CSE 232 SS13 Lab Session 9 Agenda

1. ADTs vs Implementation
	1. ex. Sequence (aka List)
		1. array-based and linked list implementations
	2. ex. String
		1. fixed/resizable array based implementations
		2. null byte vs length variable
	3. Interface (public) vs implementation (private)
2. Lab 9
	1. “retrieve” in this lab retrieves **by index**, not by value
	2. remove/retrieve return a success flag (as does insert), so the item itself is returned through a reference parameter
		1. In a test program like our driver, you can initialize the variable you feed the function to a specific value like -99 instead of checking the flag.
3. Drawing memory diagrams
	1. Primitive types
	2. Pointers
	3. Objects
	4. Arrays
4. Container classes
	1. Should almost never print anything, even error messages
	2. Returning error flag as an alternative to printing messages
5. Exceptions (not emphasized in this class) are a more common method of throwing errors:
	1. “Throw” an exception instead of printing an error
	2. Handling an exception:
		1. Try-catch mechanism – catch an exception and take some action yourself
		2. Let the exception raise – caller can take care of it instead
6. Dynamic memory allocation
	1. Should check that allocation was successful
7. Destructors
	1. Deleting dynamically allocated memory
	2. Deinitializing pointers
8. typedef
	1. Makes it easy to change the type of many variables all at once
		1. Ex. typedef double EType; -> typedef int EType;
9. Testing container classes
	1. Must test inserts/removes from front, back, middle – your algorithms may fail for only some cases
		1. Need a list of at least 2 elements in order to insert to the middle.
		2. Need a list of at least 3 elements in order to remove a middle element.