2) The call to the one argument insert function causes a compilation error, as the one argument form must compare the object to other objects in the sequence in order to determine its position. As Complex cannot be easily compared, the user must define an overloaded > operator. Thus, with the compiler being unable to compare two Complex objects, the one argument insert function gives a compilation error, as it does not know where to insert it.

3b) The constraints in part A stated that we must use recursion to solve the problem. However, to delve deeper and deeper into the submenus (in essence, splitting the problem into smaller and smaller parts), we must pass the path of the previous recursion in order to solve the problem as a whole. This is essential because we must combine all of the recursive “solutions” to create an overall solution from the algorithm. This is impossible without at least 2 parameters in the function, one which pertains to the last recursive solution.

4a) The complexity of this algorithm is O(N3). There are 3 nested loops, each of which go to N. As we go deeper and deeper into the function, each loop always operated N times. As this is compounded over and over again, it turns out to have an overall complexity of O(N3).

4b) The complexity of this algorithm is still O(N3). This is due to the fact that, in the worst case scenario, the second loop still runs N times. Changing the inner loop to go from N to I, in the most common case scenario would change the complexity of that loop to 0.5N, compounding onto the other loops this would becomes 0.5N3, which translates to an overall complexity of O(N3) as well. Thus, the complexity still lies at O(N3).

5a) The complexity of this Interleave function is O(N2). The first loop operates N times and uses the get function, which has a complexity of N. Thus, the first loop has an overall complexity of N2. The second loop, at worst case, also runs N times and also uses the get function, so it also has a complexity of N2. However, because they are both of the same complexity, the overall complexity is O(N2).

5b) This function is much better in terms of run time. The overall complexity is O(N). The first big operation, that which traverses the linked list, does N operations. The second traversal also does N operations at worst-case scenario. Finally, the swap function must traverse the linked list and swap each item, which has a complexity of O(N). Overall, the function does 3N operations, creating a complexity of O(N).