

MODULE-1 ASSIGNMENT

1. Explain in your own words what a program is and how it functions. What is programming ?

- Programming is the process of creating instructions for a computer to execute , typicall by writing code in a programming languages .
- It involves designing,writing,testing,and maintaining software to solve specific problems.

2. What are the key steps involved in the programming process?

Types of Programming Languages.

- Defining the problem, designing a solution (including algorithm development), coding the solution, testing and debugging, and finally, documentation and maintenance.
- C#,Python,Javascript,SQL,there are programming languages.

3. What are the main differences between high-level and low-level programming languages .

World Wide Web & How Internet Works.

- High-level languages are more human-readable and easier to use, with a focus on problem-solving, while low-level languages are closer to the hardware, offer more control, and require greater expertise.
- The Internet is the global network of interconnected computer networks, while the World Wide Web (WWW) is a way of

accessing information over that network, using hyperlinks and a specific set of protocols

4. Describe the roles of the client and server in web communication.

Network Layers on Client and Server.

- The client (like a web browser) initiates requests for resources (like web pages) from a server. The server, in turn, processes these requests and sends back the requested information or performs the requested action.

5. Explain the function of the TCP/IP model and its layers.

Clients and server.

- It is a conceptual framework . It divides the communication process into four layers: Application, Transport, Internet, and Network Access.
- Each layer handles specific tasks, ensuring reliable and efficient data transfer between client and server applications.

6. Explain Client Server Communication Types of Internet Connections.

- Client-server communication is a network model where clients request services or resources from a central server, which then provides the requested information or performs the requested action.
- Types of internet connections include broadband (DSL, cable, fiber , satellite, and dial-up.)

7. How does broadband differ from fiber-optic internet Protocols?

- Broadband is a general term for high-speed internet access, while fiber-optic internet is a specific technology used to deliver that high-speed access.

8. What are the differences between HTTP and HTTPS protocols?

Application Security .

- HTTP (Hypertext Transfer Protocol) transmits data in plain text, making it vulnerable to interception and manipulation by malicious actors. HTTPS (Hypertext Transfer Protocol Secure) uses encryption, typically through SSL/TLS certificates, to protect data during transmission, significantly enhancing security.

9.What is the role of encryption in securing applications? Software

Applications and Its Types

- Encryption is a critical component in securing software applications, playing a vital role in protecting sensitive data and ensuring confidentiality, integrity, and authenticity.

10. What is the difference between system software and application software? Software Architecture.

- System software manages and controls computer hardware, providing a platform for other software to run, while application software is designed for specific tasks or user needs.

11. What is the significance of modularity in software architecture?

Layers in Software Architecture

- Modularity in software architecture is significant because it enhances code organization, reusability, and maintainability, allowing for easier testing, debugging, and deployment.

12. Why are layers important in software architecture? Software Environments.

- enhancing modularity, scalability, maintainability, and testability. They promote a separation of concerns, allowing developers to focus on specific functionalities within each layer, leading to cleaner code and easier troubleshooting.

13.Explain the importance of a development environment in software production source code .

- enhancing modularity, scalability, maintainability, and testability. They promote a separation of concerns, allowing developers to focus on specific functionalities within each layer, leading to cleaner code and easier troubleshooting.

14. What is the difference between source code and machine code? Github and Introductions.

- Source code is human-readable instructions written in a programming language, while machine code is the binary code directly executable by a computer's processor.
- GitHub is a cloud-based platform where you can store, share, and work together with others to write code.

15. Why is version control important in software development?

Student Account in Github.

- It efficiently manages and tracks changes to code over time, facilitating team collaboration and project progress tracking.
- Under "Select your role in education", select Student. In the "Application" dialog, enter the name of your school. If required, select or add the email address you use for school. Click Continue, and then you will be prompted to upload proof of your academic status.

16. What are the benefits of using Github for students? Types of Software.

- GitHub Education offers students real-world experience with free access to various developer tools from GitHub's partners.
- Software can be broadly categorized into system software and application software.

17. What are the differences between open-source and proprietary software?

GIT and GITHUB Training.

- Open-source software's source code is freely available for anyone to use, modify, and distribute, while proprietary software's source code is kept secret and controlled by the developer or company.
- Git is a free, open source version control tool that developers install locally on their personal computers, while GitHub is a pay-for-use online service built to run Git in the cloud.

18. How does GIT improve collaboration in a software development team? Application Software

- providing a structured framework for version control and code management. Its branching and merging capabilities allow developers to work on separate features simultaneously

19. What is the role of application software in businesses? Software Development Process.

- Application software plays a crucial role in modern businesses by streamlining operations, improving efficiency, and facilitating better decision-making. It enables businesses to automate tasks, manage data, enhance communication, and adapt to changing market demands.

20. What are the main stages of the software development process? Software Requirement.

- Application software plays a crucial role in modern businesses by streamlining operations, improving efficiency, and facilitating better decision-making.

21. Why is the requirement analysis phase critical in software development? Software Analysis.

- It ensures that the software being built aligns with the needs and expectations of its users and stakeholders.

22. What is the role of software analysis in the development process? System Design

- Software analysis is a crucial phase in the software development process, acting as a bridge between initial requirements and the eventual implementation.

23. What are the key elements of system design? Software Testing

- System design in software engineering involves creating a blueprint for a software system structure include architecture, database design, APIs, caching, load balancing, security, scalability, and redundancy.
- Software testing, including functional and non-functional testing (like unit testing, integration testing and acceptance testing,)

24. Why is software testing important? Maintenance

- Software testing is crucial for ensuring a product works correctly, meets user expectations, and is free of bugs. It helps identify issues early, improve software quality, and enhance user experience. In the context of maintenance, it ensures the continued reliability and effectiveness of software as it evolves over time.

25. What types of software maintenance are there? Development

- There are four main types of software maintenance: corrective, adaptive, perfective, and preventive. Corrective maintenance addresses bugs and errors. Adaptive maintenance ensures the

software works with new environments. Perfective maintenance improves functionality and performance. Preventive maintenance proactively addresses potential issues to prevent future problems.

26. What are the key differences between web and desktop applications?

- Web applications run in a browser and are accessible from any device with an internet connection, while desktop applications are installed directly on a specific computer and don't require an internet connection for basic functionality.

27. What are the advantages of using web applications over desktop applications?

- Due to their accessibility, ease of updates, and platform independence. Web apps can be accessed from any device with a web browser and an internet connection.

28. What role does UI/UX design play in application development?

- UI/UX design plays a critical role in application development by shaping how users interact with an app, influencing their satisfaction, and ultimately impacting the app's success.

29. What are the differences between native and hybrid mobile apps?

- Native apps are built specifically for one platform (iOS or Android) using platform-specific languages and offer optimal performance and access to device features.

30. What is the significance of DFDs in system analysis?

- They visually represent how data moves through a system, aiding in understanding, analysis, and improvement. DFDs clarify the flow of information.

31. What are the pros and cons of desktop applications compared to web applications?

- Desktop applications generally offer faster performance and offline functionality, while web applications excel in accessibility, ease of updates, and cross-platform compatibility.

32. How do flowcharts help in programming and system design?

- Flowcharts serve as a powerful visual tool in programming and system design by providing a clear and structured representation of processes and logic.
- Their benefits include: Visualizing Logic and Flow, Planning and Design, Problem Identification and Optimization, Communication and Collaboration.