**Design notes for the drone world simulation, project 2.**

The bulk of the print statements have been left in and commented out. My hope is that if you have problems in an area the print statements in that area can be uncommented and helpful.

A few “toString()” methods have been added to try to be helpful.

At about line 157 is where the handling of extra spaces is coded.

Reference **globe** == Droneworld simulation. It is the dictionary that is drone world. All cubes are added to it. It uses the coordinates of each cube as the its key. This provides for a fast lookup.

For any cube that is **not** on the 0 level of the Y plane or has a ‘?’ as a coordinate, it is added to the **globe.floatingCubeList.** All others that are at the 0 level are just added to the **globe**. The processFile() method creates these three lists, floatingCubeList, cubeList, and specialCubes. The first two are gone through in processList(), the third is a referential for you if needed.

In the **processList()** method (line 227), the cubes are checked to make sure they are not floating in the air. Also any cubes that have one or more of the position values of “?” are added to the **globe.**

The following all depends on how you handle the second list. For project 1, I created a world from each list and checked if they matched for if done (in case of a trick question where both input lists are the same). I modified world 1 (list1) to make it like world 2(list2). If you create another drone world with the second list and use it for comparison to the first, those cubes with ‘?’ values will need position updating once placed within the second world.

A list has been added to the simulation: specialCubes. It holds all cubes that have a ‘?’ for any and all position values. The ‘?’ have been replaced with a value of 55+ (see code – starts at about line 172). The drone will be not added to that list if there are any ‘?’ for position values – but its position values will be changed to the 55+ values. (if drone is wanted in the list, copy lines 215 & 216 to around 207) These are added to the drone world - and are not validated for position

The list that holds cubes with a ‘?’ in them is printed at about line 222. Currently commented out.

The original simulation only looked for '?' values at the X and Z positions. It's been updated

to look for it at any of the coordinate positions.

For project 1, I did do a simple example of the single ‘?’ position. Also, to compare the resulting world to what it was to be, the contents of the resulting /changed world were printed out and compared to the second list visually for validation.

If you want to watch the states change as the drone moves a cube uncomment lines 321, 322, 352, 353. The Speak() method prints out state data.