

Name :

Alamzeb

ID :

14481

Section :

BS (SE-4) (A)

Dep :

Computer Science

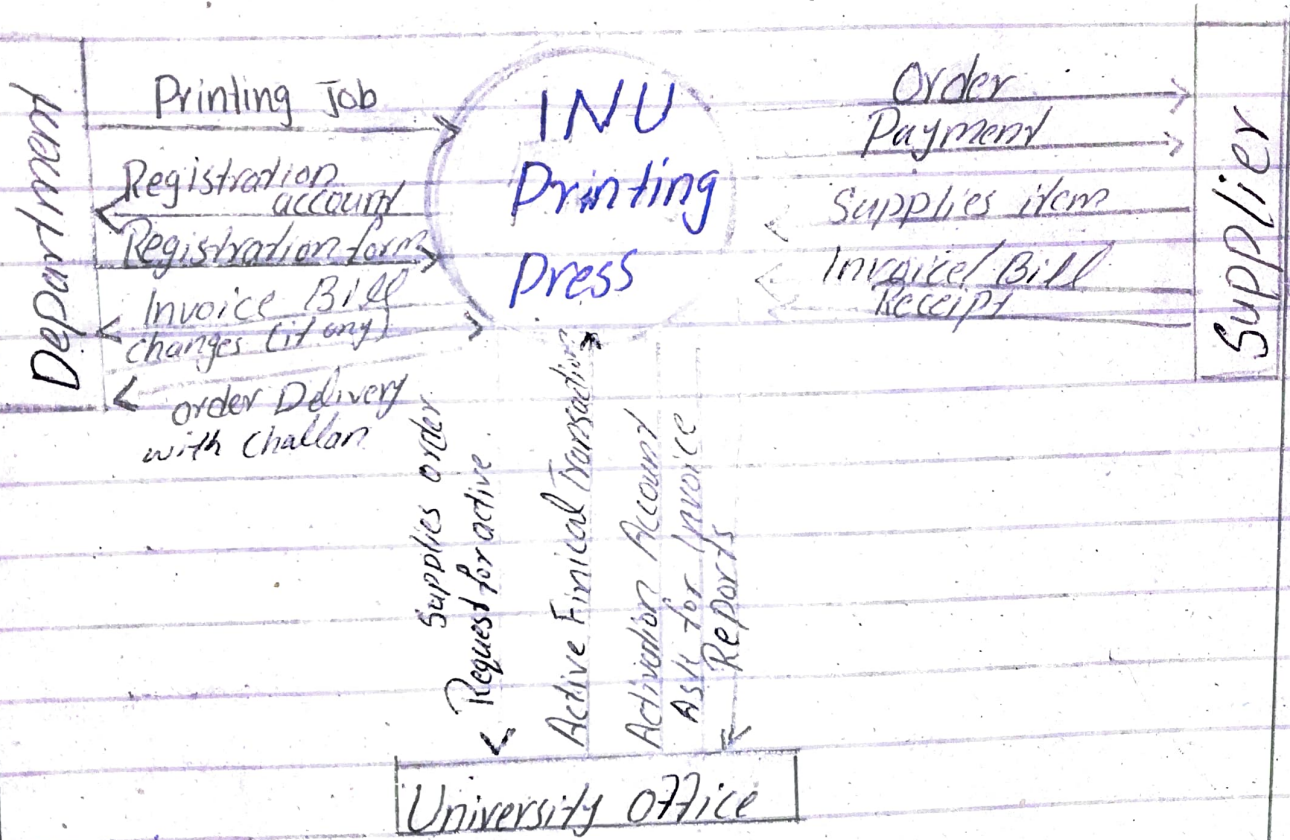
Subject :

Software Engineering

Instructor :

Engr. Ghassan Husnain

Q1: Draw a Context diagram for INU Printing press? Draw a level 1 Data Flow Diagram (DFD) for the above cases Study.
 Answer:



Q3: Draw a level 2 DFD for the Order Registration process, press production process, inventory process and Billing process.

Q2 Explain why testing can only detect the presence of errors, not their absence?

Ans: Testing can detect only the presence of errors, not their absence because the main goal of the testing is to observe the behavior of the particular software and to check whether it meet its requirement expectation or not. Testing is a part of broader process of software verification and validation. It consists of a set of activities, where the testers try to make the software behave anomalous in order to detect or anomaly to be later fix. Testing cannot demonstrate the faults other than specified in every circumstance. It always possible that a test have overlooked could discover further problem with the system.



Q2.2 Define the following terms.

1) Unit Testing :-

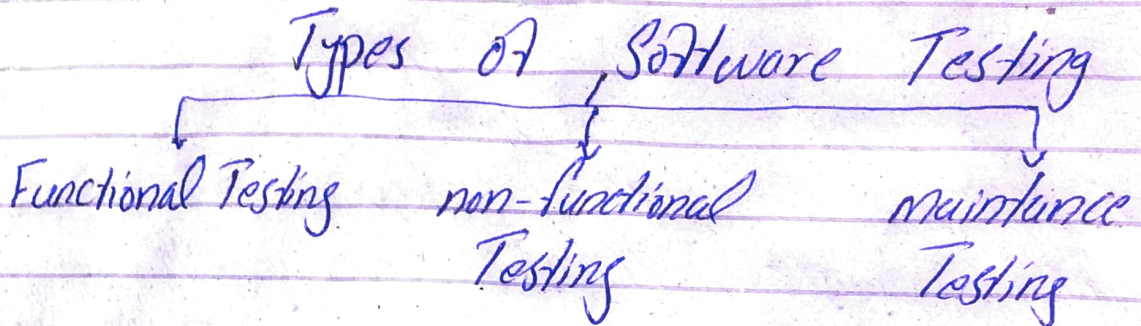
Unit testing is a software development process in which the

Smallest testable parts of an application called units, are individually and independently scrutinized for proper operation. This testing methodology is done during the development process by the software developers and sometimes QA staff. The main objective of unit testing is to isolate written code to test and determine if it works as intended.

2) Software Testing..

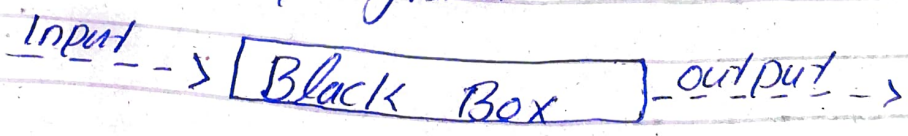
Software testing is defined as an activity to check whether the actual results match the expected results and to ensure that the software system is defect free. It involves the execution of a software component or system component to evaluate one or more properties of interest. Software testing also help to identify errors, gaps, or missing requirements in contrast to the actual requirements. It can either be done manually or using automated tools. Some prefer saying software testing as a white box.

and Black Box Testing.



3) Black Box Testing:

Black Box Testing is defined as a testing technique in which functionality of the application Under Test (AUT) is tested without looking at the internal code structure, implementation details and knowledge of internal paths of the software. This type of testing is based entirely on software requirements and specification. In Black Box Testing we just focus on inputs and outputs of the software system without bothering about internal knowledge of the program.



4) White Box Testing:

White Box Testing is Testing of a Software Solution's internal structure, design, and coding. In this type of testing the code is visible to the tester. It focuses primarily on verifying the flow of inputs and outputs through the application, improving design and usability. Strengthening security, white Box testing is also known as Clear Box testing, Open Box Testing, Structural Testing, Transparent Box Testing, Code-Based testing and Glass Box Testing. It usually performed by developers.



Q31 Briefly describe the three main types of Software maintenance. Why is it sometime difficult to distinguish between them?

Ans:

The main types of Software maintenance are as follows.

1) Fault repairs:

Coding errors are usually relatively cheap to correct, design errors are more expensive as they may involve rewriting several program components. Requirement errors are the most expensive to repair because of the expensive system redesign which be necessary.

2) Environmental adaptation:

This types of maintenance is required when some aspect of the system's environment such as the hardware, the platform operating system, or other support software changes the application system must be modified to adapt it to cope with these environmental changes.

3) Functionality addition:

This types of maintenance is necessary when the system requirements changes in response to organizational or

business changes the scale of the changes required to the software is often much greater than for the other types of maintenance.

Difficult to differentiate b/w the types of maintenance.

In practice, there is not clear-cut distinction between these types of maintenance. When the system adapts to new environment, then add functionality to take advantages of new environmental features. Software faults are often exposed because user use the system in unanticipated ways. These types of maintenance are recognized but a different person sometimes gives them different names.

'Corrective maintenance' is universally used to refer to maintenance for fault ~~and~~ repair.

'Adaptive maintenance' sometime means adapting to new environment and some times ~~that~~ means adapting the software to new requirements.

(5/16)

'Perfective maintenance' sometimes means perfecting the same software by implementing new requirements in order cases it means maintaining the functionality of the system but improving its structures and performance.



Q3.2 What are the principle factors that affects the costs of system reengineering? Also briefly explain the reengineering process with the help of diagram.

Ans:

Reverse engineering is the process of analysing software with the objective of recovering its design and specification. The program itself is unchanged by the reverse engineering process.

The cost effective to re engineer a system. re-engineering a system is cost-effective when it has a high business value but it expensive to maintain. Re-engineering improves

the System Structure, creates new System documentation and makes it easier to understand.

Re-engineering a Software System has two key advantages over more radical Reduced cost. The cost of re-engineering is significantly less than the cost of developing new Software.

Example:

Ulrich (Ulrich, 1990) quotes an example of a Commercial System where the re-implementation costs were estimated as at \$50 million. The System was successfully re-engineered for £20 £12 million. The Re-engineering Process

