Problem 2:

In this case, one difference between the one-argument and two-argument versions of insert function is that the one-argument version needs to compare the value of the complex object in the argument with the value of the complex object in the sequence. The function does so by calling the > operator. However, the complex class does not have an overloading > operator. This causes the compilation error.

Problem 3b:

To solve this problem recursively, the two parameters are both necessary. Solving this problem recursively requires breaking down the problem into smaller pieces, and solving them in the same way. Therefore, a constant pointer to MenuItem is needed to execute the listAll function on submenus. However, the information about the address of the submenu is also needed so that it can be printed out before the submenu information is printed out. Therefore, a string parameter (path) should be passed.

Problem 4a:

The time complexity of this algorithm is O(N3). This algorithm has 3 levels of for loop, and in each level the for loop iterates N times. In each for loop, except for the inner for loops, the runtime of other operations is constant no matter what N is. This leads to an overall complexity of N to the cube.

Problem 4b:

The time complexity of this algorithm is O(N3). The use of symmetry does reduce the runtime by half on average, which leads to a time complexity of O(0.5N2). However, the constant is omitted in time complexity analysis, and therefore for the inner two embedded loops the time complexity is O(N2). The outermost loop still traverses all N elements. Therefore, the overall time complexity is O(N3), the same as that in part 4a.

Problem 5a:  
 The time complexity of this algorithm is O(N2). In the interleave function, a for loop traverses all N elements of a sequence, and for each element insert function will be called. The insert function will call the nodeAtPos function, which for the worse case loops for N/2 elements. Therefore, the loop has a time complexity of O(N2), which leads to an overall time complexity of O(N2).

Problem 5b:

The time complexity of this algorithm is O(N). In this interleave function, a for loop traverses all N elements of a sequence. However, for each element, the insertBefore function does not have to traverse the sequence to insert the new node. Therefore, the time complexity of the insert step is O(1), and the time complexity of the loop, also the whole function, is O(N). It is better than the implementation in part a, which results in a time complexity of O(N2).