CS222 Advanced Programming: Overview

Course Overview

- Section Information
- Resources
- Grading
- Communication
- Miscellany

Section Information

- Instructor: Dr. Paul Gestwicki
- Section: 2

The course catalog provides this short description of the course:

> Project-intensive study of advanced topics and best practices in software development, including advanced language features, modular decomposition, and development tools.

You learned fundamentals of programming, data structures, discrete mathematics, and composition in your prerequisite coursework: these things are building blocks. In this course, you will learn how to assemble these to solve real problems. Real problems rarely have a closed form solution, where there is one "right answer." Instead, you have to balance design decisions and manage ambiguity. Learning how a Computer Scientist approaches such problems will prepare you for future project-oriented coursework and, more generally, to be a better developer.

This semester, our inquiry will be guided by the following essential questions of software development:

1. What is *software craftsmanship*?
2. How do you know when a feature is done?
3. How do small teams of developers coordinate activity?
4. How does a software developer use ubiquitous and chaotic information to be a lifetime learner?

Student Learning Outcomes

The Computer Science Department has identified the following learning outcomes for this course.

1. Explain, justify, and follow heuristics of code quality, including: coding standards and style documents; refactoring as a technique for improving code quality; clear naming and language-specific naming conventions; using abstraction and decomposition to eliminate redundancy; prudent use of comments and version control commit messages
2. Demonstrate and justify software development practices that incorporate the compiler, debugger, version control, and testing features of a modern integrated development environment
3. Apply code review techniques to identify design and implementation defects
4. Create and analyze software following conventions of object-orientation, including the Single Responsibility Principle and model-view separation
5. Apply user-centered philosophy to design simple graphical user interfaces, justify usability considerations, and implement the user-interface in a high-level programming language.
6. Plan, manage, present, and evaluate multi-week programming projects created by small teams using an incremental and iterative approach
7. Demonstrate reflective practice for professional improvement

By meeting these objectives and addressing the essential questions, you will make progress toward all eight NACE Career Readiness Competencies.

Commitment

CS222 is a three credit-hour class, and you are expected to invest nine hours of effort per week into it. This expectation is not arbitrary: it comes from federal student loan legislation. In a sense, these laws define what you pay for, or what you commit to doing, with federally-backed student loans. Hence, they are a standard for all of higher education in the United States.

It is worth thinking about the implications of this level of commitment. First, a bit of math. If you are taking 15 credit-hours of courses, that translates into an expected commitment of 45 hours per week. That's significantly more than full-time labor because you don't have a union that negotiates meal and coffee breaks.

Second, consider effort. The weekly commitment assumes focused effort. This is not to be confused with text-messaging, Instagram-checking, Netflix-streaming, browser-tab-switching effort. This level of effort is challenging to maintain and requires discipline. Consider, however, that being able to consistently perform at this level of focus is a primary outcome of a quality higher education. Learning to manage it prepares you for peak performance in your post-graduation community. Put another way, this work is formative, and the effort is moral.

When I design a course, I think carefully about where I expect your focus to be. You should be able to succeed in this course by committing nine hours of attention per week assuming that you have met the learning outcomes of the prerequisite courses. As with any aspect of the course, feel free to talk to me about it if you find that this is not the case. This discussion implies, of course, that if you did not meet learning outcomes of the prerequisite courses, then this course will take more effort, because we are building directly upon their foundations, especially CS121. That is, we will use what you learned in CS124 and your writing courses, and we are extending what you learned in CS121.

Resources

Book

The following book is required for this course.

- Robert C. Martin. *Clean Code: A Handbook of Agile Software Craftsmanship*, Second Edition. Addison-Wesley. 2026.

In writing, we will often reference chapters of the book as "CC*N*," meaning "*Clean Code* chapter *N*."

Hardware

As per the undergraduate catalog, Computer Science majors are expected to have a laptop capable of running contemporary software development tools and to bring that machine to class meetings.

Software

We will be using Flutter, Dart, Android Studio, and git. These are all available at no cost. Installing and configuring these will be part of your first week's expected work as described on the assignments page.

Web Services

You will need to sign up for a GitHub account if you don't already have one. As always, remember to use strong and unique passwords for different Web services.

You will need a Google Account for collaborating on Google Drive.

Schedule

The semester is divided into three parts. First, we will dive into fundamental topics of contemporary software development, including tools and techniques to help you become a more productive programmer. There will be regular, rigorous assignments that will have you studying best practices.

We will then apply these ideas in a two-week project that will involve programmatically manipulating data pulled from a public source. This will help you think more critically about the challenges and ambiguities of software development.

The majority of the semester will be spent on student-designed, team-based projects. These will be completed in multiple iterations. That is, you will build your project in small pieces, learning as you go.

The withdraw deadline is October 28. It is recommended that you talk to the instructor if you are considering withdrawing from the course. Students sometimes misunderstand the course goals and their progress toward learning objectives, and it cannot hurt to talk these over with the instructor before taking drastic action.

Following university policy, expect a meeting during our scheduled final exam slot, which is 12:00–2:00 on Tuesday, December 8.

Decorum

Your learning is commensurate to your participation, and so attendance is expected. You are responsible for your learning regardless of attendance. If you miss a class meeting, you should consult with trusted classmates to ensure you are caught up.

We will begin our meetings on time, which means you should arrive a few minutes beforehand. This gives you adequate time to get situated and exchange greetings. Recognize that it is inconsiderate to miss or be late to a meeting.

Stow your phone in silent mode for the duration of meetings. If you anticipate receipt of a communiqué so critical as to merit interrupting a meeting, it is certainly better not to be in that meeting at all.

If we are not actively using laptops in a class activity, stow those as well. When we are using laptops for an activity, maintain focus by shutting down social networking software and disabling notifications. If you are giving a presentation, either alone or with a team, ensure that your laptop is ready before the meeting starts.

Always have paper and a writing implement on hand during meetings so that you are ready to take notes or record questions. The most frequent lie that we tell ourselves is, “I will remember this.” Transcribing your notes after the meeting is an excellent way to study, if not the best way.

It will help you structure your experience and strengthen your understanding. If you want to do your transcription on a computer rather than on paper, that's up to you. Be careful, though, not to delude yourself into thinking that your distraction machine is an appropriate replacement for a notebook. I recommend Bill Rapaport's How to Study guide and the Cornell Note Taking System for more advice on how to succeed in your studies, and the tips page provides advice tuned to this course.

Listen actively. Assume the speaker knows something you do not. Listening this way means you are not simply waiting for your chance to speak. More often, you will ask probing questions in order to better understand.

Most of our face-to-face communication is non-verbal, and a corollary is that it is easy to misread someone's intention in text. Keep in mind The Rule of Three Interpretations: If you read something that rankles you—whether it appears to be bull-headed, offensive, or just plain wrong—pause for a moment. Think of three different possible interpretations of the text. The most charitable one is probably correct.

Grading

We will be using triage grading. This system recognizes that student work can be graded on a four-point scale.

Points	Interpretation	Letter Equivalent
3	Essentially correct	A
2	Partially correct	C
1	Essentially incorrect	D
0	No submission	F

Essential correctness includes clarity of communication, and that is why this course has a writing prerequisite. Not every submission must be a masterwork of the English language, but if a submission contains errors that impede interpretation, then it cannot be essentially correct. See the tips page for specific advice about how to write clearly and concisely.

Course Grade

Your course grade will be determined by a weighted average of the various components of the course according to the table below.

Category	Weight
Assignments	30%
Projects	60%
Final Exam	10%

If your final project score is greater than your short project score, then your Project score is simply your final project score. Otherwise, it is a weighted average of 20% short project and 80% final project.

The relative contribution of your project score may not exceed the relative contribution of your assignments score. This policy balances individual accountability with the value of teamwork.

Communication

Office Hours and Appointments

I am glad to talk to you about your work and your plans. You are welcome to email me any time with questions. I almost always respond within two working days and usually much faster.

My office hours are first-come, first-served. You don’t need an appointment for office hours.

Not every student is available during my office hours. If you want to meet at another time, simply email me your availability, and we will use that to set up a meeting. Be clear about whether you are seeking an online or face-to-face meeting, and let me know what you would like to discuss so that I can be prepared.

Email

All email communication to the instructor should be from a BSU-affiliated address. This policy ensures that senders can be correctly identified and protects your privacy. Email sent from other domains may not be answered.

You are expected to process your email at least daily.

Please do not use Canvas’ Inbox system to contact me. I recognize that some faculty have the opposite recommendation, and I apologize for the confusion this may cause. Canvas’ Inbox is a proprietary and inferior email substitute; we will instead rely on the real thing, which is a robust and open standard.

Don’t email pictures of text (for example, a screenshot of your IDE) nor embed these in assignments unless specifically instructed to do so. Digital images lack all of the affordances of

plain text. If you have a question about code, either include the code itself in the body of an email or include a link to the code on GitHub. These are the right tools for the job. In a course that is all about using the right tools for the job, there's no reason to do it any other way.

Telephone

Although my office telephone number is listed on my Web site, it is not a reliable way to reach me. Email or online meetings are strongly preferred. I will respond to every student email I receive; I may not respond to telephone messages.

Miscellany

YouTube and Advertisements

I have a large collection of tutorial videos on YouTube. When I started posting content there, non-commercial content like mine would show ad-free. Several years ago, YouTube changed their licensing scheme so that they could run ads on any video, not just for monetized channels. I apologize that the video tutorials now include ads. However, bandwidth is not free, and ads are the cost of keeping the content available to you and others outside the confines of our semester together. Should I receive enough views to be eligible for YouTube's partner program, such that I would earn a portion of the advertising revenue myself, I intend to donate that money to local charities.

Academic Integrity

Students and faculty are bound by the Student Academic Ethics Policy of the Code of Student Rights and Responsibilities. This means, among other things, that putting your name onto work and submitting it indicates that this is your original work. Any exceptions are to be clearly identified, such as through citation or in the README file that accompanies project work.

Intellectual Property

It behooves you to be familiar with fundamentals of copyright law and the university's intellectual property policies for student-created work. We will talk about some of these ideas during the semester.

The Writing Center

Want extra feedback on your papers? The Writing Center is a community of Ball State students and faculty who value writing. Come and collaborate with one of our trained peer tutors on any project for any major. The Writing Center is a comfortable, supportive environment for writers

from all communities and backgrounds. It is located in Robert Bell 295. To make an appointment, go to https://ballstate.mywconline.com.

Disability Statement

If you need course adaptations or accommodations because of a disability, please contact me as soon as possible. Ball State's Disability Services office coordinates services for students with disabilities; documentation of a disability needs to be on file in that office before any accommodations can be provided. Disability Services can be contacted at 765-285-5293 or dsd@bsu.edu.

University Statement

A university policy mandates the inclusion of this statement:

> We are committed to ensuring that all members of the community are welcome, through valuing the various experiences and worldviews represented at Ball State and among those we serve. We promote a culture of respect and civil discourse.

It is an exercise in critical thinking to consider whether this is bunkum. Feel free to bring it up in class for discussion.

CS222 Course Plans (Fall 2026) © 2026 by Paul Gestwicki is licensed under CC BY-SA 4.0