11-601 Coding & Algorithms Bootcamp
Preliminary Fall 2018 Syllabus

Instructor

Ralf Brown (ralf@cs.cmu.edu, GHC 5711, office hours see www.cs.cmu.edu/~ralf or by appointment)

Teaching Assistants

Yuchen Cao <yuchenc1@andrew.cmu.edu>, Yunhan Gao <yunhang@andrew.cmu.edu>,

Liang Zhang <liangz1@andrew.cmu.edu>, Kuo Zhao <kuoz@andrew.cmu.edu>

Course Description

This course has one goal: To ingrain as deep a mastery of fundamental algorithm and coding skills as possible in the timeframe of this course. We will seek specifically to maximize your chances of superior performance in any coding interview, improving your ability to form structured thoughts with respect to algorithmic problem solving, improve your ability to describe and plan solutions to problems, and develop further your ability to translate your thoughts into code intuitively and explain that code to others.

Prerequisites

Students should already be able to program in Java at an intermediate level. No prior experience with JavaScript or Python is necessary. Familiarity with programming and object oriented programming principles is useful, as is the abillity to install, configure, and use integrated development environments such as Eclipse.

Required Text

"Cracking the Coding Interview: 189 Programming Questions and Solutions" 6th Edition (green cover) by Gayle Laakmann McDowell
Available from your favorite bookseller and
http://www.amazon.com/Cracking-Coding-Interview-6th-Edition/dp/0984782850

Other material will be provided as needed during the course.

Class Information

- Class meets Tuesdays and Thursdays from 12:00pm to 1:20pm in POS 153.
- There is no substitute for hands-on learning with algorithms and coding. For this reason, the majority of material you will be expected to learn will be covered in depth with the readings, exercises, and tutorials you will complete outside lecture. Please complete all readings and exercises on time – they are critical to doing well on the in-class exercises.
- Expect to average a full 12 hours per week on the work for this course
    - mock interviews: 2 hours
    - homework assignments: 4-6 hours
    - plus readings and working through textbook questions
- You will receive email from HackerRank when homeworks and in-class exercises are assigned.
- Due dates for assignments will be provided at the time the assignment is given. Late assignments will be penalized 10% for every day they are late; **assignments will not be accepted more than seven days late**. If you feel that the grade given on an assignment should be reconsidered, please submit the re-grade request in writing by email to the instructor, with a brief description of why you would like the assignment reviewed.

Evaluation

Grading will be based on the following criteria:

In-Class Exercises: 20%

There will be an in-class exercise lasting about 10-15 minutes during most class meetings. These will done online using the HackerRank system, so be sure to bring your laptop and have a fully-charged battery, as POS 153 does not have electrical outlets by the seats. These exercises **can not** be made up if you miss the class, but your four lowest scores on in-class exercises will be dropped (three lowest in calculating your mid-semester grade).

Homework Assignments: 30%

There will be an assignment every week except for the weeks in which there are exams, usually due at the end of the day on the following Monday. A homework assignment will usually consist of two HackerRank exercises, each of which will typically allow 12 or 24 hours to complete from the time they are started. Extensions to the due date will only be given in case of a major emergency.

Mock Technical Interviews: 25% (Peer 15%, Instructor/TA 10%)

Each week, every student will be required to administer a peer technical interview, as well as be interviewed by another student. Interview assignments will be made by “the shuffle” which will randomly assign specific interviewers/interviewees as well as specific questions. The shuffle will change each week. Students are responsible for arranging and coordinating these interviews once the interviewer/interviewee pairings are announced. If an interview cannot be completed for any reason, a written explanation of the missed interview must be submitted. Forms for the administration and assessment of interviews will be provided and are to be uploaded to Canvas following the interview. The questions for each shuffle (drawn from the textbook) will be on the current or recent subject of study. Additional 30-minute technical interviews with a TA or the instructor may be assigned at any time in addition to the weekly peer interview; there will normally be two such interviews over the course of the semester, beginning in Week 5.

Mid-Term Exam: 10%

A closed-notes/closed-book written exam that will take place during class using HackerRank and cover all material discussed to date. Includes multiple choice, short answer, and code questions.

Final Exam: 15%

A closed-notes/closed-book written exam that will take place in **two parts during class** in the last week of the course using HackerRank and will cover all material discussed during the course. Includes multiple choice, short answer, and code questions.

Curving

As this class is **deliberately designed to be challenging**, the target for the class average score is 85% (it was ~83% in 2016 and ~87% in 2017). We will rescale the numerical scores to shift the actual average to 85% with a standard deviation of 5% (if raw scores were symmetrically distributed, this would result in letter grades in the B range being the most common). Canvas will be updated regularly with the current curve starting with Homework #3. See the 2017 grade distribution graph below.

Pass/Fail

For those selecting the Pass/Fail option as permitted by their academic program, the criterion for a Pass is a grade of at least 75% *after* curving (a C, the minimum passing letter grade for a Masters-level course, would be at least 74% after curving).

Academic Integrity

Collaboration in learning concepts is encouraged among students – feel free to share notes (but not code) and hold study groups; however attribution must be given in the form of citation, quotation, or co-authorship on documents.

IF YOU PARTICIPATE IN A STUDY GROUP FOR ANY ASSIGNMENT, YOU MUST LIST ALL MEMBERS OF THE STUDY GROUP ON YOUR ASSIGNMENT.

****** Sharing or copying code is explicitly forbidden ******

Unless specifically permitted in the instructions for a particular assignment, you may not share your code with other students, use code from another student, or use code found online. Where permitted by the assignment, you must attribute any code not written by yourself; this is most easily done with a comment preceding that code which states something like "based on code found at http://foo.com/bar/" or "code provided by <studentname>".

If a student obtains access to another student's work without their knowledge, both students will still be held accountable. ***Secure your work!*** *To protect future students (including your future self),* ***do not post your solutions publicly****, even after the end of the course.*

Failure to list all participants of a study group on assignments, to disclose all the participants on a collaborative work, or to disclose the source of any code not written by yourself is not acceptable. These violations will result in the elimination of credit for the entire assignment on the first offense, and in the assignment of a failing grade for the course on the second offense. Violations will be reported to your department, which may **impose additional penalties**.

Violations of academic integrity are very serious and can result in heavy penalties up to suspension or expulsion from the university. Make sure you review and understand the information at: http://www.cmu.edu/policies/student-and-student-life/academic-integrity.html.

Take care of yourself.

This course is a lot of work. Do your best to maintain a healthy lifestyle this semester by eating well, exercising, avoiding drugs and alcohol, getting enough sleep and taking some time to relax. This will help you achieve your goals and cope with stress.

If you find yourself struggling with the material or workload, please **ask for help**. There are many helpful resources available on campus – including your course instructor and TAs – and an important part of the college experience is learning how to ask for help. Asking for support sooner rather than later is often helpful.

Don't panic if you mess up on one assignment (we actually expect everyone to be unable to finish some of the in-class exercises)**.** Your grade in this course is based on a very large number of scores, so any individual assignment has very little effect on your final grade.

If you or anyone you know experiences any academic stress, difficult life events, or feelings like anxiety or depression, we strongly encourage you to seek support. Counseling and Psychological Services (CaPS) is here to help: call 412-268-2922 and visit http://www.cmu.edu/counseling/

If you or someone you know is feeling suicidal or in danger of self-harm, call someone immediately, day or night:

CaPS: 412-268-2922 *Re:solve Crisis Network: 888-796-8226*

If the situation is life threatening, call the police:

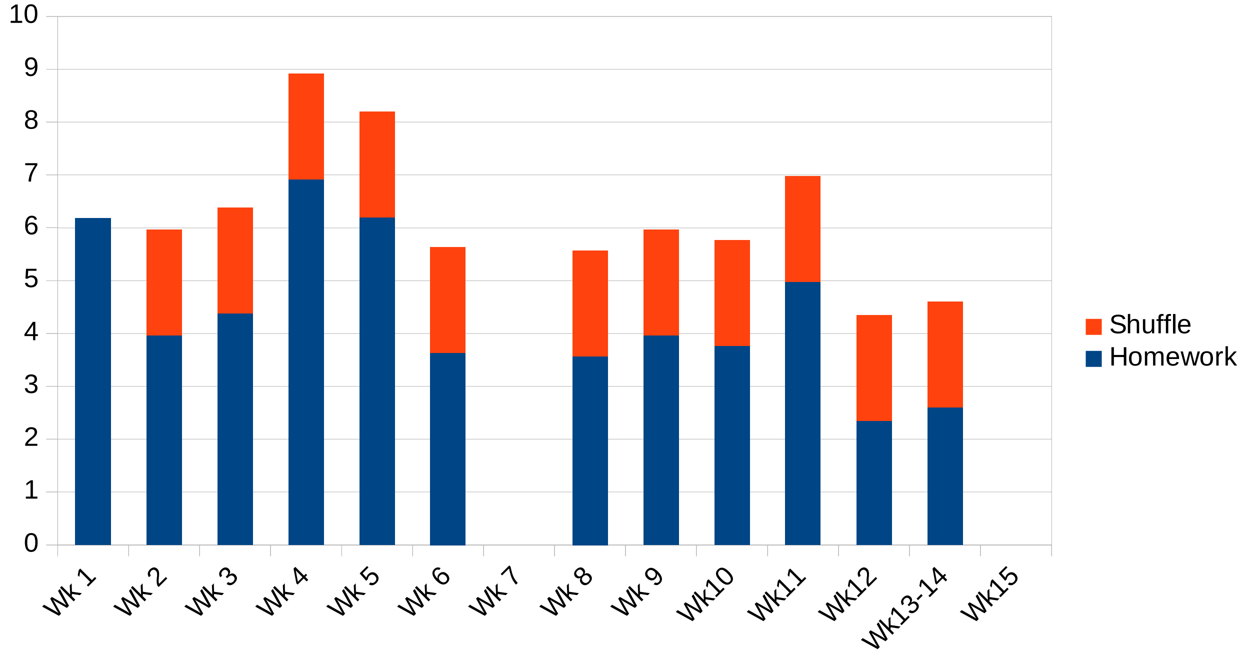
On campus: CMU Police: 412-268-2323, *Off campus: 911*

Tentative Course Schedule

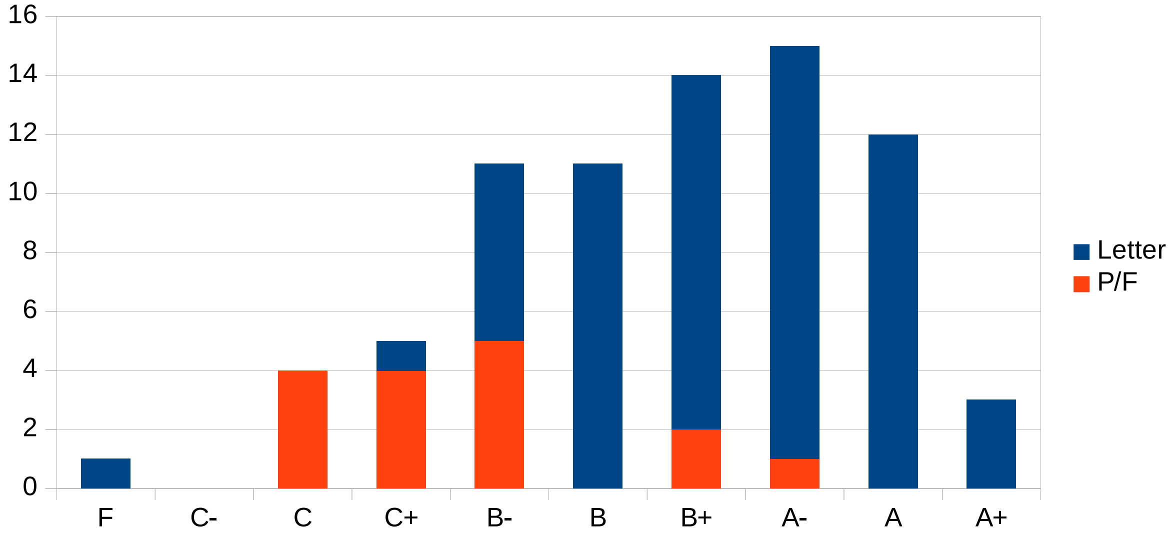
Wk	Date	Topic	Readings	Textbook Questions
1	8/28	Course Overview Java - Strings and Arrays	McDowell Ch. 1	
	8/30	Computational Complexity Java - String Manipulation Regular Expressions	McDowell Ch. VI McDowell Ch. 2	1.1-1.9 2.1-2.8
2	9/4	Interview Techniques Java – Testing Defensive Programming	McDowell Ch. VII McDowell Ch. 11 You Are Not Done Testing Until... (pages 2, 31-32)	11.1-11.6
	9/6	Java – Sorting Merge Sort, QuickSort	McDowell Ch. 10	10.1, 10.2, 10.6, 10.11
3	9/11	No Class (TOC)		
	9/13	Java – Sorting Insertion Sort, Radix Sort, ...	McDowell Ch. 3	3.1-3.6
4	9/18	Java – Searching Binary Search, BFS/DFS	McDowell Ch. 4	4.1-4.12, 10.3-10.5
	9/20	Java – Searching Path Searches	McDowell Ch. XI, pages 633-636	10.7-10.10
5	9/25	Java – Object-Oriented Programming Design Patterns	McDowell Ch. 7	7.1-7.12
	9/27	Java – Recursion / Optimization	McDowell Ch. 8	8.3, 8.4, 8.6-8.9
6	10/2	Java – Dynamic Programming		8.1, 8.2, 8.11, 8.13
	10/4	Java – Bit Manipulation	McDowell Ch. 5	5.1-5.8
7	10/9	Java – Math and Logic Puzzles	McDowell Ch. 6	6.1-6.10
	10/11	Parallel Processing and Scalability / Review	McDowell Ch. 9 and 15	
8	10/16	Midterm Exam		
	10/18	Python – Syntax	Whirlwind Tour of Python	
9	10/23	Python – String Parsing and Manipul.		
	10/25	Functional Programming Python – Sorting	McDowell Ch XI, pages 642-644 Functional Programming in Python	

Wk	Date	Topic	Readings	Textbook Questions
10	10/30	Python – Testing / Optimization Python – Searching	Monkey Patching	
	11/1	Python – Recursion and Dynamic Programming		
11	11/6	State Machines Combinatorics	TBD	
	11/8	JavaScript - Syntax	Intro to JavaScript for Java Programmers	
12	11/13	JavaScript – Strings and Sorting	JavaScript for Java Developers	
	11/15	JavaScript – Testing / Optimization JavaScript – Searching		
13	11/20	JavaScript – Recursion and DP JavaScript – Bit Manipulation		
	11/22	No Class (Thanksgiving)		
14	11/27	Self-Adjusting Trees	McDowell Ch. XI, p. 637-642	
	11/29	Flex / Review		
15	12/4	Final Exam, Part 1		
	12/6	Final Exam, Part 2		

Median Homework Load in hours, Fall 2017 class



Grade Distribution, Fall 2017 class



Notes: The failure resulted from an Academic Integrity Violation. Mid-semester grades were a very good predictor of final grades, with the majority of students receiving exactly the same final grade as midterm grade and none changing more than two steps (e.g. A- to B or B- to B+). *Past results are not a guarantee of future performance.*

About HackerRank

Homeworks, in-class exercises, and exams will be conducted using the system provided by HackerRank.com.

HackerRank provides the following features:

- an editor with code completion
- automatic scoring of submissions based on the number of test cases passed
- timed tests, with automatic submission of code when the time runs out
- logging of activity timelines and all code submissions (if you had a partially-working solution but the automatic submission was broken, let us know and we can verify your partial solution and give you credit for it)
- plagiarism detection (flagging submissions which are suspiciously similar to each other)

Sample Problem

If you would like to attempt a sample homework problem to see whether this course is for you, one of the problems from the first homework assignment in the Fall 2016 and Fall 2017 editions is available at

http://hr.gs/11601f18sample

This problem took students a median of about 3.5 hours and was generally considered the second-hardest of the course. You will have 12 hours to complete the problem once you begin.

Frequently Asked Questions

Q: I was just notified that I'm now enrolled, but I can't access the course in Canvas!

A: Canvas updates its roster from SIO several times per day rather than instantly. Please be patient – you'll have access in a few hours.

Q: What is the policy if part of a homework is submitted on time and part late?

A: The late penalty of 10% per day will only be assessed on the points for the part which is submitted late; the part which is submitted on time will receive full points earned.

Q: How do I get my assignment for the peer mock technical interviews (shuffle)?

A: The information on who to interview, who will interview you, and which questions to ask will be posted to Canvas as a comment on the appropriate assignment in your gradebook.

Q: What programming language(s) will be used during the interviews?

A: Instructor and TA interviews will all be in Java (to keep things consistent for everyone). Peer interviews will be in Java for the first half of the semester, then switch to Python and JavaScript. You will be informed which language to use when the interview is assigned in your gradebook.

Q: My laptop crashed/I fell ill/my Internet went out and the timer ran out on my homework assignment in HackerRank. What now?

A: Send an email and we can give you additional time to make up for the lost time.

Q: I realized after submitting my assignment that I forgot to cite someone or something. What do I do?

A: Send an email to your instructor as soon as possible with the missing citation.

Q: What if I panic under pressure and do something I shouldn't have?

A: Your penalty will be less severe if you confess before we find your transgression ourselves.

Q: I submitted XYZ, but the Canvas gradebook still shows a missing grade.

A: Most of the grade entries are handled by an external script which must be run manually. For example, homework assignments are normally processed shortly after the due date and again shortly after the last possible late submission date. If the grade still shows as missing more than a week after submission, send an email.

Q: What about ABC...?

A: See the FAQs posted on Piazza.