11-601 Coding Bootcamp
Fall 2015 Syllabus

Instructors	Ravi Starzl, GHC 6701 Ralf Brown, GHC 5711	E-mails	rstarzl@cs.cmu.edu ralf@cs.cmu.edu
Office Hours	By appointment	TAs:	Yiran Fei yiranf@andrew.cmu.edu Kang Huang kangh@andrew.cmu.edu Xingyu Yan xingyuy@andrew.cmu.edu

Required Texts:
"Cracking the Coding Interview 6th Edition"
by Gayle Laakmann McDowell
Available at:
http://www.amazon.com/Cracking-Coding-Interview-6th-Edition/dp/0984782850

Recommended Texts:
"Introducing Python"
by Bill Lubanovic
Available at:
http://www.amazon.com/Introducing-Python-Modern-Computing-Packages/dp/1449359361

"Web Scraping with Python"
by Ryan Mitchell
Available at:
http://www.amazon.com/Web-Scraping-Python-Collecting-Modern/dp/1491910291

Other material will be provided as needed during the 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 to 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 translating your thoughts into code intuitively.

Pre-requisites:
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. Must be able to install, configure, and use integrated development environments such as Eclipse and Java 8.

Evaluation:
Grading will be based on the following criteria:

Assignments: 33%
There will be one or more assignments every week, as well as random in-class exercises that will take place on Tuesday class meetings. Due dates will be provided at the time the assignment is given. Missed in-class exercises can be made up only if the instructors are given 24 hours advanced notice of an absence, or if the absence is due to an emergency. Unexcused absences will not be allowed to make up any work.

Peer Technical Interviews: 33%
Each week, every student will be required to administer two peer technical interviews, as well as be interviewed two times. Interview assignments will be made by "the shuffle" which will assign specific interviewers/interviewees as well as specific questions. The shuffle will change each week. All arrangements and coordination for conducting these interviews is the responsibility of the students. If an interview cannot be complete for any reason, a written explanation of the missed interview must be submitted. A form for the administration and assessment of interviews will be provided. The questions for each shuffle will be on the current or recent subject of study. Additional technical interviews with TAs and instructors may be assigned at any time.

Mid-Term: 17%
Closed-notes/closed-book written exam that will take place during class and covers all material discussed to date. Includes multiple choice, short answer, and code questions.

Final: 17%
Closed-notes/closed-book written exam that will take place during class and covers all material discussed to date. Includes multiple choice, short answer, and code questions.

Class Information:

- Class meets Tuesday and Thursday from 12:00pm to 1:20pm in SH 125.
- Tuesday classes will contain a review of important concepts related to the material we will cover that week. Thursday classes will be an open Q&A session with the instructor and TAs will be available for one-on-one assistance.
- Attendance is required on Tuesday, highly recommended Thursday. Every two unexcused absences will lower your grade one letter.
- There is no substitution 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, examples, and tutorials you will complete outside lecture. Please complete all readings and exercises on time.
- Expect to spend a full 12 hours per week on the work for this course.
- Due dates for assignments will be provided at the time the assignment is given. Late assignments will be penalized one grade for every two days it is late. All assignments are required, regardless of how late they are submitted. 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.

Academic Integrity
Collaboration is expected and encouraged among students – feel free to share notes 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.

Failure to list participants of a study group on assignments, or disclose all the participants on a collaborative work, 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 of academic integrity are very serious and can result in heavy penalties up to suspension or expulsion. Make sure you review and understand the information at:
http://www.cmu.edu/policies/documents/Cheating.html and
http://www.cmu.edu/policies/documents/GradDisc.html.
Sharing code is explicitly forbidden
Writing is a crucial skill for both academic and commercial success. Therefore we ask you to do your best possible writing in the comments for each assignment. Please use coherent arguments, good grammar, and correct spelling in your comments and solutions. If you need any assistance in editing or proofing, please make use of the resources at
http://www.cmu.edu/acadev/resources/writing.html or see the instructors for help.

Course Schedule:

Week	*Topic*
1	Java & Algorithms – Strings, Arrays, Linked Lists
2	Java & Algorithms – Stacks, Queues, Trees, Graphs
3	Java & Algorithms – Sorting
4	Java & Algorithms – Bits, Math, Logic
5	Java & Algorithms – Recursion, dynamic programming
6	Java & Algorithms – Testing, OO
7	Java & Software Engineering – Design Patterns, Agile
8	Java & Software Engineering – Design Patterns, PM
9	JavaScript – Document Based
10	JavaScript – Backend
11	Python – From Nothing to Control Structures
12	Python – From Control Structures to Objects and Classes
13	Python – Data and Web Wrangling
14	Python – Concurrent and Network Ops

Assignments:

Week	*Topic*
1	McDowell 1.1-1.9, 2.1-2.8
2	McDowell 3.1-3.6, 4.1-4.12
3	McDowell 10.1-10.11
4	McDowell 5.1-5.8, 6.1-6.10
5	McDowell 8.1-8.14
6	McDowell 7.1-7.12, 11.1-11.6
7	Pattern Problems 1-3, Agile Scenario
8	Pattern Problems 4-6, PM Scenario
9	JQuery Project
10	Node.js Project
11	Python problem set 1.1-1.36, Web Crawler Project Assigned
12	Python problem set 2.1-2.17
13	Python problem set 3.1-3.26
14	Python problem set 4.1-4.5, Web Crawler Project Due