

Course Syllabus

[Jump to Today](#)

Syllabus

CSE 310D Technical Interviews, Spring 2019

Department of Computer Science and Software Engineering

College of Engineering and Computing

Miami University

Instructor

Dr. John Femiani,

Phone: 513-529-0349

Email: femianjc@MiamiOH.edu (<mailto:femianjc@MiamiOH.edu>)

Office hours: MWF 1:30-4:30

Office Location: BEN214, or BEN205F.

→ *I will often be in the graduate student lab during office hours.*

→ *I have an "open door" policy; if my door is open please say hi*

Course Summary

This course will cover problem solving to prepare students for the technical interview process. Students will review key problem solving strategies specific to the technical or coding interview. We will explore the specific types of question one might expect when seeking an internship or a full-time job, and also the types of interaction and communication that are expected. Students will learn to demonstrate their abilities effectively in written / oral interviews. Students will work in groups to simulate the hiring and interviewing process in a friendly practice environment, and we will explore and discuss tools and techniques to prepare for the process.

Format of Meetings

[5-10 minutes] We meet and discuss the reading and recap last week's activities

[The rest of the time] We break out into groups based on "mock" companies. Half of the groups are "candidates", and the other groups are "interviewers". Repeat with the roles switched

Grading

This is a pass-fail course. Students are expected to attend and participate fully in all course activities.

Prerequisites:

This course is available by invitation. Students generally have Algorithms (CSE464), Data Structures (CSE274) and Discrete Math (MTH231).

Textbook

Cracking The Coding Interview [<http://www.crackingthecodinginterview.com/> <http://www.crackingthecodinginterview.com/>]

Tentative Plan

R 2:50 PM - 3:45 PM - 01/28/19 To 05/18/19 - 266 Engineering Building

Wk Topics	Rdg	Activity
1 Frontmatter: Introduction, Interview process, Before The Interview	I, II, IV	<i>Form Mock Companies</i> <i>Prepare Resumes</i>
2 Frontmatter: Behavioral Questions, Soft Skills, Big O, technical Questions Interview Preparation Grids, Telling your story	V,VI, VII	<i>Choose who your mock company will interview based on resumes.</i> <i>Assign phone interview for next time [will ask soft & O questions]</i>
3 Arrays, String, Linked Lists, Hash tables	Ch01, Ch02	Interviewer / Interviewee process discussion. Whiteboard breakout -- pairs of groups one question each person.
4 Stacks and Queues, Trees and Graphs	Ch03, Ch04	Whiteboard breakout -- pairs of groups one question each person.
5 Bit Manipulation, Math and Logic Puzzles	Ch05, Ch06	Whiteboard breakout -- pairs of groups one question each person. New mock companies Update resumes

6 Java, C, and C++ Question (language questions)	Ch12, Ch13 <i>Choose who your mock company will interview based on resumes.</i>
	<i>Assign phone interview for next time. [Will ask cumulative questions]</i>
	Interviewer / Interviewee process discussion.
7 Object Oriented Design Questions	Ch07 Whiteboard breakout -- pairs of groups one question each person.
8 Recursion and Dynamic programming Questions	Ch08 Whiteboard breakout -- pairs of groups one question each person.
9 System Design and Scalability	Ch09 Whiteboard breakout -- pairs of groups one question each person.
10 Sorting and Searching	Ch10 Whiteboard breakout -- pairs of groups one question each person.
11 Software Testing Questions	Ch11 Whiteboard breakout -- pairs of groups one question each person.
	Whiteboard breakout -- pairs of groups one question each person.
12 Threads and Locks Questions	Ch15 New mock companies Update resumes
	<i>Choose who your mock company will interview based on resumes.</i>
13 Database Questions	Ch14 <i>Assign phone interview for next time. [Will ask cumulative questions]</i>
	Interviewer / Interviewee process discussion.
14 Review problems	Ch16, Ch17 Whiteboard breakout -- pairs of groups one question each person.

Final Exam time TBD: *No test!*

Other concerns

Recording of classes: In order to make the classroom a safe place for discussion, for both the professor and students, all video or audio recording of class periods is prohibited. This policy may be waived, however, with the consent of the professor and the other members of the class.

Plagiarism detection software notice: Students agree, by taking this course, that all required papers may be subject to submission for textual similarity review for the detection of plagiarism. All submitted papers will be included as source documents in the a reference database solely for the purpose of detecting plagiarism of such papers.

Attendance: Your course grade includes a participation component, which is a catch-all for participating in surveys to improve the course, providing exit-cards, and attendance and actively participating in lectures. Attendance will be taken at the beginning of each class; if you miss four classes before the withdrawal deadline I will withdraw you from the course.

Computer Science and Software Engineering

Academic Integrity Expectations for

Individual and Group Problem Solving Assignments

The Department of Computer Science and Software Engineering is committed to maintaining strict standards of academic integrity. The department expects each student to understand and comply with the University's Policy on Academic Integrity: <http://www.miamioh.edu/integrity> and the [undergraduate student handbook](http://www.miamioh.edu/files/documents/secretary/Student_Handbook.pdf) and the [graduate student handbook](http://miamioh.edu/files/documents/secretary/Student_Handbook.pdf). Students may direct questions regarding academic integrity expectations to their instructor or to the department chair. All work submitted must be **original** for that class. Submitting the same project for two different classes is grounds for charging a student with academic misconduct unless prior written permission is received from **both** instructors.

“Problem Solving Assignments” are assignments that involve **programming, math, proofs, derivations, and puzzles**. The purpose of a problem solving assignment is for you to develop the skills necessary to solve similar problems in the future. To learn to solve problems you must solve the problems and write your solutions independently.

It is worth reiterating that the important aspect of the assignment is that you actually create the solution from start to finish; simply copying a solution and then **understanding it after the fact is not a substitute** for actually developing the solution.

The notion of academic integrity can be confusing in courses with substantial problem solving because certain forms of collaboration and investigation are permitted, but you are still required to complete your assignment independently. The following scenarios are meant to help distinguish between acceptable and unacceptable levels of collaboration and research, but are not all-inclusive:

ACCEPTABLE:

- Consulting solutions from the current course textbook, but not from other published sources.
- Seeking help on how to use the programming environment such as the editor, the compiler, or other tools.
- Seeking help on how to fix a program syntax error or how a certain language feature works.
- Discussing strategies with a fellow student on how to approach a particular problem. This discussion should not include significant sections of completed work or source code (including printouts, email, viewing on a monitor). Discussions should begin with a clean sheet of paper and end with conceptual drawings and/or pseudo-code.

UNACCEPTABLE:

- Looking at another solution including those written by current students, past students, or outside sources such as code or solutions found on the Web, or in publications other than the current class textbook.
- Using another solution as a starting point and then modifying the code or text as your own work.
- Providing a copy of your solution or a portion of your solution, in any form (electronic, hard copy, allowing another student to view your code on a monitor), to another student.
- Giving or receiving code fragments to fix a problem in a program.

If you are stuck on a problem and you are tempted to search for a solution on the Web or to look at another student’s solution **STOP** and email or ask your instructor for help.

Course Summary:

Date	Details	
Thu Feb 7, 2019	 Prepare for "Soft" skills and Big O Questions (https://miamioh.instructure.com/courses/92907/assignments/971030)	due by 4:25pm
Thu Feb 14, 2019	 Read Chapter 1 and Chapter 2 (https://miamioh.instructure.com/courses/92907/assignments/971032)	due by 11:59pm
Thu Feb 21, 2019	 Read Chapter 3 and Chapter 4 (https://miamioh.instructure.com/courses/92907/assignments/971033)	due by 11:59pm
Thu Feb 28, 2019	 Read Chapter 5 and Chapter 7 (https://miamioh.instructure.com/courses/92907/assignments/971034)	due by 11:59pm
Thu Mar 7, 2019	 Read Chapter 8 (https://miamioh.instructure.com/courses/92907/assignments/971035)	due by 11:59pm
Thu Mar 14, 2019	 Read Chapter 10 (https://miamioh.instructure.com/courses/92907/assignments/971036)	due by 11:59pm
Thu Mar 21, 2019	 Read Chapter 9 and Chapter 11 (https://miamioh.instructure.com/courses/92907/assignments/971037)	due by 11:59pm
Thu Mar 28, 2019	 Read Chapter 12 and Chapter 13 (https://miamioh.instructure.com/courses/92907/assignments/971038)	due by 11:59pm
Thu Apr 11, 2019	 Read Chapter 14 and Chapter 15 (https://miamioh.instructure.com/courses/92907/assignments/971041)	due by 11:59pm
Thu Apr 18, 2019	 Read Chapter 16 (Moderate Problems) (https://miamioh.instructure.com/courses/92907/assignments/971039)	due by 11:59pm
Thu Apr 25, 2019	 Read Chapter 17 (Hard Problems) (https://miamioh.instructure.com/courses/92907/assignments/971042)	due by 11:59pm