

CSCI 209 → CSCI 2100: Software Development

Sara Sprenkle

`sprenkles@wlu.edu`

<https://cs.wlu.edu/~sprenkles/cs2100>

<https://pollev.com/sprenkle>



Sept 4, 2026

How did you get to CSCI2100: Software Development?

- CSCI111 and CSCI112:
 - Recently renamed to Introduction to Computer Science and Data Structures, respectively
 - Also, renumbered!
 - Formerly known as,
Fundamentals of *Programming* I and II

What is Programming?

“If you don't think carefully, you might think that programming is just typing statements in a programming language.”

-- Ward Cunningham

“Any fool can write code that a computer can understand. Good programmers write code that *humans* can understand.”

-- Martin Fowler

“Refactoring: Improving the Design of Existing Code”

Discussion: What Is *Good* Software?

- What are its outcomes?
- What are the characteristics of the software?
- How can we write good software?
- What are short-term vs long-term goals?



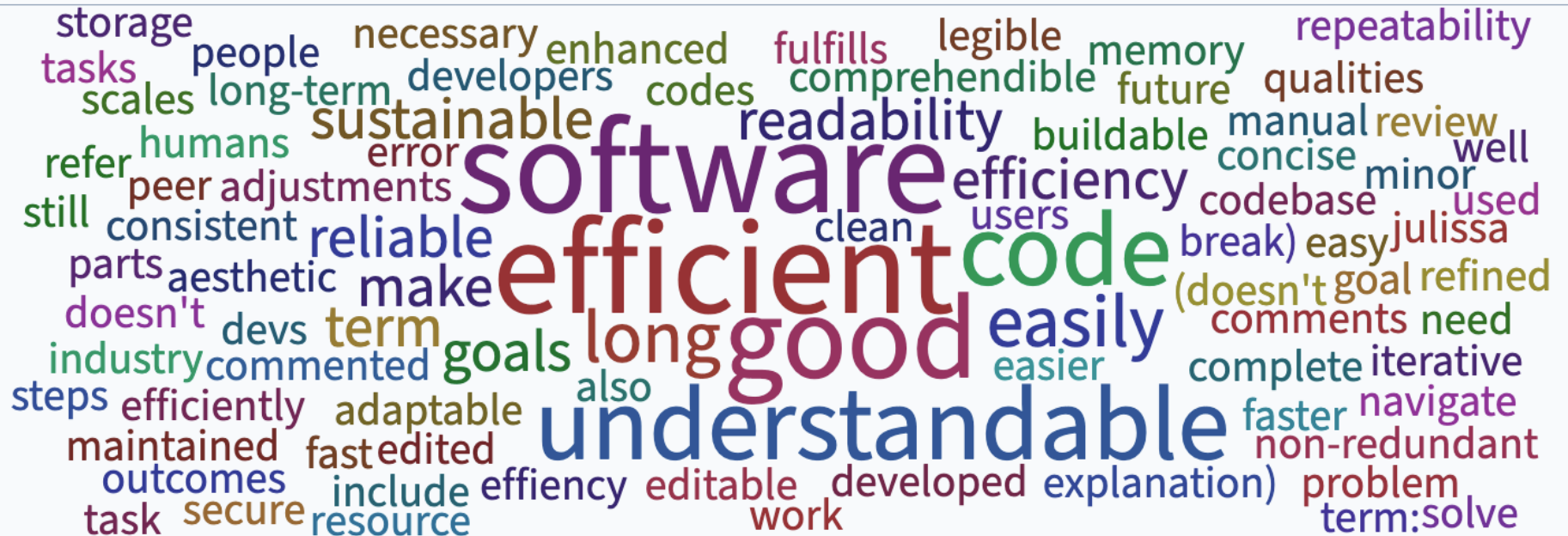
Start the presentation to see live content. For screen share software, share the entire screen. Get help at pollev.com/app



Start the presentation to see live content. For screen share software, share the entire screen. Get help at pollev.com/app

Section 1's Cloud

What are qualities of good software? What are its outcomes? What are the characteristics of the software? How can we write good software? What are short-term vs long-term goals?



Section 2's Cloud

What are qualities of good software? What are its outcomes? What are the characteristics of the software? How can we write good software? What are short-term vs long-term goals?



Characteristics of *Good* Software?

- Correct, free of bugs, does what is expected
 - Robust, reliability, stability
- Code is easy to read, extend, maintain
 - Readability, extensibility, maintainability
- Usability: Application is easy to use
- Efficiency
- Scalability

➔ Metrics of quality
referred to as the **ilities*

My Research Interests

- Broadly: Software engineering
- Automated testing of web applications
 - Develop algorithms
 - Implement in tools
 - Empirical studies
 - Try ideas out, see what actually happens, analyze
- Subject applications: [Ancient Graffiti Project](#), [ChemTutor](#)
- CSCI335→CSCI3404: Software Engineering via Web Applications

My Research Interests: Automatically Finding Bugs

- In an email:

Review Your Recent Purchases

o

write a review

a

write a review

u

write a review

My Research Interests: Automatically Finding Bugs

From Kroger <noreply@kroger.com> ☆
Subject Your null Comments Have Been Received
To Sara Sprenkle ★

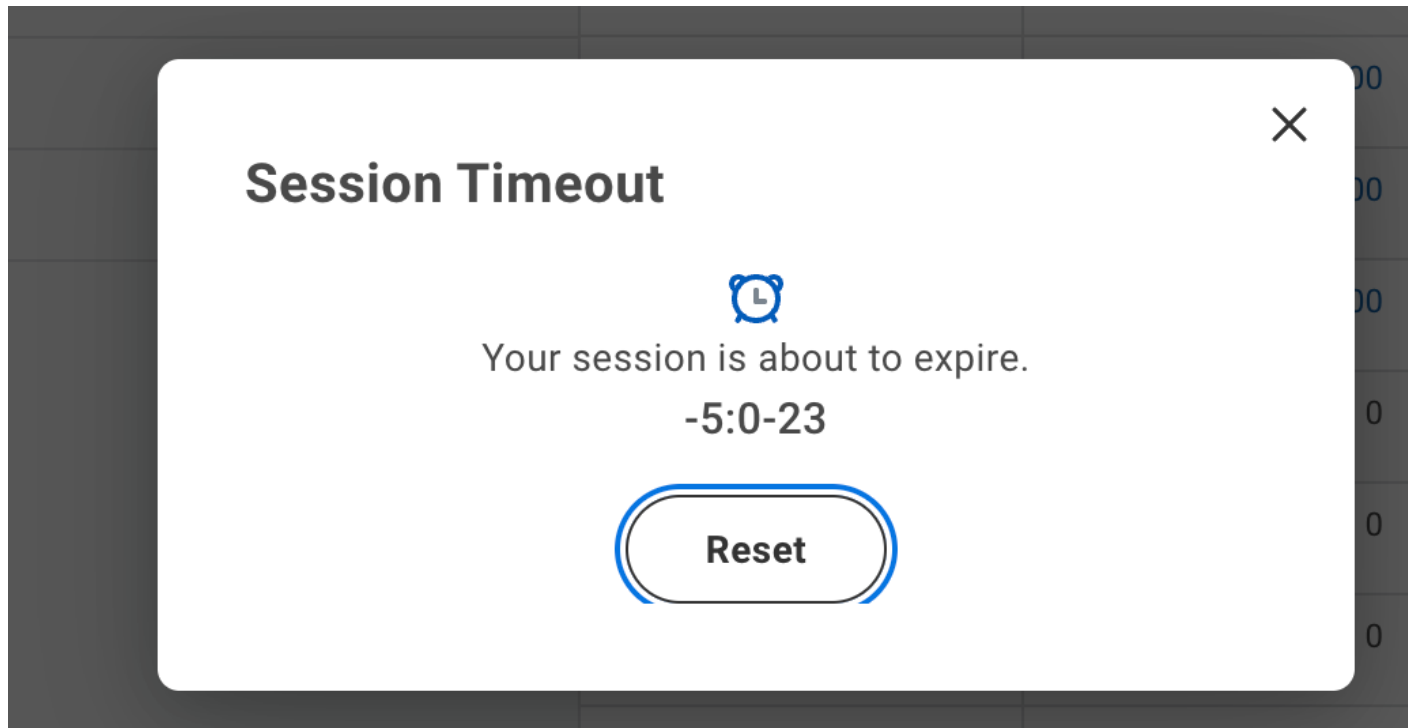
Sometimes, Errors are More Obvious

Our online textbook

```
----- Traceback -----  
Traceback (most recent call last):  
File "/home/bmiller/Runestone/web2py/gluon/main.py", line 444, in wsgibase  
    serve_controller(request, response, session)  
File "/home/bmiller/Runestone/web2py/gluon/main.py", line 179, in serve_controller  
    page = run_controller_in(request.controller, request.function, environment)  
File "/home/bmiller/Runestone/web2py/gluon/compileapp.py", line 657, in run_controller_in  
    ccode = getcfs(layer, filename, lambda: compile2(code, filename))  
File "/home/bmiller/Runestone/web2py/gluon/cfs.py", line 50, in getcfs  
    data = filter()  
File "/home/bmiller/Runestone/web2py/gluon/compileapp.py", line 657, in <lambda>  
    ccode = getcfs(layer, filename, lambda: compile2(code, filename))  
File "/home/bmiller/Runestone/web2py/gluon/restricted.py", line 205, in compile2  
    return compile(code, layer, 'exec')  
File "<fstring>", line 1  
    (searchdict=)  
      ^  
SyntaxError: invalid syntax
```

Update: “Fixed! Sorry for the inconvenience. This slipped through our automated tests and my manual testing. It is a feature that is not supported by python3.7 but is supported by python 3.8 and 3.9. Time to get all of Runestone’s servers off of 3.7!!”

Workday!



Course Content

- Software Design Principles
- Java programming language
 - **Statically** typed, compiled
- Software development, productivity tools
 - Version Control Systems (git)
 - Eclipse

What to Expect from this Class

- Programming intensive
 - Variety of assignments and projects
 - More freedom in design, *ilities
 - Larger portion of your grade
 - Correctness is **NOT** enough
 - Building on large library of classes
 - Read others' code! Learn from the good and the bad
 - Building larger applications
- Compare/Contrast with Python
 - PL design; what's the best PL for your needs
- Learning on your own
 - Online resources

Learning Objectives

- Discuss software development and practices **knowledgably**, using appropriate **terminology**
- Design, implement, test, and document efficient applications of **increasing size** and **complexity**
- Understand the designs and implementations of **others**
- Use a **version control system**
- Use many of the capabilities of the **Eclipse** IDE
- Test and debug large applications **systematically**, using standard tools
- Understand **design principles** such as DRY and shy
- Discuss the benefits and limitations of a **statically typed** language

Non-Syllabus Objectives

- Prepare you for the future: advanced-level classes, interviews, internships, jobs, ...
 - Process for solving larger problems in a sustainable way
 - Comfort/confidence to solve problems, communicate with collaborators, learn new things

I often hear from alumni a version of
“I didn’t realize how important [that] was when I took your
class.”

ALUMNI FEEDBACK

Feedback from an Alumnus

“I am schooling everyone at work on OO design and Java. Seriously, keep pounding OO design principles in. It is incredibly practical. I'm teaching CS majors and Computer Engineering grads about this. It's crazy how some (good) technical schools don't stress this more.”

Feedback from Another Alumnus

“CSCI 209 was super super helpful in my job application process. In all my interviews not only did I have to program in Java, but I also had to explain theories and design techniques used in the software development process.

Being able to explain those theories/techniques confidently and thoroughly like we had practiced during each of our class meetings made all the difference for me.”

Feedback from Yet Another Alumnus

“One advantage that I'm grateful for from W&L is that I've been able to make a lot of impact in the front-end and technical *writing spaces* of my projects.

I think the liberal-arts education paired with thinking about *design/practicing* writing documentation in your classes has given me a niche that's been really rewarding.”

One more, for good measure

“Since August 2022, I've been working at a startup software company that meets the various needs of pharmaceutical companies. I frequently take on new projects and new requirements, and our application has to be **flexible** enough to **easily inject new functionality**. Additionally, I built a strong understanding of the pharmaceutical business itself to ensure that our solutions align with the business end.

“Our backend is built in Java, and we use Eclipse and Github, so, in many ways, the development side feels like a scaled-up version of the Picasso Project. When I first started, I had a similar experience to that project: trying to make sense of an intimidating amount of existing code, figuring out where and how to inject new functionality, and learning the coding standards and practices already in place.”

OUR CLASS

Class Details

- Course Web Site <https://cs.wlu.edu/~sprenkles/cs2100>
 - Example code, slides, readings, resources, assignments
- One required online textbook
- Plentiful online resources
- Participation
 - Class discussions

Class Details

- Classes recorded via Zoom
 - Recordings posted on Canvas after class
- You can join class on Zoom (see Canvas), **but** it shouldn't be the common case
- Zoom Office Hours links in Canvas (TBD)
 - Email for appointments

Communication from me

- If I'm not in the office: often, short responses from my phone
 - Please excuse the brevity/informality
 - I prefer a short response to no response

Where Should I Go For...?

Course Web Site

- Slides
- Assignments
- Example code
- Readings
- Resources

Canvas

- Class recordings
- Some assignment submissions
- Zoom office hours links
- Zoom class links (e.g., when sort of sick)

Same info, different view

Where Should I Go For...?

I am looking for...	Find it ...
Slides	Course Web Site
Assignments	
Example code	
Readings	
Resources	
Class Recordings	Canvas
Some assignment submissions	
Zoom links for office hours, class	

Class Details

- Programming Assignments
 - Hands-on learning
 - Various sizes
 - To start, a lot of short ones
- 1 Testing Project
- 2 Exams
- Team Final Project

Course Dynamics

- Professor's Responsibilities:

- Be **prepared** for class
- Make material clear
- **Challenge** and **encourage** students
- Provide constructive feedback to students
- Treat students with **respect**

- Student's Responsibilities

- Be **prepared** for class (do readings and homework)
- Give **attention** and **effort** in class to learning
- Ask questions (**during class** and via email)
- Use professor's office hours
- Let professor know if something is going wrong
- Treat other students and professor with **respect**

Changes from CS1 and CS2 to Software Development

- No lab with student assistants!
 - Need to work on your own
 - Recommendation: Schedule your own “lab time” to work
- Emphasis changes
 - Not just about correctness and efficiency but about maintainability and human readability
- Python → Java

Embedded Tutors!

Takunda Mawere '28



UJ Nwokoye '28



Embedded Tutors!

Takunda Mawere '28

- Computer Science and Math major
- Extracurriculars: Volunteer with Campus Kitchen and Habitat for Humanity.
- Favorite activities: play soccer, walk through Lexington, and draw
- CSCI209 Favorite: learning a new language
- Office Hours: TR 7-9 p.m. and by appointment on Penji



Embedded Tutors!



UJ Nwokoye '28

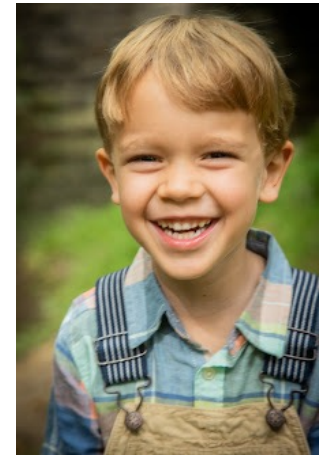
- Computer Science and Economics major
- Extracurriculars
 - Career Fellow
 - Library information desk
 - Gen Tech
- Favorite activities: Gaming, Pickleball!
- CSCI209 Favorite: Picasso!
 - GitHub merge conflicts
- Office Hours: TR 4:30-6:30 p.m.



August 2024

My Bio

- From Dallastown, PA
- Education:
B.S., Gettysburg College;
M.S., Duke University;
Ph.D., University of Delaware
- For fun: Xander, family



Sprenkle - CSCI2100

His First Self Haircut



August 2026

Sept 4, 2026

Sprenkle - CSCI2100

38

August 2023



Sept 4, 2026

Sprenkle - CSCI2100

39



October 2019

Sept 4, 2026



August 2020



September 2021



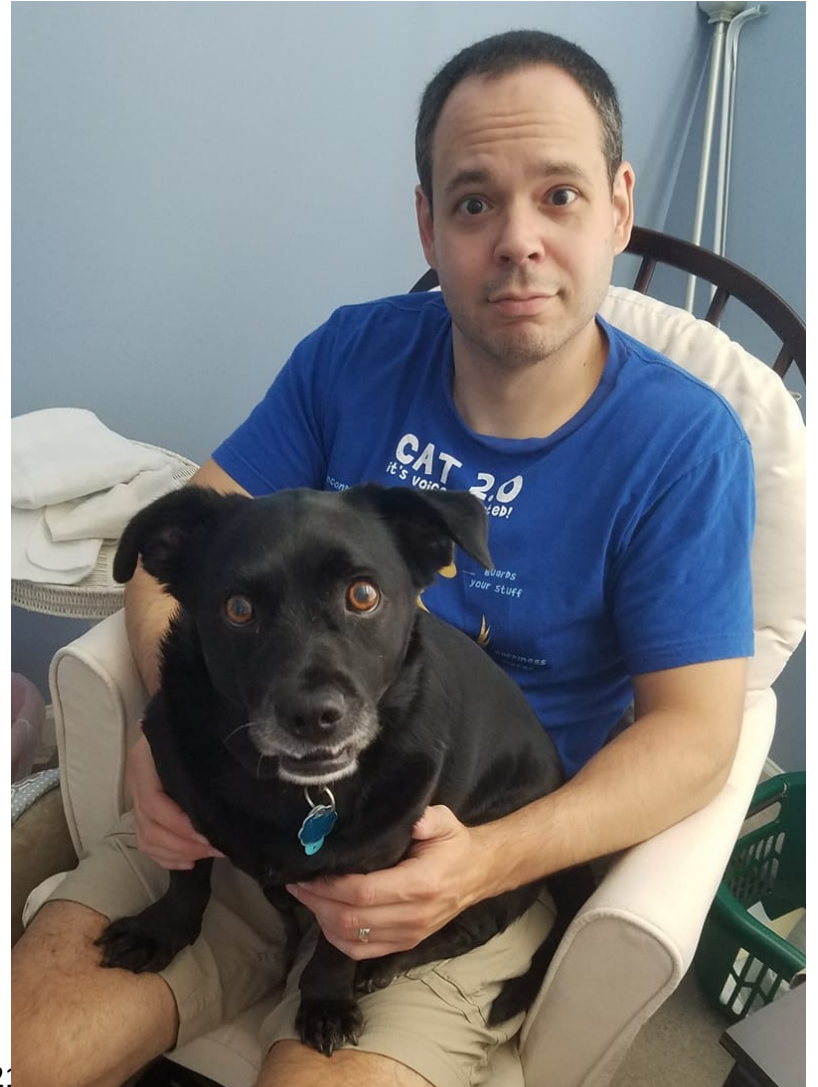
July 2022

2100

My Roommates



Sept 4, 2026



Sprenkle - CSC121

LEARNING HOW TO LEARN COMPUTER SCIENCE

The Learning Space

Not knowing
how to do
something



Being able to do
something
successfully

The Learning Space

Not knowing
how to do
something

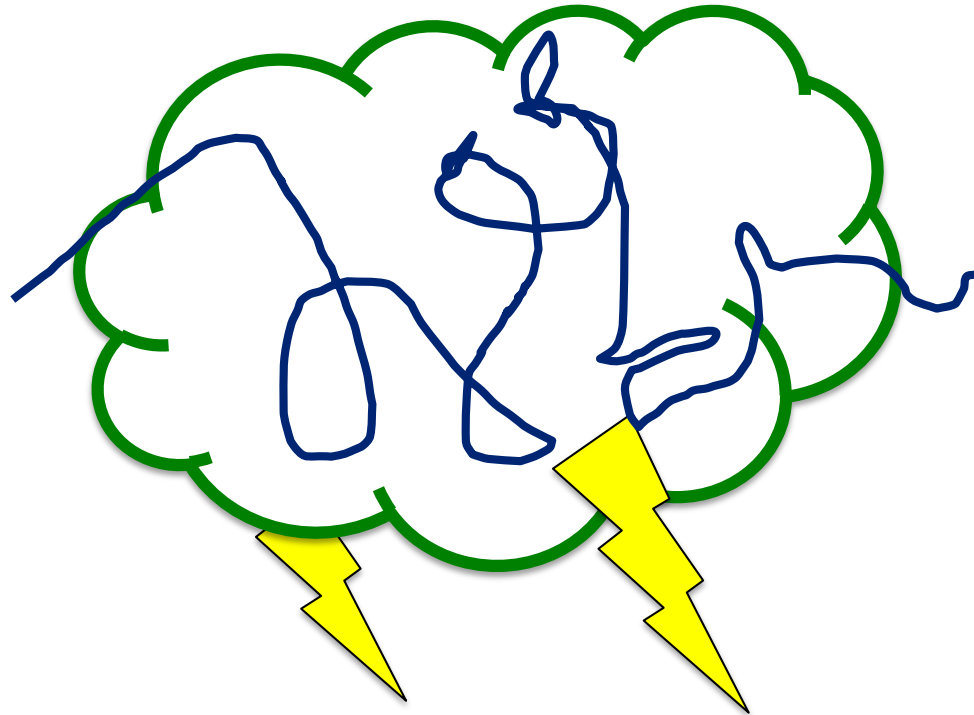


**The Learning
Space**

Being able to do
something
successfully

The Learning Space

Not knowing
how to do
something

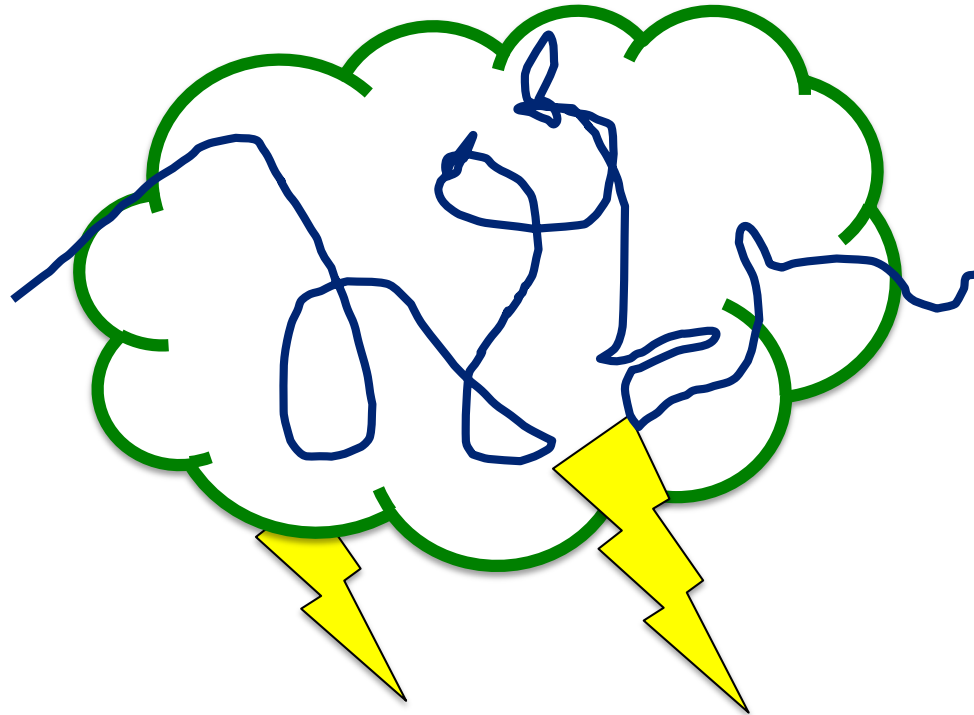


Being able to do
something
successfully

The path is not an easy, straight line.
Often frustration, uncertainty, and doubt in your abilities.

The Learning Space

Not knowing
how to do
something

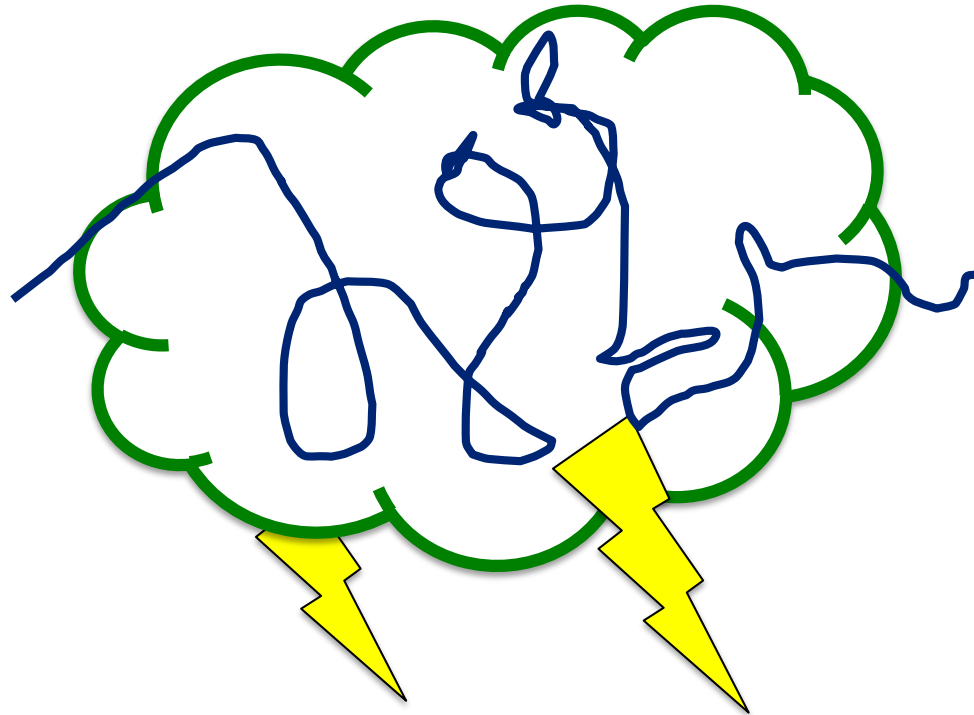


Being able to do
something
successfully

This process is where ***learning happens***
and ***resilience is built.***

The Learning Space

Not knowing
how to do
something

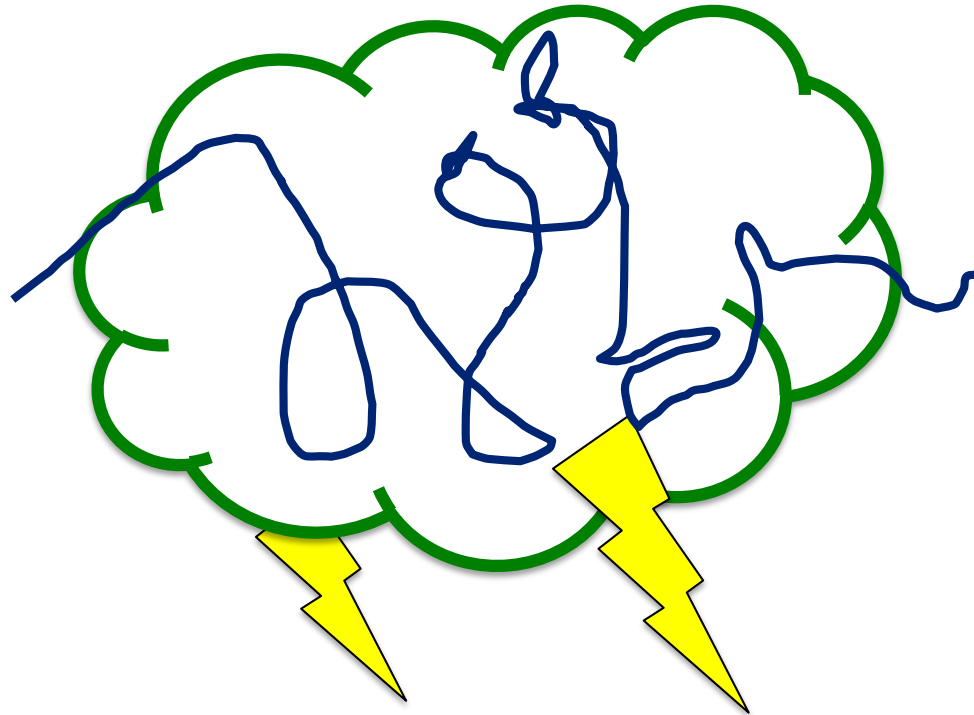


Being able to do
something
successfully

You may be incentivized to take short cuts (AI 🙄)
to reach the destination faster, but what did you **learn**?

The Learning Space

Not knowing
how to do
something



Being able to do
something
successfully

That's not to say that I don't want you to figure out productive ways to learn. I will guide you in that process.

The Learning Space

Not knowing
how to do
something



Being able to do
something
successfully

LET'S DO THIS!

Towards Being a Power User Assignment

- Goals

- 111/112 → 209/2100

- Comfort with Unix and the Command-Line

- Know your computer

- Iterative process:

- 1. Try to recall in your own mind

- 2. Confirm/check with neighbors, web

Collaborating at Pods

In parallel, I will go around
and meet everyone.

From the Lab Computers

- Automatically mirrored

From Your Computer

- USB-C
- HDMI

- Adjust the monitors, using TWO hands!
- Read the handout at the table
- Practice: Take turns projecting the lab machines and connecting your laptops and projecting
 - Check out the course web site and look at today's assignments on schedule page
 - *How will you find the web site?*
 - Start on tasks for Monday

Looking Ahead: Before Monday's Class

- Bookmark course schedule page
- Register for the online textbook
- Complete the “Towards Being a Power User” assignment
 - Due Monday before class
 - Expected to take less than 2 hours.
 - Make sure you're using resources well.
 - If more than two hours, stop and contact me.