Name:- *Yash Kumar*

Q1. Write a simple “hello world” program in two diffrent programming languages of your choice. Compare the structure and syntax.

: - .

C language

#include<stdio.h> int
main()

{

Printf(“Hello world”); return0

}

C++ language

#include<iostream>

Using namespace std;

Int main()

{

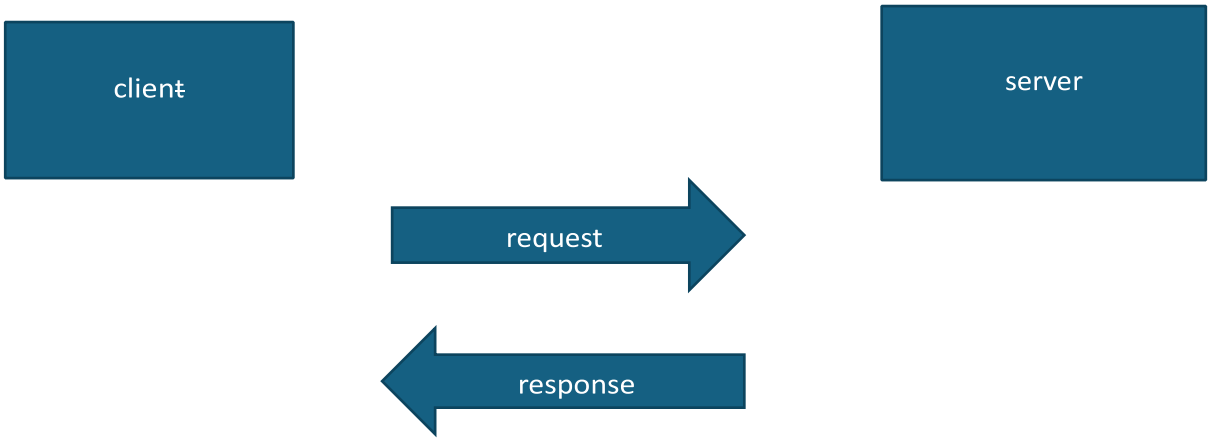
Cout<<”hello world”;

Return 0;

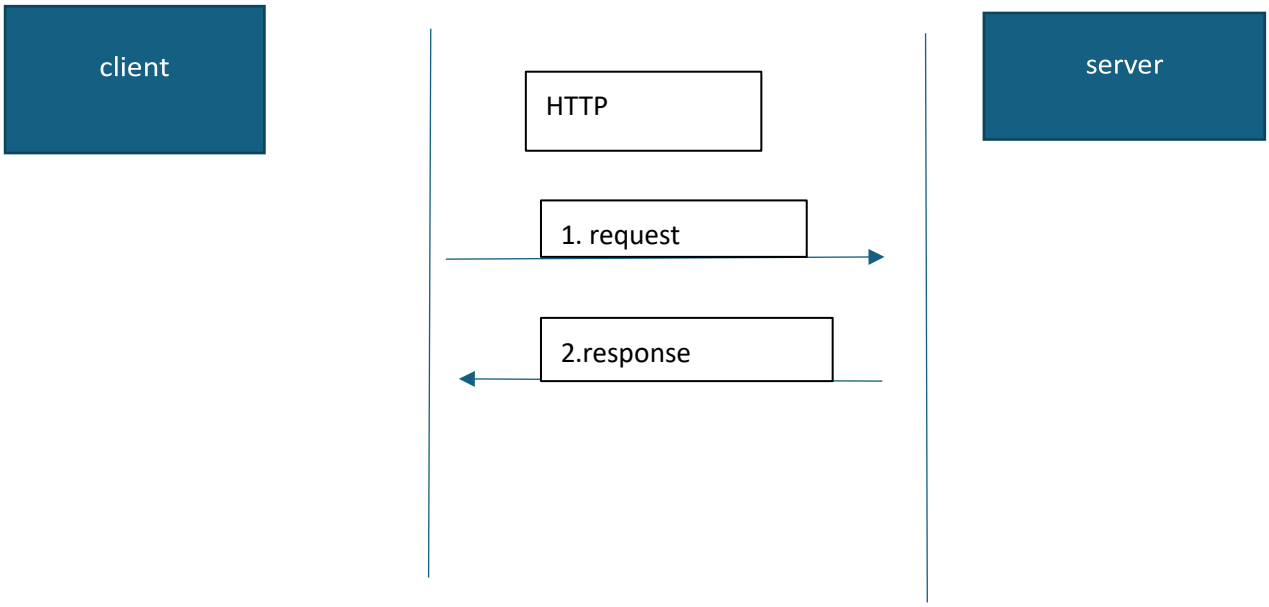
}

Q2. Research and create a diagram of how data is tr : - mitted from a client to a server over the internet.

: - . Client to server data tr : - fer over the internet involves a set of protocols and interactions that enable devices and servers to communicate and exchange information.



Q3. Design a simple HTTP client-server communication in any language.



Q4.Research different types of internet connections (e.g., broadband, fiber, Satellite)and list their pros and cons.

: - - 1.Broadband Pros-

widely available,affordable,stable

Cons-slower than fiber, speed depend on location.

2.fiber optic

Pros-very fast, reliable, low latency.

Cons- expensive , limited in rural areas.

3.satelite

Pros- avilable in remote areas,no cabl e

needed

Cons-weather dependent, costly

4. Wireless (4G/5G)

Pros-use for mobile ,fast ,easy to set up

Cons-signal issues,coverage

Q5. Simulate HTTP and FTP requests using command line tools (e.g., curl).

: - - For HTTP requests using curl:

Basic GET request.

Code- curl https://www.example.com

Post request

Code- curl -X POST -H “Content-Type: application/json” -d ‘{“key”: “value”}’ https://api.example.com/endpoint

Specifying headers

Code- curl -H “Authorization: Bearer your_token” https://api.example.com/protectedresource

For FTP requests using curl:

List directory contents

Code- curl -u username:password -T – ftp://server.com/path/to/directory

Upload a file

Code-curl -T your_file.txt -u username:password ftp://server.com/path/to/upload

Download a file

Code-curl -O -u username:password ftp://server.com/path/to/file.txt

Q6. Identify and explain three common application security vulnerabilities. Suggest possible solutions.

: - - 1.injection attack

Injection occurs when untrusted data is sent to an interpreter as part of a command or query.

Solution- input validation: validate and sanitize all user input to prevent malicious data.

Parameterized Queries: Use prepared statements with parameterized queries

2. Cross site scripting(XSS)

: - - XSS allows attackers to inject malicious scripts into web pages viewed by other users.

Solution- Libraries: Use libraries like DOMPurify for sanitizing inputs.

Output Encoding: Encode output to ensure it is displayed as text, not executed as code.

3. Broken authentication

This vulnerability occurs when authentication mechanisms are improperly implemented, allowing attackers to compromise user accounts Ex-weak password,missing multi-factor authentication.

Solution-Strong Password Policies: Enforce complex and unique passwords.

Multi-Factor Authentication (MFA): Add an extra layer of security.

Q7.Identify and classify 5 applications you use daily as either system software Or application software.

1. Google chrome
   Classification: application software.
   Reason: browse the internet and access web base service.

2.Microsoft word

Classification: application software

Reason: use for creating and editing document.

3.windows operating system

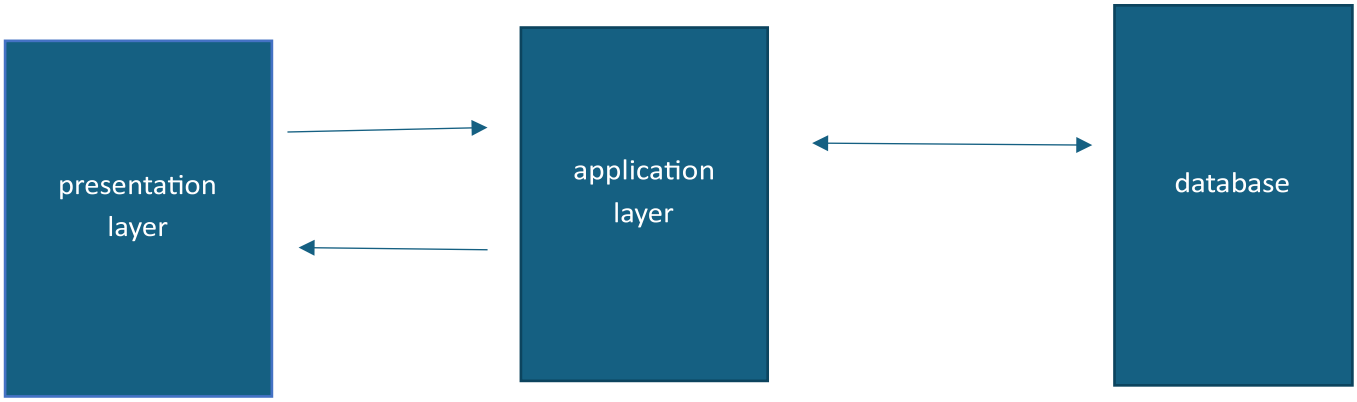
Classification: system software

Reason: it manages hardware and software.

4.Antivirus

Classification: system software

Reason: it provides security to the system.



5.Vs code

Classification: application software

Reason: it is code editor design to write and edit the code.

Q8. Design a basic three-tier software architecture diagram for a web application.

Q9. Create a case study on the functionality of the presentation, business logic, and Data access layers of a given software system.

: - .

1.presentation layer 2.business logic layer 3. Data access layer

1.presentation layer

The front end interface where users interact with the application. It handels displaying data and receiving user input and making request to business logic layer.

2.business logic layer

The core of the application that process user requests applies business rules and communicate with the data access layer.It handles tasks like order processing, payment handling, inventory management, and user authentication.

3.Data access layer

Manages all databases interactions and ensure data persistence. It retrivies and store information like product details,user profiles and order history.

Q10.Explore different types of software environments (development, testing, Production).Set up a basic environment in a virtual machine.

: - .

1.Development Environment

2.Testing environment

3.Production Environment

1. Development environment

Purpose: This is where developers write ,test and debug the application code

Tools: IDEs such as visual studio code

2.Testing environment

Purpose: The testing environment is used to verify that the application behaves as expected under different scenarios.

Tools: Testing framework such as Junit

3.production environment

Purpose: The production environment is where the final application runs and is accessible by end users. This is the live environment.

Tools:Monitoring tools like relic Setup:

1.Install Virtual Machine Software

2.Create a New Virtual Machine

3.Install the Operating System

4.Set Up Development Tools (For Development Environment)

5.Set Up Testing Tools (For Testing Environment)

6.Set Up Production Environment (For Production Environment)

7.Finalize and Test

Q10.Write and upload your first source code file to Git hub.

1. Create a GitHub account
2. Create a new repository for your project.
3. Upload a files to your project's repository.

Q11.Create a Git hub repository and document how to commit and push code changes.

: - . git push: Pushes changes from the local repository to GitHub.

Git commit -m "message":

Q12. Create a student account on Git hub and collaborate on a small project with A classmate.

: - .

1.Go to the repository you just created.

2.Click on the Settings tab in the repository.

3.Scroll down to the Collaborators section on the left sidebar.

4.Under Manage access, click Invite a collaborator.

5.Search for your classmate's GitHub username and invite them to your repository.

6.Your classmate will receive an invitation via email or GitHub notifications. They need to accept the invitation to collaborate on the repository.

Q13.Create a list of software you use regularly and classify them into the Following categories: system, application, and utility software.

: - .

1.System software provides basic functions that allow hardware and application software to work together.

Ex: operating system, device drivers

2.Application Software

Application software is used by end-users to perform specific tasks, such as creating documents, editing photos, or browsing the web.

Ex: web browser, visual studio code

3.Utility software

Utility software helps to manage, maintain, and control system resources, improve efficiency, or enhance system performance.

Ex: antivirus, file compression.

Q14.Follow a GIT tutorial to practice cloning, branching, and merging repositories : - .

1.Cloning: git clone https://github.com/yourusername/git-tutorial.git

2.Branching:

Git checkout -b feature-branch

3.Merging:

git merge feature-branch

Q15.Write a report on the various types of application software and how they Improve productivity.

: - .

1.Productivity Software: Tools like Microsoft Office and Google Workspace streamline document creation, data analysis, and collaboration.

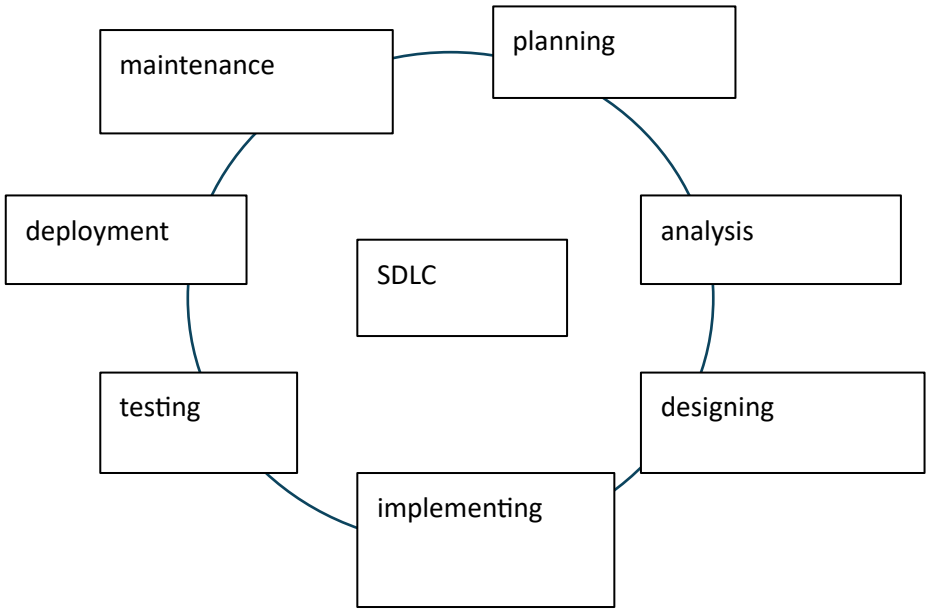
2.Utility Software: Programs like antivirus and file management tools optimize system performance and prevent downtime.

3.Web Browsers: Browsers facilitate quick access to online resources, supporting research and communication.

4.Communication Software: Platforms like Zoom and Slack enable seamless remote collaboration, saving time and resources.

5.education software : Platforms like Moodle and Coursera provide on-demand learning, reducing time spent on traditional education.

Q16. Create a flowchart representing the Software Development Life Cycle (SDLC).



Q17. Write a requirement specification for a simple library management system.

: - .

1.user management: Handel library members,staff with secure login system

2.book management:admin can add,update or remove book.user can search for the book and check for the avilability.

3.borrowing& returning: track issued book,due dates.

Q18.Perform a functional analysis for an online shopping system.

1.registration and login: new user register their information . existing user can login in system.

2.product info/details: display product image, name, price, availability.

3. shopping cart: can store multiple item in cart add or remove the product or save for it later.

4.Order management: collect delivery address, show order tracking and past order details.

5.payment option& payment gateway: support multiple payment option, secure payment gateway.

6.inventory management:

Track inventory ,add, update or delete the product and product availability.

Q19.Design a basic system architecture for a food delivery app.

1.User interface(Ui):

Customer: for browsing restaurant.

Placing &tracking order.

Delivery: managing deliveries and navigation.

2.Backend service:

Handle authentication, order management, restaurant Data.

3.database:user profiles, restaurant menu , order history.

4.payment:payment gateway for secure tr : - action.

Q20.Develop test cases for a simple calculator program.

: - : To test case calculator you can test.

1. Basic arithmetic:addition , subtraction , multiplication , division .

2.complax case: divide by zero Multiple calculation (ex :2+3*5)

3.handeling decimal input

4.error handling : invalid operator, alphabets.

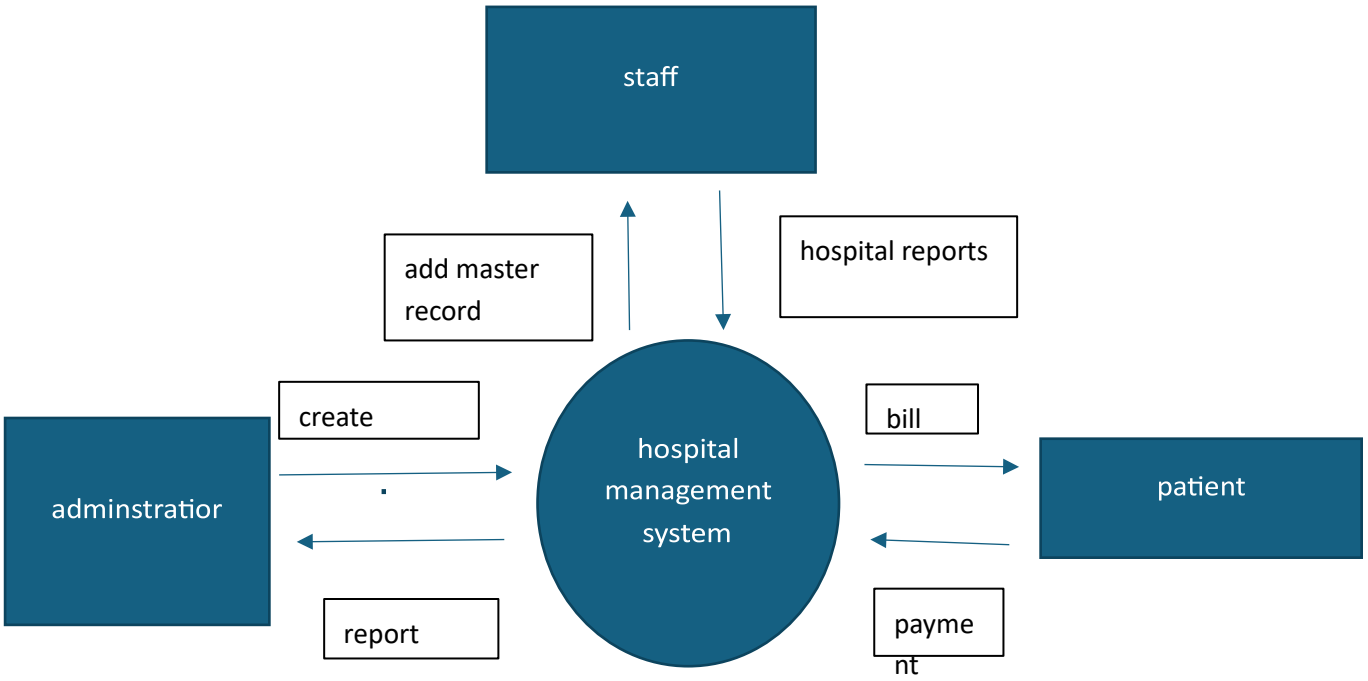
Q21. Document a real-world case where a software application required Critical maintenance.

: - . in 2017 Microsoft face critical issue with a windows 10 creator update, which cause system crashes, performance issue and driver compatibility problems for users

Response: Microsoft provided a rollback option, released patch updates to fix bugs and security flaws, and improved its update testing process.

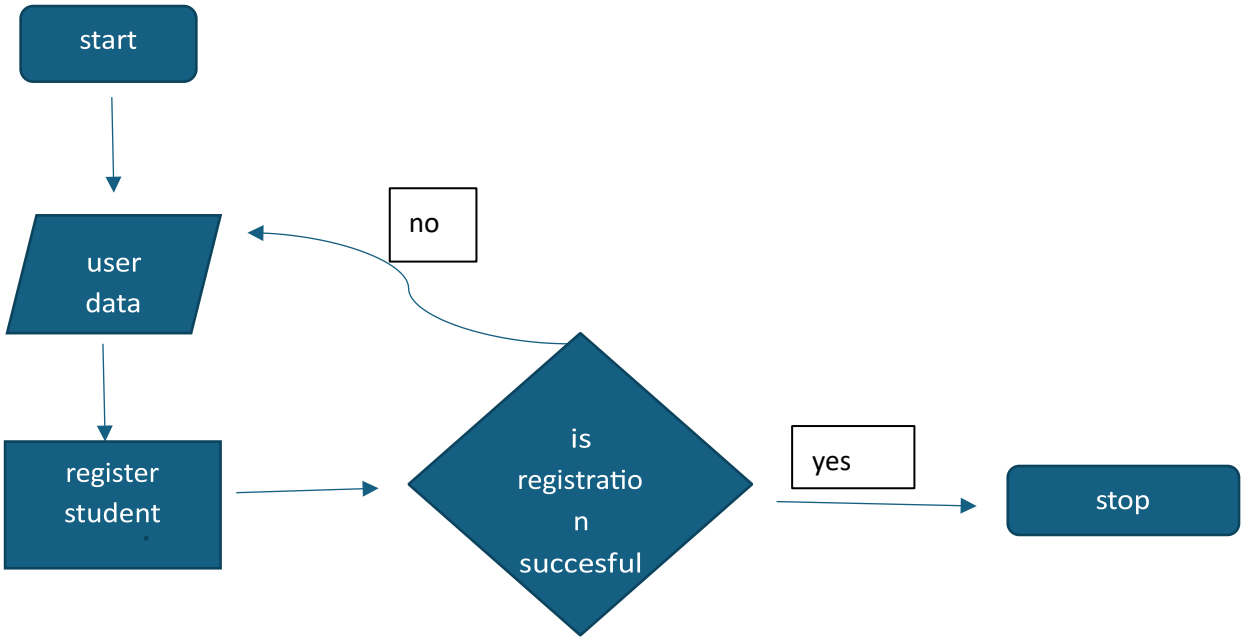
Q22. Create a DFD for a hospital management system.

: - .



Q23. Build a simple dekstop GUI library.

Q24. Draw a flow chart representing the logic of basic onlline registrartion system.


start
user data
no
register student
is registratio n succesful
yes
stop