

How Should I Approach CSE 2050?

Hey y'all, it's Akeva! I'll be reflecting about my time during CSE 2050 and how you should approach the class: exams, quizzes, homework, and all.

For some context, I took this class when it had been three solid semesters where I had not touched coding due to the nature of my concentration in Biomedical Engineering. Specifically, my last class was CSE 1010, ... I know. Anyways, I can understand how you may feel as though you are underprepared. However, I will say that the class starts off fairly slow and begins to pick up in about the second or third week in terms of introducing new topics. During that time, if you have never touched Python or haven't used it in a while, I would suggest using the following videos/resources:

- [Programming with Mosh](#)
- [W3 Schools](#)
- [Codecademy](#)

If I am being honest, I did not these and I think I had to make up for it by searching up basic functions later. It's better to do this up front and find the best method of refreshing/learning for you.

For each topic in the class, I would suggest that you watch videos to solidify the concepts, redo the algorithms done in class, and do practice problems based on the topics learned in class. These will help you tremendously in quizzes and exams. The quizzes are mainly conceptual and going through the workflow of each algorithm to test you understand. This is further tested during the conceptual part of your exams during lab and coding is done in-person during lecture, so use quizzes as a way to study for the conceptual part of exams. Yes, it will be imperative you understand the data structure; however, they begin to introduce the topic of algorithms which is using a data structure in different contexts. To be able to do this, make sure you know the basic data structure down pact and try doing practice problems based on the data structures using leetcode.com. My algorithms professor (CSE 3500) has us do homework problems with this and I think it would have been nice to know this when studying for exams because some questions require modifying data structures for a particular use. Furthermore, it helps you think more in which data structure can help you achieve an outcome. For understanding topics, I suggest the following content creators on YouTube (for me, it's how I learn best):

- Michael Sambol
- Greg Hogg
- Codebagel

For studying for exams, I suggest:

- Watching videos explaining the concepts behind the code for solidification of the topics

- Writing the code by hand (since you have to for the final exam anyways), by memory until you have it down perfectly
- Explaining to yourself what each line of code does and how it contributes to the underlying structure
- Doing easy leetcode.com problems on the topics you will be tested on

You will also be introduced to this thing called Big O notation which helps us understand the amount of space our code takes on our computer based on what the code is doing and amount of time dependent on inputs. I remember being confused by it when I first learned it but after you match structures and time, it will be simple. Use the following resources:

- [NeetCode](#)
- [Michael Sambol](#)

For homework, please do your homework or at least look at it the day its released so you can gauge about how long it will take you to complete it. I am a procrastinator at heart and with my 19 other credits I was taking at the time, I did not do this, and it left me scrambling and have to spend more time studying for exams because I was understanding less due to the amount of stress I was in. Don't be like me. Commit to doing part of the homework each day, even if you are unable to code 30 minutes a day as suggested by the professors. It will help you practice and do your homework at the same time.

During class, code with the professor and write notes about key concepts and explanations, which is helpful for quizzes and exams. Don't hesitate to ask questions.

That's all I have regarding approaching CSE 2050! Good luck on the course and know that you are able to succeed regardless of your starting point.