# 1 Introduction to Computers

## REVIEW QUESTIONS

1. Computer software is divided into two broad categories: system software and operational software.

   b. False

3. The first step in system development is to create a source program.

   b. False

5. Blackbox testing gets its name from the concept that the program is being tested without knowing how it works.

   a. True

7. Which of the following is an example of application software?

   a. Database management system

9. The computer language that most closely resembles machine language is

   a. Assembly/symbolic

11. The _________ contains the programmer's original program code.

    b. Source file

13. The _________ is a program design tool that is a visual representation of the logic in a function within a program.

    a. Flowchart

## EXERCISES

15. The two major components of a computer system are hardware and software. The hardware component of a computer system is made of five parts: The input devices, central processing unit (CPU), primary storage or main memory, output devices, and auxiliary storage devices. The software consists of system software, which includes the operating system, and application software used to solve the user's business requirements.

17. In a time sharing environment, each user has a terminal that does not have any processing capability; all processing is done by the central computer. In a client-server environment, users have terminals that have programming capabilities and a portion of the processing is done by the terminal-workstation and a portion is done by the central computer.

19. The operating system provides system services such as a user interface, file and database access, and communication services. Its primary purpose is system efficiency while providing a user interface to the hardware and applications.

21. General-purpose software can be used for more than one purpose. Examples include word processors, spreadsheets, and database management system. Application-specific software solves a specific business problem and can not be used for other purposes. Examples include personal finance systems and general ledger accounting systems.

23. Symbolic languages, often called assembly languages, provide mnemonics for machine instructions, data identifiers, and other objects such as functions. They allow the programmer to write program instructions that basically mirror the machine instructions. High-level languages, on the other hand, are machine independent. They allow for portability and let the user concentrate on the problem being solved rather than the hardware for which the software is being written. Generally, each high-level language statement generates many machine language statements.

25. The system development life cycle contains six steps:
    - a. System requirements: Gather user requirements, the “what”.
    - b. Analysis: Evaluate alternative solutions to the requirements.
    - c. Design: Describe the specific implementation for the problem, the “how”.
    - d. Code: Prepare and unit test programs based on the design.
    - e. System Test: Verify that the programs integrate and work as a system to satisfy the user requirements.
    - f. Maintenance: Keep the system working in production.

27. The four steps in program development are:
    - a. Understand the problem.
    - b. Develop a solution using structure charts and either flowcharts or pseudocode.
    - c. Write the program.
    - d. Test the program.

29. “Resist the temptation to code” means that the programmer must fully understand the problem and design a solution before beginning to write code. It is human nature to want to get to the coding step as soon as possible, but this often leads to poorly implemented and inefficient programs.

31. Software engineering is the use of sound engineering methods and principles to develop software that works.

## PROBLEMS

33. No standard answer.

83. No standard answer.