

Welcome!

COMS W3136

Essential Data Structures in C/C++

Fall 2024

Teaching staff

- 2-3 Teaching Assistants (TAs)
 - Lucas Lugones
 - lnl2116@columbia.edu
 - Tiffany Fu
 - tlf2131@columbia.edu

Teaching staff

- TA email & office hours
 - Email to cucs3136-tas@googlegroups.com goes to all teaching staff
 - TA room – 1st floor, Mudd building
 - TA calendar: <http://bit.ly/3136-cal> (will be filled by this weekend)
- Instructor email & office hours
 - Tim Paine paine@cs.columbia.edu
 - Office Hours – 4:30-5:30PM T/R 480 CSB

Prerequisites and course objective

- Prerequisites
 - One semester of introductory programming course
 - Ex) 1006 (Python), 1004 (Java), or equivalent
 - High school CS class you took 3 years ago does NOT count
 - No prior knowledge of C/C++ or UNIX is assumed
- Course objective
 - Meet the increasing demand for solid programming skills in many non-CS disciplines including EE & IEOR
 - Provide the minimum prerequisite for non-majors who are interested in taking upper-level CS courses

Course summary

- Follow-on course for ENGI E1006
 - Intended for non-CS majors
 - Bridges E1006 and many upper-level CS courses
 - Interleaves C/C++ language and data structure topics
 - Introduces professional UNIX programming tools
- Fusion of 3157 and 3134
 - 3157-lite: C & C++, but no heavy systems stuff
 - 3134-extract: only the most important data structures
 - Kill two birds with one 4-credit stone!
 - Perfect for EE & IEOR folks who came to 3157 to learn C/C++ but found it a bit too much

Review session

- Logistics
 - One topic / week, multiple sessions by different TAs
 - When:
 - Evenings – hours TBA
 - Where:
 - CLIC lab: <http://www.cs.columbia.edu/clic/directions.html>
 - Attendance optional, but recommended
- Topics
 - UNIX basics, editors, Git, etc. (in the beginning)
 - Lecture reviews
 - Lab assignment clarifications & reviews
 - Lab solutions walk-through
 - Exam preps

Participate in class, please!

- Classes are no fun (for me, at least) if we don't interact
 - Answer questions I pose
 - Ask questions anytime
 - Embarrass me when I'm wrong
- People are afraid to ask when they think:
 - “I'm the only one who doesn't know this.”
 - “I can't frame this question clearly and eloquently.”
 - “Maybe he just said it when I dozed off just now...”
- Big class, so I may not entertain all questions, but:

NEVER BE AFRAID TO ASK ANYTHING, IN THIS CLASS AND IN LIFE!

Course structure and focus

- Start with C and move toward C++, interleaving data structures throughout
- Focus on programming with data structures, not mathematical analysis
- “Job postings just say C++. Why waste time with C?”

Grading

- Grading logistics may change later
- You get overall score out of 100, comprised of:
 - HW assignments – 33%
 - Midterm exam – 33%
 - Final exam – 33%
- I look at everyone's HW & exam scores in a big spreadsheet sorted by the overall score
- I decide cutoffs for letter grades A+, ..., D, F
 - No predetermined formula for cutoffs
- Booster: I reserve the right to raise one's overall score by a small amount (typically less than 0.5%)

Booster

- Grade boost based on subjective evaluation
 - Most people will not get it
 - Have been used to boost some borderline cases
 - Can be up to 5% in theory, but never been $> 1\%$
- Based on:
 - Class participation
 - Mailing list participation
 - Beautiful code
 - Awesome documentation
 - Optional work (if any)

HW (aka lab)

- Probably 6 HW assignments
 - Mostly programming assignments
 - Some of them may have written parts
- Deadline
 - Soft deadline, and then hard deadline 2 days later
 - You use 1 late day if you submit within 24 hours after the soft deadline
 - You use 2 late days if you submit between 24 and 48 hours after the soft deadline
 - After 48 hours past the soft deadline, no submission will be accepted
 - You have 5 late days total; up to 2 can be used for a single hw
 - Check your late days by running: `/home/w3136/submit/check-late-days`
 - Absolutely no exception under any circumstances
 - After you receive grade, you have 2 weeks to send re-grade request

Lab grading

- Grading model
 - You are a software company
 - I hire you to develop a product according to spec
 - You ship the finished & polished product on time
 - TAs are the end users who will pay you with grade
- What this means:
 - Your software doesn't work, they don't pay
 - Your software didn't follow spec, they don't pay
 - Your software didn't ship on time, they don't pay
 - But you worked so hard... they sympathize, but they don't pay
- For example:
 - Your software doesn't compile – you get ZERO
 - Deductions for not following spec EXACTLY
 - Ex) Spec asked for README.txt file, not README, not README.md, not Readme.txt

Cheating. Please don't.

- **REQUIRED READING:**

<http://www.cs.columbia.edu/~paine/3136/honesty.html>

- You are cheating if you:
 - Take code from friends, or search for code on the Internet
 - Look at solutions that your friend has from previous semester
 - Upload any class materials (including your own code) to public repository (ex. GitHub) during or after this semester
- We can tell
 - We compare you submissions to **CURRENT AND PREVIOUS** submissions
 - You submit work history – **minimum 5 commits required**
- Result of cheating
 - Case 1: You get caught
 - Academic penalty – 1 letter grade down for mild cases; F for severe ones
 - Referral to the Office of Judicial Affairs
 - **I will make sure you are expelled. Your cheating devalues MY degrees.**
 - Spring 2016: 50+ cases of suspected cheating (still pending); 36 convicted cases
 - Case 2: You get away with it
 - You will keep cheating for the rest of your life – have a nice life.

Class ListServ

- Communication between all of us
 - Official announcements, lecture notes, lab assignments
 - Should be the 1st place to go for non-personal questions
- Do:
 - Ask & answer questions
 - Provide helpful tips and fun links for your classmates
 - Be considerate & friendly
- Don't:
 - Ask questions without first trying to solve it on your own
 - Post code or critical info that leads directly to solution
 - Be impatient & rude
- TAs and I respond to emails in this order:
 1. All pending questions on the listserv first
 2. All pending questions sent to cucs3136-tas@googlegroups.com
 3. Then individual emails
 4. NEVER send a same question separately to multiple people
 - You will get banned from ever sending an email if you get caught doing this.

Manage ListServ emails

- Learn to manage high volume – filter by tags in subject
 - [Cs3136] – all emails from the class listserv will have this tag
 - [ANN] – important announcements from me or TAs
 - [HW n] – information relevant on a particular hw
 - Examples:
 - [Cs3136][ANN] Sample midterm exam
 - [Cs3136][ANN][HW3] Correction on lab3 instruction
 - [Cs3136][HW4] in case you're curious about the optional part
- Setup Gmail filters
- Keep up diligently
- Yes, I know about Piazza. Thanks for your suggestion.

“Required” textbooks

- *Foundations of Computer Science: C Edition*
 - By Alfred V. Aho and Jeffrey D. Ullman
 - Out of print, but available for free
 - <http://infolab.stanford.edu/~ullman/focs.html>
- 1. *The C Programming Language* (2nd ed.) – aka K&R C
 - By Kernighan and Ritchie
 - Simply the best
- 2. *A Tour of C++*
 - By Bjarne Stroustrup

Last semester, only 4% bought them at the local bookstore
So get them wherever you usually get your textbooks

Recommended references

- Other data structures textbooks
 - *Data Structures and Algorithm Analysis in C++*
 - Mark A. Weiss
 - 4th ed. Is the latest, but any edition will do
 - Older editions based on C might be even better if you can find it
 - *Algorithms (4th Edition)*
 - Robert Sedgewick and Kevin Wayne
 - Best introductory DS & Algorithms book I have seen, but in Java

HW0 (may or may not be graded)

- **Part A (50 points): due noon Thursday**

1. Subscribe to 3136 ListServ today

- <https://lists.cs.columbia.edu/mailman/listinfo/cs3136>
 - In the textbox “Your name (optional)” put **Your Full Name (UNI)**
 - For example: Tim Paine (tkp2108)
 - **You must reply to the confirm email (which might be in your spam folder)**
 - Then receive “Welcome to the "Cs3136" mailing list”
 - This email contains your password for accessing archives of past postings
2. Get the textbooks
 - Start reading K&R chapters 1,2,3,4

HW0 continued

- **Part B (50 points): due Friday 11:59pm**
 1. Read the following two documents:
 - <http://www.cs.columbia.edu/education/honesty>
 - <http://www.cs.columbia.edu/~paine/3136/honesty.html>
 2. Send me an email containing:
 - Subject: “[3136] hw0-UNI”
 - Without the quotes, sole space before hw0, UNI replaced with your actual UNI in lowercase
 - Your name, major & school program, year
 - Ex) Tim Paine, Computer Science, Columbia College, class of 2014
 - Your pledge
 - see honesty.html above
 - CS classes taken and/or other programming background
 - Optionally anything else you want to let me know