# Biological Computation 20.181

Homework 2

Data Structures. In class, we looked at a simple data structure for representing phylogenies. A dictionary with several predefined keys: name, left, and right. This structure is sufficient to represent the topology (of a rooted tree), but in general we are sometimes interested in the length of the different branches. For this homework exercise, you will write a subroutine that outputs a tree data structure as a string in Newick format as outlined in class, but you will modify the code such that it also reports branch lengths.

For reference, the following tree:
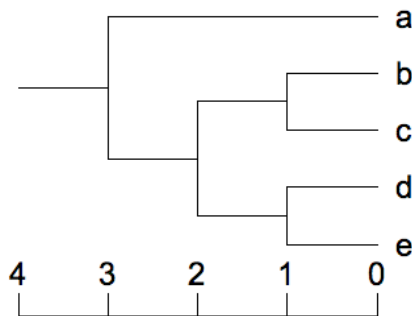
...can be represented as: "(a,((b,c),(d,e)));" in Newick format.

In Newick notation, additional information can be added to each node by following the node with a colon. To add to an internal node, follow the appropriate right paren with a colon. For example, here is the above tree with branch lengths added:
"(a:3,((b:1,c:1):1,(d:1,e:1):1):1):1;"

In this (and most subsequent) homework, you will be responsible for adding functions to an existing codebase. You can download the appropriate Python codebase from the class website. Your function, which will print out Newick formatted strings including branch lengths, will have the following form:

```
def tree2String(tree):
...your code here...
```

and will be called by the 'wrapper' function:

```
def printTree(tree):
        treeStr = tree2String(tree)
        print tree2String + ";"
```

Finally, your code should detect whether there are branch lengths to report (!= None), otherwise it should output a standard Newick formatted file without lengths.