David Li

404473282

CS32 Homework 4

2.) The one parameter insert function requires the comparison of the Complex object with others in the sequence so it can be rearranged in increasing order; however, since the Complex class does not offer an overload operator for “>” or another comparison operator, it will produce an error.

3.) b.) Creating a listAll function with only one parameter would be impossible because we need to decompose the problem into simpler pieces and that’s done by getting access to the MenuItem\* parameter to access the vector holding the additional submenus. However, we also need to access a string parameter in order to work with the paths, which could not be done without the string path parameter.

4.) a.) The time complexity of this algorithm is O(N^3). This is because each of the for loops run until length N. Since we have three for loops, with each additional one within the prior, there will be 3 levels deep of traversing to N and thus the higher order time complexity is O(N^3).

b.) The time complexity of this algorithm is also O(N^3). The for loop running from “j” to “i” has a worst-case scenario of running from “j” to “N-1”, which is essentially just N, and a best-case scenario of running from j to I would be a time of 0. We want to measure “i” for its average length case, which would be N/2. So the average of the two inner loops would be O(.5N^2). Without coefficients, combining the inner for loop with the outer would give us O(N^3). Thus there would be three levels of traversing for loops to an average of N times and the time complexity is identical to the previous question.

5.) a.) (O(N^2)) Assuming seq1 and seq2 are both length N, the first for loop will loop through N times and use the get function twice, which must iterate through the sequence the worst case about N times and average about N/2 times. This totals to O(N(N+N)) times or just O(N^2). The second portion of the code has a for loop looping through N times and also a get function inside, which also runs O(N^2); however, the highest order would still be O(N^2) since O(2N^2) equates to O(N^2).

b.) This function time complexity is O(N). There are no for loops traversing N that contain another for loop line that traverses N. InsertBefore only has a time of O(1) and the second for loop has a worst case of N. Swap is also a O(N) as well. However, overall if we disregard coefficients, the time complexity then simply remains O(N).