

SQA National 5 Computing Science Practice Exam (Sample)

Study Guide



Everything you need from our exam experts!

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

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

- 1. What is a computer virus?**
 - A. A harmful program that can self-replicate and infect files**
 - B. A hardware malfunction affecting data storage**
 - C. A program designed to enhance system performance**
 - D. A type of software designed to protect against malware**
- 2. What is pseudocode typically used for?**
 - A. To write code in programming languages**
 - B. To outline algorithms using informal language**
 - C. To compile programs into executable files**
 - D. To perform debugging**
- 3. What does 'True Colour' refer to in digital imaging?**
 - A. 16,777,216 colors**
 - B. 8-bit color depth**
 - C. Grayscale imaging**
 - D. Monochrome imaging**
- 4. What is the purpose of a control character in computing?**
 - A. To display graphics on the screen**
 - B. To execute program commands**
 - C. To represent data in a readable form**
 - D. To perform non-printing actions**
- 5. What is the role of a web browser?**
 - A. To run backend scripts on a server**
 - B. To retrieve and display web content**
 - C. To manage databases**
 - D. To create web content**
- 6. What type of data is typically represented in a character set?**
 - A. Only numerical values**
 - B. Only alphabetic characters**
 - C. A combination of letters, numbers, and symbols**
 - D. Only punctuation marks**

- 7. Which of the following is an example of a high-level programming language?**
- A. C**
 - B. Python**
 - C. Assembly**
 - D. Machine Code**
- 8. What advantage does modularity offer in software development?**
- A. It ensures faster data processing**
 - B. It allows for easier updates and maintenance**
 - C. It reduces the need for user manuals**
 - D. It increases the complexity of the code**
- 9. What is the primary function of an Address Bus in computing?**
- A. A uni directional bus that specifies the address location in memory to be read from/written to**
 - B. A communication channel for data transfer between processor and peripherals**
 - C. A bus used exclusively for power distribution to hardware components**
 - D. A protocol for addressing network devices**
- 10. Which of the following accurately describes Base 2?**
- A. A numbering system using digits 0 to 5**
 - B. A numbering system that only has the digits 0 and 1**
 - C. A system that includes both positive and negative numbers**
 - D. A decimal system commonly used in everyday counting**

Answers

SAMPLE

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

SAMPLE

Explanations

SAMPLE

1. What is a computer virus?

A. A harmful program that can self-replicate and infect files

B. A hardware malfunction affecting data storage

C. A program designed to enhance system performance

D. A type of software designed to protect against malware

A computer virus is defined as a harmful program that can self-replicate and infect files. This means that a virus is designed to attach itself to legitimate files or applications, allowing it to spread from one system or file to another. Once a virus infects a system, it can cause various types of damage, such as corrupting files, stealing sensitive information, or rendering the system inoperable. The self-replication aspect is key, as it enables the virus to spread without user intervention, making it a significant threat to computer security and data integrity. In contrast, other options present concepts that do not align with the definition of a computer virus. For example, a hardware malfunction affecting data storage refers to issues related to physical components of the computer, such as hard drives or solid-state drives, rather than software-related threats. A program designed to enhance system performance would focus on optimizing the operation of a computer, improving speed or efficiency, rather than causing harm. Lastly, software specifically created to protect against malware refers to antivirus or anti-malware programs, which are intended to defend against viruses and other malicious software, not to function as a virus itself.

2. What is pseudocode typically used for?

A. To write code in programming languages

B. To outline algorithms using informal language

C. To compile programs into executable files

D. To perform debugging

Pseudocode is primarily utilized to outline algorithms in a way that is more accessible and easier to understand than traditional programming languages. It employs informal language and structured formatting to describe the steps and logic involved in solving a problem. This helps programmers to focus on the logic and flow of an algorithm without getting bogged down by the syntax of any specific programming language. Using pseudocode allows developers to plan and visualize algorithms clearly and intuitively, making it easier to translate these outlines into actual code later on. As pseudocode is not tied to any specific language, it serves as a universal tool that can be understood by programmers regardless of their preferred programming languages. This flexibility greatly aids in the design process, allowing for efficient communication of ideas among team members before the formal implementation stage begins.

3. What does 'True Colour' refer to in digital imaging?

- A. 16,777,216 colors**
- B. 8-bit color depth**
- C. Grayscale imaging**
- D. Monochrome imaging**

True Colour refers to a system of representing images in digital form using 24 bits per pixel, which allows for the display of 16,777,216 different colors. This color depth is achieved by using 8 bits for each of the three primary colors: red, green, and blue. Each of these colors can have 256 different intensity levels, and when these are combined in various ways, they create a vast range of colors that closely approximates what the human eye can perceive. This term is significant for digital imaging because it enables high-quality and vibrant representations of images, making it ideal for applications like photography, graphic design, and multimedia. The high fidelity in color reproduction is what differentiates True Colour systems from those with lower color depths, which would have limited color representation and could result in images that appear less realistic or visually appealing.

4. What is the purpose of a control character in computing?

- A. To display graphics on the screen**
- B. To execute program commands**
- C. To represent data in a readable form**
- D. To perform non-printing actions**

A control character serves a specific purpose in computing by directing the operation of hardware or software without representing a visual symbol on the screen. Control characters are used to manage data streams and perform tasks such as formatting text, initiating commands, controlling devices, or signaling the end of a message. They do not produce a visible output but execute functions that are essential for proper data handling and communication between devices. For instance, a common control character is the line feed, which moves the cursor to the next line without printing a character. This demonstrates how control characters can influence how information is processed and displayed, but they do so behind the scenes rather than by providing readable data or graphics.

5. What is the role of a web browser?

- A. To run backend scripts on a server
- B. To retrieve and display web content**
- C. To manage databases
- D. To create web content

The role of a web browser is fundamentally about retrieving and displaying web content. A web browser acts as a client-side application that interacts with web servers to fetch HTML documents, images, stylesheets, and other resources. When you enter a URL or click on a link, the browser sends a request to the server where the site is hosted. Once the server responds with the necessary files, the browser interprets this data, rendering the web page on your screen in a format that is visually comprehensible to users. This process includes interpreting HTML, CSS, and potentially JavaScript to provide a complete and interactive user experience on the web. In contrast, options that suggest running backend scripts, managing databases, or creating web content pertain to different roles typically handled by web servers, database management systems, or web development tools, rather than the main function of a web browser itself. A web browser's primary focus is on the user's interaction with content, emphasizing retrieval and display.

6. What type of data is typically represented in a character set?

- A. Only numerical values
- B. Only alphabetic characters
- C. A combination of letters, numbers, and symbols**
- D. Only punctuation marks

The correct answer represents the nature of a character set comprehensively. A character set is designed to include a wide range of characters that can represent text in a computer system. This encompasses alphabetic characters (both uppercase and lowercase letters), numerical values (digits from 0 to 9), as well as various symbols and punctuation marks (like \$, %, &, !, etc.). For instance, the ASCII character set includes not only the letters and digits but also special characters such as space, punctuation marks, and control characters. This inclusive approach allows computers to handle a variety of textual data, making it possible to display text accurately across different applications and platforms. Hence, the representation of characters in a character set is multifaceted, allowing for a rich interaction with a variety of text forms.

7. Which of the following is an example of a high-level programming language?

- A. C
- B. Python**
- C. Assembly
- D. Machine Code

High-level programming languages are designed to be easy for humans to read and write, abstracting away much of the complexity of hardware. They allow programmers to focus on writing algorithms without needing to manage hardware details or memory directly. Python is a prime example of a high-level programming language because it uses clear and concise syntax, supports multiple programming paradigms, and comes with a rich set of libraries that simplify complex tasks. This makes it user-friendly and accessible, especially for beginners. In contrast, other options like C are often considered lower-level than Python, as they require more understanding of system operations and memory management, though they are still higher-level than assembly or machine code. Assembly language is a low-level language that is closely tied to machine code and architecture, making it less abstract and harder to read. Machine code consists of binary instructions that a computer's CPU executes directly, making it the lowest level of programming and not human-readable at all.

8. What advantage does modularity offer in software development?

- A. It ensures faster data processing
- B. It allows for easier updates and maintenance**
- C. It reduces the need for user manuals
- D. It increases the complexity of the code

Modularity in software development refers to the practice of dividing a program into separate, distinct components or modules, each performing a specific function. The primary advantage of modularity is that it simplifies updates and maintenance. When a program is broken down into smaller, manageable modules, developers can modify or replace individual parts of the software without affecting the overall functionality. This approach allows teams to work on different modules simultaneously, which enhances productivity and speeds up development. Furthermore, if a specific module needs to be updated or a bug needs to be fixed, developers can do so quickly without having to sift through the entire codebase. Such isolation of components not only facilitates easier troubleshooting but also allows for targeted testing, ensuring that changes can be validated effectively. In the context of the other options, faster data processing can be influenced by many factors unrelated to modularity, such as algorithm efficiency or hardware capabilities. The reduction of user manuals is not a direct advantage of modularity; while it might lead to better-structured code that is easier to use, it does not inherently eliminate the need for documentation. Lastly, modularity is known for reducing complexity by breaking a large and complex system into smaller, more understandable parts, rather than increasing the complexity of the code. This organization helps developers

9. What is the primary function of an Address Bus in computing?

- A. A uni directional bus that specifies the address location in memory to be read from/written to**
- B. A communication channel for data transfer between processor and peripherals**
- C. A bus used exclusively for power distribution to hardware components**
- D. A protocol for addressing network devices**

The primary function of an Address Bus in computing is to specify the address location in memory that the processor intends to read from or write to. This bus is typically unidirectional, meaning it carries information in one direction only, from the processor to the memory. When the CPU needs to access a particular memory location, it sends the corresponding address along the address bus. This is crucial for the correct operation of a computer, as it ensures that data is accurately retrieved from or stored to the right memory location. In contrast, the other options present functions that do not pertain to the role of an address bus. The choice that describes a communication channel for data transfer relates more to data buses, which facilitate the movement of actual data between components rather than specifying addresses. The option regarding power distribution concerns electrical buses, which are not responsible for data transmission or addressing. Lastly, the choice that refers to a protocol for network devices pertains to networking and device communication, not the underlying hardware architecture of how data is stored and accessed in memory systems.

10. Which of the following accurately describes Base 2?

- A. A numbering system using digits 0 to 5**
- B. A numbering system that only has the digits 0 and 1**
- C. A system that includes both positive and negative numbers**
- D. A decimal system commonly used in everyday counting**

Base 2, also known as the binary numbering system, accurately describes a system that only uses the digits 0 and 1. This is fundamental to how binary numbers work, as each digit in a binary number represents a power of 2, and only these two digits are employed to construct any number within this system. For example, the binary number 101 in Base 2 can be interpreted as $1 \times (2^2) + 0 \times (2^1) + 1 \times (2^0)$, which equates to $4 + 0 + 1$, resulting in the decimal value 5. The simplicity of having only two digits makes binary efficient for computer systems, as it aligns perfectly with the on/off states found in electronic circuits. Other options do not define Base 2 correctly. A system using digits 0 to 5 would represent Base 6, while a system including both positive and negative numbers refers more to signed numbers or possibly the representation of numbers in systems like Two's Complement. Lastly, the decimal system, which is Base 10, is what we commonly use in everyday counting and involves digits from 0 to 9.

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://sqanational5computingscience.examzify.com>

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