Collaboration Policy and Registration Form

Introduction to Computer Graphics, Fall 2016

1 Collaboration Policy

Since cs123 does not have any examinations, your grade is based solely on your homework and programs. In order to evaluate you in the course, we must be sure that your assignments are your own work.

For every assignment you should do your own thinking, design, coding, and debugging. You should never let yourself be led by another student or receive an amount of help which makes an assignment significantly easier. Conversely, you should never assist another student in a matter that would overly lead them or make their job much easier.

This is important to keep in mind: **Just "helping a friend out" is okay only if the help does not breach the collaboration contract. Otherwise, it is as much a violation as receiving that help.**

We strictly enforce this policy. Cheating has been a significant problem for the CS department in the past, but because of TA alertness and a software package called Measures of Software Similarity (MOSS), illegal collaboration is easily detected in cs123.

1.1 Permitted Collaboration

The following actions are **encouraged** and are not considered collaboration.

1. **Discussing solutions to the "algorithm" assignments before each project.**
   This is new to the course this year. If you choose to discuss the algorithm assignments with another student you must **erase/throw out all notes** from the discussion before writing up the solution on your own, and you must **write the logins of the student(s) you collaborated with** next to each problem. Your handin should not look identical to another student's.
2. **Discussing material covered in lectures or the textbook.**
3. **Discussing Qt or support code, as long as it is a technical question about the support code itself.**
4. **Discussing the requirements of an assignment**.
5. **Asking general knowledge questions about syntax and C++**.
   Examples: How do I make something public? How do I set up debugging in Qt Creator?
6. **Discussing general techniques of designing, coding, or debugging**.
   Example: "When I get a segmentation fault, I use gdb to find the line where the program crashed."
7. **Discussing the design of an assignment.**
   Design is a crucial part of the programming process, and discussing it can be very valuable. Try to work out as much as you can on your own. If you choose to discuss your program design with another student, you must **erase/throw out all notes** from the discussion before you start coding, and you must **include his/her login in your README file** along with a description of the collaboration. Your discussion should focus on conceptual aspects of the design without getting into specifics. If you find yourself discussing class names and method names, you're going too far.
8. **Discussing general traits you may notice in another's compiled, running program (without looking at the code)**.
   Example: "It looks like your lighting is a little off."
9. **Exchanging render results, scene files, and other test data.**
   Example: "I exchanged {sample renders, scene files} with psastras."

There is no penalty for permissible collaboration with another student, as long as you document the students with whom you collaborate, as noted above.

1.2 Prohibited collaboration

The following things are not allowed under any circumstances.

1. **Copying code or test cases.**
   You should not be writing down anyone else's code (including code from the internet), or allowing anyone else to write down your code. Remember, we have software designed explicitly to look for undue similarity of code.
2. **Discussing pseudocode**.
   Pseudocode in cs123 is close to discussing the code itself. If you've gotten beyond discussing the basic kinds of classes and initial relationships, you're ready to work on your own.
3. **Taking someone else's design.**
   Discussing basic design with someone else is okay. However, just taking someone else's entire design is not allowed.
4. **Debugging with another person.**
   Sitting at the same computer with someone else and trying to fix a bug is not allowed. Describing your problem to someone and asking for advice on how to fix it is okay, but you should debug it by yourself.
5. **Looking at someone else's code.**
   You should never read anyone else's code, whether it is on the screen, printed, or written out by hand.
6. **Asking for help on something you haven't thought about yourself.**
   Always make every attempt to tackle a problem yourself before asking another student or a TA. It will help you become a better programmer, as well as a better student.
7. **Having incorrect file permissions/being careless with your source files.**
   We require that all students maintain appropriate permissions on their coursework. Other students should not be able to access, view, or copy your files. If you don't know how to do this, ask a Sun Lab consultant or see a TA. You can also type man chmod in a shell to see the UNIX documentation. **If another student copies your work, you will be held accountable for negligence**.
8. **Using previously published solutions.**
   Students retaking the course may have access to course materials and solutions published the year prior. You may not use any assignment materials (solutions or otherwise) that were not published on the course website before the assignment was due. Students may not consult their own solutions from the previous year when completing the current year's work.

On a general level, you are not allowed to let yourself be led by another person to the extent that your task becomes significantly less challenging because of your discussion with that person. Always be careful not to venture into design and coding specifics.

In addition, don't sit down with someone else before you've analyzed the problem in-depth on your own. You should do your own problem solving, do your own program design and decomposition, and design your own algorithms and data structures. If you are discussing what algorithms or functions you wrote to solve the problem, or describing header files or specific lines of code, then you are breaking the Collaboration Policy.

1.3 Conclusion

We believe that this policy is explicit enough to guide your judgment and that we have not left you much gray area. If you are ever in doubt about the legality of your actions, be sure to clear them with a TA, even after the event has already occurred. When we confront a student with a case of suspected violation, an answer of "I didn't know that this is wrong" will not be met with sympathy. Suspected cases of disallowed collaboration will be referred to the deans and typically result in a directed NC on your transcript and parental notification for the first offense.

Again, note that you are expected always to initially approach a problem on your own, and seriously attempt to find a solution. You are honor-bound to preserve your independence of thought. Remember that the TAs and lecture slides should always be your first resource when you have a question or problem.

2 Registration

By signing below you are bound to these terms and will be officially registered for cs123. You will be added to the cs123 student group (and, if necessary, your account will be created) before your first lab section.

Name: _____ Date: _____
Signature: _____ CS login: _____