Sagar Doshi

UID: 604901376

Dis 1A, TA: Abani, N.

Homework 4

Question 2:

The reason this produces a compiler error is because the one-argument function insert(ItemType value) makes use of at least one of the comparison operators “<”, “>”, “>=”, “==”, “<=”. The coordinate class DOES NOT have an overloaded operator for any of these comparison operators meaning that there is no well-defined way to compare coordinate objects. Thus, you get a compiler error when the compiler tries to write a template version of the function.

Question 4b:

Every time you recursively call listAll, you are building upon the previous call’s path meaning that the function must take the string path argument so that the new recursive environment has access to the previous environment’s path to build upon as it goes deeper into the path.

In addition, listAll needs some File to perform its operation on. While iterating through a directory to only to find another directory, the File pointer passed to listAll is incremented thus requiring that you have a second Filer pointer argument.

Therefore, you need both a string argument to handle the building path, and a File pointer to iterate through multiple files recursively.

Question 5a:

* Outer (i loop)– O(N)
  + Inner(j loop)– O(N)
    - Inner(k loop) – O(N)

The time complexity of this algorithm is

Question 5b:

* Outer – O(N)
  + Inner – Will run (0, 1, 2, …, N times but already summing across iterations of the outer loop -> O(N)
    - Inner – O(N)

The time complexity of this algorithm is still

Question 6a:

The general form of the interleave function for N elements in all the passed sequences is as follows:

Execute a bunch of O(1) statements

….

* O(N) – for loop iterating through N elements
  + Insert is O(N) worst case (and interleave often appends to the end)
  + (Some O(1) statements)
  + Insert is O(N) worst case (and interleave often appends to the end)
* O(N) – for loop iterating through N elements
  + Insert is O(N) worst case (and interleave often appends to the end)
  + (Some O(1) statements)

The first outer loop has an overall time complexity of

The second outer loop also has an overall time complexity of

Combing the two, the overall time complexity of this algorithm is

Question 6b.

The general form of this new implementation of the interleave function is as follows:

Execute some O(1) Statements

* O(N) – for loop iterating N times
  + O(1) – InsertBefore() just goes to the front of the list and inserts a value
  + O(1) – InsertBefore() just goes to the front of the list and inserts a value

Some more O(1) statements

* O(N) – for loop iterating N times
  + O(1) – InsertBefore() just goes to the front he list and inserts a value.

Some final O(1) statements

Therefore, the overall time complexity of this algorithm is . This is much better than the previous algorithm in question 6a. Instead of using insert() which iterates N times to get to the end of the list, this function uses insertBefore() which only has to visit 1 element of the linkedlist. This eliminates the nested iteration over items resulting in a time complexity of just .