

scholareye.xyz

9th Class

Computer Science & Entrepreneurship

CHAPTER 04

SYSTEM TROUBLESHOOTING

Complete Exercise Solutions

MCQs • Short Questions • Long Questions

Step-by-step Answers • Highlighted MCQs • Explanations

www.scholareye.xyz

EXERCISE - Multiple Choice Questions (MCQs)

1. What is the first step in the systematic process of troubleshooting?

- A) Establish a Theory of Probable Cause
- B) Implement the Solution
- C) Identify Problem ✓**
- D) Document Findings, Actions, and Outcomes

✓ **Correct Answer: C**

Explanation: The very first step in any troubleshooting methodology is to identify the problem. You cannot solve an issue you haven't clearly defined or recognized. This involves gathering information, questioning the user, and identifying the symptoms before jumping to conclusions or attempting fixes.

2. Why is effective troubleshooting important for maintaining systems?

- A) It helps save money on repairs
- B) It prevents the need for professional help
- C) It ensures systems operate smoothly and efficiently ✓**
- D) It allows for more frequent system updates

✓ **Correct Answer: C**

Explanation: While saving money and avoiding professional help can be side effects of good troubleshooting, the primary and most important goal is to ensure the overall health, smooth operation, and efficiency of the system. Minimizing downtime and maintaining productivity are the ultimate objectives.

3. Which step involves coming up with a theory about what might be causing a problem?

- A) Test the Theory to Determine the Cause
- B) Establish a Theory of Probable Cause ✓**
- C) Implement the Solution
- D) Verify Full System Functionality

✓ **Correct Answer: B**

Explanation: Once the problem is identified, the next logical step is to brainstorm and formulate a hypothesis (a theory) about what is causing the issue. This step is literally called “Establish a Theory of Probable Cause” in standard troubleshooting frameworks.

4. After implementing a solution, what is the next step in the troubleshooting process?

- A) Document Findings, Actions, and Outcomes
- B) Test the Theory to Determine the Cause
- C) Verify Full System Functionality ✓**
- D) Establish a Plan of Action to Resolve the Problem

✓ **Correct Answer: C**

Explanation: After applying a fix, you must verify that the system is fully operational and that the solution didn't cause any new problems (regressions). Documentation (Option A) is the final step, coming *after* verification.

5. Which of the following is an example of identifying a problem in troubleshooting?

- A) Testing a laptop battery by plugging in the power cord
- B) Coming up with a plan to replace a laptop battery
- C) Noticing that a laptop does not turn on when the power button is pressed ✓**
- D) Writing down that a laptop battery was replaced

✓ **Correct Answer: C**

Explanation: Identifying the problem is about recognizing the symptom. Noticing the laptop won't turn on is the observation of the issue. Option A is testing a theory, Option B is planning, and Option D is documentation.

6. Why is documenting findings, actions, and outcomes important in troubleshooting?

- A) It helps solve problems faster
- B) It provides a record for future reference ✓**
- C) It allows for more efficient testing
- D) It ensures the solution is implemented correctly

✓ **Correct Answer: B**

Explanation: The primary purpose of documentation is to create a knowledge base. If the same problem occurs again, or a similar issue arises, having a record of what was done and what worked provides a massive shortcut for future reference.

7. What is the purpose of establishing a plan of action in troubleshooting?

- A) To identify the problem
- B) To verify full system functionality
- C) To determine the cause of the problem
- D) To decide on the steps needed to resolve the issue ✓**

✓ **Correct Answer: D**

Explanation: Once a probable cause is identified (and possibly tested), you establish a plan of action. This step is purely about strategizing and outlining the exact steps you will take to fix the problem before you actually start making changes.

8. Why is troubleshooting important in computing systems?

- A) It ensures hardware components are always up to date
- B) It prevents the need for data backups
- C) It helps keep systems running smoothly and securely ✓**
- D) It eliminates the need for software updates

✓ **Correct Answer: C**

Explanation: Troubleshooting is vital for minimizing downtime, resolving bugs, and maintaining security, which all contribute to keeping systems running smoothly. It does not, however, replace the need for routine maintenance like backups or updates.

9. What does troubleshooting help prevent by quickly identifying and resolving issues?

- A) The need for professional help
- B) The need for software updates
- C) Downtime and lost productivity ✓**
- D) The need for regular maintenance

✓ **Correct Answer: C**

Explanation: The biggest enemy of a business or user is downtime. Quick identification and resolution of issues directly minimize the amount of time a system is unusable, thereby preventing lost productivity.

10. Which of the following is an example of ensuring data integrity through troubleshooting?

- A) Identifying a software bug that causes incorrect database results ✓**
- B) Replacing a faulty printer
- C) Using a cooling pad to prevent laptop overheating
- D) Updating the operating system regularly

✓ **Correct Answer: A**

Explanation: Data integrity means the data is accurate, consistent, and uncorrupted. Finding and fixing a bug that produces incorrect database results directly protects the integrity of that data. The other options relate to hardware maintenance or general updates.

Short Questions

- 1. What is the first step in the systematic process of troubleshooting, and why is it important?**

Answer: The first step is **Identifying the Problem**. It is crucial because you cannot effectively fix what you do not understand. This step involves gathering information from the user, identifying symptoms, and determining exactly what is not working. If you skip this or misidentify the problem, all subsequent steps (theories, solutions) will be based on a flawed premise, wasting time and potentially causing further system damage.

2. After identifying a problem, what is the next step in troubleshooting, and how does it help in resolving the issue?

Answer: The next step is to **Establish a Theory of Probable Cause**. This helps resolve the issue by allowing the technician to narrow down the infinite possibilities of what *could* be wrong into a targeted hypothesis. By questioning the obvious and researching known issues, the technician can predict the root cause, which allows them to formulate a specific plan of action rather than blindly guessing.

3. Describe the importance of testing a theory during the troubleshooting process. Provide an example.

Answer: Testing a theory is important because a hypothesis is just an educated guess. Testing confirms whether the suspected cause is actually the root cause before significant resources are spent on a fix.

Example: If a user reports a laptop won't turn on, you form a theory that the battery is dead. To test this theory, you plug the laptop into a known working power outlet. If it turns on, your theory is confirmed. If it doesn't, your theory is disproven, and you must form a new one (e.g., the power adapter is faulty, or the motherboard is dead).

4. Explain what the “Implement the Solution” step entails in troubleshooting.

Answer: The “Implement the Solution” step entails actually executing the plan of action developed in the previous step. This involves making the necessary changes to the system, such as replacing a hardware component, changing a software configuration, running a malware removal tool, or applying a patch, to resolve the identified root cause. It is the active intervention phase of troubleshooting.

5. Why is it necessary to verify full system functionality after implementing a solution?

Answer: Verifying full system functionality is necessary to ensure that the implemented solution actually fixed the original problem, and crucially, that it did not introduce any new problems (known as regressions). A fix for one component might inadvertently break another. Verification ensures the system is fully operational, stable, and secure across all its normal functions before declaring the issue resolved.

Long Questions

- 1. Discuss the importance of troubleshooting in maintaining the smooth operation of systems, especially computing systems.**

Answer: Troubleshooting is the backbone of system maintenance. In computing systems, where hardware and software interact in highly complex ways, issues are inevitable. Effective troubleshooting minimizes downtime, which is critical for businesses where every minute of system failure translates to lost revenue and productivity. It ensures data integrity by quickly identifying and resolving corrupting bugs. Furthermore, it enhances security by identifying vulnerabilities or malware before they can cause catastrophic damage. Ultimately, a structured troubleshooting approach transforms chaotic technical failures into manageable, solvable incidents, extending the lifespan of hardware and software assets and ensuring a reliable user experience.

- 2. Explain the systematic process of troubleshooting. Describe each step in detail.**

Answer: The systematic process of troubleshooting typically follows these steps:

1. **Identify the Problem:** Gather information, ask the user questions, and identify the exact symptoms.
2. **Establish a Theory of Probable Cause:** Question the obvious and research the issue to form a hypothesis about the root cause.
3. **Test the Theory to Determine the Cause:** Once a theory is formed, test it. If confirmed, move to the next step. If not, form a new theory or escalate.
4. **Establish a Plan of Action to Resolve the Problem:** Outline the exact steps and resources needed to fix the issue.
5. **Implement the Solution:** Execute the plan, ensuring minimal disruption to the user.
6. **Verify Full System Functionality:** Ensure the fix worked and didn't break anything else. Implement preventive measures if necessary.
7. **Document Findings, Actions, and Outcomes:** Record everything from the initial symptom to the final solution for future reference.

3. Using a case study where a printer is not printing, explain how you would identify the problem and establish a theory of probable cause.

Answer: Case Study: A user complains their printer is not printing.

Identify the Problem: I would ask the user questions: Is the printer turned on? Are there error lights? Does it happen with all documents? I would observe the printer and the computer. I would conclude the symptom: "The printer is unresponsive to print commands."

Establish a Theory of Probable Cause: Based on the symptoms, I would form several theories: 1) The printer is offline or disconnected from the network/USB. 2) There is a paper jam or the ink/toner is empty. 3) The print spooler service on the computer has crashed. 4) The wrong printer drivers are installed. I would start with the simplest theory (connection/power) and move to the more complex ones (drivers/spooler).

4. Discuss the importance of documenting findings, actions, and outcomes during the troubleshooting process.

Answer: Documentation serves as the institutional memory of technical support. Without it, technicians waste time reinventing the wheel when a recurring problem arises. By documenting the initial symptoms, the steps taken, and the final outcome, a searchable knowledge base is created. This allows other technicians to resolve the same issue in a fraction of the time. It also provides a historical record for the specific machine, which is invaluable for identifying chronic hardware failures (e.g., a laptop that has had three hard drive failures). Furthermore, documentation provides accountability and can reveal patterns indicating a need for larger system upgrades or user training.

5. Analyze the various ways troubleshooting is vital in computing systems, particularly in preventing downtime, ensuring data integrity, and improving security. Provide specific examples and scenarios to support your analysis.

- **Answer: Preventing Downtime:** A systematic approach allows for rapid resolution of critical failures. *Example:* If a server crashes, a technician uses the OS event logs to pinpoint a faulty memory module. Replacing it quickly restores the server, keeping the business online.

- **Ensuring Data Integrity:** Bugs or hardware failures can corrupt files. *Example:* A database is returning incorrect inventory numbers. Troubleshooting reveals a software bug in the latest update. Rolling back the update fixes the data processing, ensuring data integrity.
- **Improving Security:** Troubleshooting isn't just about broken hardware; it's about identifying anomalies. *Example:* A user complains their computer is running very slowly. Troubleshooting reveals unauthorized background processes and high network usage. This leads to the discovery of malware, which is then removed, preventing a potential data breach.

6. Describe basic software-related troubleshooting strategies, including handling application freezing and unresponsive peripherals.

Answer: For software freezes, the first step is often to force-quit the application (e.g., via Task Manager or Force Quit). If the entire system is frozen, a “hard reboot” (holding the power button) may be necessary, though this risks data loss. After restarting, check for software updates, as freezes often stem from bugs.

For unresponsive peripherals, first check physical connections (cables, ports). If that fails, check the Device Manager (Windows) or System Information (macOS) to see if the OS recognizes the device. Reinstalling or rolling back the device drivers is a common strategy. Finally, checking for conflicting software or recent OS updates that may have broken compatibility is crucial.

7. Explain how to recognize hardware failures, particularly focusing on RAM and hard drive issues.

- **Answer: RAM Issues:** Symptoms of failing RAM include frequent random Blue Screens of Death (BSOD), system freezes, application crashes, and memory errors. A telltale sign is when a computer fails to boot or beeps repeatedly during the Power-On Self-Test (POST). Tools like MemTest86 can be used to diagnose specific RAM errors.

- **Hard Drive Issues:** Symptoms include strange clicking, grinding, or whirring noises (physical failure). Logically, users may experience extremely slow file access, frequent “file not found” errors, and corrupted files. The system may take a very long time to boot or hang. Tools like S.M.A.R.T. (Self-Monitoring, Analysis, and Reporting Technology) can predict imminent drive failure.

8. Elaborate on the importance of maintaining software and addressing security threats.

Answer: Software maintenance is not just about getting new features; it is a critical security necessity. Software vendors constantly release patches to fix newly discovered vulnerabilities. If a system is not updated, it remains exposed to known exploits, making it an easy target for malware, ransomware, and hackers. Maintaining software also ensures compatibility with other updated applications and hardware. Addressing security threats promptly (like installing antivirus definitions and OS security patches) creates a layered defense. Failure to do so can lead to catastrophic data breaches, financial loss, and reputational damage for an organization.

9. Describe common methods for identifying and removing malware infections and applying operating system updates for security.

Answer: Identifying malware involves looking for symptoms like sudden performance drops, unexpected pop-ups, browser redirects, unknown programs running, or disabled security tools. Technicians use tools like Task Manager to spot suspicious processes and network monitors to detect unusual outbound traffic.

Removal involves running a trusted anti-malware/antivirus scan in “Safe Mode” to prevent the malware from hiding. If the infection is severe, a clean OS reinstall may be required. For OS updates, technicians should enable automatic updates where possible, or manually check for updates regularly. Critical security patches should be applied immediately, while feature updates can be scheduled for off-hours to avoid disruption.

10. Describe the different data backup methods, including using external storage devices and cloud solutions.

Answer: Data backup methods are generally categorized into physical and cloud-based.

- **External Storage Devices:** This includes external hard drives, SSDs, and USB flash drives. They offer fast local backups and full control over the data. However, they are vulnerable to physical damage, theft, or loss if kept in the same location as the main system (e.g., a fire or flood destroys both).
- **Cloud Solutions:** Services like Google Drive, Dropbox, or AWS backup data to remote servers. This protects data from local disasters and allows access from anywhere. However, it requires a stable internet connection, may incur ongoing subscription costs, and raises data privacy concerns.
- **Best Practice:** The industry standard is the **3-2-1 rule**: Keep at least 3 copies of data, on 2 different types of media, with 1 copy stored offsite (e.g., cloud or a secure remote location).

scholareye.xyz

Thank You!

Keep Learning, Keep Growing

What You Covered

- ✓ 10 MCQs with highlighted answers
- ✓ 5 Short Questions
- ✓ 10 Long Questions

www.scholareye.xyz

Study smart. Revise well. Succeed.