I have completed my project and submitted it to the Test Dropbox.

The Compile Dropbox found nothing wrong with my code. I have been compiling and testing without error on my end, and according to the test results, make test is also working correctly.

However, the test results ultimately fail, throwing these errors:

```
gcc -Wall -std=c99 -g -o littletest-sll littletest-sll.c sll.c gold_sll.c gold_integer.c
/tmp/ccecOzcE.o: In function `insertSLL':
/home/lusth/test/stsinischo@crimson.ua.edu/sll.c:21: undefined reference to `newSLLNode'
collect2: error: ld returned 1 exit status

gcc -Wall -std=c99 -g -o littletest-dll littletest-dll.c dll.c gold_dll.c gold_integer.c
/tmp/ccTxlyMM.o: In function `insertDLL':
/home/lusth/test/stsinischo@crimson.ua.edu/dll.c:22: undefined reference to `newDLLNode'
collect2: error: ld returned 1 exit status

gcc -Wall -std=c99 -g -o littletest-stack littletest-stack.c dll.c stack.c gold_dll.c gold_stack.c
gold_integer.c
/tmp/cc4uqDmY.o: In function `insertDLL':
/home/lusth/test/stsinischo@crimson.ua.edu/dll.c:22: undefined reference to `newDLLNode'
collect2: error: ld returned 1 exit status

gcc -Wall -std=c99 -g -o littletest-queue littletest-queue.c sll.c queue.c gold_sll.c gold_queue.c
gold_integer.c
/tmp/cc0nPjhb.o: In function `insertSLL':
/home/lusth/test/stsinischo@crimson.ua.edu/sll.c:21: undefined reference to `newSLLNode'
collect2: error: ld returned 1 exit status
```

I'm noticing that it is omitting my "node" class files - sllnode.c and dllnode.c - which are necessary for SLL and DLL to compile. Is this an incorrect implementation? How can I fix this?

For reference, here is the output of my make command, which compiles as expected:

```
make
gcc -Wall -Wextra -g -std=c99 -c integer.c
gcc -Wall -Wextra -g -std=c99 -c sllnode.c sll.c
gcc -Wall -Wextra -g -std=c99 -c dllnode.c dll.c
gcc -Wall -Wextra -g -std=c99 -c stack.c
gcc -Wall -Wextra -g -std=c99 -c queue.c
gcc -Wall -Wextra -g -std=c99 -c test-all.c
gcc -Wall -Wextra -g -std=c99 -o test-all integer.o sllnode.c dllnode.c sll.o dll.o stack.o queue.o test-all.o
```
I had a utility class that held my fatal function and a few debugging print statements. Lusth said that he isn't going to know to link the utils.c and utils.h module... so the only files that you can submit are the 8 module files and your makefile to test. You are going to have to add the node creation and stuff in the actual sl1 and dll classes. I had to do the same thing.

edit: It works with your makefile because you link your node files... But his makefile that he uses to test does not.

^ Correct.

This restriction is only for this assignment. With the subsequent assignments, you will submit programs, not just modules, so you will be able to organize your code as you see fit (including as many modules as you need).