-based challenges...



A

CHALLENGE A - You need to:

Create a program named: M3A

Add the BULLDOZER (BLADE) attachment.

Place the DEBRIS in D1.

Starting at HANGAR ONE (A1), design the program to:

MOVE the BULLDOZER with the BLADE in the UP position and DETECT the DEBRIS. Stop a short distance away. Then lower the BLADE and MOVE the DEBRIS to F2 - a safe place away from the runways.

Return to HANGAR ONE (A1).

B

CHALLENGE B - You need to:

Create a program named: M3B

Add the LIFTER (FORK) attachment.

Place the SOLAR PANEL onto the FORK in the DOWN position. Starting at HANGAR ONE (A1), design the program to:

Lift the SOLAR PANEL up and deliver it to the SOLAR ENERGY AREA (B2) using the COLOUR SENSOR. Return to HANGAR ONE (A1).

C

CHALLENGE C - You need to:

Create a program named: M3C

Add the LIFTER (FORK) attachment.

Place the RADAR UPGRADE in B3. Starting at HANGAR ONE (A1), design the program to:

Detect (using the DISTANCE SENSOR) and collect the RADAR UPGRADE from B3.

Then using the COLOUR SENSOR, deliver it to the MAINTENANCE HANGAR (C3).

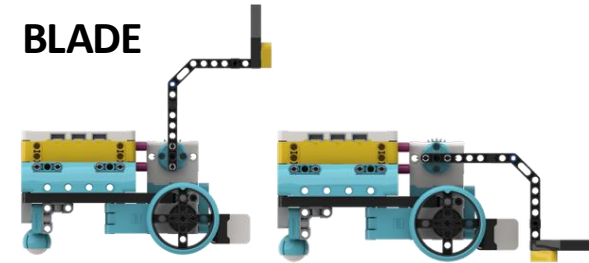
Return to HANGAR ONE (A1).

KEY PYTHON CODE REMINDER: MOVEMENT

```
1 from hub import port# "port" helps us talk to the places we plug motors and sensors in
2
3 # "runloop" helps the Hub do things one after the other in order
4 # "motor_pair" is the remote control for TWO motors working together
5 # "motor" lets us tell one motor exactly what to do
6 import runloop, motor_pair, motor
7
8 async def main():
9     # First, make sure motors A and B are not already paired (just in case)
10    motor_pair.unpair(motor_pair.PAIR_1)
11
12    # Now, pair motors A and B so they can drive together like car wheels
13    motor_pair.pair(motor_pair.PAIR_1, port.A, port.B)
14
15    # Tell the paired motors A+B to move forward about 25 cm (512 degrees turn)
16    # The velocity=500 means about 50% of top speed
17    await motor_pair.move_for_degrees(motor_pair.PAIR_1, 512, 0, velocity=500)
18
19    # Tell the paired motors A+B to turn (spin) 90 degrees right
20    await motor_pair.move_for_degrees(motor_pair.PAIR_1, 158, 100, velocity=300)
21
22    # Tell the paired motors A+B to turn (spin) 90 degrees left
23    await motor_pair.move_for_degrees(motor_pair.PAIR_1, 158, -100, velocity=300)
24
25    # Turn the motor on port C to the absolute position 88°, using the shortest rotation path
26    # The number 100 means how fast it should spin (degrees per second)
27    await motor.run_to_absolute_position(port.C, 88, 100, direction=motor.SHORTEST_PATH)
28
29 # This starts the program
30 runloop.run(main())
```

MOTOR POSITION

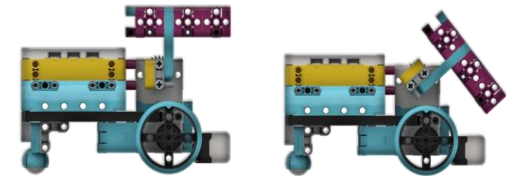
BLADE



Position 0

Position 88

TIPPER BED



Position 0

Position 45

FORK



Position 0

Position 310



Position 35

DISTANCE

```
1 import runloop, motor_pair, distance_sensor
2 from hub import port
3
4 # Set Target Distance variable in millimetres
5 TARGET_DISTANCE = 80# in millimeters (8 cm)
6
7 # Function to check if an object is closer than TARGET_DISTANCE
8 def object_detected():
9     d = distance_sensor.distance(port.E)# returns mm or -1 if nothing detected
10    if d == -1:# -1 means "no object"
11        return False
12    return d < TARGET_DISTANCE
13
14 async def main():
15     # Set up motors by unpairing any old pairing and pairing A + B
16     motor_pair.unpair(motor_pair.PAIR_1)
17     motor_pair.pair(motor_pair.PAIR_1, port.A, port.B)
18
19     # start moving in a straight line
20     motor_pair.move(motor_pair.PAIR_1, 0, velocity=500)
21     # wait until object is detected within the target distance
22     await runloop.until(object_detected)
23     # stop moving
24     motor_pair.stop(motor_pair.PAIR_1)
25
26 runloop.run(main())
```

COLOUR

```
1 import runloop, motor_pair, color_sensor, color
2 from hub import port
3
4 # Set the Target Color variable
5 # You can change this to one of:
6 # color.BLACK (0), color.MAGENTA (1), color.PURPLE (2), color.BLUE (3),
7 # color.AZURE (4), color.TURQUOISE (5), color.GREEN (6), color.YELLOW (7),
8 # color.ORANGE (8), color.RED (9), color.WHITE (10), or color.UNKNOWN (-1)
9 TARGET_COLOR = color.AZURE
10
11 # This function checks if the color sensor on port D
12 # sees the color we are looking for
13 def target_detected():
14     # color_sensor.color(port.D) returns the current color seen by the sensor
15     # Compare it to TARGET_COLOR - if they match, return True
16     return color_sensor.color(port.D) == TARGET_COLOR
17
18 async def main():
19     # Set up motors by unpairing any old pairing and pairing A + B
20     motor_pair.unpair(motor_pair.PAIR_1)
21     motor_pair.pair(motor_pair.PAIR_1, port.A, port.B)
22
23     # Start driving straight forward at speed 500
24     motor_pair.move(motor_pair.PAIR_1, 0, velocity=500)
25
26     # Keep moving until target_detected() returns True
27     await runloop.until(target_detected)
28
29     # Stop the motors when the target color is found
30     motor_pair.stop(motor_pair.PAIR_1)
31
32 # Run the main program
33 runloop.run(main())
```



SHOWCASE: TWO MULTI-MOVERS WORKING TOGETHER

It's time to showcase your skills in a timed challenge using TWO Multi-Movers on the mat at the same time! In this timed challenge, humans AND robots must work together.

Create and design a program named: M3S

START YOUR TIMER and then...

MULTI-MOVER ONE:

Starting in HANGAR ONE (A1) add an attachment to your Multi-Mover and then collect the USED TYRES from B3. MOVE to the RECYCLING ZONE (E3) and deposit the USED TYRES. MOVE to the STORAGE AREA (A4). Remove the attachment.

MULTI-MOVER TWO:

Starting in the STORAGE AREA (A4) add an attachment to your Multi-Mover and then collect the SOLAR PANEL from the SOLAR ENERGY AREA (B2). MOVE to the MAINTENANCE HANGAR (C3) and deposit the SOLAR PANEL. MOVE to HANGAR ONE (A1). Remove the attachment.

STOP YOUR TIMER once both Multi-Movers are at their final locations with their attachments removed! Good luck.



Used Tyres



Solar Panel