Course Title:

System Analysis and Design (BSCCSIT)

BIT 4th semester-Tribhuvan University (TU)

Unit–5

Implementation and Maintenance

2

System Implementation

- It is the process of building properly working system, installing it in the organization, replacing the old system and working methods, and finalizing all user documentation and system documentation, and training and supporting the users to use the system.

- *System implementation is made up of many activities. Among them, the six major activities are as follows:*

- ✓ *Coding*

- ✓ *Testing*

- ✓ *Installation*

- ✓ *Documentation*

- ✓ *Training*

- ✓ *Support.*

❑ **Coding**

- Coding is the process where physical design specification created by the analysis team are turned into the working computer code by the programming team.

- Coding is generally performed by the programmers to ensure that programs meet the design specifications.

- The coding deliverables may be code, program documentation, etc.

❑ <u>Software Application Testing</u>

- Testing is the process of finding and fixing bugs and errors. Tests are performed using various strategies. Testing is performed in parallel with the coding process.

- Testing involves using test data and scenario to verify that each component and the whole system works under normal and different situations.

- The testing deliverables may be test plan (test scenario) and test data, results of the programs, and system testing.

- Once coding has begun, the testing process can begin and proceed in parallel. As each program module is produced, it can be tested individually, then as a part of a larger program, then as a part of a target system.

- Testing software begins earlier in the SDLC process. The purpose of the written test plans is to improve communication among all the people involved in testing the application software.

- The test plan specifies what each person's role will be during testing. The ultimate goal of testing process is to ensure software quality assurance (SQA) and quality check of a software system such as functionality, reliability, usability, efficiency, portability, maintainability, etc.

❑ Different Types of Testing

✓ **Inspections**

- Inspections are the formal group activities in which participants manually examine code for occurrence of well – known errors, i.e. inspection is a testing technique in which participants examine program code for predictable language specific errors.

✓ **Walkthrough**

- In walkthrough, the correctness of the models produced is checked and the errors detected are notified for amendments (corrections).

- Unlike inspection, what the code does is an important question in the walkthrough testing, i.e. walkthrough is an effective method of detecting errors in a code.

✓ **Syntax checking**

- Syntax checking is typically done by a compiler. Errors in syntax are covered, but the code is not executed.

Contd. ….

✓ **Desk checking**

- Desk checking is an informal process where the programmer or someone else who understands the logic of the program works through the code with a paper and pencil.

- The programmer executes each instruction, using test cases that may or may not be written down, i.e. desk checking is a testing technique in which the program code is sequentially executed, manually by the reviewer.

✓ **Stub testing**

- It is a testing technique used in testing modules, specially where modules are written and tested in a top – down fashion, and where a few lines of codes are used to substitute for the subordinate modules.

- Top – level modules contain many calls to subordinate modules, we may wonder how they can be tested if the lower level modules have not been written yet. This is called stub testing.

Contd. ….

✓ Unit testing

- The first level in the testing process is called unit testing. Unit testing concerns testing the smallest components of the software i.e. modules.

- Different modules are tested against the specifications produced during design for the modules.

- Unit testing is performed by the developer before the setup is handed over to the testing team to formally execute the test cases.

- Unit testing is performed by the respective developers on the individual units of source code's assigned areas. The goal of unit testing is to isolate each part of program and show that individual parts are correct in terms of requirements and functionality.

- Developers test their piece of code which they have written like classes, functions, interfaces, procedures and methods.

- The focus of unit testing is on the performance of a single module by finding defects in that module. As the focus of this testing level is on testing the code, white box testing is best suited for this level of testing.
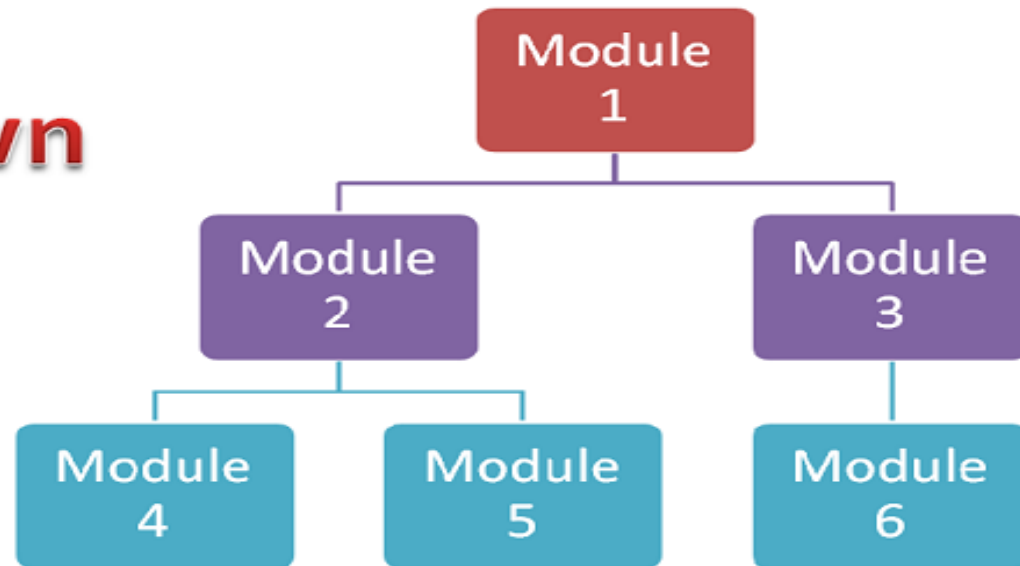
Contd. ….

✓ **Integration testing**

- The next level of testing is called integration testing.

- Integration testing is done when two modules are integrated, in order to test the behaviour and functionality of both the modules after the integration.

- It is a process of combining multiple modules systematically for conducting test in order to find the errors in the interface between the modules.

- The goal here is to see if the modules can be integrated properly. Hence, the emphasis is on testing the interface between the modules.

- Defects in the modules that have been missed during unit testing are discovered during the process of integration testing.

- **Integration testing basically follows two approaches and they are top-down integration and bottom-up integration.**

Contd. …..

Top-down integration

- In this technique, testing takes place in top-down fashion.

- In this testing technique, the higher-level modules are tested first and progressively lower-level modules are tested after that i.e. the modules at one level below are tested by calling previously tested modules till all the modules are tested.

- Components of systems are substituted by stubs during this testing technique.
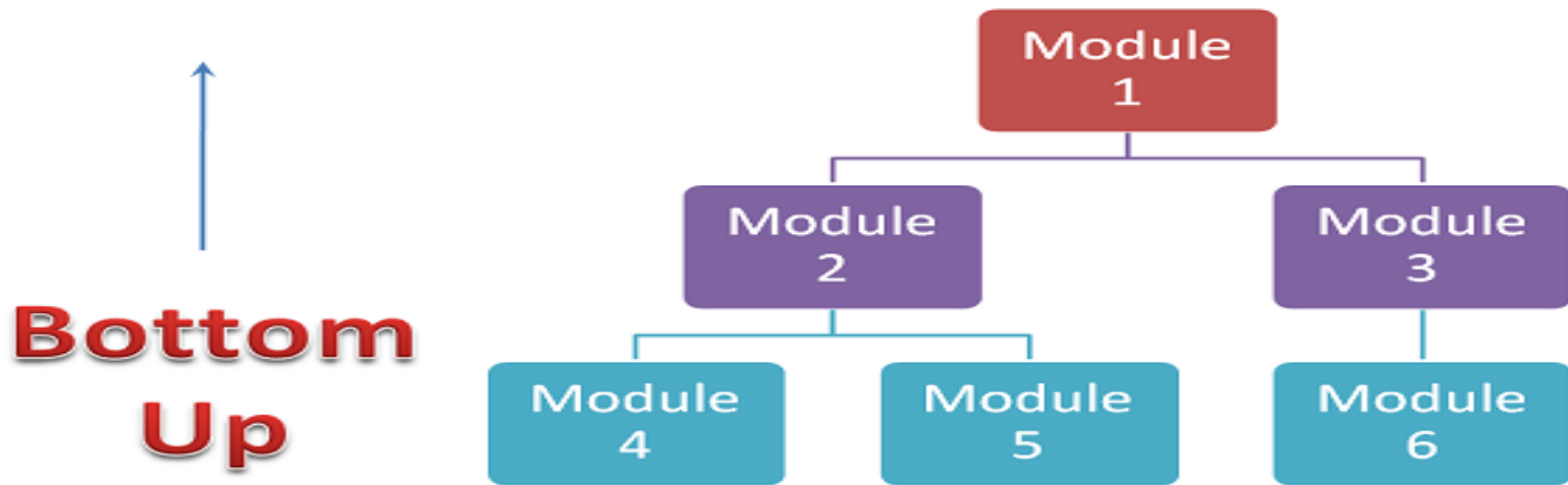
Contd. ….

Bottom-up integration

- In this technique, testing takes place in bottom-up fashion.

- In this testing technique, the testing begins with unit testing, followed by the tests of progressively higher-level combination of units called modules or builds i.e. the modules at one level above are tested by calling previously tested modules till all the modules are tested.

- Components of systems are substituted by drivers during this testing technique.

Contd. …..

✓ System testing

- This is the next level of testing and tests the system as a whole. Once all the components are integrated, the application as a whole is tested rigorously to see that it meets the quality standards.

- System testing enables us to test, verify and validate both the business requirements as well as the application architecture.

- System testing is most of the final test to verify that the system to be delivered meets the specifications and purpose. It should investigate both the functional and non-functional requirements of the application.

- It may include tests based on risks, requirement specifications, business process, other high level descriptions of system behaviour, interactions with operating system, etc. The reference document for this purpose is the requirement document, and the objective is to see if the software meets its requirements and specifications.

- System testing is carried out by the specialized testing team or independent testing team with no involvement of designers and programmers.

- *The system can be tested in two ways: Black box testing and white box testing.*

Contd. ….

Black box testing	White box testing
i. The internal workings of an application are not required to be known.	i. Tester has full knowledge of the internal workings of the application.
ii. Also known as closed box, data driven or functional testing.	ii. Also known as clear box, structural or code based testing.
iii. Performed by end users and also by testers and developers.	iii. Normally performed by testers and developers.
iv. Testing is based on external expectations; internal behavior of application is unknown.	iv. Internal workings are fully known and tester can design test data accordingly.
v. This type of testing always focuses on what is performing or carried out.	v. This type of testing always focuses on how it is performing or going on.
vi. It is performed by the testing team.	vi. It is performed by developers themselves.
vii. Basis of test case is requirement specification.	vii. Basis of test case is detail design of the system.
viii. Techniques that are used in this testing are boundary value analysis, syntax testing, equivalence class partitioning, etc.	viii. Techniques that are used in this testing are code coverage, path testing, data flow testing, etc.
ix. It is not suited for algorithm testing, and is least time consuming and exhaustive.	ix. It is suited for algorithm testing and is most time consuming and most exhaustive.
x. Programming and implementation of knowledge is not required.	x. Programming and implementation of knowledge is required.
xi. It is mainly applicable to higher levels of testing such as system testing and acceptance testing.	xi. It is mainly applicable to the lower levels of testing such as unit testing and integration testing.
xii. This can only be done by trial and error method, and it scales up, etc.	xii. Data domain and internal boundaries can be better tested and, it does not scale up, etc.

Contd. …..

✓ <u>**User Acceptance testing**</u>

- After the system test has corrected all or most of the defects, the system will be delivered to the user or customer for acceptance testing.

- The goal of acceptance testing is to establish confidence in the system, it is most often focused on validation testing.

- This is arguably the most important level of testing as it is conducted by the quality assurance (QA) team who will check whether the application meets the intended specifications and satisfies the customer's requirements.

- The QA team will have a set of pre-written scenario and test cases that will be used to test the application.

- **Acceptance testing can be performed in two phases and they are Alpha testing and Beta testing.**

Contd. …..

Alpha testing

- This test is the first stage of acceptance testing and will be performed in development phase amongst the developer and QA team to ensure that the software is fit for release and has required level of quality.

- **During this phase, the followings will be tested in the application:**

✓ Spelling mistakes

✓ Broken links

✓ The application will be tested on machines with the lowest specification to test loading time and any latency problems, etc.

Contd. ….

Beta testing

- This test is performed after alpha testing has been successfully performed.

- In this testing, a sample of intended audience tests the application.

- Beta testing is also known as pre-release testing.

- Beta version of application is ideally distributed to a wide audience, partly to give the program a real world test and partly to provide a preview of the next release.

- **In this phase, the audience will be testing the followings:**

- ✓ User will install, run the application and send their feedback to the project team.

- ✓ Typographical errors, confusing application flow, and even crashes, etc.

- Getting the feedback, the project team can fix the problems before releasing the software to the actual users or customers.

❑ <u>Installation</u>

- Installation is the process during which the current system is replaced by the new system.

- It includes installing the new system in organizational sites as well as dealing with personal and organizational resistance to the change that the new system might cause.

- The installation deliverables may be user guides, user training plans, installation and conversion plan, etc.

➢ *There are basically following four categories of installation:*

✓ <u>**Direct installation**</u>

- It means changing over from the old information system to a new one by turning off the old system when the new system is turned on.

✓ <u>**Parallel installation**</u>

- It means running the old information system and the new one at the same time until management decides the old system can be turned off.

17

Contd. ….

✓ **Single – location installation / Pilot installation**

● It means trying out an information system at one site and using the experience to decide if and how new system should be deployed throughout the organization or on the individual purpose.

✓ **Phased installation**

● It means changing from the old information system to the new one incrementally, starting with one or a few functional components, and then gradually extending the installation to cover the whole new system.

❑ <u>Documenting the System</u>

- Documentation is the process of collecting, organizing, storing, and maintaining a complete record of system and other documents used or prepared during the different phases of the life cycle of the system.

- *Proper documentation of the system is necessary because:*

✓ It solves the problem of indispensability of an individual for an organization. Even if the person, who has designed or developed the system, leaves the organization, the documented knowledge remains with the organization.

✓ It makes easier to modify and maintain in the future. The key to maintenance is proper and dynamic documentation.

✓ It helps in restarting a system development, which was postponed due to some reasons i.e. the job need not be started from scratch; we can give continuity of the software on the basis of the prepared or developed documents through the use of proper and efficient documentation methods.

❑ Types of Documentation

❖ System documentation

- System documentation records detailed information about a system's design specification, its internal working and its functionality.

- System documentation is intended primarily for maintenance programmers.

➢ **It contains the following information:**

✓ A description of the system specifying the scope of problem, the environment in which it functions, its limitations, its input requirements, and the form and type of output required.

✓ Detailed diagram of system flowchart and program flowchart.

✓ A source code listing of all the full details of any modifications made since its development.

✓ Problem definition and the objective of developing the system.

✓ Output and test report of the program development.

✓ Upgrade or maintenance history, if modification of the program is made, etc.

Contd. ….

Types of system documentation

I. <u>Internal documentation</u>: It is part of the program source code or is generated at compile time.

II. <u>External documentation</u>: It includes the outcome of structured diagramming techniques such as DFDs and ERDs.

❖ **<u>User documentation</u>**

- The next type of documentation is user documentation.

- User documentation consists of written or other visual information about an application system, how it works, and how to use it. User documentation is primarily for the users.

➢ **It contains the following information:**

✓ Set up and operational details of each system.

✓ Loading and unloading procedures, special checks and security measures.

✓ Problems which could arise, their meaning reply and operation action.

✓ Quick reference guides about operating a system in a short, concise format, etc.

❑ <u>Training and Supporting Users</u>

- **User training** is the process of providing necessary skills and knowledge to users about how to use the system.

- It may include a variety of human and computer assisted session as well as tools to explain the purpose, and use of the system.

- The training may include or can be: application specific; or general specific for operating the system, and off – the – shelf software.

- The user training deliverables include classes, tutorials, user training modules, training materials, and computer – based training aids, etc.

- **User support** is the process of building of mechanism which acts as guidelines for the users in case of any problem arises.

- It ensures that users can obtain assistance they need as questions and problems arise.

- User support deliverables may include help – desk, online help, bulletin boards, and other support mechanisms.

❑ **<u>Organizational Issues in Systems Implementation</u>**

- Here, it discusses about why implementation sometimes fails, and also about the security issues of systems implementation.

- *The systems implementation sometimes fails because of the following some of the factors:*

✓ User's personal stake.

✓ System characteristics.

✓ User demographic features.

✓ Organizational support.

✓ Performance.

✓ Satisfaction.

✓ Security issues, etc.

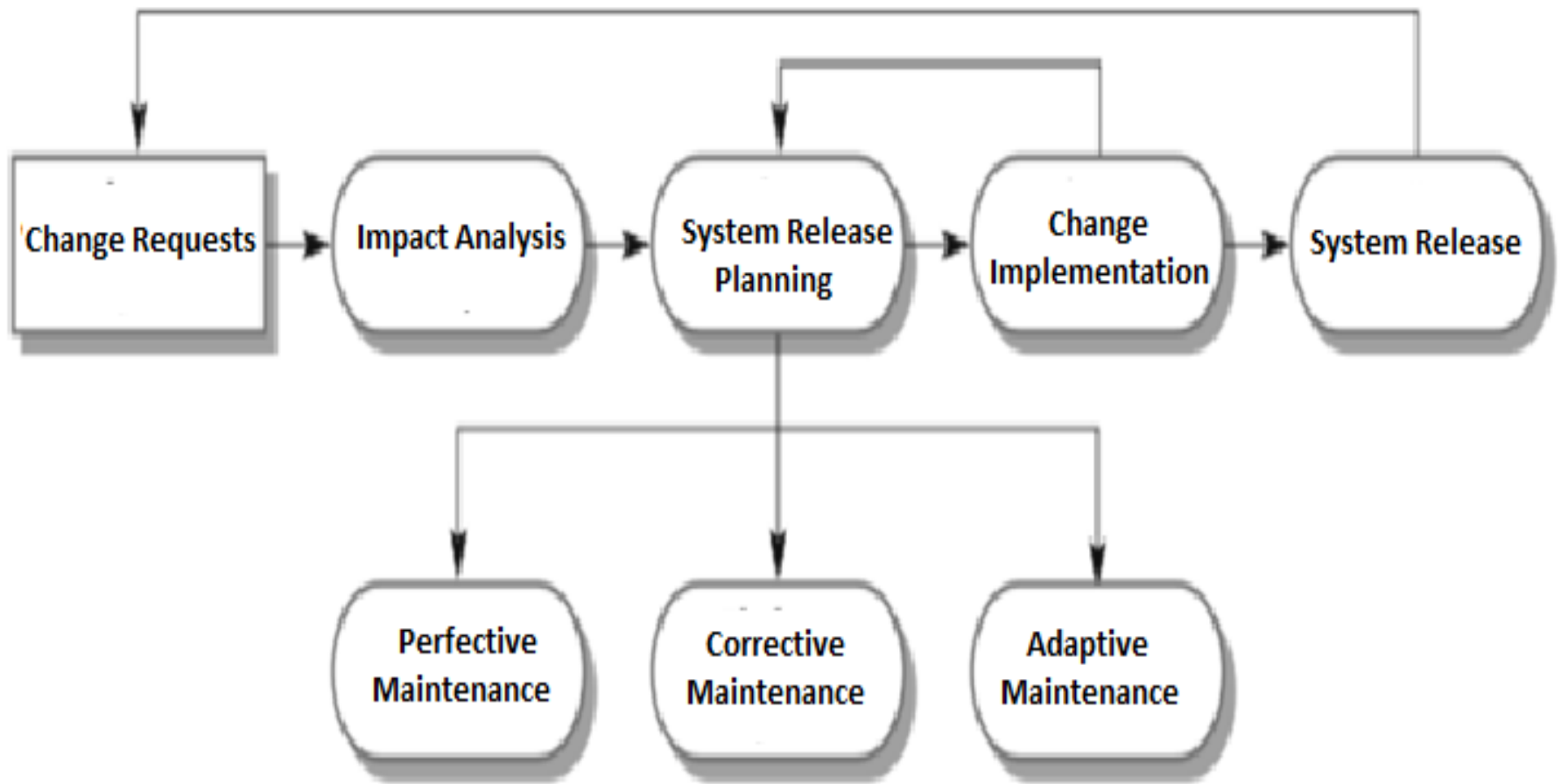# ❑ System Maintenance / Maintaining Information Systems

- System maintenance is the general process of changing a system after it has been delivered. The term is usually applied to custom system where separate development groups are involved before and after delivery.

- The changes made to the system may be simple changes to correct coding errors, more extensive changes to correct design errors or significant enhancements to correct specification errors or accommodate new requirements.

- Changes are implemented by modifying the existing system components and, where necessary, by adding new components to the system.

- **There are four major activities that take place within the maintenance process:**

- ✓ *Obtaining maintenance requests*
- ✓ *Transforming requests into changes*
- ✓ *Designing changes*
- ✓ *Implementing changes*

❑ Conducting Systems Maintenance

- System maintenance is referred to as the process of developing, maintaining and updating information systems for various reasons.

- System maintenance processes vary considerably depending on the type of IS being maintained, the development processes used in an organization and the people involved in the process.

- System maintenance is referred to as the process of developing, maintaining and updating system for various reasons. System changes are inevitable because there are many factors that change during the life cycle of a piece of system. **Some of these factors include:**

  - ✓ New requirements emerge when the software is used;
  - ✓ The business environment changes;
  - ✓ Errors must be repaired;
  - ✓ New computers and equipment is added to the system;
  - ✓ The performance or reliability of the system may have to be improved, etc.

- System maintenance is important because organizations have invested large amount of money in their IS and are now completely dependent on these systems.

Contd. ....

Fig:The Maintenance Process

❑ __Types of Maintenance__

- We can perform several types of maintenance on an information system. By maintenance, we mean fixing or enhancing the information system. These various types of maintenance are:

- ✓ Corrective maintenance

- ✓ Adaptive maintenance

- ✓ Perfective maintenance

- ✓ Preventive maintenance

- *Now, these are discussed below:*

Contd. ….

I. Corrective Maintenance

- This includes modifications and updations done in order to correct or fix problems, which are either discovered by the users or concluded by the user error reports.

- In other words, Corrective maintenance deals with the repair of faults or defects found in day-to-day system functions.

- It refers to the changes made to repair defects in the design, coding, or implementation of the system.

- It implies removing errors in a program, which might have crept in the system due to faulty design or wrong assumptions.

- Thus, in corrective maintenance, processing or performance failures are repaired.

- In short, we can say that corrective maintenance is universally used to refer to maintenance for **fault repair**.

Contd. ….

II. Adaptive Maintenance

- This includes modifications and updations applied to keep the system product up-to-date and turned to the ever changing or dynamic world of technology and business environment.

- In other words, adaptive maintenance means adapting to a new environment and can mean adapting the information system to new requirements.

- Adaptive maintenance is the implementation of changes in a part of the system, which has been affected by a change that occurred in some other parts of the system.

- Adaptive maintenance consists of adapting information system to changes in the environment such as the hardware or the operating system.

- Adaptive maintenance is usually less urgent than the corrective maintenance because business and technical changes typically occur over some period of time.

- Contrary to the corrective maintenance, adaptive maintenance is generally a small part of an organization's maintenance effort, but it adds value to the organization.

Contd. ….

III. Perfective Maintenance

- This includes modifications and updations in order to keep the IS usable for longer period of time. It includes new features, new requirements for refining the system and improve its reliability and performance.

- In other words, perfective maintenance means perfecting the IS by implementing new requirements; in other cases, it means maintaining the functionality of the system but improving its structure and its performance.

- Perfective maintenance is for adding new programs or modifying the existing programs to enhance the performance of the system.

- Perfective maintenance includes enhancing both the function and efficiency of the code and changing the functionalities of the system as per the users' changing needs.

- So, in conclusion, perfective maintenance involves making functional enhancements to the system in addition to the activities to increase the system's performance even when the changes have not been suggested by the faults.

Contd. ….

IV. Preventive Maintenance

- This includes modifications and updations to prevent problems of the information system. It aims to attend problems, which are not significant at this moment but may cause some serious issues in future.

- In other words, preventive maintenance involves performing activities to prevent the occurrence of errors. It refers to changes made to increase the understanding and maintainability of our IS in the long run.

- Preventive changes are focused in decreasing the deterioration of our information system in the long run.

- Preventive maintenance tends to reduce the software complexity thereby improving program understandability and increasing information system's maintainability.

- Preventive maintenance comprises documentation updating, code optimization, and code restructuring.

- As with adaptive maintenance, both perfective and preventive maintenance are typically a much lower priority than corrective maintenance.

❑ **Managing Maintenance**

- **Managing Maintenance Personnel.**
- **Measuring Maintenance Effectiveness.**
- **Controlling Maintenance Requests.**
- **Configuration Management.**