



Student Programs Software Development Engineering Assessments

Assessment objectives and components

At Amazon, one of our highest priorities is hiring and developing the best, and we work hard to raise the performance bar with every hire. Amazon uses online assessments as one way to help us get to know you better, and we design them to measure key characteristics required for success in a role. Assessments also allow us to assess applicants consistently and equitably, as every individual is provided with the same experience and information needed to complete the assessment. Depending on the position, we may ask you to take an assessment during the application process or send it to you separately after you've applied.

For the role of an SDE1 or an SDE intern, the assessment focuses on the technical skills needed for success as an SDE and your demonstration of the Amazon [Leadership Principles](#). The content of the assessment provides a short but realistic example of the kind of work you will do at Amazon as an SDE.

There are three parts in the assessment:

- Part 1: Code Debugging
- Part 2: Coding and Work Styles
- Part 3: Day-in-the-Life SDE Simulation

After you complete each part, you will receive an email telling you whether or not you will move forward to the next step in the application process.

In the code debugging and coding assessment components, you will have the option to code in Java, Python, C++, or C#. For a more in-depth description of the assessment and some examples, watch this video [here](#).

PART ONE: Code Debugging

At Amazon, it's important that our code is free from errors, so SDEs often have to review and troubleshoot code to identify issues. In this first section, you will complete a code debugging task which tests how you identify, understand, and address coding errors.

The code debugging component of the assessment requires you to read another person's code and identify and address compilation, logical, and syntactical errors. It takes approximately 20 minutes to complete. You will be given seven questions and asked to debug them. The screenshot below shows what the debug section looks like.

Problem | Test Cases | Output |

Compile and Run



You are required to fix all the logical error in the given code. You can click on *Compile & Run* anytime to check the compilation/execution status of the program. You can use *System.out.println()* to debug your code. The submitted code should be logically/syntactically correct and pass all testcases. Do not write the *main()* function as it is not required. **Code Approach:** For this question, you will need to complete the code as in given implementation. We **do not** expect you to modify the approach.

The function/method **checkPairSumExists** accepts four arguments:
rows, an integer representing the rows of the two-dimensional array;
cols, an integer representing the columns of the two-dimensional array;
arr, a list of integers representing the two-dimensional array;
sum, an integer representing the sum of a pair of integers to be searched in the array.

The function/method returns a Boolean value as "true" if a pair with a given sum value exists in the array. Otherwise it returns "false."

The code compiles successfully but fails to return the desired result for some test cases because of an incorrect implementation of the function/method **checkPairSumExists**. Your task is to fix the code so that it passes all the test cases.

```
1 import java.util.*;
2
3 class Pair
4 {
5     static boolean checkPairSumExists(int rows, int cols, int arr[], int sum)
6     {
7         Set<Integer> set = new HashSet<Integer>();
8
9         for (int i = 0; i < rows; i++)
10         {
11             for (int j = 0; j < cols; j++)
12             {
13                 if (set.contains(sum - arr[i][j]))
14                 {
15                     return true;
16                 }
17                 else
18                 {
19                     set.add(sum);
20                 }
21             }
22         }
23         return false;
24     }
25 }
```

SUBMIT ANSWER

PART TWO: Coding and Workstyles Assessment

Part 2 of the Online Assessment includes the Coding Assessment and Workstyles Assessment.

Coding

The coding assessment evaluates your knowledge of basic programming constructs, data structures, and algorithms. It is used to assess your computer science fundamentals (e.g., data structures and algorithms), your approach to problem solving, and the clarity, maintainability, and efficiency of your code. These are all important skills that you will apply in your everyday work at Amazon.

You will have 70 minutes to complete two coding problems. For each of the two problems, the problem statement, and the expected input and output will help you understand the objective of your code. Some test cases will also be available to test and debug your code. The screenshot below shows what the debug section looks like; you can take a practice test here: <https://www.myamcat.com/amazon-campus-demo>. The practice test will allow you to get more comfortable with the coding platform you'll use to complete the questions. There is also an opportunity to test out the coding environment when you start the assessment.

Problem | Test Cases | Output |

Code saved | Compile and Run | Java

The current selected programming language is **Java**. We emphasize the submission of a fully working code over partially correct but efficient code. Once **submitted**, you cannot review this problem again. You can use `System.out.println()` to debug your code. The `System.out.println()` may not work in case of syntax/runtime error. The version of **JDK** being used is **1.8**.

Amazon is building a way to help customers search reviews quicker by providing real-time suggestions to search terms when the customer starts typing. When given a minimum of two characters into the search field the system will suggest at most three keywords from the review word repository. As the customer continues to type in the reviews search bar the relevant keyword suggestions will update automatically.

Write an algorithm that will output a maximum of three keyword suggestions after each character is typed by the customer in the search field.

If there are more than three acceptable keywords, return the keywords that are first in alphabetical order.
Only return keyword suggestions after the customer has entered two characters.
Keyword suggestions must start with the characters already typed

Both the `repository` and the `customerQuery` should be compared in a **case-insensitive way**.

Input
The input to the method/function consists of three arguments:
`numReviews`, an integer representing the number of various keywords from the Amazon review comment section;
`repository`, a list of unique strings representing the various keywords from the Amazon review comment section;
`customerQuery`, a string representing the full search query of the customer.

Output
Return a list of a list of strings, where each list represents the keyword suggestions made by

```
1 // IMPORT LIBRARY PACKAGES NEEDED BY YOUR PROGRAM
2 // SOME CLASSES WITHIN A PACKAGE MAY BE RESTRICTED
3 // DEFINE ANY CLASS AND METHOD NEEDED
4 import java.util.List;
5 // CLASS BEGINS, THIS CLASS IS REQUIRED
6 public class Solution
7 {
8     // METHOD SIGNATURE BEGINS, THIS METHOD IS REQUIRED
9     List<List<String>> threeKeywordSuggestions(int numReviews,
10                                             List<String> repository,
11                                             String customerQuery)
12     {
13         // WRITE YOUR CODE HERE
14     }
15     // METHOD SIGNATURE ENDS
16 }
```

SUBMIT ANSWER

Workstyles Assessment

The Workstyles Assessment is centered around Amazon's [Leadership Principles](#). This assessment will include pairs of statements and you'll be asked to choose what statement is more like you. For instance, we may ask you to choose one of the following statements, "I like for things to be clearly structured," or "I look forward to the opportunity to learn new things," that best describes you.

PART THREE: Work Simulation

Lastly, you will complete an interactive simulation designed to give you a window into a typical day in the life of an SDE at Amazon. The simulation focuses on skills central to the success of an SDE at Amazon, including problem solving, dealing with ambiguity, curiosity, collaboration, troubleshooting, and code reviewing and testing.

You'll be placed in scenarios that SDEs are likely to encounter on the job. During the simulation, you will be prompted with emails, videos, and instant messages from your virtual managers and team members to solve various problems. For each scenario, you will be provided with four to five viable options of how to respond and be asked to either rate the effectiveness of each response or select the single response that you think is the best course of action. Additional work materials will also be provided to help you make your decision, such as internal wikis, code snippets, and roadmaps. The work simulation will take approximately two hours to complete.