10.4 TESTING

The aim of program testing is to help realiseidentify all defects in a program. However, in practice, even after satisfactory completion of the testing phase, it is not possible to guarantee that a program is error free. This is because the input data domain of most programs is very large, and it is not practical to test the program exhaustively with respect to each value that the input can assume. Consider a function taking a floating point number as argument. If a tester takes 1sec to type in a value, then even a million testers would not be able to exhaustively test it after trying for a million number of years. Even with this obvious limitation of the testing process, we should not underestimate the importance of testing. We must remember that careful testing can expose a large percentage of the defects existing in a program, and therefore provides a practical way of reducing defects in a system.

DR→ After the implementation phase, the testing phase begins. Testing of software is critical, since testing determines the correctness, completeness and quality of the software being developed. Its main objective is to detect errors in the software. Errors prevent software from producing outputs according to user requirements. They occur if some part of the developed system is found to be incorrect, incomplete, or inconsistent. Errors can broadly be classified into three types, namely, *requirements errors*, *design errors*, and *programming errors*. To avoid these errors, it is necessary that: requirements are examined for conformance to user needs, software design is consistent with the requirements and notational convention, and the source code is examined for conformance to the requirements specification, design documentation and user expectations. All this can be accomplished through efficacious means of software testing.

The activities involved in testing phase basically evaluate the capability of the developed system and ensure that the system meets the desired requirements. It should be noted that testing is fruitful only if it is performed in the correct manner. Through effective software testing, the software can be examined for correctness, comprehensiveness, consistency and adherence to standards. This helps in delivering high-quality software products and lowering maintenance costs, thus leading to more contented users.

6.6 LEVELS OF SOFTWARE TESTING

As mentioned earlier, the software is tested at different levels. Initially, the individual units are tested and once they are tested, they are integrated and checked for interfaces established between them. After this, the entire software is tested to ensure that the output produced is according to user requirements. As shown in Figure 6.8, there are four levels of software testing, namely, *unit testing, integration testing, system testing*, and *acceptance testing*.

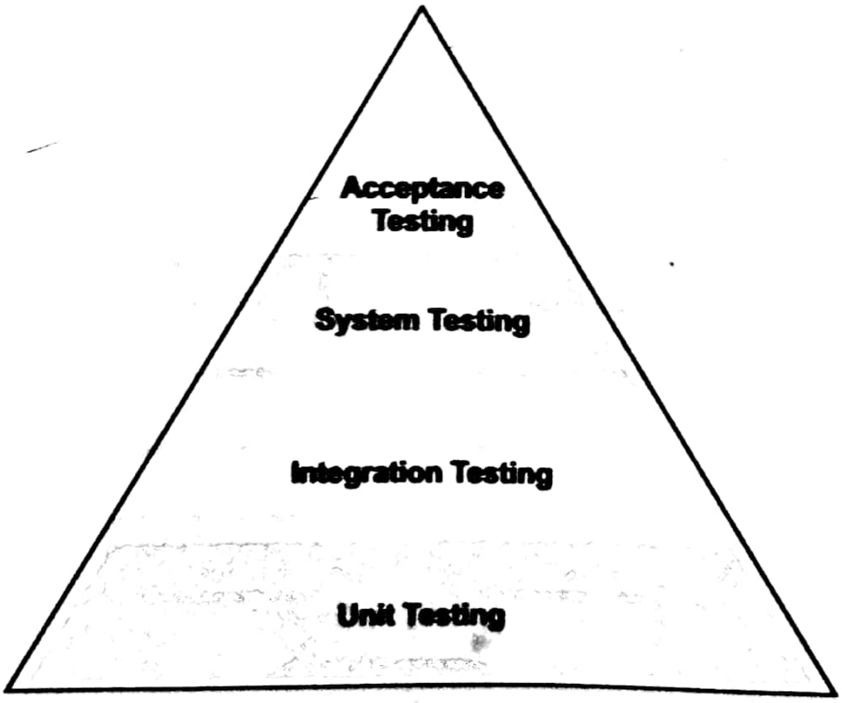

Figure 6.8 Levels of Software Testing