Madeleine Murphy

004991129

CS 32 Homework 4

Due 28 May 2019

2. Coord does not have the correct comparison operator, <=, so when Coord is passed into the one argument form of insert, the insert function does not know how to compare two coordinates in the sequence to figure out what order they need to go into. An ints, for example, can be automatically compared in this way, but without a defined <= operator for the Coord class, they cannot be properly compared to be put into correct order according to the one argument insert function.

4b. You could not have a one parameter function for this recursive problem because we have to continually add to the path string until we reach the bottom of the tree. If you did not have this second parameter, we would only ever be able to access the label we were looking at, but not any of the other parts of the tree related to it. You have to continue to add the new subdomains, but without the string parameter, there would be nowhere for us to actually add the subdomains.

5a. The time complexity is O(N3) because there are three for loops nested within each other that each run N times.

5b. The time complexity is still O(N3) because the maximum value j can have is i which is N, which is the same as the last problem. There are still 3 nested for loops each with maximum value N.

6a. Interleave calls the function get which calls the helper function nodeAtPos. Call nmin = N because the sequences are the same size, so nmin is just the number of elements in each sequence. nodeatPos runs through it N/2 = N times, so get runs through the list N times. This is then ran through the for loop N times, so the total is O(N2). The second for loop is disregarded because the sequences both have N elements, so O(N2) dominates.

6b. The first for loop is O(N). The second for is O(N) but because both sequences have about N elements this N is smaller than the first for loop’s O(N). So the overall time complexity is O(N), which means this is a better implementation than part a because it has a faster time complexity.