Name: ShahZeb

ID# 14798

Subject: Software Engineering

Section: B

**Question 1:**

**Q1.1:**

**Diagram:** 

**Question 1:**

**Q1.2:**

**Diagram:**



**Q1.3:**

**Diagram:**



**Question 2:**

**Q2.1**

**Ans:**Testing can detect only the presence of error not their absence because the main goal of the testing is a part of broader process of software verification and validation. It consists of a set of activities. Where the tester try to make the software behave anomalous in order to detect or anomaly to be later fix. Testing can, t demonstrate the faults other than specified in every circumstance. It is always possible that a test have overlook could discover further problem with the system

**Q2.2: Define the following terms.**

**Ans:**

**Unit Testing:**

**Objectives:** To test the function of unit of code such as a program as module

* To test internal logic
* To verify internal design
* To test path and condition coverage
* To test exception condition and error handling

**When**: Aftermod are coded.

**Input**: Internalapplication design.

* Master test plan
* Unit test plan

**Output:** Unit test report**.**

**Who:** Developer

**Methods:** white box testing techniques

**Tools:** Debug

* Restructure
* Code Analyzers
* Path statement coverage tools

**System Testing:**

**Objectives:**  To verify that the system component perform control function .

* To perform inter system test.
* To demonstrate that the system perform both functionally and operational as specified
* To perform appropriate types of test relating to transation flow, installation rolibility, regression etc.

**When**: After integration testing

**Input**: Detailed requirement and external application design

* Master test plan
* System test plan

**Output**: System test report.

**Who**: Development team and user

**Methods**: Problem/configuration management .

**Tools**: Depends.

**Education**: Testing methodology

**Black Box Testing:**

* No knowledge of internal design or code required .
* Test are based on requirement and functionality
* Not based on any knowledge of internal design or code
* Covers all combined part of a system
* Test are data driven

**It uncovers:**

* Incorrect or incising function
* Interface errors
* Errors in data structure or external databased access
* Performance errors
* Initialization and termination errors

**White Box Testing:**

* Based or knowledge of internal logic an application code base on coverage of code statement branches paths, conditions tests are logic given it ensures.
* All independent paths within a module have been exercised at list once
* Exercise all logical decision on their true & false sides
* Execute all loops at their boundaries and within their operational boude
* Exercise internal data structure to ensure their validity

**Question 3:**

**Q3.1:**

**Ans:**

In practice there is not a clear cut distinction between these types of maintenance, when the system adopt to new environment then add functionality to take advantages of new environmental features. Software faults are often exposed because users use the system in unanticipated way. These types of maintenance are recognized but a different person sometime give them different names

**Corrective maintenance :** Is universally used to refer to maintenance for fault repair.

**Adaptive maintenance:** Sometimes means adapting to new environment and sometimes means adapting the software to new requirements.

**Perfective maintenance**: Sometimes perfectively the software by implementing new requirements in other case it means maintaining the functionality of the system but improving its structure & performance.

**Q 3.2:**

**Ans:**

**System Re engineering:**

* Re structuring or re writing part or all of a legacy system without changing its functionality
* Some applicable where sub system of a larger system require frequent maintenance
* Re engineering involves adding effort to make then easier to maintain. The system may be re structured and redocumented
* Reduced risk
* Reduced cast

**Diagram**

Original data

**Re Engineering process:**

Original program

Program documentation

 dat

Reverse engineering

Re engineered program

Program modulanization

Source code translation

Data Re engineering

Re engineered data

Restruction program

Programe structure improvement