1. The one-argument form of insert compares the values of the Nodes in the Sequence to insert the value in the correct position. However, no comparison operator (<) has been defined for the Complex class, so this results in a compilation error.
2. b) If we had only a one-parameter listAll, implementing it as the recursive function would mean that the parameter would have to be of type MenuItem. Since it has to be of type MenuItem, we cannot pass the previous path into the function, and there would be no way to access this previous path given the constraints. Thus, we would only be able to print out the current MenuItem’s name.
   1. O(N3) because there are 3 major loops that each run N number of times (N x N x N).
   2. It is still O(N3) because i can be a maximum of N, and j can be a max of i, so therefore it is still 3 loops that each run N number of times.
   3. O(N) because the loop runs N times since the Sequences all have N elements, and the ItemType object is only visited twice per loop through the get() method.
   4. O(N) because the loop would end up running N times, since all the Sequences have roughly N elements. Thus, it’s roughly the same as the implementation in part a.