

CSE 20289 Systems Programming



Fall 2025

Instructor Collin McMillan <cmc@nd.edu>
Office 352 Fitzpatrick

Physical Location TBD
Lecture Time TBD

Cyberspace Locations
office hours and chat <https://discord.gg/tbd>
grades <https://canvas.nd.edu/courses/tbd>
homework directory [/escnfs/courses/fa25-cse-20289.01/](https://escnfs/courses/fa25-cse-20289.01/)
[/escnfs/home/your-netid/esc-courses/fa25-cse-20289.01/](https://escnfs/home/your-netid/esc-courses/fa25-cse-20289.01/)

Description

Think of this class as “how to use a computer, from a computer scientist’s perspective.” We will cover command line utilities, files, processes, Unix/C programming, shell and Python, and more!

Grading / Assignments

Your grade will come from the following. **All due dates are at the start of class** unless otherwise stated.

Description	Count	Points Each	Total Points
Shell Dailies	~7	~1-5	~20
Python Dailies	~8	~3-5	~30
C Dailies	~14	~3-5	~50
Shell Milestone			30
Python Milestone			50
C Milestone			80
Shell Exam			50
Python Exam			50
Comprehensive Exam (~70% C)			100

Total Points = **500**

Letter Assignments

100% $\geq$ <i>grade</i> $\geq$ 94%	A	80% $>$ <i>grade</i> $\geq$ 77%	C+
94% $>$ <i>grade</i> $\geq$ 90%	A-	77% $>$ <i>grade</i> $\geq$ 73%	C
90% $>$ <i>grade</i> $\geq$ 87%	B+	73% $>$ <i>grade</i> $\geq$ 70%	C-
87% $>$ <i>grade</i> $\geq$ 83%	B	70% $>$ <i>grade</i> $\geq$ 60%	D
83% $>$ <i>grade</i> $\geq$ 80%	B-	60% $>$ <i>grade</i>	F

Late Policy

Late assignments will not be accepted unless: 1) you make prior arrangements with me, or 2) you have a University-approved reason.

Honor Code

Notre Dame students are expected to abide by Academic Code of Honor Pledge. "As a member of the Notre Dame community, I will not participate in or tolerate academic dishonesty."

The "Pencils Down" Rule: You may communicate about homework with your classmates at a high-level, give general advice, and chat about common problems. You may **not** communicate specific knowledge **such as source code** or planning documents. A good litmus test is *if you would need to write it down to communicate or remember something, it is off limits*. Seek homework help from the Professor or the TA instead. We would love to help!

I encourage you to find a study group for the class, where you can discuss challenges at a high level, but remember that every grade is **individual**. You *may not* share source code or any written (paper, electronic, or whiteboard) information other than what is handed out in class. Minimum cheating penalty is 25% of total class grade.

Remote Policy

This class is in-person. It is not remote or hybrid. However, life happens and I am a reasonable person. If you are sick, please do not come to class or office hours and spread viruses. Please do me the courtesy of sending me a Discord message to let me know you will be joining remotely.

Assignment and Grading Structure

Your grade comes from three sources: small practice assignments given most lecture periods called "dailies", large programming assignments called "milestones", and exams. Dailies are only worth a few points each and are a completion grade. I design the dailies to take a couple hours max (usually much less than that). If you find yourself taking much more than that or making a very complex solution it means you are probably on the wrong track – time to get in touch. The dailies are a way to verify your understanding of the course material and practice a little bit at a time.

Milestones are longer assignments that build on the dailies. There are three but each one had multiple parts. As you work on the dailies you are practicing skills you will accumulate into each part of a milestone. Then the milestones are made from those parts.

Then there are three exams. These will be in-class, timed assessments of your learning. I expect to have a programming component where you will use a computer.

Craftsmanship Bonus

In this class I want you to learn *craftsmanship*. By “craftsmanship” I mean taking pride in your work and striving to do good work for its own sake. It involves both moral and intellectual excellence. Moral excellence because you must act with integrity and patience even when no one is looking. Intellectual excellence because you must have attention to detail and endurance to put polish on your work even when it seems “done enough” already.

So! To encourage this virtue the milestones will have a **craftsmanship bonus**. I will grade each milestone like normal up to 95% of its value. To earn the extra 5% you must apply for the craftsmanship bonus. There are not strict requirements for the bonus – you instead make a case on your own for some polish you put on your assignment and why you are especially proud of it, above and beyond the written requirements. These could include:

- Especially clean code organization
- Appropriate commenting
- Clean reports from memory check tools or other debuggers as appropriate
- Extras such as code coverage reports, speed tests, or comparisons to default solutions
- Alternative solutions you wrote that work but are inferior to your main solution

To apply for the bonus just put a file called CRAFTSMAN.TXT, all caps just like that, in your homework turn-in. In the file describe what you did as a bonus, why you are proud of it, what you did extra, etc. Make your case.

But beware! If you apply for the bonus and don’t really do anything extra, you can **lose** that 5%. That’s right, you can end up with a 90% on the milestone if you turn in a giant spaghetti code or other clearly inferior product and try to pass it off as especially clean. I will have an open mind on your ideas and will give you the benefit of any doubt, so don’t worry. My goal is to encourage great work.

Types of Errors

There are four types of errors, in increasing level of severity:

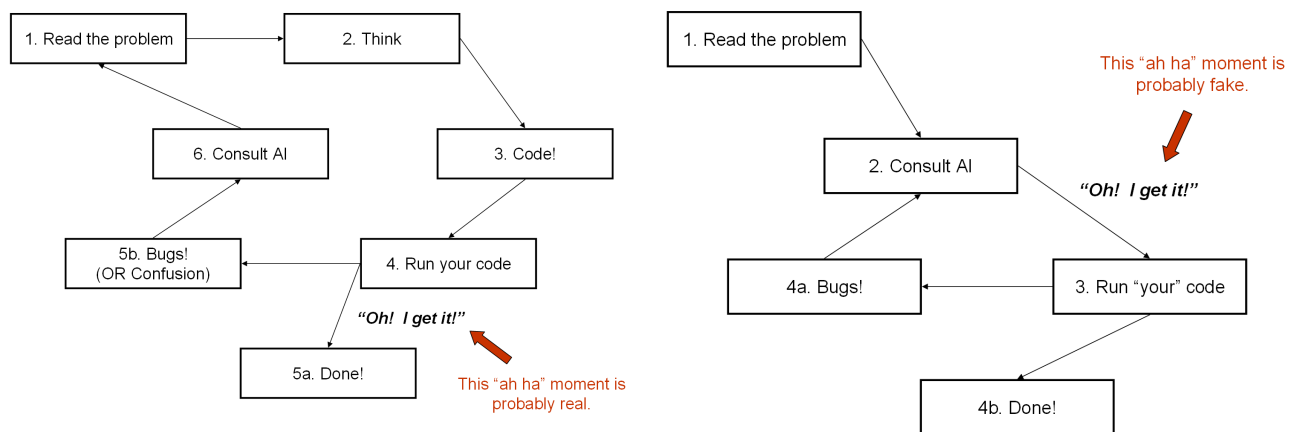
- Technical Errors. These occur when you make a mistake as a result of normal human failure. This could be forgetting something, mistyping/misspelling something, misunderstanding a relevant concept, or other honest mistake that introduced a bug in the code.
- Quasi-normative Errors. These occur when you violate local conventions or practices. For example if your organization uses a particular style coding and you insist on using a different style. Your style might not be wrong exactly, but it is not the local norm.
- Normative Errors. These occur when you violate widely-accepted best practices. For example if you put all your code logic in one giant main() function. Your solution may work but is still wrong to most people. You need a very good reason to violate an accepted best practice.
- Moral Errors. These occur when you act unethically. For example stealing credit for someone else’s work, releasing code with known bugs that you do not disclose, and laziness.

Craftsmanship is about avoiding all four types of error. It means taking pride in your work.

AI Use

You are allowed to use AI on the homeworks but you need to use it in the right way. The reality is that AI can be a great partner in learning and an amazing productivity booster in practice. But it can also hold you back – imagine an overly “helpful” family member who pays all your bills, does all your laundry, checks all your homework, sets up your social appointments, steps in and fixes disputes between you and your friends, picks out your clothes, and cooks all your meals. Hope you like feeling inadequate! On the other hand, imagine if you go out on your own, struggle and try, have problems and need advice. A truly helpful parent will give you good counsel without taking over your life.

Now think about AI. A common antipattern happens by consulting AI too soon. AI “helpfully” gives you an answer. It seems to make sense and you *think* you learned it. But it’s a false feeling. You didn’t really understand it. A better pattern is to try it on your own and consult AI when you run into problems. Then it can guide you back on track without it becoming your boss.



Two things to keep in mind: First, you will eventually develop a different process of using AI in your future job. Once you have a deep understanding of the problem, you can rely a bit more on AI to do things for you up front. But when you’re learning, you need to let it let you try.

And second, if you find yourself in this antipattern and wonder if you are having false “ah ha” moments, no shame! Get in touch! The TAs and I can help get you back on track. The whole point of the homeworks is for you to practice and learn and find weak points in your understanding where I can help you.