Chapter 13

System Implementation and Operation

Computer Fundamentals - Pradeep K. Sinha & Priti Sinha

Learning Objectives

In this chapter you will learn about:

- Main activities of implementation and operation phase
- Testing and debugging of programs
- Complete documentation of the system
- Change over to the new system
- System evaluation and
- System maintenance

Testing and Debugging

- Program errors are known as *bugs*
- Process of detecting and correcting these errors is called *debugging*
- *Testing* is the process of making sure that the program performs the intended task
- *Debugging* is the process of locating and eliminating program errors

Types of Program Errors

- *Syntax errors*
  - Occurs when the rules or syntax of the programming language are not followed
  - For example, incorrect punctuation, incorrect word sequence, undefined terms, and misuse of terms
  - Syntax errors are detected by a language processor
- *Logic errors*
  - Occurs due to errors in planning a program's logic
  - Such errors cause the program to produce incorrect output.
  - These errors cannot be detected by a language processor

Testing of a Program

- Testing procedure involves running program to process input test data, and comparing obtained results with correct results
- Test data must test each logical function of the program, and should include all types of possible valid and invalid data
- Program internally released for testing is known as *alpha version* and the test conducted on it is called *alpha testing*
- Program released for additional testing to a selected set of external users is *beta version* and test conducted on it called is *beta testing*

Debugging a Program for Syntax Errors

- Relatively easier to detect and correct syntax errors than logic errors in a program
- Language processors are designed to automatically detect syntax errors
- Single syntax error often causes multiple error messages to be generated by the language processor
- Removal of the syntax error will result in the removal of all associated error messages

Debugging a Program for Logic Errors

- Logic errors are more difficult to detect than syntax errors as computer does not produce any error message for such errors
- One or more of following methods are commonly used for locating logic errors:
  - Doing hand simulation of the program code
  - Putting print statements in the program code
  - Using a debugger (a software tool that assists a programmer in following the program's execution step-by-step)
  - Using memory dump (printout of the contents of main memory and registers)

Difference Between Testing and Debugging

Sr. No.	Testing	Debugging
1	Testing is the process of validating the correctness of a program Its objective is to demonstrate that the program meets its design specifications	Debugging is the process of eliminating errors in a program Its objective is to detect the exact cause and remove known errors in the program
2	Testing is complete when all desired verifications against specifications have been performed	Debugging is complete when all known errors in the program have been fixed Note that debugging process ends only temporarily as it must be restarted whenever a new error is found in the program

(Continued on next slide)

Difference Between Testing and Debugging

(Continued from previous slide..)

Sr. No.	Testing	Debugging
3	Testing is a definable process which can and should be planned and scheduled properly	Debugging being a reactive process cannot be planned ahead of time It must be carried out as and when errors are found in a program
4	Testing can begin in the early stages of software development. Although the test runs of a program can be done only after the program is coded, but the decision of what to test, how to test, and with what kind of data to test, can and should be done before the coding is started	Debugging can begin only after the program is coded The approach used for debugging largely depends on the personal choice of the programmer and the type of problem in the program

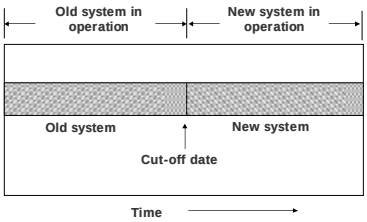
Documentation

- Process of collecting, organizing, storing, and otherwise maintaining a complete historical record of programs and other documents used or prepared during the different phases of the life cycle of a software
- Three commonly used forms of documentation are:
  - Program comments
  - System manual
  - User manual

Changeover to the New System

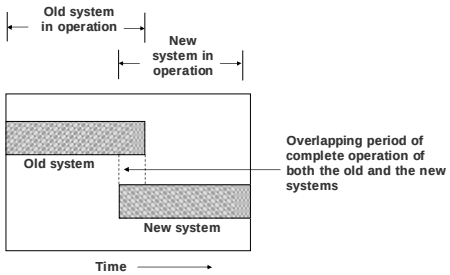
- When a software is ready for use, it is deployed at site for use by the intended users
- At this stage, a changeover from the old system of operation to the new system takes place
- Three normally followed methods to carry out the changeover process are:
  - Immediate changeover
  - Parallel run
  - Phased conversion

Changeover to the New System



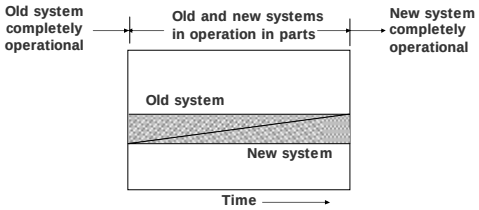
(a) Immediate changeover

Changeover to the New System



(b) Parallel run

Changeover to the New System



(c) Phased conversion

System Evaluation

- Process of evaluating a system (after it is put in operation) to verify whether or not it is meeting its objectives
- Points normally considered for evaluating a system are:
  - Performance evaluation
  - Cost analysis
  - Time analysis
  - User satisfaction
  - Ease of modification
  - Failure rate

System Maintenance

- Process of incorporating changes in an existing system to enhance, update, or upgrade its features
- On an average, maintenance cost of a computerized system is two to four times more than the initial development cost

Key Words/Phrases

- Beta testing
- Bugs
- Comments
- Debugger
- Debugging
- Documentation
- Immediate changeover
- Logic errors
- Memory dump
- Parallel run
- Phased conversion
- Syntax errors
- System evaluation
- System maintenance
- System manual
- Testing
- User manual