Lab 6

Let's build a binary search tree using Python's class keyword. The reason is that we want to look at algorithms on BST's so it's worth revisiting what a binary search tree actually is. The best way to understand something in computer science is to build it.

Write the following:

```
class Node:

      __init__(self, item)

               self.item = item

               self.left = None

               self.right = None
```

I'll talk about this in class and orient you to how Python approaches OOP. You should be thinking about how you would do (or have done) this in C++.

Next we want a binary search tree class:

```
class BST:

        __init__(self):

               self.root = None
```

Our task in the lab is to write the add_node method for the BST class. Trace through how you'd do it on paper, perhaps think about how you did it in C++, and remember that in Python you have to use self. to access data items within an instantiated object (whereas in C++ you just use their identifier.)