

Unit 3 CodeBot Python Code By Mission

Mission 6 - Line Follower	
Create a list	<pre>detected = [False, False, False, False, False]</pre>
Update a specific value in a list	<pre>detected[count] = val > thresh</pre>
Use a list with LEDs	<pre>vals = check_lines(threshold) leds.ls([False, True, True, True, False]) leds.ls(vals)</pre>
Botcore line sensors function (similar to check_lines) but faster	<pre>vals = ls.check(thresh, is_reflective) leds.ls(vals)</pre> <p>ls.check() takes 2 parameters It has a second parameter <code>is_reflective</code> that controls whether "detected" means the sensor is <code>> thresh</code> OR <code>< thresh</code>.</p> <p>It  returns a  tuple rather than a  list.</p>
Using or (logical operator)	<pre>elif vals[1] or vals[2] or vals[3]:</pre> <p>can have two or more conditions; if any of the conditions are true, the statement will evaluate to true</p>
Comparing with a tuple	<pre>elif vals == (0,1,1,0,0):</pre>
Code needed to change a global variable inside a function	<pre>global count global thresh, is_reflective</pre>
Built-in math operations	<pre>abs(x) round(x, ndigits)</pre>
Mission 7 - Hot Pursuit	
Read the proximity sensors	<pre>prox.detect()</pre> <p>returns a tuple (left, right) with values True or False</p> <pre>vals = prox.detect() left_detected = vals[0] right_detected = vals[1]</pre> <p>Index values: 0 = left 1 = right</p>

Proximity LEDs	<pre># Check proximity sensors p = prox.detect() # Show (left, right) on the PROX LEDs leds.prox(p)</pre>
Use parameters	P = prox.detect(power, threshold) Power is the “bot flashlight” with settings from 1 to 8 (high power) Threshold is the sensitivity level, with settings from 1 to 100 (how much light is needed to detect)
Another built-in function that finds the ideal thresh for a given environment	<pre>prox.range() prox.range(num_samples, power, range_low, range_high)</pre> <p>All parameters are optional</p>
Toggle the motors on and off – can be used with a button press to turn on/off the motors	<pre># Toggle a variable go_motors = False go_motors = not go_motors # (not False) == True go_motors = not go_motors # (not True) == False</pre>