Danning Yu, 305087992

CS32 Homework 4

2. It is because the greater than operator, >, is not defined for a Coordinate, so thus, the one parameter insert(…) function cannot perform the statement “value > p->m\_value,” thus causing a compilation error.

4b. The problem is not solvable because there is no way to store the value of the superdomain as you traverse down the tree, so thus, if a given domain has more than one subdomain, you will not be able to apply the superdomain to all the subdomains. It can be only printed out once in the initial recursive call.

5a. There are 3 nested for-loops that each iterate from 0 to N (for indices i, j, and k respectively), so thus, the run time is O(N3).

5b. The inner for-loop (with k as the index) will run N times, the middle for-loop (j as the index) will run a maximum of N-1 times because it is dependent on the value of i set by the outer for-loop, and the outer for-loop (i as the index) will run N times. Thus, the run time is O(N3).

6a. The first for-loop calls the get() function and insert() functions twice, and each of those functions have a worst case run time of O(N), and the for-loop itself will run N times, so thus, the total run time for the first for-loop is O(N\*4N) = O(N2) (dropping the constant). The copy constructor has a runtime of O(N), so it does not matter in this case because there is already a segment of code with an O(N2) runtime. The second for-loop also uses the get() and inset() functions, so it is O(N2) as well (dropping the constant). Thus, the overall time complexity is O(N2).

6b. The first for-loop calls the insertBefore() function N times, and the insertBefore() function has a O(1) run time because it is only switching around pointers, so thus, the first for-loop has a time complexity of O(N). The same is true for the second for-loop. Finally, the swap function has a constant run time because it is just switching around pointers. Thus, the overall time complexity is O(N). This is much better than the runtime of the function given in 6(a).