

CS 200 and CS 212

Assignment and Project Submission Guidelines

Northern Arizona University • College of Engineering, Forestry, and Natural Sciences
Department of Electrical Engineering and Computer Science

Introduction

These guidelines explain how you should be submitting your work for CS200 and CS212. Note that CS 212 uses projects, not homework assignments, so skip to the project guidelines.

General Information

Assignments and projects are presented in the Bb Learn system in both .pdf and .docx formats. It is expected that you will complete and submit your work in .pdf, .doc, .docx, or .rtf formats. You may bundle multiple files into .zip, .jar, .tar, or .rar files. This is especially important if you are submitting code and executable files. If your own computer does not already support these formats you have several choices: purchase the appropriate software (the Student version of Office is available through the Bookstore, for example), download a free package such as OpenOffice, or simply use one of the many computers available around campus – in the Library, Internet Café, or one of the Computer Labs.

Note to Mac users: .rtf files can sometimes turn out very large and may take longer to upload. You may want to experiment with other formats to see which works best for you.

Whatever software you decide to use, it is up to you to figure out how to access it and use it to upload your work into Bb Learn. There are plenty of resources starting with your fellow students and running up to the technical support for Bb Learn. I can help with specific problems but am not teaching a class in Word or Acrobat or whatever. This is especially true of program files; there are too many environments for me to be proficient in them all. One of the strengths as a computer professional that you should be developing is self-sufficiency. Employers look for that and this is a good time to practice it. It doesn't hurt your grade if you can show me a new trick or two.

Due Dates and Times

Work is due according to the assignment in Bb Learn. I strongly suggest you upload early just in case the system has a problem; you don't want to be counted late just because your upload was interrupted. If you upload early and discover you need to make a change, you can simply re-upload. That is safer than waiting until the last minute because at least you will have submitted something for grade. The last submission I get is the one that counts. A little perspective: if you are working a real job and tell your boss the work was late because the system failed, how sympathetic would he be? One boss said, "Oh yeah? Well, the payroll system's going to be a little late with your check next week..."

Projects may receive partial credit if submitted late but homework assignments **will not be accepted late**. The reason for this is that solutions for homework will be posted after the due date. Depending on your circumstances, I may provide a make-up homework, but don't count on it.

General Submission Guidelines

- Every submission **must** have a cover page, just as if it were to be printed.
- Each submission must have the class (ex. CS200), the assignment number and title, and your name, at a minimum.
- Spell and grammar check your submission. Think of it as practice for the real world; you'd never submit a badly written document to your boss.

Homework Guidelines

- If you submit via email (due to problems with BbLearn, presumably)
 - you **must** put your work (even a single file) in a .zip, .jar, .tar, or .rar file. Otherwise, things tend to get garbled.
 - If you are using an archive file, save your work in a file named *xx0cs200hw0*, where *xx0* is your NAU login and *hw0* is the homework number. *Example: I submit the 3rd homework as pek7cs200hw3.zip – note the file extension is dependent on the archive type you are using.*
- If you are submitting on BbLearn and not using an archive, naming convention is not important.
- Each answer should be preceded by the question. One good way to do this is simply open the .docx version of the assignment, save the file as your own, and modify it, inserting your answers after the questions.
- Show your work, give full answers and explanations, etc. Brief answers are hard to give partial credit to, since it's harder to tell whether you were generally right and made a simple error or were generally wrong but happened to get close to the right answer. If I can't tell, I assume the worst and count it off.
- Do your own work. It is fine to help each other out but I want to see your work and your answers in your own words. See the section on plagiarism.

Project Guidelines

- If you submit via email
 - you **must** put your work (even a single file) in a .zip, .jar, .tar, or .rar file. Otherwise, things tend to get garbled.
 - If you are using an archive file, save your project report in a file named *xx0cs200proj0*, where *xx0* is your NAU login and *proj0* is the project number. *Example: I submit the 8th project as pek7cs200proj8.zip – note the file extension is dependent on the archive type you are using.*
- If you are submitting on BbLearn and not using an archive, naming convention is not important.
- The cover page should also include the date you **submitted** the report.
- The report should consist of three sections: Project Overview, Results, and Conclusions.
 - The project overview should consist of two sections: Purpose and Approach.
 - Purpose should be a brief restatement, **in your own words**, of the problem you are solving.
 - Approach should be a statement of how you intend to solve the problem. You can draw diagrams or use pseudocode or whatever method to get your plan across.
 - The Results should be text describing what happened during your project. Not every project gets completed but I can give you partial credit if you show me that you put effort in and at least were proceeding in a reasonable direction. You can include sample output to support your text, but output alone without explanation is not sufficient.
 - The Sample Output can be screen shots, text capture, or whatever is needed to show that your program works as desired. There will generally be test data given in the project assignment; you should show that the program works with at least that data.
 - Deliverables, such as source code, screenshots, etc. do not have to be embedded in the report. They can be included separately and referred to in the report. Don't try to 'bulk up' your report with deliverables; that just makes it hard to read and actually hurts your score. Keep embedded deliverables concise and to the point – screenshots that support your report or code fragments rather than your complete code.
 - The Conclusions section should have a paragraph of what you took away from the project – did you learn something interesting or maybe your approach failed but you figured out a better way? It should also have a paragraph about any problems you encountered and, hopefully, overcame. Don't be afraid to use more than two paragraphs but there should be at least two.
- Submit your source code along with your report. If it's a single file you can submit it as is; otherwise, package your files into a .zip or .jar file. I'd like to see file names following the same convention as the report file so we don't get them lost or mixed up. However, the files inside a .zip or .jar can follow any naming convention you like.
- Do your own work. It is fine to help each other out with general design but I want to see your work and your code in your own style. See the section on plagiarism.

Report Rubric

Low Quality

- Not organized, with sections missing, running together, or out of order. Not all required topics are addressed.
- The document contains serious spelling and grammar problems.
- The content of the sections are vague or off-topic. They include generalities without support, unnecessary or irrelevant information, or incorrect statements. A reader would have difficulty following the writer's reasoning.
- Steps are omitted; a reader would be confused as to how to proceed at any point or would not be able to understand how the writer proceeded.
- The results fail to achieve the purpose of the project; a reader would not be able to use the solution for any useful purpose.
- Regarding the conclusions, a reader would not have any idea of what the report writer had difficulty with, learned, and took away from the project.

Medium Quality

- Organized in sections but with some required sections missing or not clearly labeled. All topics are touched on, even if the specific section is missing.
- All sections have been spell- and grammar-checked but there may be some awkwardness of expression that the checking tools did not catch.
- The content of the sections may be a little vague or fail to address the scope of the sections. A reader can follow the report and duplicate the steps with some difficulty to achieve similar results.
- Some steps may be left out or glossed over; a reader might be confused as to how to proceed at certain points.
- The results may not achieve the all the purpose of the project. A reader could use the solution to some effect but not without some error or caution.
- Regarding the conclusions, a reader would have a vague idea of what the report writer had difficulty with, learned, and took away from the project.

High Quality

- Organized in sections with all required sections represented and clearly labeled.
- All sections are in proper English that has been spell- and grammar-checked.
- The content of the sections are clear and address the scope of the sections. A reader can easily follow the report and duplicate the steps to achieve similar results. There is no vagueness or unnecessary prose intended to 'pad out' the report.
- Steps are not left out or glossed over; a reader would not be confused as to how to proceed at any point.
- The results achieve the purpose of the project and a reader could use the solution to similar effect.
- Regarding the conclusions, a reader would have a clear idea of what the report writer had difficulty with, learned, and took away from the project.

Plagiarism

You'd think it would be really easy to copy each other's work, especially with electronic submissions. Just change the names and who would know? Well, me, for one. You probably already know that there are programs out there for comparing documents and producing a statistical analysis of congruence – how similar they are to each other. But there are other ways, too. What's not always so easy to tell is who did the copying and who provided the material. The way we deal with that is to penalize both parties. Read the NAU policies again – the University takes it very seriously so I have to as well and so should you.

But on the other hand, I want you to work together and help each other out. How does that jibe with the proscription against copying? Here's the deal: help each other with words and pictures but don't lend anyone a copy of your work. Draw someone a diagram, share pseudocode, write an example on paper. That way your work can't be copied and potentially get you in trouble.

On the flip side, when someone is helping you, rephrase the information in your own words. Don't use the same variable names, comment style, etc. Definitely don't cut and paste someone else's work; rewrite it for yourself and make it your own. While you're doing that, you may actually find some ways to improve on it, which will make it even more your own.

If you copy code from an external source, as some of the projects encourage you to do, credit it in the comments. Use a full citation – the website or printed media you got it from. Don't take a whole program; take just the fragments you need. If a framework is provided, you can use it freely without citation but you should comment where your code begins and ends to make things easier for me or a grader.

For work that isn't source code, it should be in your own words. That doesn't mean changing a word or two in someone else's document. I had someone rename a document and make some wording changes but I was suspicious because all the headings were the same as another one – right down to the misspellings. If you're going to all that trouble you might as well re-write the document completely so it isn't so obvious. It's the same as before: help each other out on paper or whiteboard but don't share your documents.