

WHY SOFTWARE ENGINEERING:-

- 1) In the late 1960's hardware price were falling but software price rising.
- 2) Many software projects failed.
- 3) Large software project required large development loans.
- 4) Many software project late and over budget.
- 5) Complicity of software project is increased.
- 6) Demand for new software on the market.

WHY STUDY SOFTWARE ENGINEERING?

- 1) Higher productivity.
- 2) To acquire skills to develop large programs.
- 3) Ability to solve complex programming problems.
- 4) Learn techniques of specification design.
- 5) Better quality programmers.

APPLICATION OF SOFTWARE:-

- 1) System software.
- 2) Application software.
- 3) Engineering/scientific software.
- 4) Embedded software.
- 5) Product line software.
- 6) Web application software.
- 7) Artificial intelligence software (AI).

- 1) The concept analysis in the view of S.E.
- 2) System development life cycle (SDLC).
- 3) Software requirement specification (SRS).
- 4) Object- data base and flow base analysis.
- 5) Models- spiral, water fall model.

DIFFERENCE BETWEEN PROGRAM AND ENGINEERING.

PROGRAM	ENGINEERING
1) Small project. 2) You 3) Once product. 4) Cheap 5) Few sequential changes.	1) large product 2) team 3) family of product 4) costly 5) Many parallel changes.

Preview from Notesale.co.uk
Page 10 of 36

PRINCIPLES OF SYSTEM DOCUMENTATION: - it has the following specific features. These are

- 1) availability
- 2) objectivity
- 3) cross referencing
- 4) easy to maintain
- 5) completeness

- 1) *Availability:* - it refers the documentation should be available for the analyst, developer and the user. In right place at right time.
- 2) *Objectivity:* - the objectivity must be focused and clear who are making the documentation for its specific use. Objectivity refers the original method for making particular software.
- 3) *Cross referencing:* - it determines for internal communication between two or more modules. This feature includes the internal relationship between the modules.
- 4) *Easy to maintain:* - after complete the documentation some times it refers in future it may be changed at this time the documentation should be worked properly and correctly.
- 5) *Completeness:* - it refers the all phase including design, coding, testing, user manual. This must be included for making the documentation successful and complete.

MODULE 6

CODING AND PROGRAMMING

The main object of coding is to implement the design with the help of some programming environment to achieve a specific task. After writing the code we have to go through the error checking part to run the code successfully. After that the code and the design part we have to match those with the original output we may get from the original code.

Coding techniques help to increase the cost of the software in the error handling part and the implementation part.

Choice of programming language: to achieve the particular software design output in the real world the coder has to choose. Some programming language with some specific programming environment for this we have to know very well the following objectives in the software coding part

- 1) Understand very well the design approach.
- 2) Enough knowledge of programming and environment.
- 3) Select a programming language to achieve a specific task for client.
- 4) Understand the original cost and coding part in the real world scenario.

Basically we have two types of environment available in software industry

- a) structured programming
- b) object programming

The both programming environment has some specific norms in its own area. For example the structured programming environment deals with some specific function with its own parameter and an OOP environment based on the objectives which is sharable from method to another.

Mixed language programming: - it is basically an integrated approach for the software coder to direct and indirect they can change the environment as it required at the time of operation.

MODULE- 7

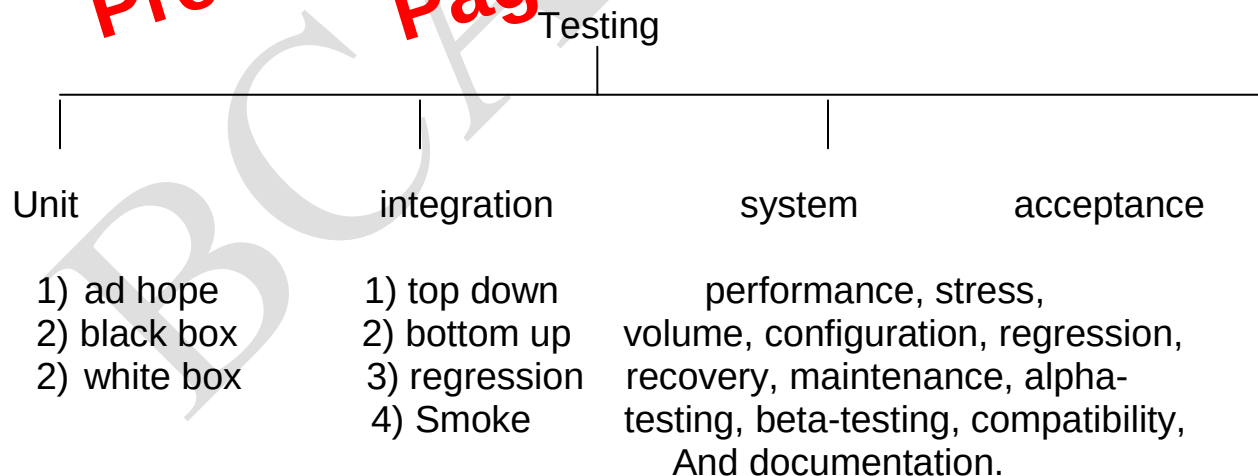
SYSTEM TESTING

Definition of testing: - according to Myers (1979) testing is technique of software to find the error or mistake in a particular software design.

The term over we can define as some unconditional or conditional, direct or indirect mistakes from our design phase to implementing phase in particular software area. In testing the main fault and failure are two main corresponding factors in testing. Fault is a condition for required achievement of a specific function where as failure is the inability to achieve a desired goal.

Software testing is a very essential part in S/E to make a product success. Its basic objective for checking all parts connecting through a specific project.

Types of testing.



Unit testing: - in testing methodology first testing is unit testing. It has basically three parts a hope testing, white box testing, black box testing.

WBS (WORK BREAK DOWN STRUCTURE): - a WBS in project management and system engineering is use to define and group a project. Work elements in a way that helps organization and define that total work scope of the project.

A WBS element may product, data service or any combination. A WBS also provide the necessary from was for detailed cost estimating and control along with providing guidelines for schedule development control. Additionally the key is a dynamic tool that can be developed and received as needed by project manager. A WBS is structure which shows sol division of effort required to achieve an objective.

For example- a program a project and contract the project or contract the WBS is developed by starting with the end objectives and solve dividing into manageable components in tare-of-size, duration and responsibility.

One of the most important WBS, structure design principle is called 100% rule. The 100% rule state that WBS includes 100% of the work defined by the project scope and capture for internal and external process in term of work to be completed including Project management.

Preview from Notesale.co.uk
Page 33 of 36