



Hi everyone, Congratulations on clearing the coding test.

As you already know we will be starting mock interviews based on the topics of Module 2. This mock is to ensure that you are completely ready to continue in module 3. You will be thoroughly evaluated and will be transferred back to Module 2 in case you do not score up to the mark.

Topics:

- Oops
- Linked Lists
- Binary Search
- Stack and Queues

Please go through this video to understand how to book a mock interview session.

<https://share.vidyard.com/watch/DDwyRyp5GTpirZCQHbZyJc>

There are a few important things you need to take care of once you book a session:

1. Be available on the date and time of the session.
2. If your mentor doesn't show up in the session please mark it as "NO SHOW" and inform your batch manager.
3. If there is any emergency then please reach out to the mentor and ask them if the interview can be rescheduled and also inform your batch manager.
4. Make sure your camera and microphone are working properly.
5. Do not abruptly leave the ongoing session.
6. Please behave appropriately with the mentor. If you face any issues then please inform your batch manager.

We have made the marking scheme in the real company interview format. You will be evaluated on these parameters in the real interview as well so better to get evaluated on the same.

How to give a mock?

<https://www.youtube.com/watch?v=11QQjcMPVTQ>

These are the following parameters based on which you will get your score:

1. Theory-based questions
2. 2 coding questions
3. Approach: Discussing your approach is very important with the interviewer as it gives them an idea of how you approach solving problems.
 - It is very important to discuss both brute force and the optimal approach with the interviewer, You will be getting scores accordingly. If you just discuss the brute force approach or the optimal approach, your scores will get deducted.
4. Code evaluation: Your final code should be completely working. If it is not, your marks will get deducted.
5. Code explanation: You need to explain the code properly. It is better if you do it while writing the code as it keeps the flow of the interview.
 - Practice explaining the code while writing code, then only you will be able to explain it well while giving the interview.
6. Hints: If you are stuck on a problem, you can ask the interviewer to provide hints. But some marks will be deducted if you solve the question after taking a hint. The marks deducted will depend on the kind of hint.
7. Code quality: It is very important to have good code quality, as interviewers are very observant and tend to deduct marks if your code is not clean. These are a few checks you can keep track of to ensure good code quality:
 - Variable Naming: It is very important to give meaningful names to the variables being used in your code as it increases the readability of the whole code. The interviewer will better understand your code and the variables used.
 - Comments: It is a must to write comments as they are added with the purpose of making the source code easier for humans to understand.
 - Functions and Modular code: Try to break your code down into small functions that do one thing. For example, if you need to write the logic of merge sort, break it into a separate merge function and a separate function to break the array.
It increases the readability and usability of the code across an organization or a team of developers that's why it is very important to take care of this.
8. Pseudo code: Write a short pseudo-code while explaining the approach. It is a text-based form of your code that helps you in writing the actual code.
For example:

```
If student's grade is greater than or equal to 60
    Print "passed"
else
    Print "failed"
```