Problem 2:

The Set class does not understand comparisons between Coord data members using binary expressions such as == and >, which work for library data types like integers and strings. Thus, the Set class is unable to insert a Coord as it attempts to check if the Coord already exists in the Set, returning an error in the process.

Problem 4b:

The recursive implementation of listAll had to have two parameters in order for each recursive call to remember where it is located in its iterations. The “path” parameter gives the recursive function the previous paths it had taken to reach the current call.

Problem 5:

1. O(N^3), because there are three nested loops that all run to N and the function calls inside the for loops are all of constant time complexity.
2. O(N^3), because the second loop still eventually reaches a maximal value of N. Despite halving the number of calls to the functions inside the loops, the time complexity is not affected by a constant multiple.

Problem 6:

1. O(N^2), because the loop runs at most N times, and the nested functions get() and insert() loop at most N/2 and N times, respectively. The worst case scenario takes N^2 steps to insert an item.
2. O(N log N), because the sort function, which runs on logarithmic time, has the highest time complexity.
3. O(N), because every step has linear time complexity.