

Collaboration Policy

Fall 2015

Introduction

In order to help the course staff evaluate each student in CS 33 as fairly and individually as possible, we have written a collaboration policy by which we expect all students to abide. **Please read this policy carefully**, as it may differ from collaboration policies in CS classes you have taken previously. Please print the last page of this document, sign it, and turn it in at your first lab section to acknowledge your understanding of the policy.

We strongly encourage teaching and learning with your peers. At the same time, your work must, in the end, represent your own understanding of the material. In that spirit, our overall policy is that you can talk about almost anything and work out solutions together, but your final write-up must be your own work. Debugging is one area where we particularly encourage you to help one another. See the Testing and Debugging section for more information.

You may (and are encouraged to) talk to other students about the content of the lectures and of the textbook, and about high-level concepts in general, but please ensure that in doing so you are not violating any of the following policies.

Note that the point of this policy is to prevent discussion about the content of the assignments. It is not, however, meant to destroy the camaraderie of the department. There is no harm in talking to your classmates and friends about how you're doing, how far along into an assignment you are, and so on. Inter-person communication is important, and we want to preserve our great student culture. Use your discretion, but as always, if in doubt, talk to the course staff (even if it's after a possible violation).

Permissions

You must ensure that your course documents have permissions which do not permit other students to view them. Ask a consultant or a TA for help with permissions if necessary, or use `man chmod`. Failure to do this may be seen as a violation on your part.

Projects

While working on projects, you are allowed to discuss how to approach an assignment with other CS 33 students. However, you should never sit down to discuss an assignment with someone else before you've analyzed the problem in depth on your own. In addition, we ask that you write up your code independently and separately from any joint discussions, and that you not take away any notes from such discussions (please erase the whiteboard and don't take a photograph of it!). When you are coding the assignment, you may not code along with another student. You may only discuss your code or allow others to look at your code once you have moved on to debugging. While you may search the internet for documentation, such as looking up C syntax or how certain

instructions work, you may not search the internet for solutions to problems. Copying code from the internet is a clear violation of the academic code. Additionally, **do not post code on Piazza**.

Testing and Debugging

Testing ideas and a testing plan are part of project design, and you may collaborate to come up with them. Test code, however, must not be shared. Saying something like “Oh, have you tried doing an add after a jump? I had a strange problem with that.” is fine, but “Try compiling this chunk of assembly code” or “Load this in to your register” are not.

You may not look at or copy anyone else’s code for the purpose of helping you write your own, and you may not permit anyone to look at or copy your code for that purpose. **You are permitted to work together on debugging**, and may look at someone’s code or work together to modify code for the purpose of helping him/her debug, *as long as your code is not open concurrently*. Note that this policy is significantly more lenient than that of most CS courses. You all have different skills when it comes to problem solving, so we encourage you to help each other. You can often learn just as much from your peers as you can from TAs.

If you receive help on debugging, you must cite this help in your README

ex: *“I received debugging assistance from dj and dcorrea”*

Labs

Collaboration during lab periods is allowed, but each student must still turn in their own solution to the assignment. Discussion and debugging are permitted, but pair programming is not. Please note that, especially if you get checked off after your lab section is over, we reserve the right to do a “wire-pull test” (i.e., ask you to explain your program) and to use all of the tools at our disposal to compare your code to that of other students. This also applies to all other assignments and work for the course.

Clickers

A small portion of your grade is based on your participation in class using clickers. Your clicker is registered to you, and thus your responses to in-class questions are recorded in your name. It is a violation of Brown’s academic code to give your clicker to someone else to use on your behalf, or to use a clicker on behalf of someone else.

Collaboration Policy Agreement

For each of the below items, please check the appropriate box based on whether the described is an appropriate level of collaboration. These answers are required for us to accept your collaboration policy. You should be able to explain your answers to a TA when you turn it in.

1. Bob has been struggling with Malloc. He tells Alice this, so she sends him her code so he can see how it's done.

☐ Acceptable

☐ Unacceptable

2. Alice knows GDB really well. When she sees Bob struggling with his project, she goes over to him and offers to teach him how to debug multithreaded programs and helps him figure out the right places to put breakpoints. She reminds him to cite her help, and he writes her name in his README.

☐ Acceptable

☐ Unacceptable

3. When Alice and Bob are discussing breakpoints, Eve walks by and stops for a second to look at what they are working on. While she's there, she notices a conditional in Bob's code that she forgot and goes to edit her own code.

☐ Acceptable

☐ Unacceptable

4. Carol and Bob have finished the coding portion of Shell and have moved on to testing. In order to expedite the testing process, they each write 5 test case scripts and then share them with each other.

☐ Acceptable

☐ Unacceptable

5. Alice, Bob, and Carol are brainstorming ideas for Maze together. They use whiteboards to brainstorm and then erase the whiteboards and go to the Sunlab to work individually.

☐ Acceptable

☐ Unacceptable

6. Alice and Carol are sitting next to each other while they work on Snowcast. They are both working, when Alice finds a bug that she's having a hard time locating. Carol leans over to look at Alice's code from her workstation, helps Alice debug, and then looks back to her own screen.

☐ Acceptable

☐ Unacceptable

By signing this document, you agree to abide by the CSCI0330 Collaboration Policy, as published above and available on the course website. You must sign this page and hand it in by your first lab section in order to receive credit for the first lab.

Name _____

Brown AuthID (used to log into Banner) _____

CS Login (leave blank if none) _____

Email Address _____

Signature _____

Date _____