

Tilecoding.py

Result:

```
In [10]: run Tilecoder.py
Tile indices for input ( 0.1 , 0.1 ) are : [0.0, 121.0, 242.0, 363.0, 484.0, 605.0, 726.0, 859.0]
Tile indices for input ( 4.0 , 2.0 ) are : [39.0, 160.0, 281.0, 403.0, 524.0, 645.0, 777.0, 898.0]
Tile indices for input ( 5.99 , 5.99 ) are : [108.0, 241.0, 362.0, 483.0, 604.0, 725.0, 846.0, 967.0]
Tile indices for input ( 4.0 , 2.1 ) are : [39.0, 160.0, 281.0, 403.0, 535.0, 656.0, 777.0, 898.0]
```

1. Every tile has a different tile number (index). Assuming that you number the tiles in the natural way, the tiles in the first tiling will run from 0 to 120, and the tiles in the second tiling will run from 121 to 241 (why?)

My array is 11x11, it means there are 121 components in it. Index 0-120 means there are 121 components, which is the space for my array. And the second tiling starts from index 121 and end at 241.

2. A given input point will be in exactly one tile in each tiling. For example, the point from the first example in the training set above, $x = 0.1$ and $y = 0.1$, or $(0.1, 0.1)$, will be in the first tile of the first seven tilings, that is, in tiles 0, 121, 242, 363, 484, 605, 726 (why?)

For the first seven tiles, the xcoord and ycoord will always round to 0, so it will be useless when we try to use them to calculate the index.

3. In the eighth tiling this point will be in the 13th tile (why?)

This time, the xcoord and ycoord finally round to 1, so when it will affect the index now. By $11 * ycoord + x$, which is 12. So instead of the first tile, it increases 12 to 13th.

4. In the eighth tiling this point will be in the 13th tile, which is tile 859 (why?)

The first tile of eighth tiling is 847, and increases by 12 is 13th. Which is 859 as well.

5. If you call `tilecode(0.1, 0.1, tileIndices)`, then afterwards `tileIndices` will contain exactly these eight tile indices. The largest possible tile index is 967 (why?)

Because the biggest number we can choose is 5.99 in the range[0,6). And 5.99, 5.99's

biggest index is 967, therefore, we cannot get an index which is bigger than 967.

6. Another check on your code is that none of the indices should be negative or greater than 967. Finally, the second and fourth examples should produce very similar sets of indices (they should have many tiles in2 common) (why?)

Because the point they pick are really close with each other. It will be no reason to have too much difference.