Amazon Software Development Engineer II Online Assessment Prep via Hacker Rank

Overview

Let's walk you through the assessment to better prepare you for it.

- The assessment will take about 2 hours to complete - we appreciate your time!
- The assessment are designed to help you showcase your coding skills and work style. We are interested in your demonstration of problem solving, writing correct code, applying patterns/data structures, and optimizing for algorithmic performance on the tests. Don't take preparing for this lightly, it will require you to study.

The 3 sections of the assessment **(Remember, ALL THREE SECTIONS PLAY A ROLE IN THE OVERALL EVALUATION)**

1. Coding challenge (2 scenarios) and Coding approach (elaborate why you code the way you do) - 105 minutes
2. Work style survey - 15 minutes
3. Feedback survey - 5 minutes

What about trying out a demo first?

The Coding Assessment Demo is optional to help you get familiar with the platform before starting the actual assessment. You can use it as part of your preparation.

Instructions (please read before you start the actual test)

- Do not click the **Start the assessment** link below until you're ready to start and complete the assessment.
- Set aside at least 2.5 hours in a place where you can focus.
- Make sure your internet connection is stable.
- Use the latest version of Google Chrome, Firefox or Safari.
- Acknowledge that you'll be the sole person who completes the assessment, no third parties or other resource are consulted and used.

During the coding challenge:

- You may select from any of the following languages: C, C++, C++14, C#, Go, Java7, Java8, JavaScript, Kotlin, Objective-C, PyPy2, PyPy3, Python2, Python3, Ruby, Scala, Swift
- Find supported compiler versions here.
- Manage your time effectively by checking the on-screen timer regularly.
- Complete the entire assessment in one sitting - the timer can't be paused once the assessment starts.
- ***Dishonest behavior will be caught.*** The assessment records everything once you begin. If you copy and paste code from outside it will be recorded.

From the coding challenge to the surveys:

- Take the work style survey and feedback survey after completing the coding challenge.
- Look for the confirmation that will appear once you've completed the entire assessment.

Still got questions?

Refer to our **FAQ** below if you need more information.

FAQs

Want to catch a glimpse of how it's like working at Amazon?

You can refer to the software development topics on amazon.jobs.

What resource may I use during the coding challenge?

You may use the resource that are available for all candidates (e.g. JDK or STL). Use the in-browser editor to work on the coding challenge and elaborate your coding approach as much as possible. Your coding and elaboration skills are both considered in our evaluation.

Do I have to complete both scenarios in the coding challenge?

Yes, you're expected to complete both scenarios. Manage your time effectively by checking the on-screen timer regularly during the coding challenge. Don't get stuck on one question. Move on to the next one and return to it later if you need to.

Important Assessment topics

- The time remaining will be clearly displayed on the screen. **You will not be able to stop the timer once you have started each test**, so we recommend that you complete each test in one sitting.
- Note that efficiency and optimization, as opposed to brute force solutions, earn more points! **Your code must compile for all code questions in order to move forward in the interview process. Be sure to test your code and ensure it runs before you submit your code or before time runs out.**
- You can compile your code as many times as you like during the assessment, but there must a 15 second interval between consecutive compilations.
- If you feel like your solution is mostly correct but you are facing some final compile issues, do not worry. Attempt to solve both of the challenges to the best of your ability.
- Edge Cases: Ensure your solutions consider all edge cases and handle large inputs effectively. This is key to doing well in the assessment.
- Your code is being auto-saved periodically, and you can also save it clicking on the SAVE button. In case of a system failure you can resume from the last saved instance. Your code will also auto-save when you click on NEXT QUESTION if you are going back and forth.
- If you need to take a break, the best time would be *between* the Coding Test and Work Styles Assessment. If you choose to take a break, log out, and when you're ready to start, log back in with the credentials below and the test will take you where you left off last.

TECHNICAL TOPICS

Programming Languages

Familiarity with a prominent language is generally a prerequisite for success. Be familiar with the syntax and nuances of common languages – Java | Python | C# | C | C++ | Ruby | JavaScript

Data Structures

Storing and providing access to data in efficient ways. Understand the inner workings of common data structures and be able to compare and contrast their usage in various applications. Know the runtimes for common operations as well as how they use memory. Wikipedia is a great resource for brushing up on data structures.

Algorithms

Basic implementation strategies of different classes of algorithms is more important than memorizing the specific details of any given algorithm. Consider reviewing traversals and divide and conquer algorithms. Consider knowing how and when to use a breadth-first search vs. a depth-first search, and what the trade-offs are. Knowing the runtimes, theoretical limitations, and basic implementation strategies of different classes of algorithms > memorizing specific details of any given algorithm.

Coding

The most important thing a Software Development Engineer does at Amazon is write scalable, robust, and well-tested code. Be comfortable coding by hand. Expect to be asked to write syntactically correct code—no pseudo code. Check for edge cases and validate that no bad input can slip through. The goal is to write code that's as close to production-ready as possible. This is your chance to show off your coding ability.

Object-Oriented Design

Good design is paramount to extensible, bug free, long-lived code. Using object-oriented design best practices is one way to build lasting software. Have a working knowledge of a few common and useful design patterns. Know the appropriate use of inheritance and aggregation. Expect to defend and describe your design choices.

Databases

Most of the software that we write is backed by a data store, somewhere. Many of the challenges we face arise when figuring out how to most efficiently retrieve or store data for future use. The more you know about how relational and non-relational databases work and what trade-offs exist between them, the better prepared you will be.

Operating Systems

Be familiar with some OS topics that can affect code performance. You won't need to know how to build your own operating system from scratch, but you should have an understanding of: memory management, processes, threads, synchronization, paging, and multithreading.

Preparation for the Online Assessment

Strategy:

1. Solve the coding problem to pass the test/edge cases - this can be a brute force solution. The reason for this is that the test grades the code by testing the edge/test cases.

2. AFTER you have solved a solution that passes the test/edge cases, THEN go back and optimize the solution.

3. Work Style Survey - 15 minute Question: The second part of the online assessment is our Work Style assessment. Here, you will review and rate a series of statements that are aligned with our Leadership Principles. This is an important way for us to learn more about you, and must be completed for your assessment to be viewed as complete.

Areas of Focus:

1. Algorithms: Basic Searching, Basic Sorting, Tree Traversal, Graph Traversal

2. Data Structures: Heaps, linked lists, arrays, trees (esp binary search trees), hash tables, stacks, and recursion

3. Fundamentals: Go back and re-educate yourself on all data structures and data structure algorithms. Understand all time and space complexity. Make sure you even get into more exotic things like hashmaps, b+trees (and variants), caches (and associated algorithms). Visit www.codechef.com for practice.

4. Understand high scale architecture. Go look at how other big sites are structured: http://highscalability.com/blog/category/example

Coding Assessment Preparation Videos:

1. What is Amazon Assessing? Learn about Amazon's online coding assessment and how it differs from other companies.
2. How to Navigate Amazon's Coding Assessment Platform? Before starting on your coding assessment, make sure you understand the ins and outs of the assessment platform.
3. How to Break Down the Coding Problem? After getting comfortable with the coding platform, the first step is to analyze the problem statement you're presented with.
4. Solving the Coding Problem. Now that you've broken down the problem statement, let's get started on the solution
5. Running the Solution and Test Cases. Now, you'll want to test and adjust your solution to ensure it passes all test cases.
6. Fine Tuning the Solution. Now that you've verified a working solution, consider the runtime complexity of our solution, and how we can continue improving upon it.

Additional preparation resources:

- **Fundamentals:** Go back and re-educate yourself on all data structures and data structure algorithms. They come up all the time. Understand all time and space complexity. Make sure you even get into more exotic things like hashmaps, b+trees (and variants), caches (and associated algorithms). You may want to visit www.codechef.com
- **Project Euler** - http://projecteuler.net/. If you've been out of practice, do up to problem 60 or so. You should be forced to build useful libraries, which will help give practical experience for interview like problems. Don't cheat or take shortcuts.
- **Google Coding Contest** has additional practice questions from past archived questions and results: https://codingcompetitions.withgoogle.com/codejam/archive
- **A 2021 Useful Blog Post:** https://www.jobtestprep.com/amazon-online-assessment
- **Must do Coding questions for Amazon:** https://www.geeksforgeeks.org/must-coding-questions-company-wise/#amazon

- **Pramp** - Pramp is a site where you can do mock phone/video interviews with another person. You take turns interviewing with another professional. You can split up the time however you deem fit.
- **Some discussions / examples from Leetcode:**
    - https://leetcode.com/list/xyknlrg2/
    - https://aonecode.com/amazon-online-assessment-questions
    - https://www.teamblind.com/post/Amazon-online-assessment---how-to-prepare-3yuoAgNQ