

Q.1) Explain coupling and cohesion and explain different types of cohesion.

Ans) ① Independence is assessed using two Qualitative Criteria: Cohesion and Coupling.

② Cohesion is an indication of the relative functional strength of a module.

③ Coupling is an indication of the relative interdependence among modules.

④ Cohesion is a natural extension of the information-hiding. A cohesive model performing a single task, requiring little interaction with other components in other parts of a program. Stated simply, a Cohesive model should do just one thing.

⑤ Coupling depends on the interface complexity between modules, the point at which entry or reference is made to a module, and what data pass across the interface.

⑥ In software design, you should strive for the lowest coupling and highest cohesion.

⑦ Types of cohesion: Below we stated 7 types of cohesion from best to worst.

(i) Functional cohesion: Functional Cohesion is said to exist if the different elements of a module, cooperate to achieve a single function.

(ii) Sequential Cohesion: A module is said to have Sequential Cohesion if the elements of a module form the components of the sequence, where the output from one component of the sequence is input to next.

3. Communicational Cohesion: A module is said to have communicational cohesion, if all tasks of the module refer to or update the same data structure.
e.g.: Linear and Quadratic Probing applied to same hashing data structure.
4. Procedural Cohesion: A module is said to be procedural cohesion if the set of purpose of the module are all parts of a procedure in which particular sequence of steps has to be carried out for achieving a goal, e.g.: the algorithm applied for decrypting a message.
5. Temporal Cohesion: When a module includes functions that are associated by the fact that all the methods must be executed in the same time, the module is said to exhibit temporal cohesion.
6. Logical Cohesion: A module is said to be logically cohesive if all the elements of the module perform a similar operation. For example Error handling, data input and data output etc.
7. Coincidental Cohesion: A module is said to have coincidental cohesion if it performs a set of tasks that are associated with each other very loosely, if at all.

Q.) Explain Black Box testing.

- Ans)
- ① Black-box testing, also called behavioral testing or functional testing, focuses on functional requirements of the software.
 - ② Black-box testing attempts to find errors in the following categories:
 - (i) Incorrectness or missing functions,
 - (ii) interface errors,
 - (iii) errors in data structures or external database access
 - (iv) behavior or performance errors, and
 - (v) initialization and termination errors.
 - ③ Black box testing focuses on the information domain and disregards control structure.
 - ④ Tests are designed to answer the following questions:
 - (i) How is functional validity tested?
 - (ii) How are system behaviour and performance tested?
 - (iii) What classes of input will make good test cases?
 - (iv) Is the system particularly sensitive to certain input values?
 - (v) How are the boundaries of a data class isolated?
 - (vi) What data rates and data volume can the system tolerate?
 - (vii) What effect will specific combinations of data have on system operation?

- ⑤ By applying black-box techniques, a set of test cases can be derived that satisfy the following criteria: Reduction in test cases, number of additional test cases that must be designed to achieve reasonable

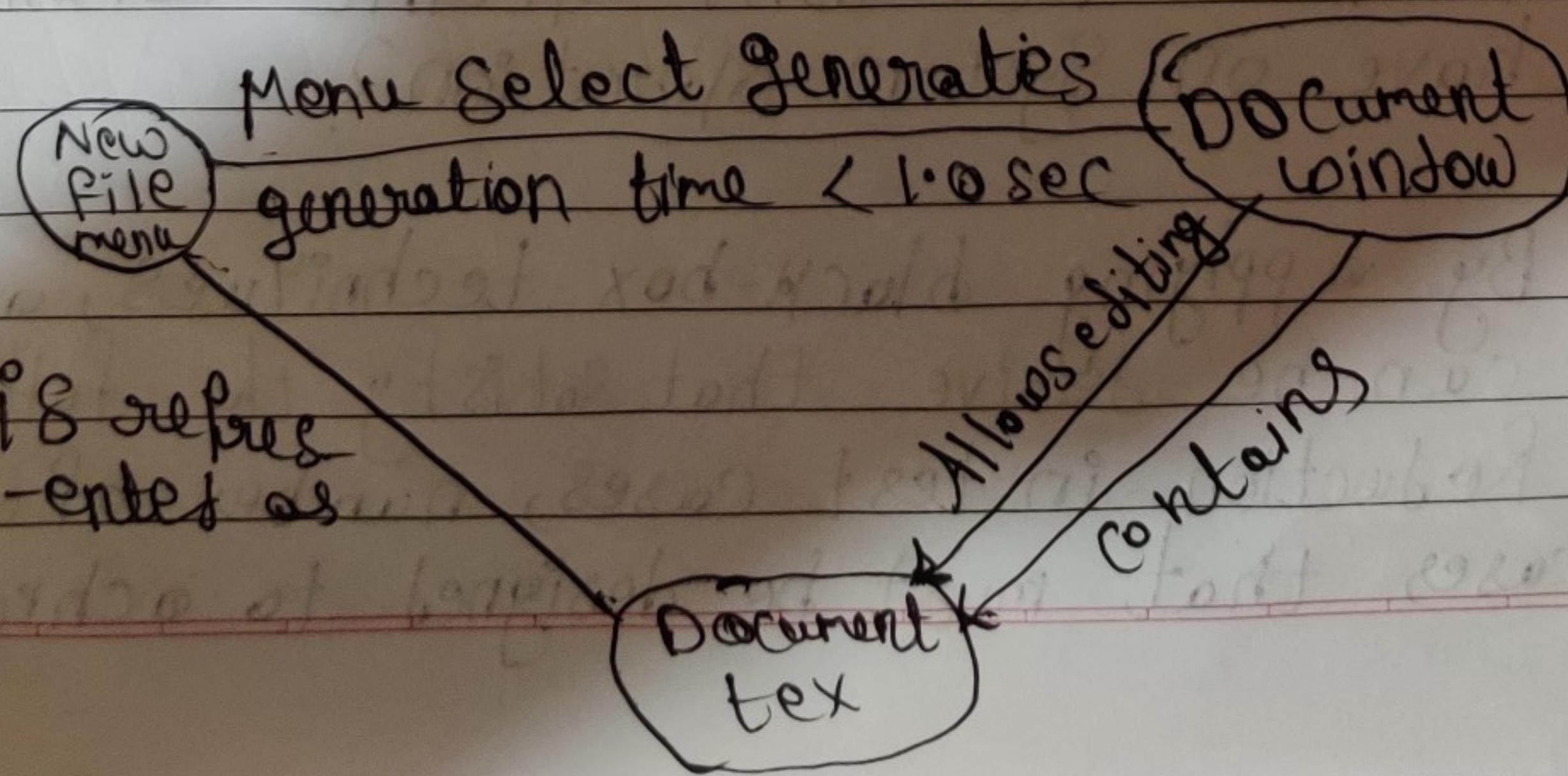
testing, and test cases that tell about presence or absence of classes of errors.

⑥ Methods that can be applied collectively for black box testing:

(7) Graph-based testing Methods: The first step in black-box testing is to understand the objects that are modeled in software and the relationships that connect these objects.

(ii) Once this has been accomplished, the next step is to define a series of tests that verify "all objects have expected relationship to one another". This can be achieved through by creating a graph - a collection of nodes representing objects, links (edges) represents relationship between objects, node weights describe properties of node and link weight describe some characteristic of link. Directed link from tail to head represent relationship in one direction only, Bidirectional or undirected link represents in both direction and parallel link represents more than one relationships among objects.

e.g:-



Boundary value Analysis: A greater number of errors occurs at the boundaries of the input domain rather than in the "center". It is for this reason that boundary value analysis (BVA) has been developed as a testing technique. BVA complements equivalence partitioning by selecting input values at the edges of equivalence classes. This reduces the number of inputs to be tested per test case.

Guidelines for BVA:

- 1) If an input condition specifies a range bounded by values $a & b$, test cases should be designed with values $a & b$ and just above & just below $a & b$.
- 2) If an input condition specifies a number of values, test cases should be designed that exercise the min. & max. numbers and numbers just above and below them.
- 3) Apply guidelines 1 & 2 to output conditions. Input those values which yields max. & min. output values.
- 4) If internal program data structures have prescribed boundaries (e.g. hashing table with 100 entries), be certain to design a test case to exercise the data structure at its boundary.

By applying these guidelines, there is a higher likelihood for error detection.

(iii) Test cases can be derived by traversing the graph. Test cases attempt to finds errors in any of relationships shown.

Different behavioral testing methods that can make use of graphs:

(i) Transaction flow modeling.

(ii) Finite State Machine.

(iii) Data flow modeling.

(iv) Timing modeling.

Equivalence Partitioning:- If a set of objects can be linked by relationships that are symmetric, reflexive and transitive, an equivalence class is present.

An equivalence class represents a set of valid or invalid states for input conditions. Typically, an input condition is either a specific numeric value, a range of values, a set of selected values, or a Boolean condition. Equivalence classes may be defined according to the following guidelines:

1. If an input condition specifies a range one valid and two invalid equivalence classes are defined.
2. If an input condition requires a specific value, one valid and two invalid equivalence classes are defined.
3. If an input condition specifies a member of a set, one valid & one invalid equivalence class are defined.
4. If an input condition is Boolean, one valid & one invalid class are defined.

Test cases are selected so that largest number of attributes of an equivalence class are exercised at once.

Q. Write note on RMMM.

Ans) (1) Risk management technique is usually seen in the Software Project plan. This can be divided into Risk Mitigation, Monitoring and Management Plan.

(2) In this plan, all works are done as part of risk analysis. As part of the overall project plan project manager generally uses this RMMM plan.

(3) In some software teams, risk is documented with help of a Risk Information Sheet (RIS).

(4) RIS is controlled by using database system for easier management of information. i.e. Creation, Priority ordering, Searching, and other analysis.

(5) After documentation of RMM and starts of project, risk mitigation & monitoring commences.

(6) Risk Mitigation: It is an activity used to avoid problems (Risk avoidance)

Steps for mitigating the risks as follows:

(i) Finding out the risks.

(ii) Removing causes that are the reason for risk creation.

(iii) Controlling the corresponding documents from time to time.

(iv) Conducting timely reviews to speed up the work.

(7)

Risk Monitoring: It is an activity used for project tracking. It has the following primary objectives as follows:

- (i) To check if predicted risks occur or not.
- (ii) To ensure proper application of risk aversion steps defined of risk.
- (iii) To collect data for future risk analysis.
- (iv) To allocate what problems are caused by which risks throughout the project.

(8)

Risk Management and Planning: It assumes that the mitigation activity failed and the risk is a reality. This task is done by project manager when risk becomes reality and causes severe problems. If the project manager effectively uses project mitigation to remove risks successfully then it is easier to manage the risk. This shows that the response that will be taken for each risk by a manager. The main objective of the risk management plan is the risk register. This risk register describes and focuses on the predicted threats to a software project.

MCQ - 2

1) Coupling is a measure of degree of interdependence between modules.

- a) Cohesion
- b) Coupling
- c) Both a & b
- d) None

2) Function oriented design technique start with functional requirement specified in SRS

- a) SDD
- b) SRS
- c) Both a & b
- d) None

3) What DFD notation is represented by Rectangle?

Data Structure

- a) Transform
- b) Data Structure
- c) Function
- d) None

4) What incorporates data, architecture, interface and procedural representation of software?

Design model

- a) Design model
- b) User model
- c) System model
- d) Data flow model

5) Which of the following is a type of architectural model? All of the above (UML, AXML, ADL)

- a) Static Structural Model
- b) Dynamic Process Model
- c) Distribution Model
- d) All of the above

6) What is cyclomatic complexity? White box testing

- a) Black box testing
- b) Yellow box testing
- c) White box testing
- d) Green box testing

7) White Box testing also classified as control structure testing.

- a) Functional testing
- b) Control Structure testing
- c) Error Guessing technique
- d) None

Analysis

- 8) Boundary value belongs to Black box and Complements Equivalence classes
- a) white box & equivalence b) black box & equivalence
 - c) white box & orthogonality d) black box & orthogonality.
- 9) Core of reverse engineering is an activity called
- a) Restructure code b) directionality
 - c) Extract abstraction d) interactivity
- Extract abstraction
- 10) Quality Management is software engineering is also known as SQA
- a) SQA b) SCM c) SQT d) SQR
- 11) Which one is not Risk management activity?
Risk Generation
- a) Risk assessment b) Risk generation
 - c) Risk control d) None
- 12) Risk management is one of important roles for Project Manager
- a) Senior Manager b) Software practitioner.
 - c) Project Manager d) Customer.
- 13) Which of the following doesn't affect SW quality & performance? Market
- a) Market b) Product c) Technology d) People.
- 14) What assess the risk & your plans for risk mitigation & revise these when you learn more about risk? Risk Monitoring
- a) Risk planning b) Risk Analysis
 - c) Risk monitoring d) Risk Management.

- 15) Which of the Following is a collection of component Version that make up System? Baseline
- Version
 - Code line
 - Baseline
 - Milestone

MCQ - I

- 1) CMM Stands For Capability Maturity Model
- Capability Maturity Model
 - Captiability Maturity Model
 - Conservative Maturity Model
 - Capability Management Model.
- 2) Efficiency in software Product doesn't include Licensing
- Responsiveness
 - Licensing
 - Memory utilization
 - Processing time.
- 3) RAD Stands for Rapid Application Development
- Relative application development
 - Rapid Application development
 - Rapid Application documentation
 - None
- 4) RAD model has 5-phases
- 2 phases
 - 3 phases
 - 5 phases
 - 4 Phases
- 5) which model doesn't allow defining requirements early in cycle? Prototyping
- water fall
 - XP
 - Incremental
 - Prototyping
- 6) The user requirements are parts of which document? SRS
- SDD
 - SRS
 - DFD
 - SRD

Q.7) Which level of ERD includes all entities & relationship ? Level 2

- a) Level-1 b) Level-2 c) Level-3 d) Level-4

Q.8) Which model in system modelling depicts the static nature of system? Structural Model

- a) Behavioral model b) Context model c) Data Model
d) Structural model

Q.9) The UML supports event based modelling using Static chart diagrams.

Q.10) Which of the following diagram is not supported by UML considering Data driven modeling? Data flow Diagram

- a) Activity b) State Chart c) Data flow
d) Component

Q.11) Which of the following is not an effective Software project management focus? Popularity

- a) People b) Product c) Process
d) Popularity

Q.12) Which of the following is not Project Manager's activity? Project Design

- a) Project Management b) Project control
c) Project Design d) Project planning.

Q.13) COCOMO Stands for Constructive cost - Model

- a) Constructive cost Model b) Comprehensive cost Model
c) constructive cost estimation model.
d) complete cost estimation model.

Q.14) Sprint concept is used in which Agile model? SCRUM

- a) Agile unified model
- b) SCRUM
- c) XP & IXP
- d) Kanban.

Q.15) Which of the two are cost estimation metrics? LOC & FP

- a) Class Point & LOC
- b) LOC & FP
- c) CRC & FP
- d) CRC & Class Point.