2. The one-argument insert attempts to find a position in which value is less than or equal to a value in a Sequence. But there is no way for the compiler to know *how* to compare members of the Complex class with each other. It doesn’t know what makes one complex number less than another, which is what causes the error.

3b. The reason why the one argument implementation of listAll wouldn’t work would be because you wouldn’t have any way to alter what it prints out – there would be no way for a pointer to a menuItem to access the name of a menuItem higher in the tree.

4a. The time complexity of the algorithm is O(N3), since the algorithm has three for loops which each iterate N times that are nested inside each other.

4b. The time complexity is still O(N3). There are still three for loops, two of which run N times. The second for loop runs i times (that is, N-1). Since 1 is a constant, we’re free to disregard it, so the time complexity is the same.

5a. The time complexity is O(N2). There are two for loops in the function that run through the nodes N times. One of which calls *get*, which, in the worst-case scenario, runs through the nodes N times, meaning the complexity is N\*N (N2).

5b. The time complexity is O(N). There are two for loops in the function that run through the nodes N times. However, since these for loops are not nested, you would add the complexities together (N+N = 2N) and disregard the coefficient, so you’d be left with N.