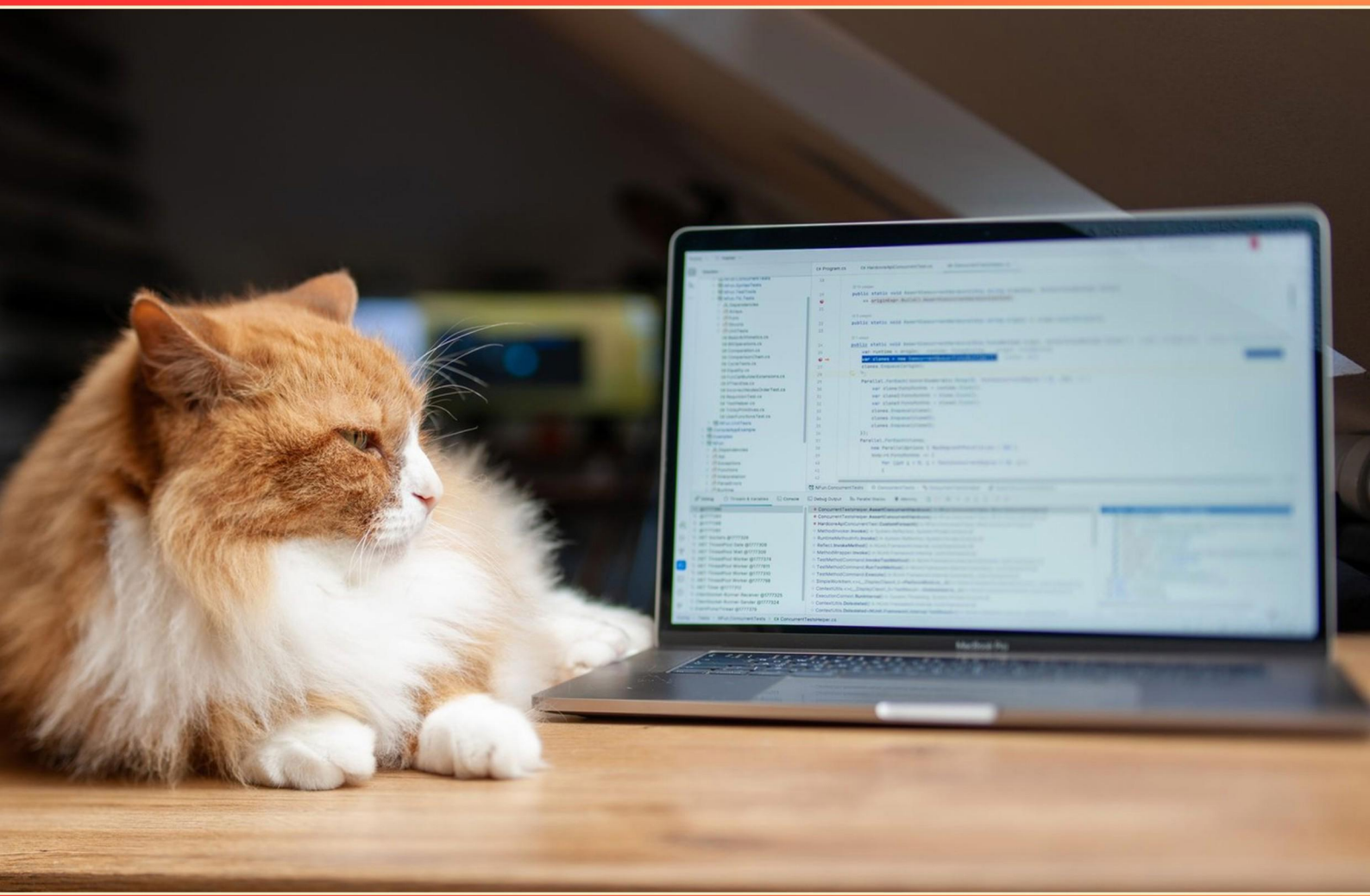
ARIZONA STATE UNIVERSITY (ASU) CSE110 PRINCIPLES OF PROGRAMMING EXAM 1 PRACTICE (SAMPLE)

STUDY GUIDE



SAMPLE STUDY GUIDE WITH LIMITED QUESTIONS

VISIT US ONLINE FOR THE FULL VERSION STUDY GUIDE

https://asu-cse100exam1.examzify.com

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 15

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.

SAMPLE

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!

SAMPLE

Questions

SAMPLE

1. **The Java programming language was originally designed to be simple, secure, and compatible with what?**

   A. Only Windows systems

   B. Many different processor types

   C. Only Linux systems

   D. Exclusively for embedded systems

2. **Why is hand-tracing code considered valuable?**

   A. It is usually faster than running the code.

   B. It assists in debugging by revealing hidden errors.

   C. You must already have a working program to do it.

   D. It gives valuable insight that you do not get by running the code.

3. **What type of loop is used in this given code?**

   A. For loop

   B. While loop

   C. Do-while loop

   D. Infinite loop

4. **Which of the following is the mathematical equivalent of the following Java expression: h = (4.0 * a * b - Math.pow(b, 2)) / c;**

   A. h = 4ab - 2b / c

   B. h = (4ab - 2b) / c

   C. h = 4ab - b^2 / c

   D. h = (4ab - b^2) / c

5. **What is likely the main reason to use pseudocode in programming?**

   A. It eliminates all errors before coding.

   B. It clarifies the design of the algorithm.

   C. It automatically translates into programming languages.

   D. It defines the complete syntax needed.

SAMPLE

6. Which operator would you use to get the remainder of a division in Java?

A. +

B. -

C. *

D. %

7. How many times does the following code fragment display "Hi"?

A. 9 times

B. 10 times

C. 11 times

D. 12 times

8. What will be the output of the code snippet when the user enters 23 and 45?

A. 23

B. 4523

C. 68

D. 2345

9. Which statement best describes the function of a do-while loop?

A. It must be executed at least once.

B. It can skip the execution based on the condition early.

C. It functions the same as a while loop.

D. It cannot be nested within other loops.

10. What is the result of evaluating the expression (45 / 6) % 5?

A. 2

B. 7

C. 3

D. 0

SAMPLE

Answers

SAMPLE

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

SAMPLE

Explanations

SAMPLE

1. The Java programming language was originally designed to be simple, secure, and compatible with what?

A. Only Windows systems

B. Many different processor types

C. Only Linux systems

D. Exclusively for embedded systems

The correct choice reflects the goal of the Java programming language's design to be versatile and adaptable across various platforms. Java was created with the intention of being able to run on many different processor types through its "Write Once, Run Anywhere" (WORA) philosophy. This means that code written in Java can be executed on any device that has a Java Virtual Machine (JVM), regardless of the underlying hardware architecture. This compatibility is achieved through Java's architecture-neutral design and the use of bytecode, which allows the same Java program to run on different systems without modification. The language was not limited to a specific operating system like Windows or Linux or designed exclusively for embedded systems; rather, it embraced a broader compatibility that enhances its utility and widespread adoption in diverse computing environments.

2. Why is hand-tracing code considered valuable?

A. It is usually faster than running the code.

B. It assists in debugging by revealing hidden errors.

C. You must already have a working program to do it.

D. It gives valuable insight that you do not get by running the code.

Hand-tracing code is a method where programmers manually step through the logic of a program, tracking variable values and flow of control. This technique provides valuable insights that are not always available through simply running the code. When running a program, especially in complex applications, it might behave as expected due to specific inputs or conditions that are not fully understood or anticipated. Hand-tracing allows the programmer to break down the code into smaller parts, which helps in understanding how the program operates at each step. It reveals the logic of the code and allows for deeper analysis of how data moves through the program, how functions interact, and how conditions are evaluated. This thorough understanding can lead to better design, optimization, and debugging, as programmers become familiar with the inner workings of the code. Ultimately, hand-tracing is a valuable educational tool, enhancing problem-solving skills and promoting a more nuanced understanding of programming concepts.

SAMPLE

3. What type of loop is used in this given code?

A. For loop

B. While loop

C. Do-while loop

D. Infinite loop

The code in question employs a do-while loop, which is distinguished by its structure that allows the loop body to execute at least once before evaluating the condition to determine if it should continue. This is because the condition check occurs after the loop's statements have been executed. The defining characteristic of a do-while loop is the placement of the condition check at the end of the loop, which guarantees that the statements within the loop will run a minimum of one time, regardless of whether the condition is initially true or false. In contrast, a for loop typically involves initialization, a condition that is checked before each iteration, and incrementation, making it suited for scenarios where the number of iterations is known in advance. A while loop also checks its condition before entering the loop, which can lead to scenarios where the loop might not execute at all if the condition is false initially. An infinite loop would continuously execute without terminating under any condition, which is not the case here since the loop is structured to stop based on a conditional check at the end. Thus, the presence of the condition at the conclusion of the loop block combined with its guaranteed initial execution supports the identification of the loop type as a do-while loop.

4. Which of the following is the mathematical equivalent of the following Java expression: h = (4.0 * a * b - Math.pow(b, 2)) / c;

A. h = 4ab - 2b / c

B. h = (4ab - 2b) / c

C. h = 4ab - b^2 / c

D. h = (4ab - b^2) / c

The expression in question is $h = \frac{4.0 \cdot a \cdot b - \text{Math.pow}(b, 2)}{c}$. To understand the correct mathematical equivalent, it is essential to parse the components of the Java expression clearly. Firstly, $\text{Math.pow}(b, 2)$ calculates b^2, which is crucial for correctly representing that part of the equation mathematically. Therefore, the expression simplifies to $h = \frac{4ab - b^2}{c}$. Looking at the options, the correct form retains the structure of the original equation: both the numerator and denominator are clearly defined. In the choice that is correct, $4ab - b^2$ is enclosed in parentheses, indicating that the entire expression $4ab - b^2$ is considered as a single entity before division by c. This grouping is vital to preserving the intended order of operations, ensuring that the computation is done in the same sequence as in the Java expression. Hence, the choice reflects the need for clarity in mathematical expressions, particularly when those expressions involve multiple operations, ensuring that calculations adhere to

SAMPLE

5. What is likely the main reason to use pseudocode in programming?

A. It eliminates all errors before coding.

B. It clarifies the design of the algorithm.

C. It automatically translates into programming languages.

D. It defines the complete syntax needed.

The main reason to use pseudocode in programming is that it clarifies the design of the algorithm. Pseudocode serves as an intermediary between the problem-solving stage and actual coding. By allowing programmers to outline their logic and flow of control in a simplified, human-readable format, pseudocode helps to refine and solidify the algorithm's structure before any syntactical elements of programming languages come into play. This helps ensure that the overall design is effective and that logical steps are clear, making the transition to actual coding more straightforward and effective. Utilizing pseudocode can effectively highlight areas of complexity or potential issues in the algorithmic design, leading to better overall planning and implementation. While it doesn't eliminate errors and doesn't directly convert to a programming language, its primary function revolves around clarifying thought processes and enabling better algorithm design, which is essential for successful programming.

6. Which operator would you use to get the remainder of a division in Java?

A. +

B. -

C. *

D. %

The operator used to obtain the remainder of a division in Java is the modulus operator, which is represented by the percentage sign (%). This operator performs division and returns the remainder of that division. For instance, using the expression `5 % 2` would yield a result of `1`, since 5 divided by 2 equals 2 with a remainder of 1. This operator is particularly useful in various programming scenarios, such as determining if a number is even or odd, or when performing operations in cycles (like wrapping around a value). In contrast, the other operators perform different arithmetic operations: the addition operator (+) sums two numbers, the subtraction operator (-) calculates the difference between them, and the multiplication operator (*) produces the product. None of these yield a remainder; therefore, they do not serve the purpose of the modulus operator.

7. How many times does the following code fragment display "Hi"?

A. 9 times

B. 10 times

C. 11 times

D. 12 times

To determine how many times the code fragment displays "Hi," it is essential to analyze the structure of the code involved. If the code includes a loop, such as a for loop or a while loop, it will execute based on the specified range or condition. A common setup might involve a loop that runs through a range of numbers or until a particular condition is met. For example, if the loop iterates from 0 to 10, it would execute a total of 11 times because it starts from 0 and goes up to but not including 10. Each iteration would output "Hi," resulting in the final total of 11 displays. Understanding the specifics of how loops work—such as their starting and ending conditions—is crucial. In many programming contexts, especially with zero-based indexing, the loop will iterate one more time than the limit specified, hence resulting in "Hi" being displayed 11 times when the upper boundary is exclusive. This kind of logical assessment of loop behavior leads to concluding that "Hi" is displayed 11 times, aligning with the provided answer.

SAMPLE

8. What will be the output of the code snippet when the user enters 23 and 45?

A. 23

B. 4523

C. 68

D. 2345

The output of the code snippet given that the user enters 23 and 45 will be 2345. This occurs because the code likely concatenates the two input values as strings rather than performing a mathematical addition. When the user enters the values 23 and 45, they are typically read in as string data types in many programming languages by default. When two strings are concatenated, they are combined end-to-end. Thus, "23" concatenated with "45" results in "2345". It's important to recognize that if the intention was to perform mathematical addition, the input values would need to be converted from strings to integers. If converted, the output would represent the sum of the two numbers, which would be 68. However, without that conversion, concatenation occurs, leading to the result of 2345.

9. Which statement best describes the function of a do-while loop?

A. It must be executed at least once.

B. It can skip the execution based on the condition early.

C. It functions the same as a while loop.

D. It cannot be nested within other loops.

The statement that a do-while loop must be executed at least once is fundamentally correct. This is a defining characteristic of the do-while loop compared to other types of loops, such as the while loop. In a do-while loop, the block of code contained within the loop is executed first before the condition is evaluated. This guarantees that the loop body runs a minimum of one time, regardless of whether the condition is true or false on the first evaluation. This behavior allows developers to implement scenarios where at least one execution is necessary, such as prompting a user for input and validating it afterward. This attribute is particularly useful in cases where you want to ensure that a specific action is taken before checking any conditions that might prevent further executions in subsequent iterations. Understanding this principle is critical for writing effective loop structures in programming.

10. What is the result of evaluating the expression (45 / 6) % 5?

A. 2

B. 7

C. 3

D. 0

To evaluate the expression (45 / 6) % 5, you need to follow the order of operations, which dictates that division occurs before modulus. First, perform the division: 45 / 6 equals 7.5. However, since we're dealing with integer division in this context (as would typically be expected in programming), the result will be truncated to 7. Next, take this integer result and perform the modulus operation: 7 % 5 calculates the remainder when 7 is divided by 5. When we divide 7 by 5, it goes in once (1 time) with a remainder of 2. Therefore, 7 % 5 gives us 2. This makes the overall result of evaluating the expression (45 / 6) % 5 equal to 2, confirming that the correct choice is indeed 2.

SAMPLE

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