**SOFTWARE ENGINEERING – 1MARK**

**2019**

**(a) Define the term software reliability.**

**Software Reliability** refers to the probability of a software system functioning correctly under specified conditions for a given period. It is a measure of how consistently a software application performs its intended functions without failure.

**(b) Write down some advantages of the waterfall model.**

**Advantages of the Waterfall Model:**

* **Simplicity and Ease of Use**: The linear approach is straightforward and easy to understand.
* **Structured Approach**: Clear phases help in organizing and managing the project efficiently.
* **Effective for Small Projects**: Works well for smaller projects where requirements are well understood from the beginning.
* **Documentation**: Strong emphasis on documentation at each stage enhances clarity and maintainability.

**(c) State various components of a software.**

**Various Components of Software:**

* **User Interface**: The means by which users interact with the software.
* **Database**: Where data is stored, managed, and retrieved.
* **Business Logic**: The rules and algorithms that determine how data is processed.
* **Application Programming Interface (API)**: Interfaces that allow different software components to communicate.
* **Security Features**: Mechanisms that ensure data protection and user privacy.

**(d) Differentiate between verification and validation.**

**Verification vs. Validation:**

* **Verification**: The process of ensuring that the software meets specified requirements and is being built correctly. It answers the question, "Are we building the product right?"
* **Validation**: The process of ensuring that the software fulfills its intended use and meets the needs of the end-users. It answers the question, "Are we building the right product?"

**(e) What is process framework?**

**Process Framework** refers to a structured set of processes that provide guidance and best practices for managing and executing software development projects. It typically includes phases, activities, tasks, and roles necessary to deliver high-quality products systematically.

**(f) Define software measurement.**

**Software Measurement** is the quantitative assessment of software characteristics or attributes. It includes measuring aspects like size, complexity, performance, quality, and productivity to enable better management, improved performance, and informed decision-making.

**(g) What is a data dictionary?**

**Data Dictionary** is a centralized repository that contains definitions and descriptions of the data elements used in a database or software application. It includes details such as data types, formats, relationships, and usage to ensure consistency and understanding among stakeholders.

**(h) What is software prototyping?**

**Software Prototyping** is the process of creating early samples or models of the software application to demonstrate its functionality and user interface. Prototypes can range from low-fidelity wireframes to high-fidelity interactive models, allowing stakeholders to refine requirements and identify potential issues before full-scale development.

**2021**

**A) What is Software Engineering?**

* **Definition**: A discipline that involves the application of engineering principles to software development.
* **Key Aspects**:
  + Requirements analysis
  + Design
  + Implementation
  + Testing
  + Maintenance

**B) What is DFD?**

* **Full Form**: Data Flow Diagram
* **Purpose**: Visual representation of data flow within a system.
* **Components**:
  + Processes
  + Data stores
  + External entities
  + Data flows

**C) What is the Full Form of SDLC?**

* **SDLC**: Software Development Life Cycle
* **Phases**:
  + Planning
  + Analysis
  + Design
  + Implementation
  + Testing
  + Deployment
  + Maintenance

**D) What is Computer Software?**

* **Definition**: A collection of programs and related data that instructs a computer on how to perform tasks.
* **Types**:
  + System software (e.g., operating systems)
  + Application software (e.g., productivity applications)

**E) What are Function Points?**

* **Definition**: A standardized unit of measurement that quantifies the functionality delivered by a software application.
* **Use**: Helps in estimating the size, complexity, and delivery of software.

**F) What is SRS?**

* **Full Form**: Software Requirements Specification
* **Purpose**: A document that outlines all expected functionalities, features, and constraints of a software system.

**G) Define CFG.**

* **Full Form**: Context-Free Grammar
* **Definition**: A set of recursive rules used to generate patterns of strings in a formal language.

**H) Write Down One Disadvantage of the Spiral Model.**

* **Disadvantage**: Can be complex and costly to manage due to its iterative nature, leading to potential challenges in planning and predicting project costs and timelines.

**2022**

**a) Full Form of COCOMO**

* **COCOMO**: Constructive Cost Model

**b) Difference Between Computer Program and Computer Software**

* **Computer Program**:
  + A specific set of instructions written to perform a task.
  + Typically refers to a single application or script.
* **Computer Software**:
  + A broader term that encompasses all types of programs, libraries, and systems that facilitate computer operations.
  + Includes applications, operating systems, and utilities.

**c) Finding the Size of a Software Product**

* **Methods**:
  + **Lines of Code (LOC)**: Counting the number of lines in the program.
  + **Function Points**: Assessing the functional requirements based on inputs, outputs, and user interactions.
  + **Use Case Points**: Evaluating the complexity and number of use cases.

**d) Modular Cohesion**

* **Definition**:
  + A measure of how closely related and focused the responsibilities of a single module are.
  + Higher modular cohesion indicates a well-structured module, where all components contribute to a single task or purpose.

**e) Software Scope**

* **Definition**:
  + Refers to the boundaries of a software product, including the range of functions, features, and limitations.
  + Defines what is included in the project and what is excluded, guiding the development process.

**f) Names of Two Project Management Tools**

* **Examples**:
  + **Trello**
  + **Asana**

**g) Modularization**

* **Definition**:
  + The process of dividing a software system into small, manageable, and interchangeable components (modules).
  + Aids in simplifying development, maintenance, and understanding of the software.

**h) CASE Tools**

* **Definition**:
  + Computer-Aided Software Engineering tools that help in software development processes.
  + Includes tools for automated code generation, design modeling, testing, and project management.

**2023**

**a) What is software engineering?**

* Software engineering is the application of engineering principles to software development in a methodical way. It involves the use of systematic methods, tools, and techniques to create software that meets requirements.

**b) What is the full form of CMMI?**

* Capability Maturity Model Integration.

**c) Write down one advantage of the waterfall model.**

* One advantage of the waterfall model is its straightforwardness and clarity, making it easy to understand and manage since each phase has specific deliverables.

**d) What is computer software?**

* Computer software is a collection of data or computer instructions that tell the computer how to work. It can be categorized into system software, application software, and middleware.

**e) What are function points?**

* Function points are a standardized unit of measure that quantifies the functional requirements of software based on the business functions it delivers to users.

**f) What is SRS?**

* SRS stands for Software Requirements Specification, a document that describes the intended purpose and environment for software under development.

**g) Define ERD.**

* ERD stands for Entity-Relationship Diagram, which visually represents the relationships between entities in a database, illustrating the structure of the data.

**h) What is the full form of SDLC?**

* Software Development Life Cycle.

**2023-24**

**a) What do you mean by software engineering?**

* **Definition**: Software engineering is a systematic approach to the development, operation, maintenance, and retirement of software. It involves applying engineering principles to software creation, emphasizing processes, methods, and project management.

**b) Define SDLC.**

* **SDLC (Software Development Life Cycle)**: A framework defining the stages of software development, including:
  + **Requirement Analysis**
  + **Design**
  + **Implementation**
  + **Testing**
  + **Deployment**
  + **Maintenance**

**c) What is a Function Point?**

* **Definition**: Function points are a standardized unit of measurement that quantifies the functionality of software based on the requirements and the user's perspective. They help in estimating development effort and productivity.

**d) What is Programming?**

* **Definition**: Programming is the process of creating a set of instructions that a computer follows to perform specific tasks. It involves writing code in various programming languages, such as Python, Java, C++, etc.

**e) Define SRS.**

* **SRS (Software Requirements Specification)**: A document that describes the intended behavior of a software system. It details functional and non-functional requirements, providing a blueprint for development.

**f) Define Waterfall model.**

* **Waterfall Model**: A sequential software development process where progress flows in one direction — resembling a waterfall. Key phases include:
  + Requirement Analysis
  + System Design
  + Implementation
  + Verification (Testing)
  + Maintenance

**g) Define COCOMO model.**

* **COCOMO (Constructive Cost Model)**: An algorithmic software estimation model used to predict the cost, effort, and schedule associated with software development. It considers project size, complexity, and team capability.

**h) What is DFD?**

* **DFD (Data Flow Diagram)**: A graphical representation used to visualize the flow of information within a system. It shows how data moves between processes, data stores, and external entities, helping in system analysis and design.

**INTERNAL 2023-24**

1. **Important Features of Software**
   1. Software should be reliable, efficient, and maintainable.
   2. User-friendly interface and performance issues are vital.
   3. It must be adaptable and scalable.
2. **Technical Feasibility**
   1. It assesses whether a project can be developed with the available technology and resources.
   2. Involves evaluating hardware, software, and operational requirements.
3. **Limitations of Classical Waterfall Model**
   1. Rigid structure; difficult to accommodate changes once a phase is completed.
   2. Poor for projects where requirements are unclear or evolve.
   3. Late testing may lead to discovering fundamental issues too late.
4. **Software Metric**
   1. A standard of measurement used to quantify various properties of software products.
   2. Metrics could include lines of code, function points, complexity measures, etc.
5. **Thrown-Away Prototyping**
   1. A development approach where a prototype is built, used to gather feedback, and then discarded.
   2. Helps refine requirements before building the actual product.
6. **Coupling**
   1. A measure of how closely connected different modules or components of a software system are.
   2. Low coupling is desirable for better maintainability and flexibility.
7. **Functional Independence**
   1. The degree to which a module performs a single task without relying on other modules.
   2. Enhances module reusability, maintainability, and reduces the complexity of interactions.