

Timed Lab 4

Due Date and Time

Day: Wednesday, April 15th, 2015

Time: Before the end of your lab section

Policy

Submission

TURN IN THIS ASSIGNMENT ELECTRONICALLY USING T-SQUARE. SUBMISSIONS WHICH ARE LATE WILL NOT BE ACCEPTED. IN ADDITION IF YOU FORGET TO HIT THE SUBMIT BUTTON YOU WILL GET A ZERO.

What's Allowed

- The assignment files
- Your previous Homework and Lab submissions
- Blank paper for scratch work

What's Not Allowed

- The Internet (except the T-Square Assignment page to submit)
- Any resource on T-Square that is not given in the assignment.
- Dropbox (if your harddrive crashes we will let you retake it!)
- Notes on paper or saved on your computer.
- Textbook
- Email
- IM
- Contact in any form with any other person besides TA's
- If you have any questions on what you may not use then assume you can't use it and ask a TA.

Restrictions

- You may not leave the classroom until we have verified that you have submitted the lab. If you leave the classroom without submitting you will receive a zero.
- **YOU MUST SUBMIT BY THE END OF YOUR LAB PERIOD.** Bear in mind that the clock on your computer may be a few minutes slow. You are supposed to have a full class period to work, and we are letting you use the 10 minutes between classes to make sure you have submitted your work. **WE WILL NOT ACCEPT LATE SUBMISSIONS**, be they 1 second or 1 hour late.
- The timed lab has been configured to accept one submission. If you accidentally submit or submit the wrong version, call one of the TA's and we will reopen submission for you. But PLEASE PLEASE PLEASE submit the right thing the first time. The TA's get busy at the end of the lab making sure everyone submitted, and it's tough doing that AND re-opening submissions for 5 students. Yes, it does happen. Don't let it happen to you.

Violations

Failure to follow these rules will be in violation of the Georgia Tech Honor Code. **AND YOU WILL RECIEVE A ZERO** and you will be reported to the professor and the Office of Student Integrity. We take cheating and using of unauthorized resources **VERY SERIOUSLY** and you will be in serious trouble if you are caught.

Remember

- Please don't get stressed out during a timed lab. You have plenty of time however use your time effectively
- Partial credit is given. If you don't know something at least **TRY** do not just walk out of the lab or submit an empty file. Do the best you can!
- Make sure your code can compile. Your code must compile to get any points for this assignment!
- Remember what you can and can't use. If you don't know, ask a TA if you can use it. If we catch you with unauthorized resources we will give you a zero, so better to be safe than sorry.

The Assignment

In today's timed lab, you will be making an integer linked list in C. It must conform to the following requirements:

- 1) The list must be singly linked, i.e. each node only points to the next node in the list.
- 2) The list must have head and tail pointers store in the LIST struct.
- 3) The tail of the list must have its next pointer set to NULL
- 4) The number of nodes currently in the list must be kept track of in the size field in the LIST struct.
- 5) The list must only hold integer values as data. This means you will to store the integer in the node without calling malloc/free to allocate space for the integer.
- 6) The list will function similarly to a set. There should only be **one** node with a given data value in the list at at time. So for example:
 - a) If push_front(list, 5) was called on a list that already contains a 5, the function would return without changing the list in any way. It would **not** add any new nodes, remove any nodes or change the size of the list.
 - b) If push_front(list, 5) was called on a list that does not already contain a 5, then a node with a data value of 5 would be added to the head of the linked list and the size of the list would be incremented.
- 7) The code must implement create_list, create_node, push_front, peek_index, pop_back and print_list.
- 8) Your code must not crash, run infinitely or produce any memory leaks.
- 9) **Your code must compile with the Makefile that we have provided!** If it does not compile with our Makefile you will receive a 0.

Running your code

We have provided a makefile for this assignment that will build your project. Your code must compile with our makefile. Here are the commands you should be using with this makefile:

- 1) To run the tests in test.c: **make run-test**
- 2) To run the tests in test.c in debug mode: **make run-debug** (be sure to run this command first: **make clean**)
- 3) To debug your code using gdb: **make run-gdb** (run the clean target first)
- 4) To run your code with valgrind: **make run-valgrind** (run the clean target first)

Hints

- 1) Note that in this timed lab you will NOT be using any function pointers. You don't need to worry about that.
- 2) Make sure that you test your list thoroughly in your test.c file. Our test cases will be exhaustive.

Deliverables

You must submit only the files listed below. We will not accept any Internet links. We want the files below and only these files!

- 1) list.c
- 2) list.h
- 3) test.c
- 4) Makefile