Amir Ali Yazdi-Nejad

UID: 204846462

CS 32

Carey Nachenberg

HW4

2. The reason that the call to the one-argument form of Sequence<Complex>::insert causes at least one compilation error is the fact that ‘>’ comparison operator has not yet been defined when comparing a complex object with a const complex object. If we define an overloaded operator, the problem will be solved.

3b. If we did not have the two-parameter overload of listAll, we would not be able to save the changed string after each step, since it is not passed by reference. Therefore, we had to make a new string every time, and the problem could not be solved.

4a. The time complexity of this algorithm is The time complexity of the outer loop (for (int i = 0; i < N; i++)) is In addition, the time complexity of the inner loop (for (int j = 0; j < N; j++)) is also The same is true for the most inner loop (for (int k = 0; k < N; k++)). Therefore, the overall time complexity of this algorithm will be

4b. The time complexity of this algorithm is The time complexity of the outer loop (for (int i = 0; i < N; i++)) is In addition, the time complexity of the inner loop (for (int j = 0; j < **i**; j++)) is also because in the worst case scenario i = N. The same is true for the most inner loop (for (int k = 0; k < N; k++)). Therefore, the overall time complexity of this algorithm will be

5a. The time complexity of this algorithm is The time complexity of the first loop (for (int k = 0; k < nmin; k++)) is , since nmin = seq1.size() = seq2.size() = N. In addition, the time complexity of the get function is in the worst case scenario. However, the insert function’s complexity is . Therefore, the overall complexity of the first loop is The second for loop (for (int k = nmin; k < n; k++)), will never run. Therefore, the overall time complexity of this algorithm will be

5b. The time complexity of this algorithm is Both sequences, have about N elements. Therefore, the first loop (for ( ; p1 != seq1.m\_head && p2 != seq2.m\_head; p1 = p1->m\_next, p2 = p2->m\_next)) runs about N times. Both insertBefore and swap have a complexity of In addition the second for loop runs a limited amount of times. Therefore, the overall time complexity of this algorithm will be