Collaboration Policy
CS224 - Interactive Computer Graphics
Spring, 2015

General

Since CS224 has no exams, your grade in CS224 is determined by projects, homeworks, and class participation. Because of this, for all homework and projects you turn in, it should be clear to us exactly what part of the work is yours.

We want you to learn and understand more about the topics presented in this course by talking with your peers. Therefore, CS224 allows broad collaboration, as described below, on all homeworks and projects, except for the cases listed in the next section. If you do collaborate with other students, you must list their logins in the project’s README file or on the top of your homework, and make clear the details of their contributions, as in “Hubert showed me how to change variables in the integral, because I’d forgotten how to do that. The algebra of actually doing the change of variables, and all the probability calculations, are all mine,” or “this implementation of Russian-roulette sampling was copied straight from Pharr and Humphrey's book. But the ray-chasing and scattering computations are my own.”

What’s allowed:

• Discussion of problems, solution methods, coding strategies, even software organization.

• Use of external (and credited) resources, except when they provide the answer directly. If we ask you to write a ray-intersect-sphere procedure, you’re free to read some other book’s description of how to find the intersection point. But you should stop reading when they start writing code.

What’s not allowed:

• Shared code. If you and Robin want to discuss raytracing, do it on paper or at a whiteboard. When you’re done, erase everything. Then go reproduce the ideas yourself later — a long enough time later that no particular piece of code remains in your head. If you’re doing this right, your code should be so different that you’re generally not using the same variable names or procedure names.

• Shared equations. You can work on math together. But then you should tear up your notes and do it again by yourself, enough later that your solution and your friend’s solution don’t look identical.

• Fixing someone else’s code. If you’ve finished a project, and your friend is stuck in debugging hell, you can help him debug. You can play the role of the rubber duck in rubber duck debugging [http://en.wikipedia.org/wiki/Rubber_duck_debugging](http://en.wikipedia.org/wiki/Rubber_duck_debugging).

We’ve made some modifications to last year’s collaboration policy, so this bit is important: when rubber duck debugging for a friend, you are allowed to ask questions to understand what
their code is doing, but you cannot tell them where their code went wrong. It’s up to your friend to have the “ah-hah! I know what I did wrong” moment midway through explaining their own code.

What if neither of you has finished the project yet? Well, it’s awfully hard to sit at adjacent machines and not take back to your machine some of what you just saw on your friend’s screen, violating the “wait until later to recreate it” rule. You therefore should not do this.

What if you’ve finished, you help your friend, and in doing so, realize that you missed something? You can go back and fix it (after waiting until later, so that you’re not just copying, but recreating), but you should be very careful to give appropriate credit to your friend. What if it’s just 10 minutes before handin? Too bad. You missed your chance, because 10 minutes isn’t long enough.

Copying from other students, or even from your joint work with them, is not an acceptable form of collaboration. In cases like the one above, where a piece of code was copied from a text, you’re still responsible for knowing every detail of the code, as in “What happens if we change this “less than” to a “less than or equals”? or “Why use this variable declared \texttt{uint8} rather than \texttt{int}?”. We (the CS224 staff) reserve the right to quiz you in detail about any work you hand in, so make sure that you \textit{really} understand anything you turn in. Failure to be able to explain something you submitted will be treated as evidence that you didn’t produce it yourself.

You may use outside resources such as books, journals, and the internet to get information but not to get answers: if you’re asked to read a paper and offer your insights on it, finding someone else’s insights and slightly modifying them is \textit{not} OK. If you were asked to implement Russianroulette sampling from a description in a homework assignment, looking up Pharr and Humphreys’ implementation is not OK either. On the other hand, if you’re told to write a photon-mapping renderer, and decide that you want to use Russian-roulette sampling, which you’ve already discussed in class and in a HW, using the P&H implementation (credited in the code!) is perfectly OK. (You’re still responsible for understanding that copied code, however.) If you have any doubts about what’s OK, ask a TA.

Whenever you do use information found in outside sources (not directly linked to from the course website) you \textit{must} mention which source(s) you used on your homework or in your project README. And once again, you’re expected to understand any information you do use. If you can’t explain it, you didn’t understand it, and you don’t deserve credit. As a guide, ask yourself, ”If I were confronted with this material and my own answer side by side, would I be embarrassed when asked detailed questions?” If so, you’re making the wrong choices.

\textbf{Other forms of non-permissable Collaboration}

1. Incorrect File Permissions / Being Careless with Source Files
   Maintain proper permissions on your work. You must make certain that other students cannot access, view, or copy any of your files. Use chmod to adjust the permissions on your files. If another student copies your work, both you and that student will be held responsible.

2. Being Careless with Homework Solutions / Discussions
   Do not leave material you have worked on lying around for other students to copy. For
example, after working through a problem on a whiteboard, you should erase what you have done.

Final Project

For the final project, you are allowed to use 3rd party libraries and other people's code, so long as you give the authors appropriate credit. Of course, your work will be evaluated on the basis of your contributions above and beyond that external material.

By signing below, you agree to the CS224 Collaboration Policy as stated above.

Name______________________________________________

CS Login____________________________________________

Signature____________________________________________

Date_______________________________________________

Final Comments

The goal of this course is that you learn. While we allow broad collaboration, we hope that you work through projects and homeworks first by yourself. Going to someone else immediately for the answer will not allow you to fully understand the material. Keep in mind that we reserve the right to question you about any work you turn in.

If you have questions at any time about the collaboration policy, do not hesitate to talk to a TA or the professor.