CS 162
Intro to Computer Science II

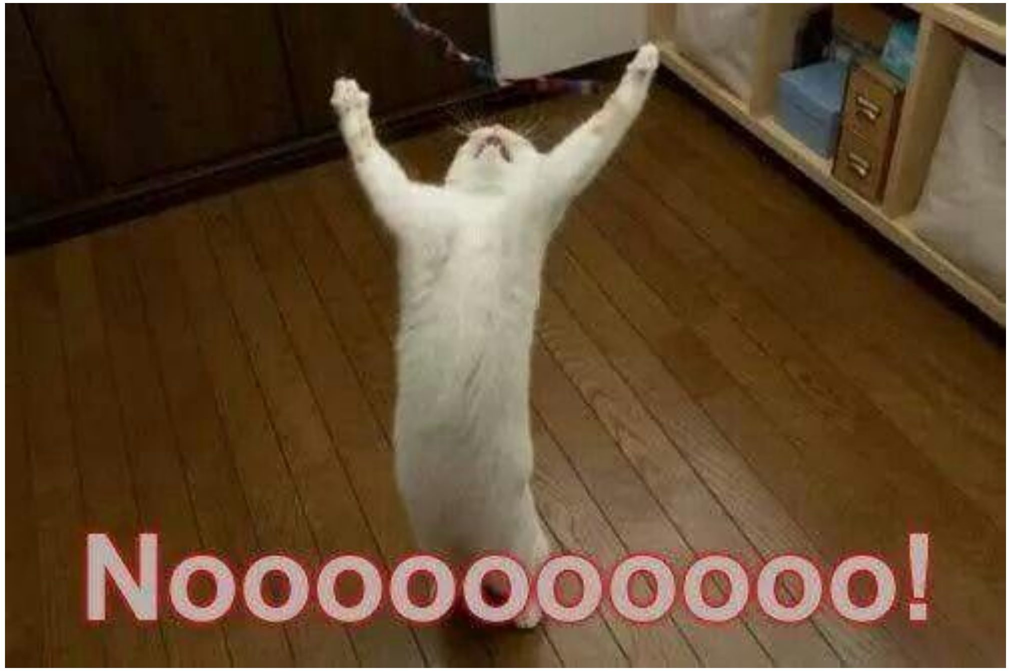
Lecture 0

Introduction and Course Syllabus

1/8/24

Oregon State University

Back to School



Course Description

- Goal:
  - provides an overview of the fundamental concepts of computer science.
  - You will study basic data structures, computer programming techniques and application of software engineering principles.
  - provides an introduction to analysis of programs.

- Topics:
  - Pointers
  - Memory model (stack vs. heap)
  - File Input/Output
  - Object-oriented programming principles
  - Program design, debugging and testing
  - Algorithm analysis
  - Recursion
  - Sorting and searching
  - Linear data structures
  - Debugging and testing

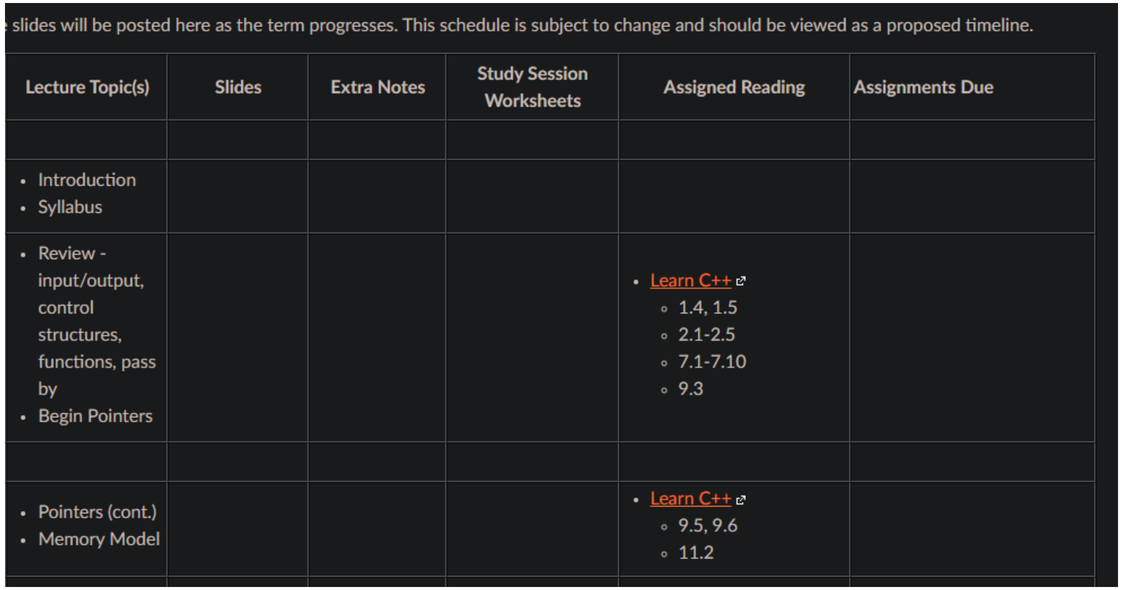
Course Learning Outcomes

- After taking this course, you would be able to …
  1. Design and implement programs that require:
     1. multiple classes and structures
     2. hierarchies of classes that use inheritance and polymorphism
     3. an understanding of abstraction, modularity and separation of concerns
  2. Construct and use basic linear structures (arrays, stacks, queues, and various linked lists) in programs, and be able to describe instances appropriate for their use.
  3. Classify moderately complicated algorithms in these complexity classes: O(1), O(log n), O(n), O(n log n), and O(n^2).
  4. Develop test-data sets and testing plans for programming projects.
  5. Produce recursive algorithms and choose appropriately between iterative and recursive algorithms.

What do people say about 162?

A SMOOTH SEA NEVER MADE A SKILLED SAILOR!

Course Structure



slides will be posted here as the term progresses. This schedule is subject to change and should be viewed as a proposed timeline.

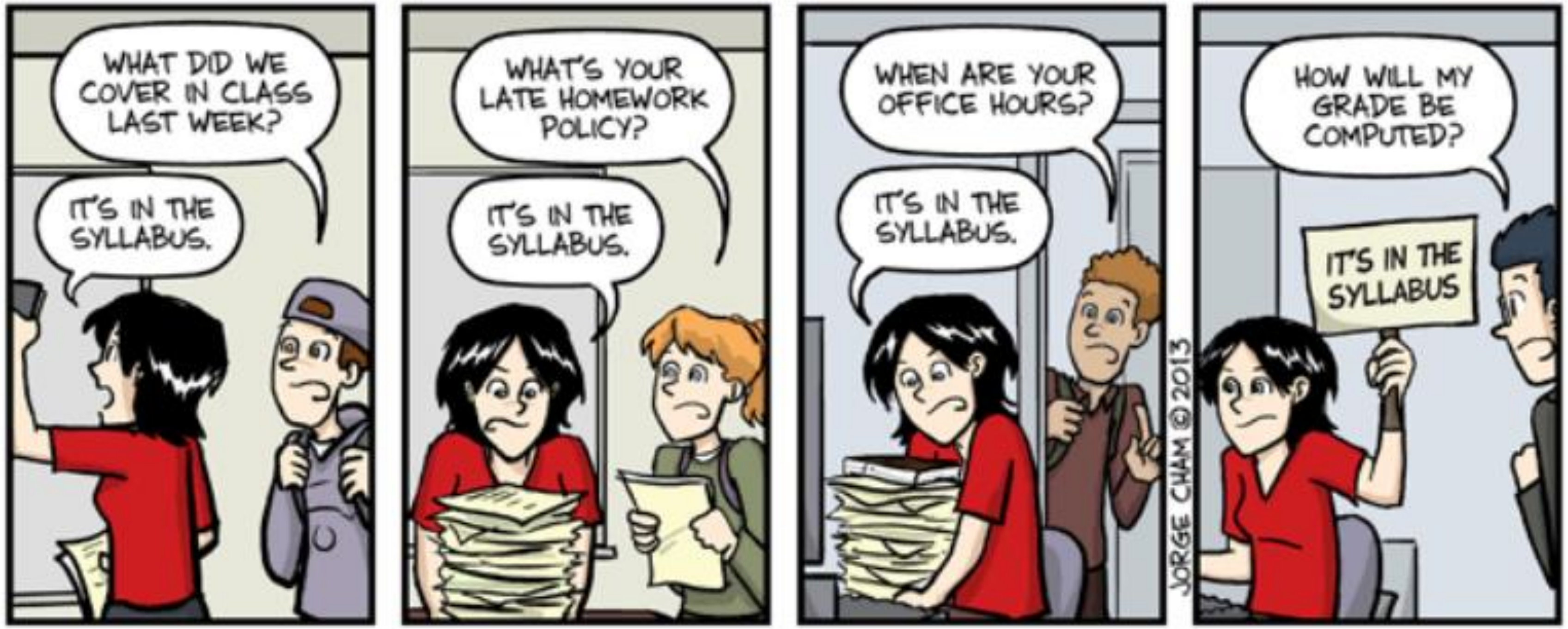
Lecture Topic(s)	Slides	Extra Notes	Study Session Worksheets	Assigned Reading	Assignments Due
• Introduction • Syllabus					
• Review - input/output, control structures, functions, pass by • Begin Pointers				• Learn C++ ∘ 1.4, 1.5 ∘ 2.1-2.5 ∘ 7.1-7.10 ∘ 9.3	
• Pointers (cont.) • Memory Model				• Learn C++ ∘ 9.5, 9.6 ∘ 11.2	

- 10 weeks schedule
  - Review (Week 1)
  - Pointers and Memory Model (Week 1-2)
  - 1D and 2D arrays (Week 2)
  - File I/O and File separation (Week 3)
  - Class composition (Week 4-5)
- After midterm
  - Inheritance and Polymorphism (Week 6-7)
  - Templates (Week 8)
  - Linked List, Complexity (Week 9)
  - Linear Data Structure and sorting (Week 10)
- Midterm: Monday of Week 6 (Feb 12th) during lecture time
- Final: Wednesday, Mar 20th at 12 pm

Textbook

- Not required
- Online resources:
    - (Learn CPP) Learn C++
    - (Wikibooks) Wikibooks C++ Programming
    - (Miller) An Introduction to the Imperative Part of C++
    - (Downey) How to think like a computer scientist

Syllabus


WHAT DID WE COVER IN CLASS LAST WEEK?
IT'S IN THE SYLLABUS.
WHAT'S YOUR LATE HOMEWORK POLICY?
IT'S IN THE SYLLABUS.
WHEN ARE YOUR OFFICE HOURS?
IT'S IN THE SYLLABUS.
JORGE CHAM ©2013
HOW WILL MY GRADE BE COMPUTED?
IT'S IN THE SYLLABUS

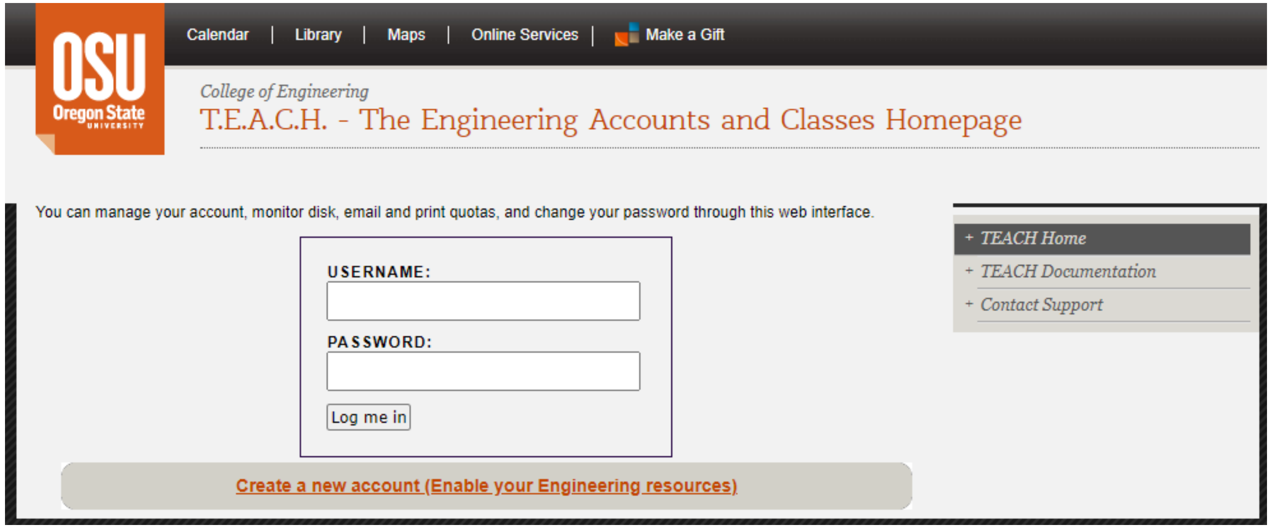
IT'S IN THE SYLLABUS

This message brought to you by every instructor that ever lived.

WWW.PHDCOMICS.COM

Course Information

- Canvas website:
  - All course material (lecture slides, labs, assignments, office hours, etc.)
  - Non-code submission
- TEACH:
  - Code submission



College of Engineering

T.E.A.C.H. - The Engineering Accounts and Classes Homepage

You can manage your account, monitor disk, email and print quotas, and change your password through this web interface.

USERNAME:

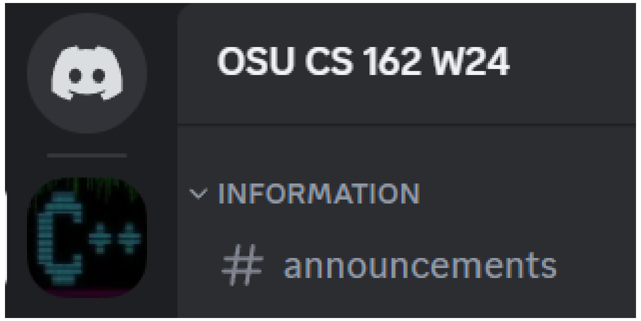
PASSWORD:

Log me in

Create a new account (Enable your Engineering resources)

- *TEACH Home*
- *TEACH Documentation*
- *Contact Support*

- Discord:
  - Online discussion forum



OSU CS 162 W24

INFORMATION

announcements

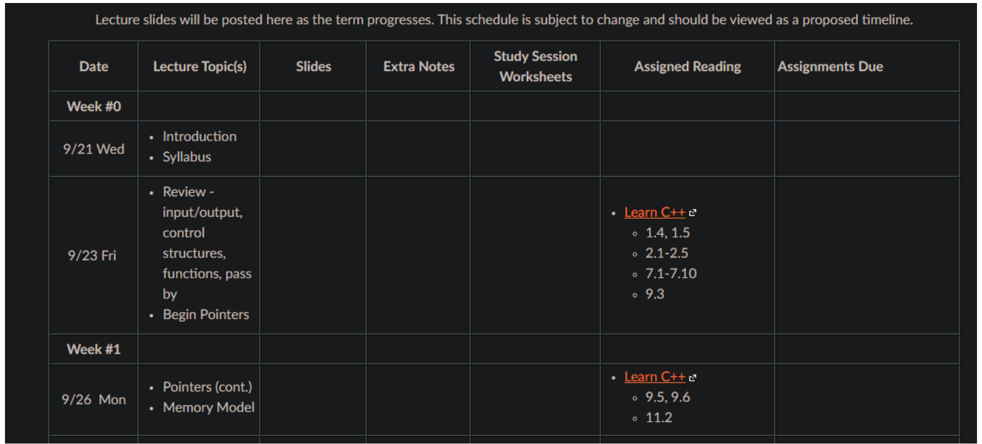
Basics

- Instructor: Yipeng (Roger) Song
  - I go by Roger ☺
- Email
  - Instructor: songyip@oregonstate.edu
  - TAs: cs162-ta@engr.orst.edu (TAs and me)
- Office Hours: TBD @ TBD
- Requirements: Laptop/PC
- Programming Language: C++

More Basics…

- Be respectful (Establishing a Positive Community)
- Have a growth mindset
  - Most abilities could be developed through dedication and hard work
  - "*The more you learn, the more you realize you don't know.*"
- Don’t cheat (0 tolerance!!)
  - https://studentlife.oregonstate.edu/studentconduct/student-info
  - *"...it is far more honorable to fail than to cheat."*
- Be Proactive
  - Take control and cause something to happen, rather than just adapt to a situation or wait for something to happen

Attendance



Lecture slides will be posted here as the term progresses. This schedule is subject to change and should be viewed as a proposed timeline.

Date	Lecture Topic(s)	Slides	Extra Notes	Study Session Worksheets	Assigned Reading	Assignments Due
Week #0						
9/21 Wed	• Introduction • Syllabus					
9/23 Fri	• Review - input/output, control structures, functions, pass by • Begin Pointers				• Learn C++ ◦ 1.4, 1.5 ◦ 2.1-2.5 ◦ 7.1-7.10 ◦ 9.3	
Week #1						
9/26 Mon	• Pointers (cont.) • Memory Model				• Learn C++ ◦ 9.5, 9.6 ◦ 11.2	

- **Lecture**: Strongly Encouraged
  - I will post lecture slides, demoed code, and additional resources on Canvas → Calendar
  - You are expected to be present during exam dates and registered lab times!!!
- **Labs**: Required
  - Labs start this week (Week 1) ☺
  - Lab 1 is posted on Canvas → Labs
  - Bring your laptops (Windows/Linux/MacOS)
- Missed labs result in a zero for that lab
  - Email TA mailer BEFORE the end of lab
  - Subject: "[CS162] Missing a Lab"
  - Lab you are missing
  - Excuse for missing lab
  - Plan for making up the lab

Lab assignments

Lab #1 -

Lab #2 -

Lab #3 -

Lab #4 -

Lab #5 -

Lab #6 -

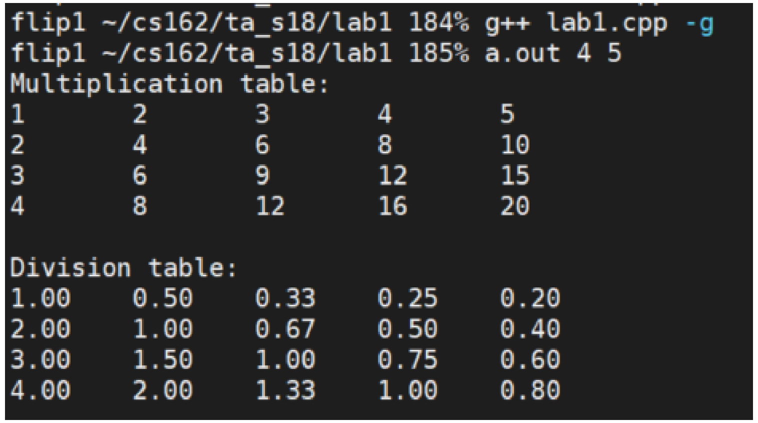
Lab #7 -

Lab #8 -

Grade Breakdown

- 10% - Labs
- 10% - Quizzes
- 35% - Assignments
- 15% - Design
- 30% - Exams
    - 15% - Midterm
    - 15% - Final

Labs – 10%



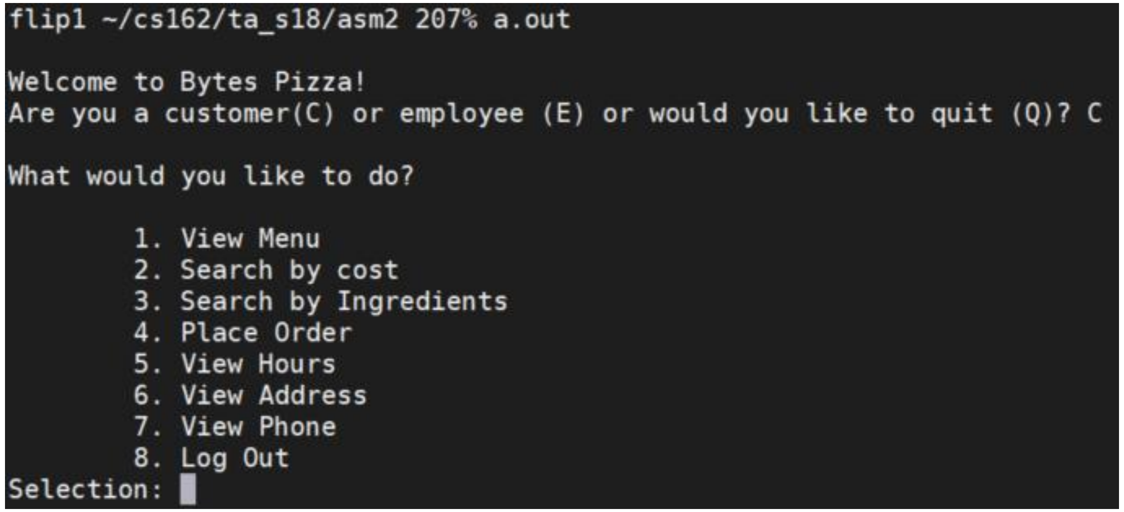
```
flip1 ~/cs162/ta_s18/lab1 184% g++ lab1.cpp -g
flip1 ~/cs162/ta_s18/lab1 185% a.out 4 5
Multiplication table:
1       2       3       4       5
2       4       6       8       10
3       6       9       12      15
4       8       12      16      20

Division table:
1.00    0.50    0.33    0.25    0.20
2.00    1.00    0.67    0.50    0.40
3.00    1.50    1.00    0.75    0.60
4.00    2.00    1.33    1.00    0.80
```

- 10 labs in total
- Each lab breakdown:
  - 1 hour worksheet (5 pts) + 2 hours hands-on practice (10 pts)
  - Lab materials will focus on understanding and implementing topics from class
- 15 pts per lab, effort + correctness, check off with TAs during lab to get points
  - If unable to finish, can make up additional 3 pts before the next lab (only for the hands-on portion)
  - 0-pt labs cannot be made up
  - Submit your lab work to TEACH for backup purposes

Bi-Weekly Quizzes – 10%

- Due every other Sunday midnight (5 in total, on Canvas)
- Quiz 1 is due this Sunday (1/14) midnight
- Available from: Fri 1 pm (after lecture) to Sun 11:59 pm
  - Canvas is very unforgiving about due times -- don't push it.
- T/F, and multiple choices, covering materials in that week
- 5 to 10 questions on each quiz, with a 60-minutes time limit
- 2 attempts for each quiz, keep the highest score

Assignments – 35%



```
flip1 ~/cs162/ta_s18/asm2 207% a.out

Welcome to Bytes Pizza!
Are you a customer(C) or employee (E) or would you like to quit (Q)? C

What would you like to do?

        1. View Menu
        2. Search by cost
        3. Search by Ingredients
        4. Place Order
        5. View Hours
        6. View Address
        7. View Phone
        8. Log Out
Selection:
```

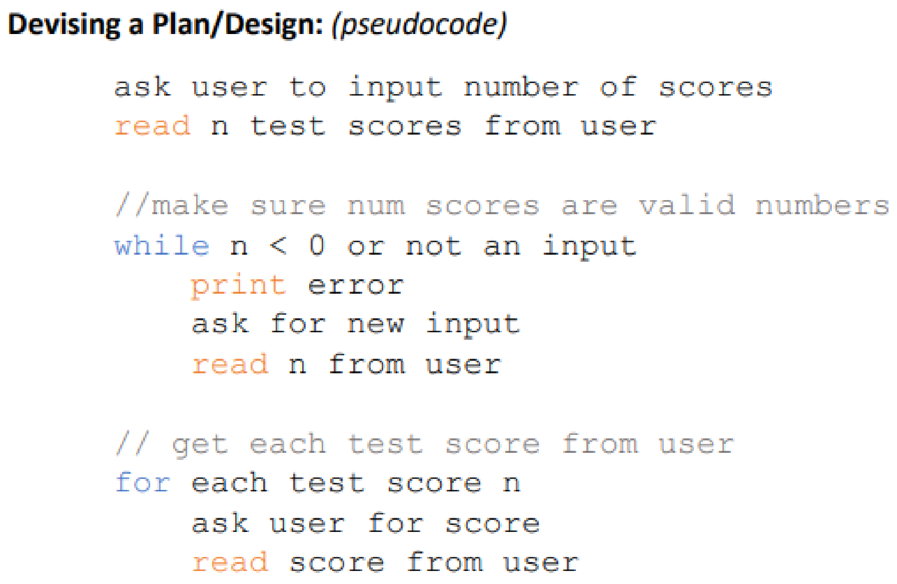
- 5 in the term, 100 pts each
- Two-week assignments
- **Due Sunday midnight**
- **All code must compile on ENGR server – otherwise 0 (implementation)**
- **All written work submitted as PDFs – otherwise 0**
- Late Policy (only for coding portion!!!)
    - 1 day late (< 24 hrs): 10% off (i.e., your score x 90%)
    - 2 days late (24-48 hrs): 30% off (i.e., your score x 70%)
    - 3 days or more (> 48 hrs): not accepted → 0
    - Begin with 3 grace days throughout the term (up to 2 grace days per assignment!)
        - Grace days can be used to remove/reduce late penalties (more info on the Canvas → Syllabus)

Design – 15%

- 5 designs in total, one per assignment
- Each design = Canvas exercise + design document
- Canvas exercise
  - Guided questions towards the assignment, quiz format, T/F, MC
  - No limit on attempts, as long as it is taken before the due date
  - These questions will help you:
  1. Understand what the assignment is asking for
  2. fully understand each requirement
  3. decompose the project into smaller parts
- Design document
  - Contains 3 parts:
  1. Understand the Problem
  2. Devising a Plan
  3. Testing

Design – 15%

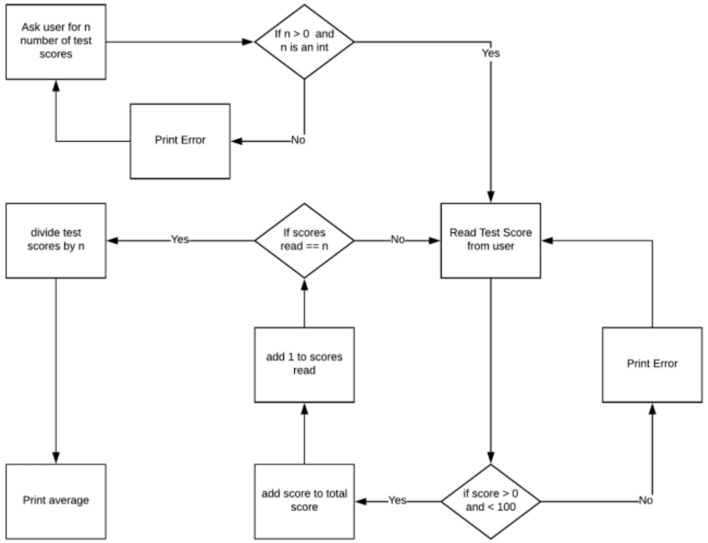
- Designs are due 1 week in advance of each programming assignment due date
- Late design are not accepted!!!
- See Canvas → Design Guide for additional details and expectations



Devising a Plan/Design: *(pseudocode)*

```
ask user to input number of scores
read n test scores from user

//make sure num scores are valid numbers
while n < 0 or not an input
    print error
    ask for new input
    read n from user

// get each test score from user
for each test score n
    ask user for score
    read score from user
```

Devising a Plan/Design: *(flowchart)*



Assignment Grading

- Assignment 1-4 are demoed (in person)
- Assignment 5 will be graded by the TAs on their own during final’s week
- Sign up for a demo for assignment 1-4 (TA Hours page)
- Demo within 2 weeks of the code due date, even if submitted late
    - Missing a demo, **-10 pts**
    - Demoing outside 2 weeks w/o permission, **-30%**
    - **Assignments without being demoed at the end of the term, 0 pts**

Look at the bi-weekly:

Mon	Tue	Wed	Thur	Fri	Sat	Sun
						1) Asm N Due
Lecture		Lecture		Lecture		1) Design N+1 Due 2) Quiz N+1 Due
Lecture		Lecture		Lecture Asm N Demo Due		1) Asm N+1 Due

Exams – 30%

- Exam I – Week 6 Monday (Feb. 12th) during lecture
- Exam II – Final’s Week Wednesday (Mar 20th) at 12pm

- Non-cumulative (but it builds on…)
- Same classroom

Grading Philosophy*

- A (93 or greater) mastery
- A- (90 – 92)
- B+ (87 – 89)
- B (83 – 86) stable/proficient
- B- (80 – 82)
- C+ (77 – 79)
- C (73 – 76) passable
- C- (70 – 72)

*Note: I do round ☺ (i.e. 89.45 → 89.5 → 90 ☺)

Extra Credit Opportunities

- In class exercise(s), randomly
- Some assignments/labs will have extra credit opportunities
- Exams may have extra credit questions as well

How to Be Successful

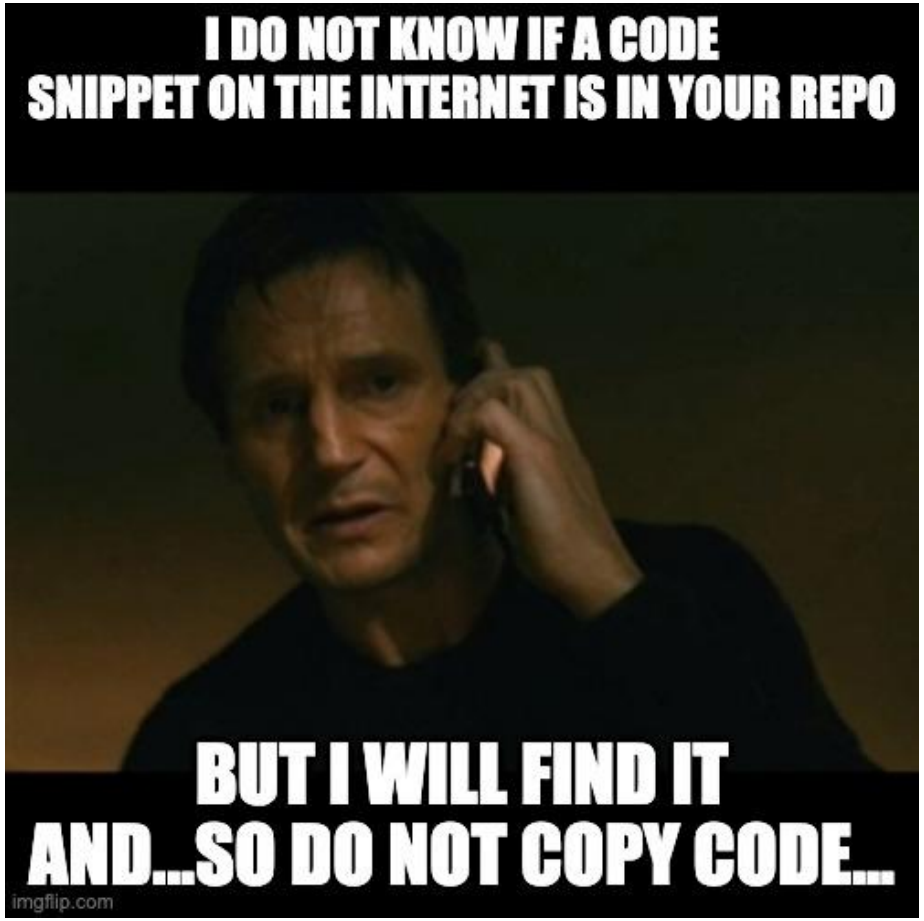
- Read and listen carefully
- Start assignments early
- Be proactive with absences and issues that arise in the term
- Get help when you need it
  - Make use of Discord and Office Hours

Lab and Assignment Rules

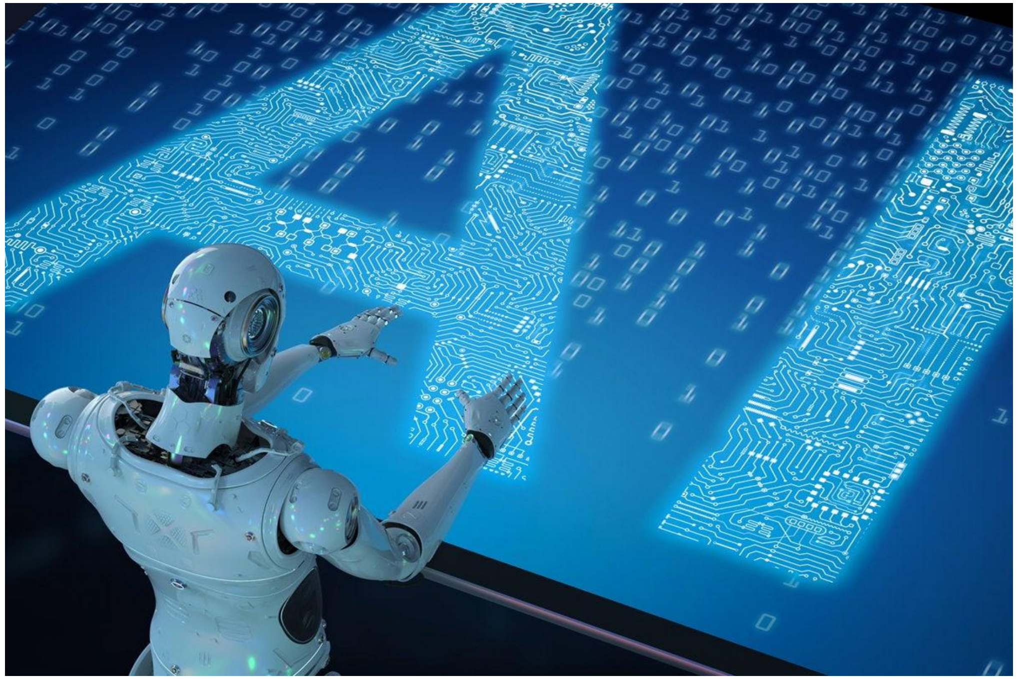
- DO NOT SHARE YOUR WORK OR CODE WITH OTHER STUDENTS
  - **You are encouraged to discuss with others** about the assignments but do not ask/give your work to the others
  - **Do not copy** other students' work or resources available (without citations) in online
  - **Do not publish** your work online

Lab and Assignment Rules

- Plagiarism will be punished via the Office of Student Life..
  - E.g., getting F or zero point for the lab/assignment that matters with plagiarism…
- Please refer the Code of Student Conduct


I DO NOT KNOW IF A CODE SNIPPET ON THE INTERNET IS IN YOUR REPO
BUT I WILL FIND IT
AND...SO DO NOT COPY CODE...
imgflip.com

AI Tool Usage in this class



- You must be the author of **all work**
- You may use AI to:
  - generate abstract ideas
  - polish or edit text you have drafted
  - quiz yourself
  - explain new or confusing concepts
  - generate code snippets to solve **unassigned example tasks**
- You may **NOT** use AI to
  - generate code snippets to solve a problem presented in a quiz, lab, assignment, or exam
  - draft a design document or the code implementation for an assignment
- If used, add a citation just like you would when you copy language or code from human authors.

Tips to the Labs/Assignments

- Study in a group (discussion is highly encouraged!)
  - But please **write code / do the labs individually**!
- Read the document thoroughly and follow the instructions
- Ask questions (Discord)
- Understand your time budget
  - **Plan ahead** to finish the labs/assignments on time
- Learn/refresh CS 161/ENGR 103 basic concepts (e.g., Linux/Unix commands, if/else, loops, functions, and arrays, etc.) ASAP

TAs

- Go see your TAs!!!
- Where: TBD
- When: Varies – check the TA Hours page on Canvas
    - No in-person office hours in week 1 unless otherwise noticed

Help Hierarchy

- Reread assignment, lecture slides, labs, syllabus
- Google/Bing/Open a textbook
- Ask a friend
- Check Discord for relevant posts or create a new question
- Ask a TA
  - You can attend office hours in person
  - TAs will also be monitoring Discord
- Ask Roger

To-dos before Wednesday’s lecture:

- Read the syllabus thoroughly
- Learn/refresh CS 161/ ENGR 103 basic concepts
  - Linux/Unix commands
  - If/else
  - Loops
  - Functions
  - 1D Arrays
  - Pass by value vs. Pass by reference
- We will review some on Wednesday ☺