1. Explain preprocessor commands for redefining NULL. I understand why it is necessary, but not what is actually happening in this section of code.
   1. Not necessary-this actually is a “quick and dirty method” for instantiating several different data types for the first node set to NULL. NULL becomes the appropriate data type.
   2. This section of code being run can be modified by changing just Line 31 :

#define C\_DT 1

1. Line 79. If err is the string being passed in for the specific error that is being thrown, then it is acting as a parameter in errMsg, which is declared as a string in the SLLException class. Is this simply exception syntax? Hard to follow – first time I’ve seen exception handling in code.
   1. The err message is passes to the METHOD for errMsg, which is defined inside the class itself. The constructor is an initializer list.
2. Line 235 – NOTE – What made you include option 2 if you were going to comment it out and note why it is not the best option. Could this have been covered in a review or simply excluded? Will we ever need the option?
   1. Just used to clarify code and see the two differences. Teaching moment vs. standardized coding.
3. Line 278. Why use the return-by address method if it is not as robust and causes complications in the exceptions?
   1. Depends on if you know the context of when the program will be used. If we KNOW the values are small, we’re good, but if we don’t know, then we need to use references.
4. Line 384. Style question. Why the underscore? It is the only variable named in such a way. Why not just use size?
   1. This is to differentiate from the METHOD size().