**MODULE – 1 QUESTIONS:**

Q.1: What is Program?

Ans: Program is a set of instruction.

Q.2: Explain in your own words what a program is and how it functions?

Ans: Program is set of instruction. And its work to take a particular output.

Q.3: What is Programming?

Ans: Programming is a process to give a program.

Q.4: What are the key steps involved in the programming process?

Ans:

1. Problem Definition
2. Program Design
3. Coding
4. Debugging
5. Testing
6. Documentation
7. Maintenance

Q.5: Types of Programming Languages.

Ans:

1. C and C++
2. Java
3. Python
4. Pascal
5. Basic

Q.6: What are main differences between high-level and low-level programming languages?

Ans:

|  |  |  |
| --- | --- | --- |
|  | High Level | Low Level |
|  | High Level languages are more user-friendly, closer to natural language and platform-independent. | Low Level languages are more machine-friendly, directly related to hardware, and can be more challenging to use. |
| Abstraction | High Level languages abstract away the complexities of the hardware making them easier to learn and use. | Low Level languages on the other hand, are closer to the machine’s instruction set and require a deeper understanding of hardware. |
| Machine friendliness | High Level languages are designed for human readability and are easier to understanding and debug. | Low Level languages are more machine-friendly and optimized for direct hardware control. |
| Portability | High Level languages generally portable, meaning they can compile and run of different platforms with minimal changes. | Low Level languages are often machine-specific and not easily portable. |
| Development time | High Level languages often allow for faster development time due to their ease of use and readily available libraries | Low Level languages may require more coding and debugging, increasing development time. |
| Performance | High Level languages may sacrifice some performance for ease of use and platform independence. | Low Level languages can be faster and more memory-efficient because they provide more different control over hardware. |
| Examples | Python, Java, c#, JavaScript, Ruby | Assembly Language, Machine code. |

Q.7: World Wide Web & How Internet Works

* **World Wide Web:**
* WWW is known as web.
* WWW is a collection of websites or webpages stored in web server.
* Connected to load computers through the internet.
* These websites contains text pages, digital images, audios, videos etc.
* Users can access the content of these sites from any part of the world over the internet using their devices such as computers, laptop, cell phones etc.
* The WWW along with internet enables the retrievals and displays of text and media to your devices.
* **How Internet Works:**
* Computers connect to each other and to the internet via wires, cables, radio waves, and other types of networking infrastructure.
* All data sent over the internet is translated into pulses of light or electricity, also called “bits”, and then interpreted by the receiving computer.

Q.8: Describe the roles of the client and server in web communication & Network layers on Client and Server.

Ans:

* **Client “Sometimes On”**
* Initiate a request to the server when interested.
* Eg. Web browser on your laptop or cell phone.
* Doesn’t communicate directly with other clients.
* Need to know the server’s address.
* **Server is “Always On”**
* Services request from many clients hosts.
* Eg. Web servers for the [www.example.com](http://www.example.com) websites.
* Doesn’t initiate contract with the clients.
* Needs a fixed, well known address.

Q.9: Explain the functions of the TCP/IP model and its layers.

Ans: The TCP/IP model is a layered model that facilitates data communication across networks. It comprises Four Layer. Application, Transport, Internet and Network Access. Each layer performs specific functions to ensure reliable and efficient data transmission.

Q.10: Client and Servers

Ans:

**Client**: A system that uses remote services from a server. Some clients have limited disk storage capacity, or perhaps none at all, and they have to rely on remote file systems from a server to function.

Server: A system that provides services to other systems in its network. There are file servers, boot servers, database servers, license servers, print servers, installation servers, and even servers for particular applications. This chapter uses the term server to mean a system that provides file systems and installation software for other systems on the network..

Q.11: Explain Client Server Communication.

Ans: Client Server communication describes a network architecture where a client requests resources or services from a server.

The server then processes the request and sends a response back to the client. This model enables a client to interact with central point for data management and application services.

Q.12: Types of Internet Connections.

Ans:

1. Digital Subscriber Line(DSL)
2. Cable Internet
3. Fiber Optic
4. Satellite Internet
5. Wireless
6. Broadband over Power Line(BPL)

Q.13: How does broadband differ from fiber-optic internet?

Ans: The broadband connection is less reliable compared to fibre optic.

The broadband connection is fast speed internet, but the speed is slower when it is compared to a fibre connection.

Q.14: Protocols:

Ans:

* A network protocol is a group of rules accompanied by the network.
* Network protocols will be formalized requirements and plans composed of rules procedures and types that describe communication among a couple of devices over the network.

Q.15: What are the differences between HTTP and HTTPS protocols?

Ans: HTTP and HTTPS are both protocols for transferring data over the web, but HTTPS is the secure version of HTTP. The key difference is in the encryption used for data transmission.

Q.16: Application Security

Ans:

* Application security refers to security precautions used to the application level to prevent the theft or hijacking of data or code within the application.
* It includes securing concerns made during application development and design, as well as methods and procedures for protecting applications once they’ve been deployed.

Q.17: What is the role of encryption in securing applications?

Ans: Encryption plays a crucial role in securing applications by transforming data into an unreadable format, known as cipher text, making it inaccessible to unauthorized individuals. This protection ensures data confidentiality, prevents data breaches, and safeguards sensitive information.

Q.18: Software Applications and its types

Ans: Application software is software designed for users to perform specific tasks, such as creating documents, browsing the web, or managing data. It's categorized into various types, including word processors, web browsers, spreadsheets, multimedia software, and more. These applications can be further classified by their purpose (business, educational, etc.) or by how they are distributed (freeware, shareware, etc.)

Types of Application Software:

1. General purpose Application
2. Business Applications
3. Educational Applications
4. Other Applications

Q.19: What is the difference between system software and application software?

Ans:

|  |  |
| --- | --- |
| System Software | Application Software |
| Manages computer resources and operations. | Perform specific tasks for end-users. |
| Ex: Windows, Linux, macOS, antivirus software. | Ex: Microsoft Office, web browsers, media players. |

Q.20: Software Architecture:

Ans:

* Software architecture is blueprint of building software.
* It shows the overall structure of software, the collection of component in it, and how they interact with one another while hiding the implementation.

Q.21: What is significance modularity in software architecture?

Ans: Modularity in software architecture is significant because it promotes organization, reusability, maintainability, and scalability. By breaking down complex systems into smaller, independent modules, developers can create more manageable, understandable, and easily extensible code. This approach simplifies development, testing, and maintenance, leading to faster development cycles and higher quality software.

Q.22: Layers in software architecture

Ans: There are 5 types of layers in software architecture

1. Presentation layer
2. Application layer
3. Business layer
4. Persistence layer
5. Database layer

Q.23: Why are layers important in software architecture?

Ans: Layers are crucial in software architecture because they promote modularity, separation of concerns, and scalability, making software easier to develop, maintain, and modify.

By organizing code into distinct layers with specific responsibilities, developers can create more maintainable flexible systems.

This also simplifies testing and debugging, as issues can be isolated to specific layers.

Q.24: What is difference between source code and machine code?

Ans:

|  |  |  |
| --- | --- | --- |
| **Feature** | **Source Code** | **Machine Code** |
| Readability | Human-readable, written in programming languages like Python, Java, C++. | Not human-readable, consisting of binary code (0s and 1s). |
| Target | For programmers to write and understand programs. | For the computer's CPU to directly execute instructions. |
| Creation | Written by programmers using a text editor or IDE. | Generated by a compiler or assembler from source code. |
| Execution | Needs to be translated into machine code before execution. | Executed directly by the CPU. |

Q.25: What are the main stages of the software development process?

Ans:

1. Planning
2. Analysis
3. Design
4. Development
5. Testing
6. Deployment
7. Maintenance

Q.26: What types of software maintenance are there?

Ans:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance
4. Preventive Maintanance

Q.27: What are the advantages of using web application and desktop applications?

Ans:

* **Web Application:**

Accessibility

Easy Updates

Cost-effective Development

Scalability

Cross-platform Compatibility

* **Desktop Application:**

Performance

Official Functionality

Hardware Compatibility

Control and Customization

Increased Security

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

Ans: UI/UX Design, referred to as “User Interface/User Experience design,” has become very important in the field of application design focusing on creating a visual interface and improving the experience of the user. The role of a UI/UX designer is to ensure that the application or software is user-friendly and visually appealing for the user.

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

Ans:

Native:

Native apps are built specifically for a particular operating system (like iOS or Android), utilizing platform-specific languages and features.

Hybrid:

Hybrid apps are built using web technologies (HTML, CSS, JavaScript) and can run on multiple platforms with a single codebase.

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

Ans: Data Flow Diagram (DFDs) are crucial in system analysis because they provide a visual structured way to understand and model the flow of data within a system or process.

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

Ans:

Web Application:

Pros:

Offline functionality

Higher performance

Enhance security

Greater control

Cons:

Limited accessibility

Manual update

Higher development

Maintanance cost

Desktop Application:

Pros:

Accessibility

Automatic updates

Cost-effective

Flexibility

Cons:

Internet dependency

Performance limitation

Security concerns

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

Ans: Flowcharts aid in programming and system design by visually representing the logic and flow of processes, helping to clarify steps, identify potential issues, and facilitates communication among developers.