END OF YEAR 8 (EOY8) COMPUTER SCIENCE PRACTICE EXAM (SAMPLE)

STUDY GUIDE



SAMPLE STUDY GUIDE WITH LIMITED QUESTIONS

VISIT US ONLINE FOR THE FULL VERSION STUDY GUIDE

https://eoy8compsci.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. What defines a boundary test?

A. A test with typical input

B. A test to ensure correct error messages

C. A test that checks values at the edge of the valid input range

D. A test that measures run time

2. Translate the following natural language task into code-like logic: 'Calculate the sum of all even numbers in a list'.

A. Sum of all numbers in the list.

B. Multiply each number by two and sum.

C. For n in list: if n % 2 == 0: sum += n

D. Compute the average of the list.

3. What does a linear search do?

A. It sorts the array before searching

B. It scans each element until the target is found

C. It uses binary search to split the search space

D. It builds a lookup table for fast access

4. What is the result of A OR B when A is false and B is true?

A. Always true.

B. Always false.

C. Depends on A and B.

D. True.

5. Is secondary storage volatile or non-volatile?

A. Volatile

B. Non-volatile

C. Both

D. None

SAMPLE

6. Authentication

A. A method of proving your identity, ensuring only certain users can access a given system

B. A backup encryption process

C. A network monitoring technique

D. A data backup procedure

7. Which of the following is an example of a dynamic data structure?

A. Array

B. List

C. Static Hash Table

D. Static Array

8. Which statement best describes metadata for a digital image?

A. The actual color values of each pixel in the image

B. The compression ratio of the image file

C. The primary content or subject of an image

D. Data about the data that is required to encode or decode the bits e.g. resolution, colour depth, file type, date

9. Which statement about a static data structure is accurate?

A. It has a fixed size determined at allocation.

B. It can grow automatically when elements are added.

C. It stores data in a linked list internally.

D. It changes size based on the number of operations.

10. The operator != stands for which relation?

A. Greater than

B. Less than

C. Equal to

D. Not equal to

SAMPLE

Answers

SAMPLE

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

SAMPLE

Explanations

SAMPLE

1. What defines a boundary test?

A. A test with typical input

B. A test to ensure correct error messages

C. A test that checks values at the edge of the valid input range

D. A test that measures run time

Boundary testing focuses on edge values of the input range to catch problems that happen at the limits. When inputs are defined within a range, it's essential to check the smallest and largest valid values, as well as values just inside and just outside those limits, to see how the system handles them. For example, if a program accepts 1 to 100, boundary tests would look at 1 and 100 (the edges) and nearby values like 2 and 99, plus outside values like 0 or 101 to ensure proper rejection or handling. This helps reveal off-by-one errors, input validation gaps, and how limits are enforced. This is why the option describing boundary values is the best choice. Other options address testing normal, typical input, or focus on error messages or performance, which don't specifically target the edge of the valid input range.

2. Translate the following natural language task into code-like logic: 'Calculate the sum of all even numbers in a list'.

A. Sum of all numbers in the list.

B. Multiply each number by two and sum.

C. For n in list: if n % 2 == 0: sum += n

D. Compute the average of the list.

Think of the task as a two-step process: go through every number, decide if it's even, and if so, add it to a running total. The key idea is using a loop to inspect each item and a conditional to filter only the evens. The modulo operation helps identify even numbers: a number is even when dividing by two leaves no remainder, which is written as remainder equals zero. So you start with a total of zero, loop through the list, and whenever the current number meets the even condition, you add it to the total. After finishing the loop, the total becomes the sum of all even numbers in the list. This matches the other choices for the task: summing all numbers would include odd ones, doubling each number and summing would produce a sum that doesn't reflect only the evens, and computing an average would give a mean rather than a total.

3. What does a linear search do?

A. It sorts the array before searching

B. It scans each element until the target is found

C. It uses binary search to split the search space

D. It builds a lookup table for fast access

A linear search checks each element in turn until it finds the target. It doesn't require the data to be sorted, and it's simple to implement: start at the first item, compare it with the target, and move to the next until a match is found or you reach the end. Because you might have to look at every item, the time it takes grows linearly with the number of elements. This differs from sorting the data first and then searching, which isn't what the linear search does. It's also different from binary search, which repeatedly splits the search space and needs the data to be sorted. And it isn't about building a lookup table, which is a preprocessing step for fast access rather than scanning for a single value.

SAMPLE

4. What is the result of A OR B when A is false and B is true?

A. Always true.

B. Always false.

C. Depends on A and B.

D. True.

The main idea is how the OR operation works: it outputs true if at least one input is true. With A false and B true, A OR B is true because there is a true input. So the result is true. This matches the option that describes True. (If both were false, it would be false; if either is true, it stays true.)

5. Is secondary storage volatile or non-volatile?

A. Volatile

B. Non-volatile

C. Both

D. None

Secondary storage is non-volatile because its purpose is to keep data even when the computer is turned off. This lets you save files, programs, and the operating system so you can access them later. In contrast, primary storage like RAM is volatile, meaning its contents are lost when power is removed. So for long-term storage and persistence, non-volatile devices such as hard drives, solid-state drives, USB sticks, and optical discs are used. If it were volatile, you would lose everything after shutting down; if it were both, it would blur the difference between storage types; if it were none, there wouldn't be a real category for persistent storage. Therefore, non-volatile is the correct description.

6. Authentication

A. A method of proving your identity, ensuring only certain users can access a given system

B. A backup encryption process

C. A network monitoring technique

D. A data backup procedure

Authentication is the process of proving who you are so that access to a system can be granted only to the right people. It involves checking credentials or attributes—like a password, a fingerprint, or a security token—to verify identity before entry. This matches the option because it describes the act of proving identity to ensure that only authorized users can access the system. Once identity is verified, the system can then decide what that user is allowed to do (authorization). The other ideas focus on protecting data or monitoring rather than proving identity: encrypting backups protects the data in storage, monitoring the network looks for unusual activity, and backing up data is about creating copies for recovery.

SAMPLE

7. Which of the following is an example of a dynamic data structure?

A. Array

B. List

C. Static Hash Table

D. Static Array

Dynamic data structures can grow or shrink as the program runs. A List fits this idea because you can add or remove elements and the structure automatically adjusts its size to hold whatever number of items you store. This flexibility is what makes it dynamic. In contrast, an array has a fixed size chosen when it's created, so it doesn't automatically resize. A static hash table and a static array are also fixed in size and layout, meaning they don't change capacity during execution. So the List is the example that best demonstrates a dynamic data structure.

8. Which statement best describes metadata for a digital image?

A. The actual color values of each pixel in the image

B. The compression ratio of the image file

C. The primary content or subject of an image

D. Data about the data that is required to encode or decode the bits e.g. resolution, colour depth, file type, date

Metadata is data about data. For a digital image, metadata includes details like resolution, colour depth, file type, and date. This information helps programs interpret and handle the image without needing to inspect every pixel. The actual color values of each pixel are the image data, not metadata, so describing the pixel values doesn't capture metadata. The compression ratio is a property of how the image is encoded rather than a description of the image's properties. The primary content or subject of the image is the image data, not metadata. The statement listing resolution, colour depth, file type, and date describes the data about the image rather than the image itself, which is exactly what metadata is.

9. Which statement about a static data structure is accurate?

A. It has a fixed size determined at allocation.

B. It can grow automatically when elements are added.

C. It stores data in a linked list internally.

D. It changes size based on the number of operations.

Static data structures have a fixed size determined at allocation, and that size stays the same for their entire lifetime. You can store only up to that many elements; if you need more space, you must allocate a new, larger structure. They don't grow automatically when elements are added, which is why they're considered fixed in capacity. The internal storage isn't required to be a linked list; a common example is a fixed-length array, though implementations can vary. Because the size isn't adjusted during use, it does not change with the number of operations.

SAMPLE

10. The operator != stands for which relation?

A. Greater than

B. Less than

C. Equal to

D. Not equal to

Inequality checks whether two values are different. The symbol not equal to is read as "not equal to" and it yields true when the two values differ, and false when they are the same. For example, 5 not equal to 3 is true, whereas 4 not equal to 4 is false. This is different from greater than or less than, which compare order rather than equality. It's also different from the equals operator, which checks if values are the same. Some languages use variations like a strict not equal (for example, !== in JavaScript) to differentiate both value and type, but the core idea remains: not equal to means the values do not match.

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://eoy8compsci.examzify.com

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

SAMPLE