

Course website: <https://jhu-ip.github.io/cs220-s22>

My info: David Hovemeyer
daveho@cs.jhu.edu
Malone 240A

Zoom

601.220 Intermediate Programming

Homework 0
due Friday 2/4

Introduction to the course

Ugrad account

GitHub account
→ google form

Outline

- Overview of course
- Signing up for Piazza
- CS account

One Course, Multiple Sections

- Multiple sections taught in parallel
- Course materials (exercises, homeworks, etc.) usually the same
- Check course site/syllabus for your section details (time/location/instructor)

Course goal

- By end of course, you'll be able to create **large***, complex, correct programs in **C and C++**
 - For some students, this comes easily; for others, not so much
 - Differences in background play a large part
 - If you're struggling, don't panic We're here to help.
 - To become a strong programmer, you need to practice, practice, practice

Content Delivery and Course Format

"flipped"

- In person class meetings as scheduled
- Pre-recorded videos, slides, recap questions, and exercises will be posted
- **Must watch** videos before attending the class (recap questions recommended too)
- Class sessions on MWF:
 - We review solutions for previous session's exercises, review main concepts from the assigned materials, and go over recap questions. We also answer your questions, discuss homeworks & projects, and finally, work on the new exercise.
 - recordings of the review parts of lectures will be posted on the course website afterwards

Programming is more than coding

Construction
design

- In introductory courses, you learn to write code
- You also spend a good deal of time debugging it
- In fact, the larger your programs get, the larger the percentage of time you'll spend debugging it
- But debugging isn't really (always) fun - how can we avoid it?
- Short-term lazy vs. long-term lazy
 - How far in the future are you thinking when you consider the consequences of your actions?
 - In some cases, extra work up front can reduce total time spent
- Sometimes its difficult to see the benefits in a single short homework assignment, but real commercial software is developed over years by large teams of people
- We aim to help you build skills that will allow you to contribute on large-scale projects

Building skills

- This course is primarily intended to help you build skills (rather than just increase knowledge)
- Building skills takes practice, and meaningful practice takes time
- Please ask for help when you need it!

Grade calculation

- Handwritten homework assignments (done individually) - 6%
- Coding homework assignments (done individually) - 30%
- Midterm coding project (in teams) - 14% C
- Midterm exam (date TBD) - 17%
- Final coding project (in teams) - 16% C++
- Final exam (TBD) - 17%
- Participation - 0% (strongly recommended to fully participate in class sessions)
- In class exercises - 0% (strongly recommended to complete them all)

Advice about coding homework

- A significant chunk of your grade (30%) is individual coding assignments
- These form an essential part of the learning experience
- Take these seriously!
 - Start early, ask questions early
 - Make steady progress
 - Strive to create robust, understandable, and elegant code
 - Do *not* share code or copy code we will report violations to the student conduct office
- If you don't take these assignments seriously, you are unlikely to have a good experience in the course

Advice about in-class exercises

- Throughout the semester we will work on exercises during class sessions
- These don't count towards your grade directly
- But they are *very* important for mastering course topics!
- Recommendations:
 - Complete all of these
 - If you do not finish them in class, finish on your own outside of class
 - Submit to **gradescope** for autograder feedback
 - Past students have repeatedly reported that finishing exercises has saved time when completing the homeworks and projects!
- We generally *won't* post solutions to the exercises
 - Completing these on your own is far more valuable than just looking at our solution
 - If you need help, ask for it in class, in office hours, or on Piazza

Course resources

- **Gradescope:** where you'll submit homework and receive grades
 - You'll receive an invitation to Gradescope site via email later this week
- **Piazza:** [see Blackboard and Syllabus for link](#)
 - We'll use Piazza as our primary form of course communication; you're expected to check it regularly!
 - Please ask questions using Piazza, rather than sending us email
 - Can make posts which are anonymous to other students
 - Can make posts which are targeted to Instructors (includes instructors and CAs) only, or just to the instructor of your section
 - Please read the post on posting guidelines
 - Sign up for Piazza right now!

CS account

- You will need a CS account for this class
- Obtain a ugrad cs account
 - if you have one already, then use that
 - if you are a cs major/minor, get a “permanent one” from CS IT (https://support.cs.jhu.edu/wiki/Obtaining_CS_Computer_Accounts)
 - otherwise, send the “instructors” a *private* post on Piazza with subject “Request for a temp cs account”
 - will get back to you as soon as possible with a user and password