

ARIZONA STATE UNIVERSITY (ASU) CSE110 PRINCIPLES OF PROGRAMMING EXAM 1 PRACTICE (SAMPLE) STUDY GUIDE



Prepare with structure. Pass with confidence



Copyright © 2026 by Examzify - A Kaluba Technologies Inc. product.

ALL RIGHTS RESERVED.

No part of this book may be reproduced or transferred in any form or by any means, graphic, electronic, or mechanical, including photocopying, recording, web distribution, taping, or by any information storage retrieval system, without the written permission of the author.

Notice: Examzify makes every reasonable effort to obtain accurate, complete, and timely information about this product from reliable sources.

SAMPLE

Table of Contents

Copyright	1
Table of Contents	2
Introduction	3
How to Use This Guide	4
Questions	5
Answers	8
Explanations	10
Next Steps	16

SAMPLE

Introduction

Preparing for a certification exam can feel overwhelming, but with the right tools, it becomes an opportunity to build confidence, sharpen your skills, and move one step closer to your goals. At Examzify, we believe that effective exam preparation isn't just about memorization, it's about understanding the material, identifying knowledge gaps, and building the test-taking strategies that lead to success.

This guide was designed to help you do exactly that.

Whether you're preparing for a licensing exam, professional certification, or entry-level qualification, this book offers structured practice to reinforce key concepts. You'll find a wide range of multiple-choice questions, each followed by clear explanations to help you understand not just the right answer, but why it's correct.

The content in this guide is based on real-world exam objectives and aligned with the types of questions and topics commonly found on official tests. It's ideal for learners who want to:

- Practice answering questions under realistic conditions,
- Improve accuracy and speed,
- Review explanations to strengthen weak areas, and
- Approach the exam with greater confidence.

We recommend using this book not as a stand-alone study tool, but alongside other resources like flashcards, textbooks, or hands-on training. For best results, we recommend working through each question, reflecting on the explanation provided, and revisiting the topics that challenge you most.

Remember: successful test preparation isn't about getting every question right the first time, it's about learning from your mistakes and improving over time. Stay focused, trust the process, and know that every page you turn brings you closer to success.

Let's begin.

How to Use This Guide

This guide is designed to help you study more effectively and approach your exam with confidence. Whether you're reviewing for the first time or doing a final refresh, here's how to get the most out of your Examzify study guide:

1. Start with a Diagnostic Review

Skim through the questions to get a sense of what you know and what you need to focus on. Your goal is to identify knowledge gaps early.

2. Study in Short, Focused Sessions

Break your study time into manageable blocks (e.g. 30 – 45 minutes). Review a handful of questions, reflect on the explanations.

3. Learn from the Explanations

After answering a question, always read the explanation, even if you got it right. It reinforces key points, corrects misunderstandings, and teaches subtle distinctions between similar answers.

4. Track Your Progress

Use bookmarks or notes (if reading digitally) to mark difficult questions. Revisit these regularly and track improvements over time.

5. Simulate the Real Exam

Once you're comfortable, try taking a full set of questions without pausing. Set a timer and simulate test-day conditions to build confidence and time management skills.

6. Repeat and Review

Don't just study once, repetition builds retention. Re-attempt questions after a few days and revisit explanations to reinforce learning. Pair this guide with other Examzify tools like flashcards, and digital practice tests to strengthen your preparation across formats.

There's no single right way to study, but consistent, thoughtful effort always wins. Use this guide flexibly, adapt the tips above to fit your pace and learning style. You've got this!

Questions

SAMPLE

1. What is the output of the code snippet that checks the substring from index 1 to less than 5?
 - A. abcd
 - B. bcde
 - C. bcbcd
 - D. cdef
2. What do libraries in Java provide to developers?
 - A. Graphics and UI components only
 - B. Pre-written code and functions for various tasks
 - C. Only mathematical functions
 - D. A way to bypass the need for compilers
3. What differentiates an applet from a standard Java application?
 - A. Applets are compiled into native code
 - B. Applets can run in a browser
 - C. Applets can only be executed by the CPU
 - D. Applets have more features than applications
4. In a switch statement, what happens if the fall-through behavior occurs?
 - A. Only the first matching case executes.
 - B. All matching cases execute without breaks.
 - C. It results in a compile-time error.
 - D. Execution jumps to the default case.
5. Which of the following statements displays price = 20.00?
 - A. `System.out.print("price = "); System.out.printf(price);`
 - B. `System.out.print("price = "); System.out.printf("%f", price);`
 - C. `System.out.print("price = "); System.out.printf("%10.2f", price);`
 - D. `System.out.print("price = "); System.out.printf("%2.10f", price);`
6. What will be the output of the following code snippet?
 - A. No output.
 - B. Hello will be displayed 10 times.
 - C. Hello will be displayed 9 times.
 - D. Hello will be displayed infinite times.

- 7. A loop inside another loop is called?**
- A. A sentinel loop
 - B. A nested loop
 - C. A parallel loop
 - D. A do/while loop
- 8. What will the loop output if the condition is set to repeat until a negative number is entered?**
- A. All positive numbers will be printed.
 - B. The loop will never execute.
 - C. No output will be displayed if negative is input first.
 - D. All input will be stored and displayed after loop ends.
- 9. What term refers to a collection of programs that a computer executes?**
- A. Hardware
 - B. Software
 - C. Firmware
 - D. Network
- 10. Which of the following statements is correct about constants?**
- A. Constants are written using capital letters because the compiler ignores constants declared in small letters.
 - B. The data stored inside a constant can be changed using an assignment statement.
 - C. You can make a variable constant by using the final reserved word when declaring it.
 - D. Constant variables can only be changed through the Math library.

Answers

SAMPLE

1. C
2. B
3. B
4. B
5. C
6. B
7. B
8. A
9. B
10. C

SAMPLE

Explanations

SAMPLE

1. What is the output of the code snippet that checks the substring from index 1 to less than 5?

- A. abcd
- B. bcde
- C. bcbcd**
- D. cdef

To determine the output of the code snippet that checks the substring from index 1 to less than 5, it's essential to understand how substring indexing works in programming languages like Python, Java, or similar languages. When a substring is taken from the original string, the start index specifies the first character to include in the substring, while the end index specifies the first character to exclude from the substring. So, when you check from index 1 to less than 5, the substring will start at the second character (since indexing usually starts at 0) and will include characters up until the character just before the fifth index. To illustrate this with an example: if the original string is "abcde", taking a substring from index 1 to 5 will result in "bcd". However, if the string were different and contained more characters, the substring could change accordingly. Given that the answer selected is "bcbcd", it indicates that there is some misunderstanding in interpreting how substring extraction works in terms of character inclusion and index range. The only correct interpretation of the indexes would yield a substring of the characters that actually fall in the defined range. In this case, closely analyzing the provided options and the requirements of indexing will lead to identifying

2. What do libraries in Java provide to developers?

- A. Graphics and UI components only
- B. Pre-written code and functions for various tasks**
- C. Only mathematical functions
- D. A way to bypass the need for compilers

Libraries in Java provide pre-written code and functions for various tasks, which significantly enhances the efficiency and productivity of developers. These libraries include a wide range of functionalities, such as data structures, algorithms, input/output processing, networking, and much more. By using these libraries, developers can leverage existing solutions rather than writing code from scratch, allowing them to focus on implementing unique features of their applications. This approach not only speeds up the development process but also helps in maintaining code quality, as these libraries are often reliable and have been tested extensively. Utilizing libraries allows for code reuse and helps in establishing consistent methods for common programming tasks, which is fundamental in software development. Other options do not fully capture the breadth of what libraries offer. For example, while graphics and UI components are part of some Java libraries, they do not encompass the wide array of functionalities that libraries provide overall. Similarly, limiting libraries to only mathematical functions neglects the comprehensive nature of libraries available in Java. Lastly, suggesting that libraries allow one to bypass compilers misrepresents their role; libraries assist in development but do not change the need for compilation to translate Java code into executable format.

3. What differentiates an applet from a standard Java application?

- A. Applets are compiled into native code
- B. Applets can run in a browser**
- C. Applets can only be executed by the CPU
- D. Applets have more features than applications

An applet is specifically designed to be embedded in a web browser and run in the context of a web page. This capability enables applets to provide interactive content right within the browser environment, allowing users to experience dynamic functionality without requiring the installation of additional software on their devices. This integration with web browsers is a key distinguishing factor between applets and standard Java applications, which typically run in a standalone environment and do not have the same level of web integration. The choice that applets can run in a browser highlights their intended purpose and utility in web development, particularly in the early days of web programming. This unique feature of applets ultimately emphasizes their role in enhancing web pages with interactive elements, while standard Java applications operate independently from the browser context.

4. In a switch statement, what happens if the fall-through behavior occurs?

- A. Only the first matching case executes.
- B. All matching cases execute without breaks.**
- C. It results in a compile-time error.
- D. Execution jumps to the default case.

In a switch statement, fall-through behavior refers to the scenario where, after executing a matching case, the control continues to execute the subsequent cases until it encounters a break statement or reaches the end of the switch block. This means that if no break statements are provided in several cases, all of those cases will execute in sequence. This behavior can be particularly useful when multiple cases share the same execution code or when you want to handle similar logic for multiple conditions. However, it can also lead to unintentional executions if the programmer does not carefully manage break statements. In contrast, if the switch statement were to only allow the first matching case to execute, the potential shared logic across multiple cases would be lost. Additionally, allowing fall-through does not cause compile-time errors; rather, the switch statement executes as per defined logic. Finally, jump execution directly to the default case would only occur if none of the specified cases match, which is different from the behavior described in this context. Thus, the correct interpretation of fall-through is that all matching cases execute sequentially until the end of the switch structure or a break statement is encountered.

5. Which of the following statements displays price = 20.00?

- A. `System.out.print("price = "); System.out.printf(price);`
- B. `System.out.print("price = "); System.out.printf("%f", price);`
- C. `System.out.print("price = "); System.out.printf("%10.2f", price);`
- D. `System.out.print("price = "); System.out.printf("%2.10f", price);`

The correct choice effectively formats the output of the variable `price` to display it as a floating-point number with two decimal places, ensuring that it appears in the required format of "20.00". When using `System.out.printf`, the format specifier `%10.2f` indicates that the output should be a floating-point number (f) with a total width of 10 characters, including 2 decimal places. This means it will display the number with two digits after the decimal point, padded with spaces if necessary to reach the total width of 10. As a result, if `price` is assigned the value `20.00`, it would be displayed as " 20.00", centered in a field of 10 characters. Additionally, the use of the format specifier ensures that the output will be consistent and properly formatted, making the code robust for any floating-point value assigned to `price`. This is particularly useful in situations where alignment of numerical output in reports or logs is important. Other choices vary in their formatting specifiers, which would not necessarily produce the correct output: - The first choice simply attempts to print the variable directly without specifying how to format it, which would likely lead to an error unless

6. What will be the output of the following code snippet?

- A. No output.
- B. Hello will be displayed 10 times.
- C. Hello will be displayed 9 times.
- D. Hello will be displayed infinite times.

The output of the code snippet is indeed that "Hello" will be displayed 10 times. This typically stems from the structure of a loop in programming, such as a `for` loop that iterates a specified number of times. In many programming languages, a common implementation might look something like this: ``python for i in range(10): print("Hello") `` Here, the loop is set to run 10 iterations, starting from 0 up to, but not including, 10. During each iteration, the instruction to print "Hello" is executed, leading to the output being displayed 10 times. This clear structure of specifying a finite number of iterations is what results in exactly 10 outputs. Other options do not align with the described behavior of a properly structured loop, making them less relevant to the functioning of the code snippet in question.

7. A loop inside another loop is called?

- A. A sentinel loop
- B. A nested loop
- C. A parallel loop
- D. A do/while loop

A loop inside another loop is referred to as a nested loop. This structure allows for multiple iterations over a set of items, which is particularly useful when dealing with multi-dimensional data structures, such as arrays or matrices. For instance, in a nested loop, the outer loop may iterate over rows while the inner loop iterates over columns, enabling you to access each element in the matrix systematically. Nested loops facilitate complex operations, as you can execute one loop's iterations completely for every instance of the outer loop. This is critical in scenarios such as matrix multiplication, where each element of a row needs to be multiplied by each element of a column. Understanding nested loops is essential for solving problems that involve repeated execution of code blocks where the number of iterations is determined by the conditions of both the outer and inner loops. This concept is foundational in programming and helps in managing more intricate iterations within data structures.

8. What will the loop output if the condition is set to repeat until a negative number is entered?

- A. All positive numbers will be printed.
- B. The loop will never execute.
- C. No output will be displayed if negative is input first.
- D. All input will be stored and displayed after loop ends.

The situation describes a loop that continuously executes until a negative number is entered. The purpose of the loop is to process input values, specifically focusing on numbers that meet the given condition (i.e., positive numbers). Since the loop is set to stop executing as soon as it encounters a negative input, all positive numbers entered prior to that negative input will be printed. When analyzing this behavior, it's clear that as long as the numbers being input are positive, they will be displayed as part of the loop's output. The loop only halts when a negative number is provided, thus preventing that negative number from being part of the output. If the first input is negative, the loop does not start, leading to the absence of any output which aligns with another choice, but it does not relate to the correct answer here. Choosing all positive numbers being printed encapsulates the correct behavior of the loop when it operates under the given condition. This outcome highlights the loop's design to capture and exhibit only those values that meet the specified criteria of being non-negative while terminating upon receiving a negative input.

9. What term refers to a collection of programs that a computer executes?

- A. Hardware
- B. Software
- C. Firmware
- D. Network

The term that refers to a collection of programs that a computer executes is software. Software encompasses all the programs and applications that run on a computer, including operating systems, utilities, and application software. It is essentially the non-tangible component of a computer that enables it to perform specific tasks or functions. Software can be categorized into various types, such as system software, which manages the hardware components and provides a platform for running application software, and application software, which performs tasks for the user, like word processing or web browsing. This collection of programs is crucial for the functioning of computers, allowing users to interact with the hardware and perform operations efficiently. In contrast, hardware refers to the physical components of a computer, firmware is specific software that is closely tied to hardware, often embedded in devices, and a network pertains to the connection of multiple devices for communication and data exchange. Each of these terms describes different aspects of computing and does not capture the concept of executable programs collectively like software does.

10. Which of the following statements is correct about constants?

- A. Constants are written using capital letters because the compiler ignores constants declared in small letters.
- B. The data stored inside a constant can be changed using an assignment statement.
- C. You can make a variable constant by using the final reserved word when declaring it.**
- D. Constant variables can only be changed through the Math library.

Constants in programming are entities whose values cannot be altered after they have been assigned. The statement that correctly describes this characteristic is that a variable can be made constant by using the `final` reserved word when declaring it. This ensures that once a value is assigned to the variable, it cannot be modified throughout the program. For instance, if you declare a constant like `final int myConstant = 10;`, any subsequent attempt to change `myConstant` will result in a compile-time error. Using `final` is a common practice in many programming languages, as it provides a way to define immutable values, promoting safer and more readable code. This is particularly useful in contexts where certain values are meant to remain constant, such as configuration settings or mathematical constants. Other choices do not accurately describe the nature of constants. The misunderstanding that the compiler ignores constants declared in lowercase highlights a misconception, as the compiler recognizes constants regardless of their case, but using capital letters is just a convention to signal that a value should not be changed. The other options suggest that constants can be modified or have specific limitations that aren't aligned with the fundamental principle of how constants function. Thus, making a variable constant through the use of `final` is the correct approach.

Next Steps

Congratulations on reaching the final section of this guide. You've taken a meaningful step toward passing your certification exam and advancing your career.

As you continue preparing, remember that consistent practice, review, and self-reflection are key to success. Make time to revisit difficult topics, simulate exam conditions, and track your progress along the way.

If you need help, have suggestions, or want to share feedback, we'd love to hear from you. Reach out to our team at hello@examzify.com.

Or visit your dedicated course page for more study tools and resources:

<https://asu-cse100exam1.examzify.com>

We wish you the very best on your exam journey. You've got this!

SAMPLE