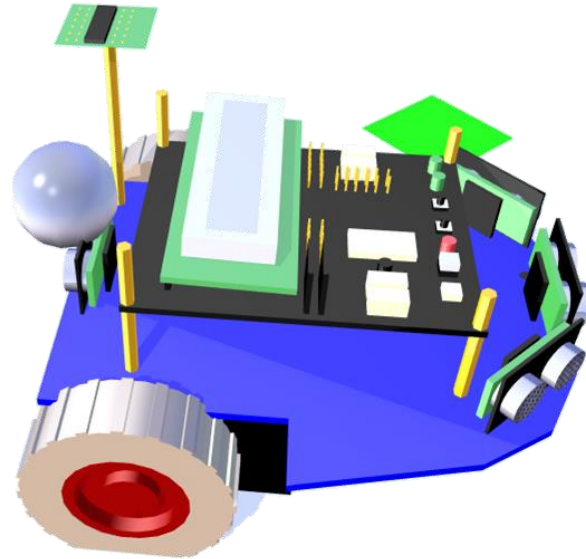


# Improving Your Program

COSPACE ROBOTICS



[www.cospace.org.uk](http://www.cospace.org.uk)  
cospace.rcj@gmail.com

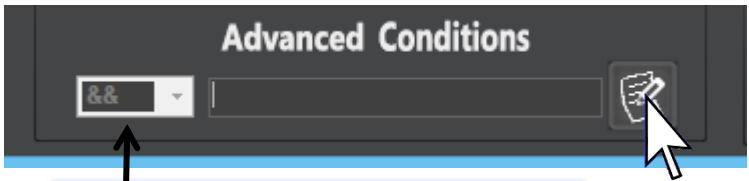
# TASK 1: IGNORE TRAPS WHEN EMPTY

## COSPACE ROBOTICS

When the robot is empty, the robot can go over the traps as no objects are lost! Ignoring the traps allows you to search the field quicker.

Add an **advanced condition** to the 'avoid the trap' statements that the variable **LoadedObjects** must be **greater than 0**

Make sure you open advanced conditions NOT advanced actions!!



This symbol means AND – both the advanced condition and the colour condition must be true!

**Hint, you condition should look like this:**

```
LoadedObjects > 0
```

> = greater than

**Test your program: Save, build, load & run**

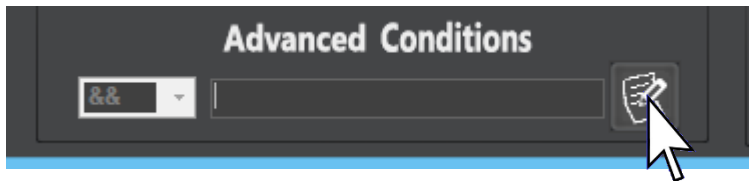


# TASK 2: IGNORE OBJECTS WHEN FULL

## COSPACE ROBOTICS

When the robot has collected 6 objects, it can not collect any more, so should ignore further objects to save time.

Add an **advanced condition** to all of detecting coloured objects statement that the variable **LoadedObjects** must be **less than 6**



**Hint, you condition should look like this:**

```
LoadedObjects < 6
```

< = greater than

**Test your program: Save, build, load & run**

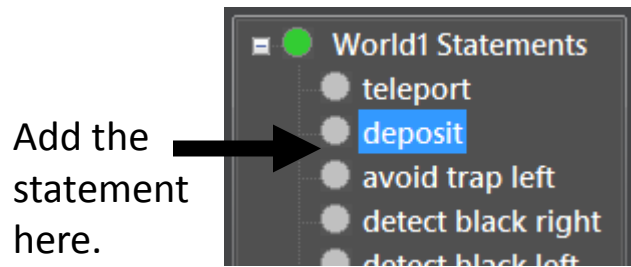


# TASK 3: TURN INTO ORANGE DEPOSITION AREA

## COSPACE ROBOTICS

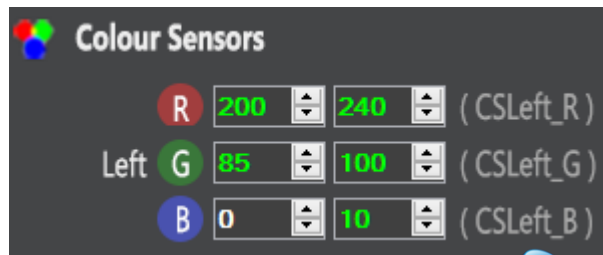
In order to deposit objects, both colour sensors must detect orange. Statements can be added so that if just one colour sensor detects orange, the robot spins until both sensors detect orange, increasing the chances of the robot depositing.

**Step 1:** Add a new statement, which is below the deposition statement. This statement is for detecting orange on left.



Remember you can right click statements and select move up/down

**Step 2:** Add the condition for detecting orange on **just** the left colour sensor:



Make sure you set the condition on the LEFT colour sensor variables only

**Step 3:** Set the action to pivot around the left wheel to turn into the orange area:



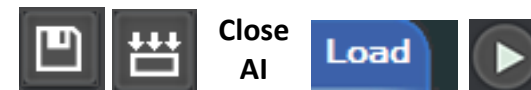
**Step 4:** Add an advanced condition that LoadedObjects must be greater than zero, so the robot only turns in if it contains objects:  
**LoadedObjects > 0**

**Step 5:** Check the program works. Save, build, load & run



**Step 6:** Add another statement this time for detecting orange on the right sensor, and pivoting on the right wheel.

**Step 7:** Check the program works. Save, build, load & run.

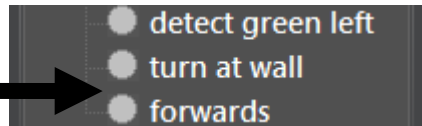


## TASK 4: ADD STATEMENTS FOR DETECTING OBJECTS USING THE LEFT & RIGHT ULTRASOUND SENSORS

In order to deposit objects, both colour sensors must detect orange. Statements can be added so that if just one colour sensor detects orange, the robot turns on the spot until both sensors detect orange. Increasing the chances of the robot depositing.

**Step 1:** Add a statement for detecting objects on the left ultrasound sensor

Add the statement here above forwards.



**Step 2:** Set the condition to be when the left ultrasound sensors is between 0-10



**Step 3:** Set the action to rotate **clockwise**, away from the wall for 60ms.

**Step 4:** Now add a statement for detecting walls on the right, and turning the robot anti-clockwise.

**Step 5:** Save, build, load & run



**Step 6:** You now need to experiment with speed and ultrasound readings to make sure the robot doesn't get stuck!!

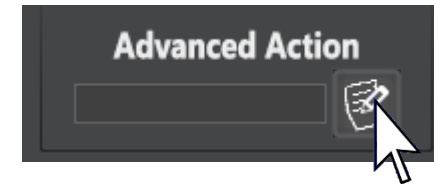
## TASK 5: WHEN FULL, SPEED UP TO FIND THE ORANGE AREA QUICKER

When full, you can travel faster to find the orange deposition area quicker.

**Step 1:** Open your 'forwards' statement.

**Step 2:** Add an advanced action so that if the robot is not full (LoadedObjects <6) the robot go slower, else faster.

To add an advanced action click here:



**Add:**

```
Code Editor
if (LoadedObjects <6) {
  WheelLeft = 3;
  WheelRight = 3;
}
else {
  WheelLeft = 5;
  WheelRight = 5;
}
```

You may want to experiment with the slow and fast speeds to find which speeds result in the most points

# TASK 6: OPTIMIZE YOUR PROGRAM

## COSPACE ROBOTICS

Here are something to experiment with to optimize your program:

- Can you turn closer to the walls to pick up objects near the walls?
- If you move quicker/slower can you pick up more objects
- Adjust the time taken turning when detect a wall or object – does your score increase?
- Make avoiding traps more efficient (turn quicker, but for a shorter amount of time)
- Which direction to turn in – can you use advanced actions so the direction of turning is dependant on the ultrasound sensor to detect on which side you detect an obstacle?
- How late can you teleport within the 3 minutes?
- If you move around in a side to side sweeping pattern or arcs, opposed to in a straight line can you find more objects?