

TESTING, MAINTENANCE AND AUDITING

Dr. Krishnendu Guha

Assistant Professor (On Contract)

National Institute of Technology (NIT), Jamshedpur

Email: krishnendu.ca@nitjsr.ac.in

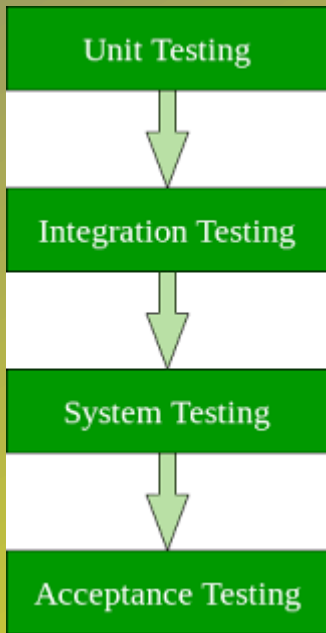
INTRODUCTION



- ▶ Once the analyst has designed and coded the system, then
- ▶ testing,
- ▶ maintenance, and
- ▶ auditing of it are prime considerations.



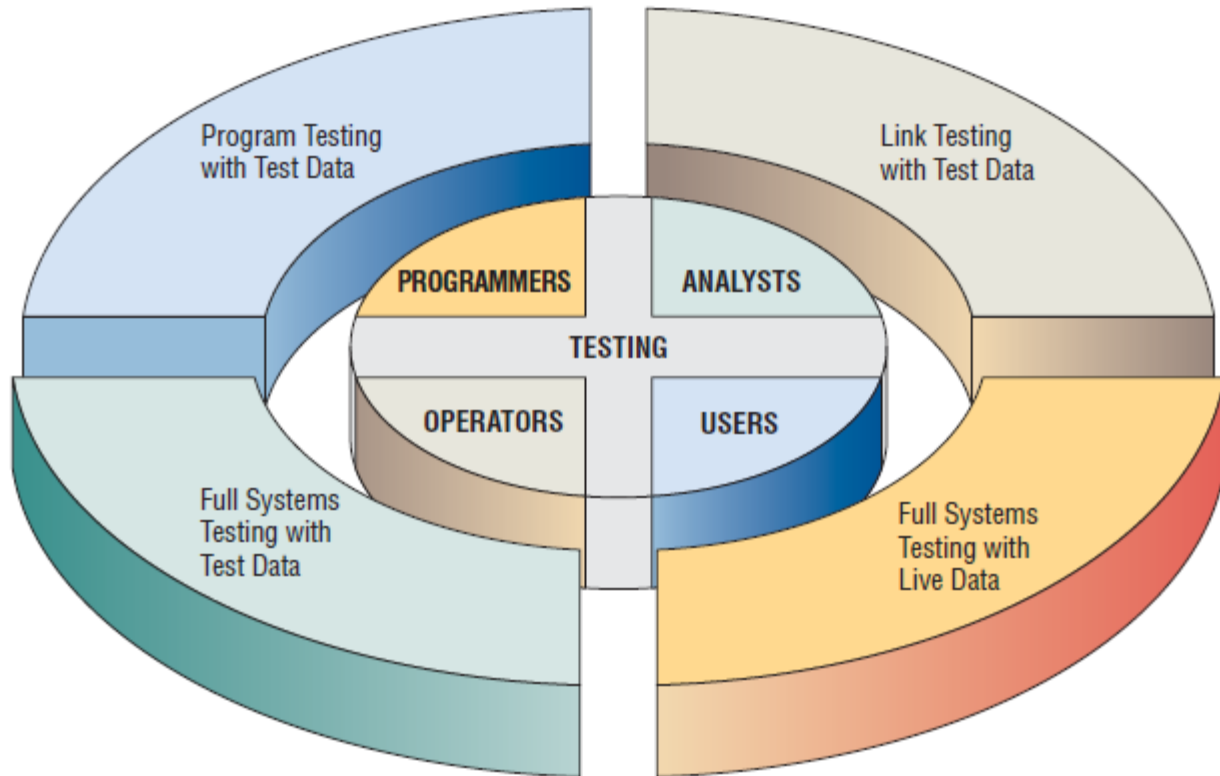
THE TESTING PROCESS



```
test_unittest.py x
1 import unittest
2 import inc_dec
3
4 class TestIncrementDecrement(unittest.TestCase):
5     def test_increment(self):
6         self.assertEqual(inc_dec.Increment(3), 4)
7
8     def test_decrement(self):
9         self.assertEqual(inc_dec.Decrement(3), 4)
10
11 if __name__ == '__main__':
12     unittest.main()
13
```



- ▶ All the system's newly written or modified application programs—as well as new procedural manuals, new hardware, and all system interfaces—must be tested thoroughly.
- ▶ Haphazard, trial and- error testing will not suffice.
- ▶ Testing is done throughout systems development, not just at the end.
- ▶ It is meant to turn up heretofore unknown problems, not to demonstrate the perfection of programs, manuals, or equipment.
- ▶ Although testing is tedious, it is an essential series of steps that helps ensure the quality of the eventual system.
- ▶ It is far less disruptive to test beforehand than to have a poorly tested system fail after installation.
- ▶ Testing is accomplished on subsystems or program modules as work progresses.



- ▶ Programmers, analysts, operators, and users all play different roles in the various aspects of testing
- ▶ Testing is done on many different levels at various intervals.
- ▶ Before the system is put into production, all programs must be desk checked, checked with test data, and checked to see if the modules work together with one another as planned.
- ▶ The system as a working whole must also be tested. Included here are testing the interfaces between subsystems, the correctness of output, and the usefulness and understandability of systems documentation and output.
- ▶ Testing of hardware is typically provided as a service by vendors of equipment, who will run their own tests on equipment when it is delivered onsite.

PROGRAM TESTING WITH TEST DATA

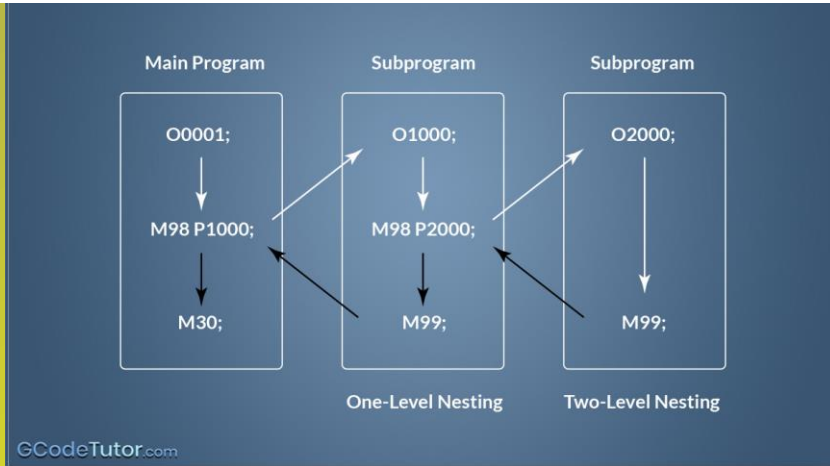


- ▶ Much of the responsibility for program testing resides with the original author(s) of each program.
- ▶ The systems analyst serves as an advisor and coordinator for program testing.
- ▶ At this stage, programmers must first desk check their programs to verify the way the system will work.
- ▶ In desk checking, the programmer follows each step in the program on paper to check whether the routine works as it is written.
- ▶ Next, programmers must create both valid and invalid test data.
- ▶ These data are then run to see if base routines work and also to catch errors.
- ▶ If output from main modules is satisfactory, you can add more test data so as to check other modules.
- ▶ Created test data should test possible minimum and maximum values as well as all possible variations in format and codes.
- ▶ Throughout this process, the systems analyst checks output for errors, advising the programmer of any needed corrections.

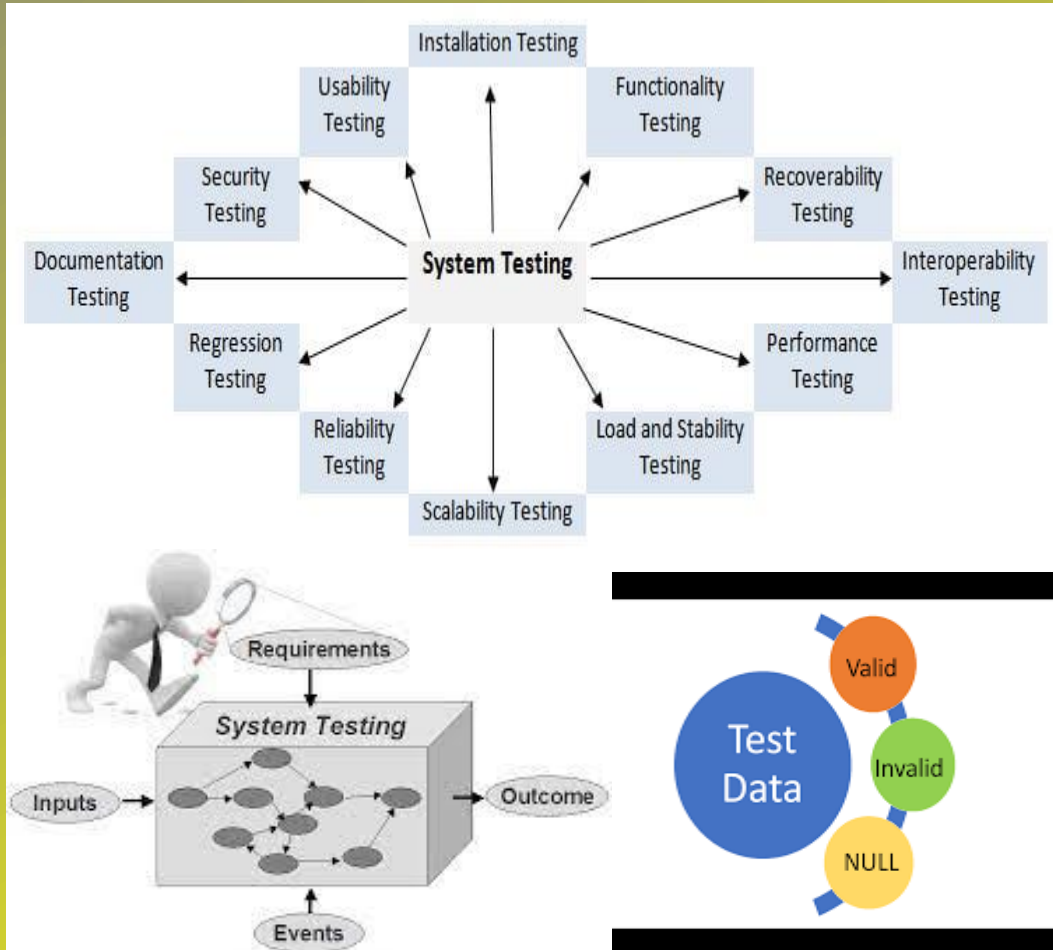
LINK TESTING WITH TEST DATA



- ▶ When programs pass desk checking and checking with test data, they must go through link testing, which is also referred to as string testing.
- ▶ Link testing checks to see if programs that are interdependent can actually work together as planned.
- ▶ The analyst creates special test data that cover a variety of processing situations for link testing.
- ▶ First, typical test data are processed to see if the system can handle normal transactions, those that would make up the bulk of its load.
- ▶ If the system works with normal transactions, variations are added, including invalid data used to ensure that the system can properly detect errors.



FULL SYSTEMS TESTING WITH TEST DATA



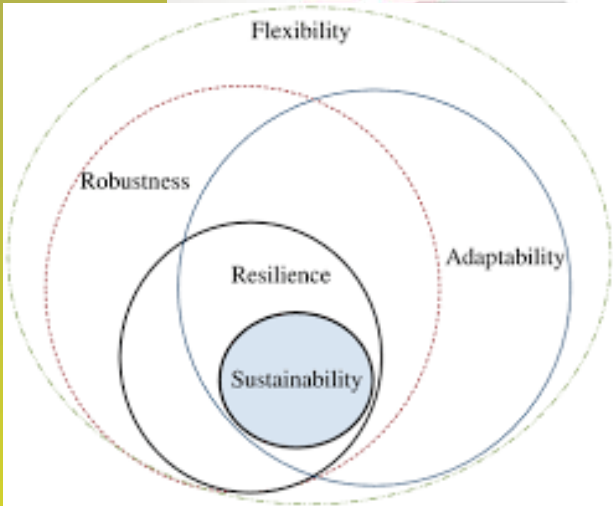
- ▶ When link tests are satisfactorily concluded, the system as a complete entity must be tested.
- ▶ Test data, created by the systems analysis team for the express purpose of testing system objectives, are used.
- ▶ As can be expected, there are a number of factors to consider:
 - ▶ **1.** Examining whether operators have adequate documentation in procedure manuals (hard copy or online) to afford correct and efficient operation.
 - ▶ **2.** Checking whether procedure manuals are clear enough in communicating how data should be prepared for input.
 - ▶ **3.** Ascertaining if work flows necessitated by the new or modified system actually “flow.”
 - ▶ **4.** Determining if output is correct and whether users understand that this output is, in all likelihood, as it will look in its final form
- ▶ Systems testing includes reaffirming the quality standards for system performance that were set up when the initial system specifications were made.

FULL SYSTEMS TESTING WITH LIVE DATA



- ▶ When systems tests using test data prove satisfactory, it is a good idea to try the new system with several passes on what is called live data, data that have been successfully processed through the existing system.
- ▶ This step allows an accurate comparison of the new system's output with what you know to be correctly processed output, as well as a good idea for testing how actual data will be handled.
- ▶ Obviously, this step is not possible when creating entirely new outputs (for instance, output from an ecommerce transaction from a new corporate Web site).
- ▶ As with test data, only small amounts of live data are used in this kind of system testing.
- ▶ Items to watch for are ease of learning the system and user reaction to system feedback
- ▶ Procedure manuals, just like computer software, also need to be tested, the only real way to test them is to have users and operators try them, preferably during full systems testing with live data. Consider user suggestions, and incorporate them into the final versions of Web pages, printed manuals, and other documentation.

MAINTENANCE PRACTICES

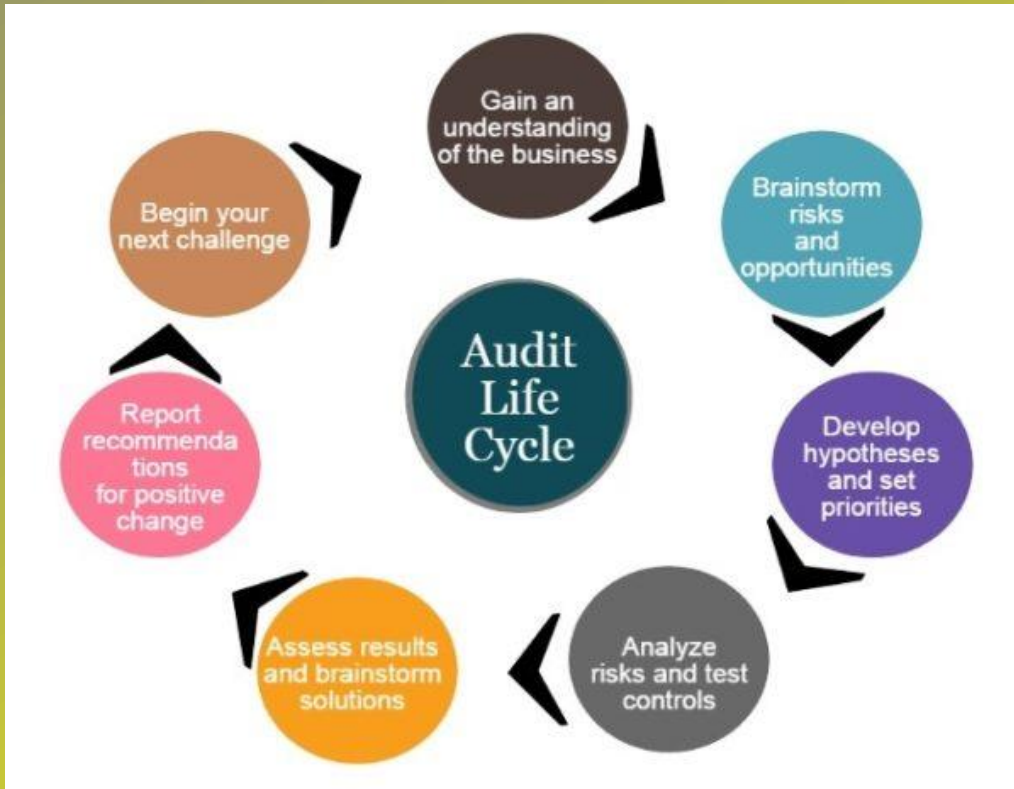


- ▶ Your objective as a systems analyst should be to install or modify systems that have a reasonably useful life.
- ▶ You want to create a system whose design is comprehensive and farsighted enough to serve current and projected user needs for several years to come.
- ▶ Part of your expertise should be used to project what those needs might be and then build flexibility and adaptability into the system.
- ▶ The better the system design, the easier it will be to maintain and the less money the business will have to spend on maintenance.
- ▶ Reducing maintenance costs is a major concern, because software maintenance alone can devour upward of 50 percent of the total data processing budget for a business



- ▶ Excessive maintenance costs reflect directly back on the system's designer, because approximately 70 percent of software errors have been attributed to inappropriate software design.
- ▶ From a systems perspective, it makes sense that detecting and correcting software design errors early on is less costly than letting errors remain unnoticed until maintenance is necessary.
- ▶ Maintenance is performed most often to improve the existing software rather than to respond to a crisis or system failure.
- ▶ Maintenance is also done to update software in response to the changing organization.
- ▶ Emergency and adaptive maintenance comprises less than half of all system maintenance.
- ▶ Part of the systems analyst's job is to ensure that there are adequate channels and procedures in place to permit feedback about—and subsequent response to—maintenance needs.
- ▶ Users must be able to communicate problems and suggestions easily to those who will be maintaining the system.
- ▶ Solutions are to provide users email access to technical support, as well as to allow them to download product updates or patches from the Web.

AUDITING



- ▶ Auditing is yet another way of ensuring the quality of the information contained in the system.
- ▶ Broadly defined, auditing refers to having an expert who is not involved in setting up or using a system examine information in order to ascertain its reliability.
- ▶ Whether or not information is found to be reliable, the finding on its reliability is communicated to others for the purpose of making the system's information more useful to them.
- ▶ For information systems, there are generally two kinds of auditors: internal and external.
- ▶ Whether both are necessary for the system you design depends on what kind of system it is.
- ▶ Internal auditors work for the same organization that owns the information system,
- ▶ External (also called independent) auditors are hired from the outside.

External Audit



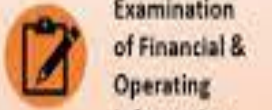
External Audit is the audit of the company's financial records in which the external auditors perform out the task of examining the validity of the company's financial records to evaluate whether there is any mistake in the records due to fraud, error or embezzlement and then reporting the same to the company's stakeholders.

- ▶ External auditors are used when the information system processes data that influences a company's financial statements.
- ▶ External auditors audit the system to ensure the fairness of the financial statements being produced.
- ▶ They may also be brought in if there is something out of the ordinary occurring that involves company employees, such as suspected computer fraud or embezzlement.

Internal Audit



Monitoring of Internal Control



Examination of Financial & Operating Information



Risk Management



Review of Compliance with Laws & Regulation



Governance



Review of Operating Activities

- ▶ Internal auditors study the controls used in the information system to make sure that they are adequate and that they are doing what they are purported to be doing.
- ▶ They also test the adequacy of security controls.
- ▶ Although they work for the same organization, internal auditors do not report to the people responsible for the system they are auditing.
- ▶ The work of internal auditors is often more in-depth than that of external auditors.

