

Marking Scheme-Practice Paper 4

COMPUTER SCIENCE (CPU)

(SUBJECT CODE: 906)

Note: Apart from the marking instructions, a teacher can evaluate at his discretion.

नोट: अंकन निर्देशों के अलावा, एक शिक्षक अपने विवेकानुसार मूल्यांकन कर सकता है।

Maximum Marks: 40

Time: 2:30 hours

General Instructions:

- This question paper is divided into 4 Sections - A, B and C and D.
- Section A** consists of 1 question (10 parts -Objective Type of 1 mark each).
- Section B** consists of 4 questions (2-5). Each question carries 1 mark.
- Section C** consists of 5 questions (6-10). Each question carries 2 marks.
- Section D** consists of 4 questions (11-14). Each question carries 4 marks.

Internal choices are given in all 4 questions.

		SECTION A (Each part of the question carries 1 Mark)	
1(i)		Which of the following is not a web browser? निम्नलिखित में से कौन सा एक वेब ब्राउज़र नहीं है? a) Chrome / क्रोम b) Firefox / फ़ायरफ़ॉक्स c) Photoshop / फोटोशॉप d) Safari / सफारी	1
	Ans	c) Photoshop / फोटोशॉप	
		1 Mark for correct identification.	
1(ii)		Which SQL function is used to retrieve the minimum value from a given column? किसी दिए गए column से न्यूनतम मान प्राप्त करने के लिए किस SQL फ़ंक्शन का उपयोग किया जाता है? a) AVG() b) MIN() c) MAX() d) COUNT()	1
	Ans	b) MIN()	
		1 Mark for correct identification.	
1(iii)		What does the HTML tag represent? HTML टैग क्या दर्शाता है? a) Bold / बॉल्ड	1

		b) Brown / ब्राउन c) Break / ब्रेक d) Black / ब्लैक	
	Ans	c) Break / ब्रेक	
		1 Mark for correct identification.	
1(iv)		If the elements "G", "I", "R", and "L" are placed in a stack and are removed one at a time, in what order will they be removed? यदि element "G", "I", "R", और "L" को एक stack में रखा जाता है और एक बार में हटा दिया जाता है, तो उन्हें किस क्रम में हटाया जाएगा? a) L,R,I,G b) G,I,R,L c) L,I,G,R d) R,I,G,L	1
	Ans	a) L,R,I,G	
		1 Mark to identify the order of removal from stack.	
1(v)		_____ tag is used to add hyperlink in HTML. _____ टैग का उपयोग HTML में हाइपरलिंक जोड़ने के लिए किया जाता है।	1
	Ans	<a>	
		1 Mark for writing the correct HTML tag. Note: Deduct ½ mark for not writing < > symbol.	
1(vi)		C++ uses the _____ operator for performing AND logical operations. C++ में AND लॉजिकल ऑपरेशन को करने के लिए _____ ऑपरेटर का उपयोग होता है।	1
	Ans	&&	
		1 Mark for writing correct operator. Do not give marks if only single & is given.	
1(vii)		Destructor is a function that initializes an object when it is created. True/ False डिस्ट्रक्टर एक फ़ंक्शन है जो किसी ऑब्जेक्ट को बनाने पर उसे प्रारंभ करता है। सत्य/असत्य	1
	Ans	False असत्य	
		1 Mark for correct identification.	
1(viii)		Images in an HTML webpage are added using the tag. True/ False	1

		HTML में टैग का उपयोग करके इमेज जोड़ी जाती हैं । सत्य/असत्य	
	Ans	False असत्य	
		1 Mark for correct identification.	
		<p>Direction: In the questions given below, there are two statements marked as Assertion (A) and Reason (R). Choose the correct option out of the choices given below in each question:</p> <p>निर्देश: नीचे दिए गए प्रश्नों में, कथन (A) और कारण (R) के रूप में चिह्नित दो कथन हैं। प्रत्येक प्रश्न में नीचे दिए गए विकल्पों में से सही विकल्प का चयन कीजिए:</p>	
1(ix)		<p>Assertion (A): Using VARCHAR in SQL helps save storage space.</p> <p>Reason (R): Unlike some other types, VARCHAR adjusts storage based on how much data is actually stored, preventing unnecessary space usage.</p> <ol style="list-style-type: none"> Both (A) and (R) are correct and (R) is correct explanation of (A). Both (A) and (R) are correct and (R) is not the correct explanation of (A). (A) is true but (R) is false. (A) is false but (R) is true. <p>कथन (A): SQL में VARCHAR डेटा प्रकार का उपयोग स्टोरेज स्थान को बचाने में मदद करता है।</p> <p>कारण (R): कुछ अन्य प्रकारों के विपरीत, VARCHAR वास्तविक डेटा के आधार पर स्टोरेज को समायोजित करता है, जिससे अनावश्यक स्टोरेज का उपयोग नहीं होता है।</p> <ol style="list-style-type: none"> (A) और (R) दोनों सही हैं और (R), (A) की सही व्याख्या है। (A) और (R) दोनों सही हैं और (R), (A) की सही व्याख्या नहीं है। (A) सत्य है लेकिन (R) असत्य है। (A) असत्य है लेकिन (R) सत्य है। 	1
	Ans	<p>I. Both (A) and (R) are correct and (R) is correct explanation of (A)</p> <p>I. (A) और (R) दोनों सही हैं और (R), (A) की सही व्याख्या है।</p>	
		1 Mark for correct identification.	

1(x)		<p>Assertion (A): LAN (Local Area Network) is a type of network that covers a limited geographical area.</p> <p>Reason: LANs are designed to provide fast data transfer, resource sharing, and efficient communication in a very large area.</p> <ol style="list-style-type: none"> Both (A) and (R) are correct and (R) is correct explanation of (A). Both (A) and (R) are correct and (R) is not the correct explanation of (A). (A) is true but (R) is false. (A) is false but (R) is true <p>कथन (A): LAN (लोकल एरिया नेटवर्क) एक प्रकार का नेटवर्क है जो एक सीमित भौगोलिक क्षेत्र को कवर करता है।</p> <p>कारण (R): LAN को एक बहुत बड़े क्षेत्र में तेजी से डेटा हस्तांतरण, संसाधन साझाकरण और कुशल संचार प्रदान करने के लिए डिज़ाइन किया गया है।</p> <ol style="list-style-type: none"> (A) और (R) दोनों सही हैं और (R), (A) की सही व्याख्या है। (A) और (R) दोनों सही हैं और (R), (A) की सही व्याख्या नहीं है। (A) सत्य है लेकिन (R) असत्य है। (A) असत्य है लेकिन (R) सत्य है। 	1
	Ans	<p>I. (A) is true but (R) is false.</p> <p>I. (A) और (R) दोनों सही हैं और (R), (A) की सही व्याख्या है।</p>	
		1 Mark for correct identification.	
		<p style="text-align: center;">SECTION B</p> <p style="text-align: center;">(Each question carries 1 mark)</p>	
2.		<p>Write syntax of "while" loop in C++.</p> <p>C++ में "व्हाइल" लूप का सिंटैक्स लिखें।</p>	1
	Ans	<p>Syntax of "while" loop in C++:</p> <p>C++ में "व्हाइल" लूप का सिंटैक्स:</p> <pre>while (condition) { // code to be executed or loop statements // updating the loop variable or increment }</pre>	

	<p>For Example:</p> <p>उदाहरण के लिए:</p> <pre>int i=1; while (i < 5) { printf("I am an Indian") // loop statements i++; // updating the loop variable }</pre> <p>This while loop will print the output i.e. "I am an Indian" 4 times.</p> <p>यह लूप आउटपुट यानी " I am an Indian " को 4 बार प्रिंट करेगा।</p>	
	<p>1 Mark for any correct syntax of while loop.</p> <p>(Deduct ½ mark if there is any syntactical error)</p> <p>Do not deduct any marks if example is not given.</p> <p>Do not deduct any marks if any correct example of the syntax is given.</p>	
3.	<p>Define object in C++.</p> <p>C++ में ऑब्जेक्ट को परिभाषित करें।</p>	1
Ans	<p>An object is a variable of a class, encapsulating data and behaviours.</p> <p>एक ऑब्जेक्ट एक क्लास का एक वेरिएबल है, जो डेटा और व्यवहार को समाहित करता है।</p> <p style="text-align: center;">OR</p> <p>In C++, an object is a real-world entity represented by a class</p> <p>C++ में, एक ऑब्जेक्ट एक वास्तविक दुनिया की इकाई है जिसे एक क्लास द्वारा दर्शाया जाता है</p> <p style="text-align: center;">OR</p> <p>An object in C++ is an instance of a class with associated data and functions.</p> <p>C++ में एक ऑब्जेक्ट संबंधित डेटा और फ़ंक्शंस के साथ एक क्लास का एक उदाहरण है।</p>	

		1 Mark for any correct definition of object other than above also.																
4.		Define Database. डेटाबेस को परिभाषित करें।	1															
	Ans	Database is a collection of records (organized data or structured information) stored in a computer system. डेटाबेस रिकॉर्ड का (व्यवस्थित डाटा का या संरचित जानकारी का) संग्रह है जो एक कंप्यूटर में स्टोर किया जाता है।																
		1 Mark for any correct definition of Database.																
5.		Define array. ऐरे को परिभाषित करें।	1															
	Ans	An array is a collection of items/elements of the same data type stored at contiguous memory locations. समान डाटा टाइप के आइटम/एलिमेंट के संग्रह को ऐरे कहा जाता है जो निरंतर मेमोरी स्थानों पर स्टोर होते हैं।																
		1 Mark for any correct definition of Array.																
		SECTION C (Each question carries 2 marks)																
6.		Consider a table named StudentDetails with the following attributes: StuID , StuName , and the Game in which he is interested. Check whether this table satisfies the First Normal Form (1NF) requirements. If not, provide a modified version of the table to be in 1NF. एक टेबल जिसका नाम StudentDetails है और जिसमें निम्नलिखित गुण हैं: StuID , StuName , और Game जिसमें वह रुचि रखता है। यह जांचें कि क्या यह टेबल पहली नार्मल फॉर्म (1NF) की आवश्यकताएं पूरी करती है। अगर नहीं, तो टेबल का संशोधित संस्करण प्रदान करें जो 1NF का पालन करता है। <table><tr><th>StuID</th><th>StuName</th><th>Game</th></tr><tr><td>101</td><td>Rahul Sharma</td><td>Cricket</td></tr><tr><td>102</td><td>Priya Patel</td><td>Baseball, Cricket</td></tr><tr><td>103</td><td>Sanjay Verma</td><td>Football , Volleyball</td></tr><tr><td>104</td><td>Aishwarya Singh</td><td>Volleyball</td></tr></table>	StuID	StuName	Game	101	Rahul Sharma	Cricket	102	Priya Patel	Baseball, Cricket	103	Sanjay Verma	Football , Volleyball	104	Aishwarya Singh	Volleyball	2
StuID	StuName	Game																
101	Rahul Sharma	Cricket																
102	Priya Patel	Baseball, Cricket																
103	Sanjay Verma	Football , Volleyball																
104	Aishwarya Singh	Volleyball																
	Ans	The given table StudentDetails does not satisfy the First Normal Form (1NF) condition of “atomic (single) values in each column”.																

The "**Game**" column contains multiple values separated by commas.

To bring it into 1NF, we can create a new table. Here's the modified version:

StuID	StuName	Game
101	Rahul Sharma	Cricket
102	Priya Patel	Baseball
102	Priya Patel	Cricket
103	Sanjay Verma	Football
103	Sanjay Verma	Volleyball
104	Aishwarya Singh	Volleyball

The modified version of the "**StudentDetails**" table is now in the First Normal Form (1NF) by ensuring atomic values in each column. This normalization will give better data organization and avoids data redundancy.

दिए गए टेबल "**StudentDetails**" पहली नॉर्मल फॉर्म (1NF) की शर्त "प्रति कॉलम में एटॉमिक (एकल) वैल्यू" होनी को पूरा नहीं करता है। "Game" कॉलम में एक से अधिक वैल्यू का समूह है जो अल्पविराम से बांटे गए हैं।

1NF में ले जाने के लिए, हमने एक नई टेबल बना सकते हैं। संशोधित संस्करण है:

StuID	StuName	Game
101	Rahul Sharma	Cricket
102	Priya Patel	Baseball
102	Priya Patel	Cricket
103	Sanjay Verma	Football
103	Sanjay Verma	Volleyball
104	Aishwarya Singh	Volleyball

ऊपर दिया "StudentDetails" टेबल का संशोधित संस्करण अब पहली नॉर्मल फॉर्म (1NF) में है, जिससे सभी कॉलम में एक ही यानि एटॉमिक वैल्यू है। इस सुधार से डेटा को बेहतर से संगठित किया जा सकता है और डेटा की पुनरावृत्ति (रीडंडेंसी) से बचा जा सकता है।

½ marks for identifying that the table is not in 1NF.

½ marks for giving the reason that why the table is not in 1NF.

1 Mark for correct modified table which is now in 1NF.

		Do not deduct any marks if no explanation is provided after modifying the table. Avoid spelling mistake.	
		OR	
		What do DDL commands do in SQL? Explain CREATE TABLE command with an example. SQL में DDL कमांड्स क्या काम करती हैं? CREATE TABLE कमांड को उदाहरण सहित समझाएं।	
	Ans	<p>DDL, or Data Definition Language, commands in SQL are used for defining the structure of a database. They allow users to create, modify, and delete database tables.</p> <p>One example of CREATE TABLE, which is used to create a new table in the database. Here's an explanation with an example:</p> <pre>CREATE TABLE Students (StudentID INT, StuName VARCHAR(50), Age INT,);</pre> <ul style="list-style-type: none"> • CREATE TABLE is the DDL command used to create a new table. • Students is the name of the table being created. • The columns StudentID, StuName and Age are defined along with their data types. • INT and VARCHAR are data types. <p>This command creates a table named "Students" with columns for student information.</p>	

		<p>SQL में DDL, या डेटा डेफिनिशन लैंग्वेज, कमांड का उपयोग डेटाबेस की संरचना को डिजाइन करने के लिए किया जाता है। वे यूजर को डेटाबेस टेबल बनाने, संशोधित करने और हटाने की अनुमति देते हैं।</p> <p>DDL कमांड CREATE TABLE का उपयोग डेटाबेस में एक नई टेबल बनाने के लिए किया जाता है। यहाँ एक उदाहरण के साथ एक स्पष्टीकरण है:</p>	
		<pre>CREATE TABLE Students (StudentID INT, StuName VARCHAR(50), Age INT,);</pre>	
		<p>CREATE TABLE एक नई टेबल बनाने के लिए उपयोग किया जाने वाला DDL कमांड है।</p> <ul style="list-style-type: none"> • Students बनाई जा रही टेबल का नाम है। • कॉलम छात्र StudentID, StuName और Age को उनके डेटा प्रकारों के साथ परिभाषित किया गया है। • INT और VARCHAR डेटा प्रकार हैं। <p>यह कमांड "Students" नाम का एक टेबल उसके कॉलम के साथ बनाती है।</p>	
		<p>½ marks for explaining use of DDL Commands.</p> <p>1 mark for using correct CREATE TABLE Syntax in any example.</p> <p>½ marks for explaining the example.</p> <p>Deduct ½ marks for any syntax error.</p> <p>Do not deduct marks for any spelling mistake.</p>	
7.		<p>What is Default constructor in C++?</p> <p>C++ में डिफॉल्ट कंस्ट्रक्टर क्या है?</p>	2
		<p>In C++, A constructor without any arguments or with the default value for every argument is said to be the Default constructor. They are used to create objects, which do not have any specific initial value.</p>	

C++ में, एक कंस्ट्रक्टर जिसमें कोई भी आर्गुमेंट नहीं होता या प्रत्येक आर्गुमेंट के लिए डिफॉल्ट वैल्यू होती है, उसे 'डिफॉल्ट कंस्ट्रक्टर' कहा जाता है। इसका उपयोग उन ऑब्जेक्ट्स को बनाने के लिए किया जाता है, जिनमें कोई विशिष्ट प्रारंभिक वैल्यू नहीं होती।

The **syntax** of a default constructor:

डिफॉल्ट कंस्ट्रक्टर का सिंटैक्स:

```
class ClassName
{
public:
    ClassName( )
    {
        // Constructor body
    }
};
```

OR / या

Programming example of default constructor:

डिफॉल्ट कंस्ट्रक्टर का प्रोग्रामिंग उदाहरण:

```
#include <iostream>
using namespace std;
// declare a class
class Wall {
private:
    double length;
public:
    // default constructor to initialize variable
    Wall() {
        length = 5.5;
        cout << "Creating a wall." << endl;
        cout << "Length = " << length << endl;
    }
};

int main() {
    Wall wall1;
    return 0;
}
```

The output of the program

```
Creating a Wall
Length = 5.5
```

		<p>1 Mark for any correct definition/ explanation of default constructor.</p> <p>1 Mark to write correct syntax.</p> <p>Deduct ½ marks if there is any syntax error.</p> <p>Do not deduct any marks for correct explanation even without programming.</p> <p>Give full marks attempting the programming example with correct syntax and output even without theoretical explanation.</p>	
8.		<p>Explain COUNT () function in SQL, providing its syntax and an example.</p> <p>SQL में COUNT () फंक्शन को समझाएं, उसका सिंटैक्स और एक उदाहरण प्रदान करें।</p>	2
	Ans	<p>The COUNT () function in SQL is used to count the number of rows in a table which matches a specific criteria. It is used to count a field/column/attribute in different expressions. It takes the name of a column as its argument.</p> <p>SQL में COUNT () फंक्शन का उपयोग टेबल में पंक्तियों की संख्या को गिनने के लिए किया जाता है। इसका उपयोग विभिन्न एक्सप्रेशन में किसी फ़ील्ड/कॉलम/एट्रिब्यूट की गणना के लिए किया जाता है। यह अपने आर्गुमेंट के रूप में एक कॉलम का नाम लेता है।</p> <p>Syntax of COUNT () function:</p> <p>COUNT () फंक्शन का सिंटैक्स</p> <pre>SELECT COUNT(column_name) FROM table_name;</pre> <p>This query will return the number of non-null values in the specified column of the specified table.</p> <p>For Example:</p> <p>If you want to count the number of orders made by a customer in a shop, you can use the following query:</p> <p>यह क्वेरी निर्दिष्ट तालिका के निर्दिष्ट कॉलम में गैरनल मूल्यों की संख्या - लौटाएगी।</p> <p>उदाहरण के लिए: यदि आपको दुकान में किसी ग्राहक द्वारा किए गए ऑर्डरों की संख्या गणना करनी है, तो आप निम्नलिखित क्वेरी का उपयोग कर सकते हैं:</p> <pre>SELECT COUNT(order_id)</pre>	

		FROM orders WHERE customer_id = 123;	
		1 mark for any correct definition of COUNT () function. ½ mark for correct syntax. ½ mark for any correct example of COUNT () function. Deduct ½ marks for any syntax error. Even if there is a spelling mistake other than syntax, do not deduct any marks if the concept is understood.	
9.		Explain the concept of container tags in HTML with an example. एक उदाहरण के साथ HTML में container टैग को समझाएं।	2
	Ans	<p>In HTML, container tags, also known as paired or nested tags, are used to define the beginning and end of an HTML element. They come in pairs - an opening tag and a closing tag. The opening tag marks where the content begins, and the closing tag marks where it ends. Together, they wrap around the content.</p> <p>Some of the container tags in HTML are:</p> <ul style="list-style-type: none"> • <h1> and </h1> create a container for a level 1 heading. • <p> and </p> define a container for a paragraph. • <a> and create a container for a hyperlink. <p>For example:</p> <pre><p>This is a paragraph.</p> <p>This is another paragraph.</p></pre> <p>In above example, when the browser renders this HTML, it will display the first paragraph, then move to the next second paragraph.</p> <p>HTML में, कंटेनर टैग्स, जिन्हें पेयर्ड या नेस्टेड टैग्स भी कहा जाता है, किसी HTML एलिमेंट की शुरुआत और समाप्ति को परिभाषित करने के लिए इस्तेमाल होते हैं। ये पेअर होते हैं - एक ओपनिंग टैग और एक क्लोजिंग टैग। ओपनिंग टैग उस स्थान को चिह्नित करता है जहां कंटेंट शुरू होता है, और क्लोजिंग टैग उस स्थान को चिह्नित करता है जहां कंटेंट समाप्त होता है। इन्हें मिलाकर, वे कंटेंट को घेरते हैं।</p> <p>कुछ HTML कंटेनर टैग्स हैं:</p> <ul style="list-style-type: none"> • <h1> और </h1> एक स्तर 1 हेडिंग के लिए कंटेनर बनाते हैं। • <p> और </p> एक पैराग्राफ के लिए कंटेनर को परिभाषित करते हैं। 	

		<p>• <code><a></code> और <code></code> एक हाइपरलिंक के लिए कंटेनर बनाते हैं।</p> <p>उदाहरण के लिए:</p> <p><code><p></code>यह एक पैराग्राफ है।<code></p></code></p> <p><code><p></code>यह एक और पैराग्राफ है।<code></p></code></p> <p>इस उदाहरण में, जब ब्राउज़र इस HTML को रेंडर करता है, तो यह पहला पैराग्राफ दिखाएगा, फिर दूसरे पैराग्राफ में जाएगा।</p>	
		<p>1 mark for any correct definition of container tags.</p> <p>1 mark for explaining the concept of Container tags with an appropriate example of any one container tag.</p> <p>½ marks if only tag name are given without explanation.</p> <p>½ marks for explanation of any one tag.</p>	
10.		<p>In a green village in India, Nisha and Aryan used a computer to send messages to their friends. With the help of something called SMTP, their messages went really fast through the internet, like a cool wind. They found out about HTTPS, a special guard for their online talks, keeping them safe, just like the trees kept their village cozy. In their small, nature-loving village, using the computer felt like a gentle way to connect with friends and take care of the Earth at the same time.</p> <p>भारत के एक हरे-भरे गांव में, निशा और आर्यन अपने दोस्तों को संदेश भेजने के लिए एक कंप्यूटर का इस्तेमाल करते थे। SMTP नामक किसी चीज़ की मदद से, उनके संदेश इंटरनेट के माध्यम से वास्तव में तेजी से चले गए, एक ठंडी हवा की तरह। उन्हें HTTPS के बारे में पता चला, जो उनकी ऑनलाइन बातचीत के लिए एक विशेष गार्ड है, जो उन्हें सुरक्षित रखता है, ठीक उसी तरह जैसे पेड़ उनके गांव को आरामदायक रखते थे। अपने छोटे, प्रकृति-प्रेमी गांव में, कंप्यूटर का उपयोग करना दोस्तों के साथ जुड़ने और एक ही समय में पृथ्वी की देखभाल करने का एक सौम्य तरीका था।</p> <p>I. What role does SMTP play in the email delivery process?</p> <ol style="list-style-type: none"> Receiving emails Storing emails Sending emails Encrypting emails <p>ईमेल वितरण प्रक्रिया में SMTP क्या भूमिका निभाता है?</p> <ol style="list-style-type: none"> ईमेल प्राप्त करना ईमेल संग्रहीत करना ईमेल भेजना ईमेल को एन्क्रिप्ट करना 	2

		<p>II. What is the purpose of HTTPS?</p> <p>a. Secure email transmission</p> <p>b. Secure web communication</p> <p>c. Secure file transfer</p> <p>d. Secure database access</p> <p>HTTPS का उद्देश्य क्या है?</p> <p>a. सुरक्षित ईमेल ट्रांसमिशन</p> <p>b. सुरक्षित वेब संचार</p> <p>c. सुरक्षित फ़ाइल स्थानांतरण</p> <p>d. सुरक्षित डेटाबेस पहुँच</p>	
	Ans	<p>I. c. Sending emails c. ईमेल भेजना</p> <p>II. b. Secure web communication b. सुरक्षित वेब संचार)</p>	
		<p>1 mark for any one correct identification.</p> <p>2 marks for both correct identifications.</p> <p>Do not deduct any marks for any spelling mistakes if option name is correctly written.</p>	
		<p>SECTION D</p> <p>(Each question carries 4 Mark)</p>	
11.		<p>What is Data Item in data structure? Write down the algorithm for inserting and deleting data in stack.</p> <p>डेटा संरचना में डेटा आइटम क्या है? स्टैक में डेटा डालने और हटाने के लिए एल्गोरिदम लिखें।</p>	4
	Ans	<p>Data item refers to the basic unit of information that can be stored, processed, or manipulated by a computer. For example Name is a data item of text or string data type and age is a data item of integer data type etc.</p> <p>डेटा आइटम उस बेसिक जानकारी को संदर्भित करता है जो किसी कंप्यूटर द्वारा स्टोर या प्रोसेस किया जा सकती है। उदाहरण के लिए, नाम एक डेटा आइटम है जो text या स्ट्रिंग डेटा प्रकार का है और आयु एक डेटा आइटम है जो integer डेटा प्रकार का है आदि।</p> <p>Whereas</p> <p>Stack is a linear data structure that follows the LIFO (Last-In-First-Out) principle. For Eg. Stack of books.</p> <p>Inserting a new element in the stack is termed a push operation.</p>	

	<p>Removing or deleting elements from the stack is termed pop operation.</p> <p>स्टैक एक लीनियर प्रकार की डेटा स्ट्रक्चर है जो LIFO (लास्ट-इन-फर्स्ट-आउट) सिद्धांत का पालन करती है उदाहरण किताबों का स्टैक</p> <p>स्टैक में एक नया एलिमेंट डालने को PUSH ऑपरेशन कहा जाता है।</p> <p>स्टैक से एलिमेंट को हटाने या हटाने को POP ऑपरेशन कहा जाता है।</p> <p>Push algorithm:</p> <p>Push operation includes various steps, which are as follows :</p> <p>Step 1: First, check whether the stack is full or not</p> <p>Step 2: If the stack is full, then exit</p> <p>Step 3: If not, increment top by one</p> <p>Step 4: Insert a new element where the top is pointing</p> <p>Step 5: Exit</p> <p>PUSH ऑपरेशन एल्गोरिदम:</p> <p>PUSH ऑपरेशन में विभिन्न चरण शामिल हैं, जो इस प्रकार हैं:</p> <p>चरण 1: सबसे पहले, जांचें कि स्टैक भरा हुआ है या नहीं</p> <p>स्टेप 2: अगर स्टैक फुल हो गया है, तो exit करें</p> <p>चरण 3: यदि नहीं, तो टॉप में एक की वृद्धि करें</p> <p>चरण 4: एक नया एलिमेंट डालें जहां TOP पॉइंट कर रहा है</p> <p>चरण 5: बाहर निकलें</p> <p style="text-align: center;">OR / या</p> <hr/> <pre> begin procedure push: stack, data if stack is full return null endif top ← top + 1 stack[top] ← data end procedure </pre> <p>POP algorithm:</p> <p>Pop operation includes various steps, which are as follows:</p> <p>Step 1 – Checks stack has some element or stack is empty.</p>	
--	--	--

	<p>Step 2 – If the stack has no element means it is empty then display “underflow”</p> <p>Step 3 – If the stack has element some element, accesses the data element at which top is pointing.</p> <p>Step 4 – Decreases the value of top by 1.</p> <p>POP ऑपरेशन एल्गोरिदम:</p> <p>POP ऑपरेशन में विभिन्न चरण शामिल हैं, जो इस प्रकार हैं:</p> <p>चरण 1 - चेक स्टैक में कुछ तत्व है या स्टैक खाली है।</p> <p>चरण 2 - यदि स्टैक में कोई तत्व नहीं है जिसका मतलब है कि यह खाली है तो "अंडरफ्लो" प्रदर्शित करें</p> <p>चरण 3 - यदि स्टैक में तत्व कुछ तत्व है, तो उस डेटा तत्व तक पहुंचता है जिस पर शीर्ष इंगित कर रहा है।</p> <p>चरण 4 – शीर्ष के मान को 1 से कम करता है।</p> <p>चरण 5 - पीओपी ऑपरेशन सफलतापूर्वक किया गया।</p> <p style="text-align: center;">OR / या</p> <pre> If TOP=-1 return "Underflow" endif item=Stack[Top] top=top-1 return Item end </pre>	
	<p>1 mark for any correct definition of data item in Data Structure with example.</p> <p>1 and 1/2 Marks for each correct algorithm for push and pop.</p> <p>Deduct ½ marks if there is any syntax error while writing algorithm.</p>	
	OR	
	<p>What is Bubble Sort? Explain with example.</p> <p>बबल सॉर्ट क्या है? उदाहरण से समझाइए।</p>	
	<p>Bubble Sort:</p> <p>Bubble Sort is a sorting algorithm that repeatedly steps through the list, compares adjacent elements, and swaps them if they are in the</p>	

wrong order. The pass through the list is repeated until the list is sorted. For Example:

Let's say we have an unsorted list: **5, 2, 9, 1, 5**.

Pass 1:

Compare 5 and 2 (swap): 2, 5, 9, 1, 5.

Compare 5 and 9 (no swap): 2, 5, 9, 1, 5.

Compare 9 and 1 (swap): 2, 5, 1, 9, 5.

Compare 9 and 5 (swap): 2, 5, 1, 5, 9.

Pass 2:

Compare 2 and 5 (no swap): 2, 5, 1, 5, 9.

Compare 5 and 1 (swap): 2, 1, 5, 5, 9.

Compare 5 and 5 (no swap): 2, 1, 5, 5, 9.

Pass 3:

Compare 2 and 1 (swap): 1, 2, 5, 5, 9.

Compare 2 and 5 (no swap): 1, 2, 5, 5, 9.

No more swaps are needed, and the list is sorted.

Sorted List: 1, 2, 5, 5, 9.

Bubble Sort is not the most efficient sorting algorithm for large datasets, but it's easy to understand and implement.

बबल सॉर्ट: बबल सॉर्ट एक साधारित सॉर्टिंग एल्गोरिदम है जो सूची में बार-बार चलता है, तत्वों की तुलना करता है, और यदि वे गलत क्रम में हैं, तो उन्हें आपस में बदलता है। सूची के सभी तत्वों पर यह प्रक्रिया बार-बार दोहराई जाती है, जब तक सूची क्रमबद्ध नहीं हो जाती।

उदाहरण:

चलिए मान लें हमें एक क्रमबद्ध नहीं सूची है: **5, 2, 9, 1, 5**

1. पास 1:

- 5 और 2 की तुलना करें (बदलें): 2, 5, 9, 1, 5
- 5 और 9 की तुलना करें (बदलाव नहीं हुआ): 2, 5, 9, 1, 5
- 9 और 1 की तुलना करें (बदलें): 2, 5, 1, 9, 5
- 9 और 5 की तुलना करें (बदलें): 2, 5, 1, 5, 9

2. पास 2:

- 2 और 5 की तुलना करें (बदलाव नहीं हुआ): 2, 5, 1, 5, 9
- 5 और 1 की तुलना करें (बदलें): 2, 1, 5, 5, 9
- 5 और 5 की तुलना करें (बदलाव नहीं हुआ): 2, 1, 5, 5, 9

3. पास 3:

		<ul style="list-style-type: none"> 2 और 1 की तुलना करें (बदलें): 1, 2, 5, 5, 9 2 और 5 की तुलना करें (बदलाव नहीं हुआ): 1, 2, 5, 5, 9 <p>और कोई बदलाव की आवश्यकता नहीं है, और सूची क्रमबद्ध है।</p> <p>सॉर्टेड सूची: 1, 2, 5, 5, 9</p> <p>बबल सॉर्ट बड़े डेटासेट्स के लिए सबसे दक्ष नहीं है, लेकिन यह समझने और लागू करने में सरल है।</p>	
		<p>1 mark for any correct definition of bubble sort in data structure.</p> <p>3 marks for correctly explaining the concept of bubble sort with any example.</p> <p>Give marks in the example stepwise accordingly if needed.</p>	
12.		<p>Explain any 4 phases of SDLC.</p> <p>SDLC के किन्हीं 4 चरणों की व्याख्या कीजिए।</p>	4
		<p>Different phases of SDLC are as follows:</p> <ol style="list-style-type: none"> 1) Requirements gathering and analysis or Survey 2) Feasibility study or Investigation and Fact Recording 3) Software design 4) Development or Coding 5) Testing 6) Implementation 7) Maintenance <p>Requirements gathering and analysis <u>or</u> Survey: First phase of SDLC is to collect the requirements from customer. Survey is the part of Requirement gathering phase of SDLC. Survey is the best technique to collect the requirements. We can do survey with following techniques:</p> <ol style="list-style-type: none"> 1. Interviews 2. Questionnaires 3. Observation 4. Sampling <p>Feasibility study <u>or</u> Investigation and fact recording: It is a part of Requirement analysis phase of SDLC. All the requirements collected through surveys are investigated in this part. Feasibility study means to check if it is possible to make the software or not. After investigation of the requirements, final facts are recorded for analysis.</p>	

Software Design: The next stage of SDLC is the Design phase. During the design phase, developers and technical architects start the high-level design of the software and system to be able to deliver each requirement.

Development or Coding

This phase of SDLC comes when we have gathered the requirements and when our design is ready. The software developer starts coding according to the requirements and the design.

Testing

Testing is done before the software is delivered to customers so that the software exactly matches their requirements. Testers usually find errors or bugs in the software in this phase. There are two types of testing methods:

1. Black Box testing: The technique of testing without having any knowledge of the interior workings of the software is called black-box testing. The tester cannot see the code but only can use the software to find any bugs or errors.

2. White Box testing: White-box testing is the detailed investigation of internal workings of the code. White-box testing is also called glass testing or open-box testing. In order to perform white-box testing on an application, a tester needs to know the internal workings of the code.

Implementation

In this phase, the system is installed to support the required function for which it is developed. It includes user notification, user training, installation of hardware, installation of software onto production computers, and integration of the system into daily work processes. This phase continues until the system is operating in production in accordance with the defined user requirements.

Maintenance

The maintenance phase of the SDLC occurs after we have delivered the software to the customers. Maintenance of software can include software upgrades or any error or bug fixing which the customer finds.

सॉफ्टवेयर इंजीनियरिंग इंजीनियरिंग की वह शाखा है जो सॉफ्टवेयर विकसित करने के लिए इंजीनियरिंग सिद्धांतों और तकनीकों का उपयोग करती है।

SDLC के विभिन्न चरण इस प्रकार हैं:

- 1) आवश्यकताएँ एकत्र करना और विश्लेषण या सर्वे
- 2) फिजिबिलिटी अध्ययन या जांच और फैक्ट रिकॉर्डिंग
- 3) सॉफ्टवेयर डिजाइन
- 4) डेवलपमेंट या कोडिंग
- 5) टेस्टिंग
- 6) इम्प्लीमेंटेशन
- 7) मेंटेनेंस

आवश्यकताएँ एकत्र करना और विश्लेषण या सर्वे

SDLC का पहला चरण ग्राहक से आवश्यकताओं को एकत्र करना है। सर्वे SDLC की आवश्यकता संग्रह चरण का हिस्सा है। आवश्यकताओं को एकत्र करने के लिए सर्वे सबसे अच्छी तकनीक है। हम निम्नलिखित तकनीकों से सर्वे कर सकते हैं:

1. साक्षात्कार
2. प्रश्नावली
3. ऑब्जरवेशन
4. सैम्पलिंग

फिजिबिलिटी अध्ययन या जांच और फैक्ट रिकॉर्डिंग

यह SDLC की आवश्यकता विश्लेषण चरण का एक हिस्सा है। इस भाग में सर्वे के माध्यम से एकत्र की गई सभी आवश्यकताओं की जांच की जाती है। फिजिबिलिटी अध्ययन का अर्थ यह जांचना है कि सॉफ्टवेयर बनाना संभव है या नहीं।

आवश्यकताओं की जांच के बाद, विश्लेषण के लिए अंतिम तथ्य दर्ज किए जाते हैं।

सॉफ्टवेयर डिजाइन

एसडीएलसी का अगला चरण डिजाइन चरण है। डिजाइन चरण के दौरान, डेवलपर्स और तकनीकी आर्किटेक्ट प्रत्येक आवश्यकता को पूरा करने में सक्षम होने के लिए सॉफ्टवेयर और सिस्टम का उच्च-स्तरीय डिज़ाइन शुरू करते हैं।

डेवलपमेंट या कोडिंग

	<p>SDLC का यह चरण तब आता है जब हम आवश्यकताओं को इकट्ठा कर लेते हैं और जब हमारा डिज़ाइन तैयार हो जाता है। सॉफ्टवेयर डेवलपर आवश्यकताओं और डिज़ाइन के अनुसार कोडिंग शुरू करता है।</p> <p><u>टेस्टिंग</u></p> <p>सॉफ्टवेयर को ग्राहकों तक पहुँचाने से पहले टेस्टिंग किया जाता है ताकि सॉफ्टवेयर उनकी आवश्यकताओं से बिल्कुल मेल खाता हो। परीक्षक आमतौर पर इस चरण में सॉफ्टवेयर में त्रुटियाँ या बग ढूँढते हैं। टेस्टिंग विधियाँ दो प्रकार की होती हैं:</p> <p>1. <u>ब्लैक बॉक्स टेस्टिंग</u>: सॉफ्टवेयर की आंतरिक कार्यप्रणाली के ज्ञान के बिना टेस्टिंग की तकनीक को ब्लैक-बॉक्स टेस्टिंग कहा जाता है। परीक्षक कोड नहीं देख सकता है लेकिन केवल बग या त्रुटियों को खोजने के लिए सॉफ्टवेयर का उपयोग कर सकता है।</p> <p>2. <u>व्हाइट बॉक्स टेस्टिंग</u>: व्हाइट-बॉक्स टेस्टिंग कोड की आंतरिक कार्यप्रणाली की विस्तृत जांच है। व्हाइट-बॉक्स टेस्टिंग को ग्लास टेस्टिंग या ओपन-बॉक्स टेस्टिंग भी कहा जाता है। किसी एप्लिकेशन पर व्हाइट-बॉक्स टेस्टिंग करने के लिए, एक परीक्षक को कोड की आंतरिक कार्यप्रणाली को जानने की आवश्यकता होती है।</p> <p><u>इम्प्लीमेंटेशन</u></p> <p>इस चरण में, सिस्टम को आवश्यक फ़ंक्शन का समर्थन करने के लिए स्थापित किया गया है जिसके लिए इसे विकसित किया गया है। इसमें उपयोगकर्ता अधिसूचना, उपयोगकर्ता प्रशिक्षण, हार्डवेयर की स्थापना, उत्पादन कंप्यूटर पर सॉफ्टवेयर की स्थापना और दैनिक कार्य प्रक्रियाओं में सिस्टम का एकीकरण शामिल है। यह चरण तब तक जारी रहता है जब तक सिस्टम परिभाषित उपयोगकर्ता आवश्यकताओं के अनुसार उत्पादन में काम नहीं कर रहा है।</p> <p><u>मैटेनेंस</u></p> <p>एसडीएलसी का रखरखाव चरण हमारे द्वारा ग्राहकों को सॉफ्टवेयर वितरित करने के बाद होता है। सॉफ्टवेयर के रखरखाव में सॉफ्टवेयर अपग्रेड या कोई त्रुटि या बग फिक्सिंग शामिल हो सकती है जो ग्राहक को मिलती है।</p>	
	<p>1 mark each for any correct explanations of any 4 phases of SDLC</p> <p>Ignore any spelling mistakes in Hindi as well as in English if the meaning of the word is understood.</p> <p>Do not deduct any marks if the English version of the word is written in Hindi. i.e. consider Software or सॉफ्टवेयर both as correct.</p>	
	<p>OR</p>	
	<p>What is software testing? Explain any two types of testing methods used in software development.</p>	

	<p>सॉफ्टवेयर परीक्षण क्या है? सॉफ्टवेयर विकास में प्रयुक्त किन्हीं दो प्रकार की परीक्षण विधियों को स्पष्ट कीजिए।</p>	
	<p>Software testing is the process of evaluating and verifying that a software does what it is supposed to do.</p> <p>There are two methods of testing</p> <ol style="list-style-type: none"> 1) White Box testing 2) Black Box testing <p>White Box Testing: White box testing is a testing technique that focuses on the internal logic, structure, and code of the software. Testers who perform white box testing have access to the source code, and they design test cases based on the knowledge of how the software is implemented. The goal is to ensure that each individual component and the entire system work as expected. White box testing involves checking the internal code and logic of the software, and an example could be testing a sorting algorithm to ensure it correctly arranges elements.</p> <p>Example of White Box Testing: Imagine you are testing a function in a calculator application that adds two numbers. In white box testing, you would look at the actual code of the "add" function. You might design test cases to check if it correctly handles positive and negative numbers, if it works with decimals, or if it handles edge cases like adding zero.</p> <p>Black Box Testing: Black box testing, on the other hand, is a testing method where the internal logic, structure, and implementation details of the software are not known to the tester. Test cases are designed based on the specifications and requirements of the software. The goal is to ensure that the software functions as intended, regardless of its internal workings. Black box testing, on the other hand, focuses on the external functionality without knowledge of internal code, and an example might be testing a search feature on a website to ensure it returns relevant results without knowing the underlying search algorithm.</p> <p>Example of Black Box Testing: Consider testing a login functionality in a website. In black box testing, you don't need to know how the</p>	

login code is implemented. You would design test cases to check if the login works with valid and invalid credentials, if it handles password recovery, and if it redirects users to the correct pages upon login.

Both testing methods are crucial for comprehensive software quality assurance.

सफेद बॉक्स परीक्षण:

सफेद बॉक्स परीक्षण एक परीक्षण तकनीक है जो सॉफ्टवेयर के आंतरिक तर्क, संरचना, और कोड पर केंद्रित है। वह टेस्ट्स जो सफेद बॉक्स परीक्षण करते हैं, उन्हें स्रोत कोड तक पहुंच होती है, और उन्होंने सॉफ्टवेयर के कैसे अमल हो रहा है की जानकारी के आधार पर टेस्ट केस डिज़ाइन किए जाते हैं। इसका उद्देश्य है सुनिश्चित करना कि प्रत्येक व्यक्ति घटक और पूरे सिस्टम को उम्मीद के अनुसार काम करते हैं। सफेद बॉक्स परीक्षण में सॉफ्टवेयर के आंतरिक कोड और तर्क की जांच होती है, जैसे कि तत्वों को सही ढंग से क्रमबद्ध करने के लिए सॉर्टिंग एल्गोरिदम का परीक्षण करना।

सफेद बॉक्स परीक्षण का उदाहरण:

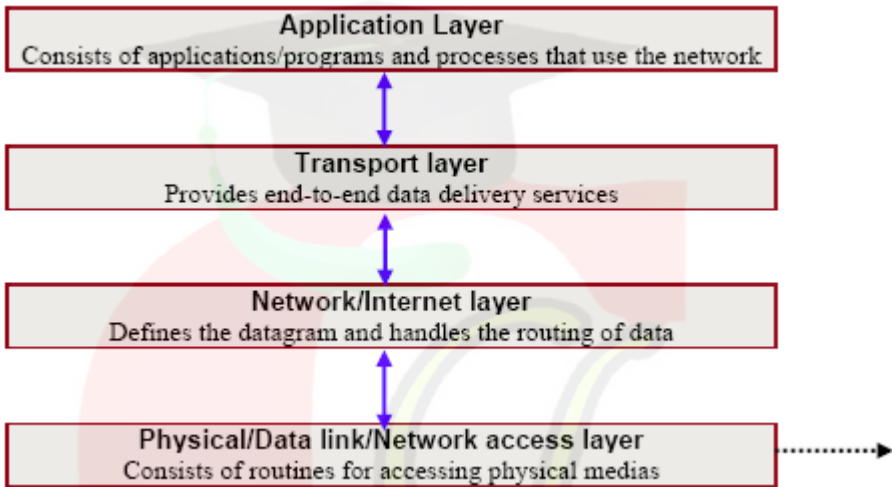
सोचें कि आप एक कैलकुलेटर एप्लिकेशन में दो नंबर जोड़ने वाले एक फ़ंक्शन को टेस्ट कर रहे हैं। सफेद बॉक्स परीक्षण में, आप "जोड़" फ़ंक्शन के वास्तविक कोड पर नजर डालेंगे। आप यह तय कर सकते हैं कि यह सही तरह से सकारात्मक और नकारात्मक संख्याओं का सही से उपयोग करता है, या यह जीरो को जोड़ने के साथ कैसे सामर्थ्यपूर्ण है।

ब्लैक बॉक्स परीक्षण:

ब्लैक बॉक्स परीक्षण, दूसरी ओर, एक परीक्षण विधि है जिसमें सॉफ्टवेयर के आंतरिक तर्क, संरचना, और अंकुश का ज्ञान टेस्टर के पास नहीं होता है। टेस्ट केस विशेषज्ञता और आवश्यकताओं के आधार पर डिज़ाइन किए जाते हैं। इसका उद्देश्य है सुनिश्चित करना कि सॉफ्टवेयर इच्छित रूप से काम करता है, चाहे इसके आंतरिक काम कुछ भी हों। ब्लैक बॉक्स परीक्षण, दूसरी ओर, आंतरिक कोड के बिना बाह्य क्रियावली पर केंद्रित है, जैसे कि एक वेबसाइट पर खोज सुनिश्चित करना कि यह अंशकों का नीति से मेल खाता है बिना उदाहरण के सुनिश्चित करता है कि इसके परिचालन में कोई बाधा नहीं है।

ब्लैक बॉक्स परीक्षण का उदाहरण:

एक वेबसाइट में लॉगिन कार्यक्षमता का परीक्षण करना कोई उदाहरण है। ब्लैक बॉक्स परीक्षण में, आपको यह नहीं पता होता कि लॉगिन कोड कैसे लागू किया गया है। आप यह सुनिश्चित करने के लिए टेस्ट केस डिज़ाइन करेंगे कि सही और गलत प्रमाण पत्रों के साथ लॉगिन करता है, क्या यह पासवर्ड पुनर्प्राप्ति को

	संभालता है, और क्या यह लॉगिन करने पर उपयोगकर्ताओं को सही पृष्ठों पर पुनर्निर्देशित करता है।	
	ये दोनों परीक्षण विधियाँ समर्थन में महत्वपूर्ण हैं।	
	<p>1 mark for any correct definition of software testing.</p> <p>1½ marks each for any correct explanation of two methods of testing.</p> <p>Ignore any spelling mistakes in Hindi as well as in English if the meaning of the word is understood.</p> <p>Do not deduct any marks if the English version of the word is written in Hindi. i.e. consider Testing or टेस्टिंग both as correct.</p>	
13.	<p>Explain briefly all the layers of the TCP/IP model.</p> <p>TCP/IP मॉडल की सभी परतों को संक्षेप में समझाइए।</p>	4
	 <p>The TCP/IP model serves as a roadmap for effective communication among devices on a network, ensuring a seamless exchange of information. Comprising four layers, each layer performs distinct functions to facilitate efficient communication:</p> <ol style="list-style-type: none"> 1. Application Layer: <ul style="list-style-type: none"> • Importance: The application layer acts as the interface between our everyday applications and the network, enabling smooth interactions. Browsing the web, sending emails, or sharing files, this layer ensures that user-facing applications can effectively communicate with the network. • Key Functions: It manages communication protocols (such as HTTP, FTP), and allows applications to access network services. 2. Transport Layer: 	

- **Importance:** The transport layer ensures the reliable delivery of data from one point to another. It maintains the integrity and accuracy of the transmitted information.
- **Key Functions:** This layer check end-to-end communication, performs error checking, and manages flow control. Protocols like TCP and UDP is used in this layer.

3. Internet Layer:

- **Importance:** The internet layer serves in routing data between different networks. It include device communication with each other to form network.
- **Key Functions:** It assigns IP addresses and routing of data packets. The Internet Protocol (IP) operates at this layer.

•

4. Link Layer:

- **Importance:** The physical connection within a local network is the responsibility of the link layer.
- **Key Functions:** The layer manages the physical connection, addresses devices in local network, and also do data framing. It include Ethernet and Wi-Fi.

TCP/IP मॉडल एक नेटवर्क पर उपकरणों के बीच प्रभावी संचार के लिए एक रोडमैप के रूप में कार्य करता है, जो सूचना का आदान-प्रदान सुनिश्चित करता है। चार परतों को मिलाकर, प्रत्येक परत कुशल संचार की सुविधा के लिए अलग-अलग कार्य करती है:

1. आवेदन परत:

- **महत्व:** एप्लिकेशन परत हमारे रोजमर्रा के अनुप्रयोगों और नेटवर्क के बीच इंटरफ़ेस के रूप में कार्य करती है, जिससे सुचारू बातचीत हो सकती है। वेब ब्राउज़ करना, ईमेल भेजना, या फ़ाइलों को साझा करना, यह परत सुनिश्चित करती है कि उपयोगकर्ता-सामना करने वाले एप्लिकेशन नेटवर्क के साथ प्रभावी ढंग से संवाद कर सकें।
- **मुख्य कार्य:** यह संचार प्रोटोकॉल (जैसे HTTP, FTP) का प्रबंधन करता है, और अनुप्रयोगों को नेटवर्क सेवाओं तक पहुंचने की अनुमति देता है।

2. परिवहन परत:

- **महत्व:** परिवहन परत एक बिंदु से दूसरे तक डेटा की विश्वसनीय डिलीवरी सुनिश्चित करती है। यह प्रेषित जानकारी की अखंडता और सटीकता को बनाए रखता है।

		<ul style="list-style-type: none"> • मुख्य कार्य: यह परत एंड-टू-एंड संचार की जांच करती है, त्रुटि जांच करती है, और प्रवाह नियंत्रण का प्रबंधन करती है। इस परत में टीसीपी और यूडीपी जैसे प्रोटोकॉल का उपयोग किया जाता है। <p>3. इंटरनेट परत:</p> <ul style="list-style-type: none"> • महत्व: इंटरनेट परत विभिन्न नेटवर्क के बीच डेटा को रूट करने में कार्य करती है। इसमें नेटवर्क बनाने के लिए एक दूसरे के साथ डिवाइस संचार शामिल है। • मुख्य कार्य: यह आईपी पते और डेटा पैकेट के रूटिंग प्रदान करता है। इंटरनेट प्रोटोकॉल (IP) इस परत पर काम करता है। <p>4. लिंक परत:</p> <ul style="list-style-type: none"> • महत्व: एक स्थानीय नेटवर्क के भीतर भौतिक कनेक्शन लिंक परत की जिम्मेदारी है। • मुख्य कार्य: परत भौतिक कनेक्शन का प्रबंधन करती है, स्थानीय नेटवर्क में उपकरणों को संबोधित करती है, और डेटा फ्रेमिंग भी करती है। इसमें ईथरनेट और वाई-फाई शामिल हैं। 	
		<p>½ marks for full form of TCP/IP.</p> <p>1½ marks for naming all the layers of TCP/IP.</p> <p>1/2 marks each for correct explanation of all layers of TCP/IP.</p>	
		OR	
		<p>Explain the following: निम्नलिखित की व्याख्या कीजिए:</p> <p>i. HTTPS</p> <p>ii. Web Address</p> <p>iii. Search Engine</p> <p>iv. WWW</p>	
	Ans	<p>i. HTTPS (Hypertext Transfer Protocol Secure):</p> <ul style="list-style-type: none"> • Internet Bodyguard: HTTPS acts as a bodyguard for your internet activities. • Encryption Expert: It encrypts (puts in a secret code) the information you share online, such as passwords and personal details. • Safe Online Transactions: Essential for secure online transactions, like shopping or banking, HTTPS ensures your sensitive data remains private and protected. 	

Trusted Connection: When you see "https://" in a website's address, it's a sign that your connection to that site is safe and secure. Without HTTPS, personal information could be at risk during online interactions.

ii. Web Address:

Web Address or also named as URL (**Uniform Resource Locator**) is a unique identifier used to locate a resource on the Internet. Every webpage on internet have unique address to help people locate it. It is also referred to as a URL. It consist of multiple parts i.e. including a protocol and domain name.

iii. Search engine:

A search engine is a tool on the internet that helps you find information by looking through a huge collection of websites. It is like a super-smart librarian for the internet. When you type something into a search engine, like Google, it looks through billions of web pages and quickly finds and shows you all the websites, articles, and pictures related to what you want to know.

iv. WWW

WWW is a collection of documents and other web resources which are identified by URLs and can be accessed and searched by web browsers via the Internet. WWW is also called the Web and it was invented by Tim Berners-Lee in 1989. All the web pages can be viewed only with the help of web browser.

i. HTTPS (हाइपरटेक्स्ट ट्रांसफर प्रोटोकॉल सुरक्षित):

एचटीटीपीएस एक सुरक्षित इंटरनेट प्रोटोकॉल है जो वेबसाइट्स से डेटा डाउनलोड और अपलोड करने की सुरक्षित अनुमति देता है। जब आप किसी सुरक्षित वेबसाइट पर होते हैं (जिसका यूआरएल "https://" से शुरू होता है), तो आपका डेटा एन्क्रिप्ट होता है, जिससे अनधिकृत व्यक्तियों को इसे पढ़ना कठिन हो जाता है।

ii. वेब एड्रेस:

वेब एड्रेस एक विशिष्ट पहचानकर्ता है जिसका उपयोग इंटरनेट पर किसी संसाधन का पता लगाने के लिए किया जाता है। इंटरनेट पर हर वेबपेज का अपना विशिष्ट पता होता है जिससे लोगों को उसका पता लगाने में मदद मिलती है। इसे URL (Uniform Resource Locator) भी कहा जाता है। वेब एड्रेस में कई भाग होते हैं: एक प्रोटोकॉल और डोमेन नाम सहित।

iii. सर्च इंजन :

		<p>एक सर्च इंजन इंटरनेट पर एक उपकरण है जो आपको वेबसाइट्स के बड़े संग्रह से जानकारी ढूँढने में मदद करता है। यह इंटरनेट के लिए एक उच्चगुणवत्ता वाले पुस्तकालय की तरह होता है। जब आप कुछ एक सर्च इंजन, जैसे कि Google, में टाइप करते हैं, तो यह अरबों वेब पेज्स की जाँच करता है और शीघ्रता से आपके जिज्ञासा से संबंधित सभी वेबसाइट्स, लेख और चित्रों को ढूँढ़कर दिखाता है।</p> <p>iv. WWW (वर्ल्ड वाइड वेब): यह दस्तावेजों और अन्य वेब संसाधनों का एक संग्रह है जो URL द्वारा पहचाने जाते हैं और इंटरनेट के माध्यम से वेब ब्राउज़र द्वारा एक्सेस और खोजे जा सकते हैं। WWW को वेब भी कहा जाता है और इसका आविष्कार टिम बर्नर्स-ली ने 1989 में किया था। सभी वेब पेजों को केवल वेब ब्राउज़र की मदद से ही देखा जा सकता है।</p>	
		<p>½ marks for full form of HTTPS without spelling mistake</p> <p>½ marks for any correct explanation of HTTPS</p> <p>1 mark for any correct explanation of Web Address</p> <p>1 mark for any correct explanation of Search Engine</p> <p>½ marks for full form of WWW without spelling mistake</p> <p>½ marks for any correct explanation of WWW</p>	
14.		<p>What is Inheritance? Explain Single, Multiple and Multilevel inheritance in C++.</p> <p>इन्हेरिटेंस क्या है? C++ में सिंगल, मल्टीपल और मल्टीलेवल इन्हेरिटेंस की व्याख्या कीजिए।</p>	4
	Ans	<p>Inheritance is the ability of a class to inherit all the properties and characteristics of another class. The class that inherits the properties from another class is known as child or derived class. The class from which the properties is being derived and inherited is known as the parent or base class. Derived class can inherit data members, member functions of the base class.</p> <p>Real life example of Inheritance: A child inherits properties and characteristics from his/her parents, like, color, height, weight etc.</p> <p>The 3 types of inheritances which we have to explain are:</p> <ol style="list-style-type: none"> 1) Single Inheritance 2) Multiple Inheritance 3) Multilevel Inheritance <p>इन्हेरिटेंस एक क्लास की अन्य क्लास के सभी गुणों और विशेषताओं को प्राप्त करने की क्षमता है। वह क्लास जो किसी अन्य क्लास से गुण प्राप्त करती है, उसे चाइल्ड या डीराईव्ड क्लास के रूप में जाना जाता है। जिस क्लास से गुण प्राप्त किए जा रहे हैं और विरासत में मिले हैं, उन्हें पैरेंट या बेस क्लास के रूप में जाना</p>	

जाता है। चाइल्ड क्लास डेटा सदस्यों, बेस क्लास के मेम्बर और फंक्शन को प्राप्त कर सकता है।

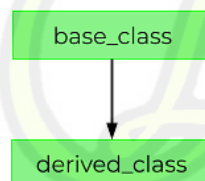
इनहेरिटेंस का वास्तविक जीवन उदाहरण: एक बच्चे को अपने माता-पिता से गुण और विशेषताएं विरासत में मिलती हैं, जैसे रंग, ऊंचाई, वजन आदि।

3 प्रकार के इनहेरिटेंस जिन्हें हमें समझाना है:

- 1) सिंगल इनहेरिटेंस
- 3) मल्टीपल इनहेरिटेंस
- 4) मल्टीलेवल इनहेरिटेंस

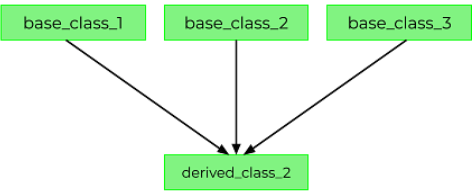
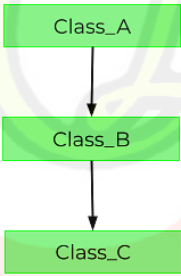
Single Inheritance: In this inheritance, a single class inherits the properties of a base class. All the data members of the base class are accessed by the derived class according to the visibility mode (i.e., private, protected, and public).

सिंगल इनहेरिटेंस: इस इनहेरिटेंस में, एक सिंगल क्लास बेस क्लास के गुणों को प्राप्त करता है। बेस क्लास के सभी डेटा सदस्यों को विसिबिलिटी मोड (यानी प्राइवेट, प्रोटेक्टेड और पब्लिक) के अनुसार डीराइव्ड क्लास द्वारा एक्सेस किया जाता है।



Multiple Inheritance: The inheritance in which a class can inherit or derive the characteristics of multiple classes, or a derived class can have over one base class, is known as Multiple Inheritance. The derived class can derive the joint features of all these classes and the data members of all the base classes.

मल्टीपल इनहेरिटेंस: वह इनहेरिटेंस जिसमें एक क्लास कई क्लास की विशेषताओं को प्राप्त कर सकती है, या एक डीराइव्ड क्लास में एक से अधिक बेस क्लास हो सकती हैं, मल्टीपल इनहेरिटेंस के रूप में जानी जाती है। डीराइव्ड क्लास इन सभी क्लास की संयुक्त विशेषताओं और सभी बेस क्लास के डेटा सदस्यों को प्राप्त कर सकती है।

		 <p>Multilevel Inheritance: The inheritance in which a class can be derived from another derived class is known as Multilevel Inheritance. Suppose there are three classes A, B, and C. B is the base class that is derived from class A. Now, C is the class that is derived from class B. This makes class B, the base class for class C but is the derived class of class A. This scenario is known as the Multilevel Inheritance.</p> <p><u>मल्टीलेवल इनहेरिटेंस:</u> वह इनहेरिटेंस जिसमें एक क्लास किसी अन्य डीराईव्ड क्लास से प्राप्त की जा सकती है, मल्टी लेवल इनहेरिटेंस के रूप में जानी जाती है। मान लीजिए कि तीन क्लास A, B और C हैं। B एक क्लास है जो क्लास A से प्राप्त होती है। अब, C वह क्लास है जो क्लास B से प्राप्त होती है। क्लास B, क्लास C के लिए बेस क्लास है लेकिन क्लास A की डीराईव्ड क्लास है। इसे मल्टीलेवल इनहेरिटेंस के रूप में जाना जाता है।</p> 	
		1 Mark for correctly defining Inheritance concept. 1 mark for correctly explaining Single inheritance. 1 mark for correctly explaining Multiple inheritance. 1 mark for correctly explaining Multilevel inheritance. Do not deduct any marks for not drawing diagrams. 1 mark for only making correct diagrams of all 3 inheritances.	
		OR	
		Write a program in C++ defining a class named Student with public member studentID , studentName , and studentmarks . Subsequently,	

		<p>implement a public member function named printStudentInfo that prints information for an object of the Student class.</p> <p>एक C++ प्रोग्राम लिखो जिसमें एक class को student के नाम से परिभाषित किया जाए, जिसमें studentID, studentName, और studentMarks नामक public सदस्य हों। इसके बाद, एक public member function printStudentInfo को लागू करें जो 'Student' कक्षा के ऑब्जेक्ट के लिए जानकारी प्रिंट करता है।</p>	
		<p>A C++ program defining a class named Student with public member variables (studentID, studentName, and studentMarks) and a public member function (printStudentInfo) that prints information for an object of the Student class:</p> <p>एक C++ प्रोग्राम जो public member variables (studentID, studentName, और studentMarks) के साथ Student नामक एक class को परिभाषित करता है और एक public member function (printStudentInfo) जो Class student के ऑब्जेक्ट के लिए जानकारी प्रिंट करता है:</p> <pre>#include <iostream> class Student { public: // Public member variables int studentID; char studentName[50]; // Using a character array for the name double studentMarks; // Public member function to print student information void printStudentInfo() { cout << "Student ID: " << studentID << endl; cout << "Student Name: " << studentName << endl; cout << "Student Marks: " << studentMarks << endl; } }; int main() { // Creating an object of the Student class Student student1; // Assigning values to the member variables student1.studentID = 101;</pre>	

	<pre> student1.studentName[0] = 'R'; student1.studentName[1] = 'a'; student1.studentName[2] = 'm'; student1.studentMarks = 95.5; // Calling the printStudentInfo function to display information student1.printStudentInfo(); return 0; } </pre> <p><u>Output of the above c++ program:</u> <u>उपरोक्त c++ प्रोग्राम का आउटपुट:</u></p> <p>Student ID: 101 Student Name: Ram Student Marks: 95.5</p>	
	<p>½ marks for correct header file inclusion. ½ marks for defining class Student. ½ marks for declaring data members. ½ marks for defining printStudentInfo() function. ½ marks for defining main() function. ½ marks for creating object in main function. ½ marks for calling printStudentInfo() function. ½ marks for correct output of the program.</p> <p>Do not deduct any marks for case sensitivity issue (if any). Do not deduct any marks for spelling mistakes in variable names.</p>	