

**GOVERNMENT POLYTECHNIC, PUNE**  
**'120 – NEP' SCHEME**

<b>PROGRAMME</b>	<b>DIPLOMA IN CM</b>
<b>PROGRAMME CODE</b>	<b>06</b>
<b>COURSE TITLE</b>	<b>COMPUTER NETWORK</b>
<b>COURSE CODE</b>	<b>CM31206</b>
<b>PREREQUISITE COURSE CODE &amp; TITLE</b>	<b>NA</b>
<b>CLASS DECLARATION COURSE</b>	<b>NO</b>

**I. LEARNING & ASSESSMENT SCHEME**

Course Code	Course Title	Course Type	Learning Scheme						Credits	Paper Duration	Assessment Scheme										Total Marks
			Actual Contact Hrs./Week			SLH	NLH	Theory			Based on LL & TSL				Based on SL						
			CL	TL	LL						FA-TH	SA-TH	Total	FA-PR		SA-PR		SLA			
														Max	Min	Max	Min		Max	Min	
CM31206	COMPUTER NETWORK	DSC	3	1	2	--	6	3	3	30	70	100	40	25	10	25@	10	--	--	150	

**Total IKS Hrs for Term:** 0 Hrs

**Abbreviations:** CL-Classroom Learning, TL-Tutorial Learning, LL-Laboratory Learning, SLH-Self Learning Hours, NLH-Notional Learning Hours, FA - Formative Assessment, SA -Summative assessment, IKS – Indian Knowledge System, SLA- Self Learning Assessment

**Legends:** @-Internal Assessment, #- External Assessment,\*# - Online Examination,@\$ - Internal Online Examination

**Note:**

**FA-TH** represents an average of two class tests of 30 marks each conducted during the semester.

- If a candidate is not securing minimum passing marks in **FA-PR** (Formative Assessment - Practical) of any course, then the candidate shall be declared as '**Detained**' in that course.
- If a candidate does not secure minimum passing marks in SLA (Self Learning Assessment) of any course, then the candidate shall be declared as '**fail**' and will have to repeat and resubmit SLA work.
- Notional learning hours** for the semester are **(CL + LL + TL + SL) hrs. \* 15 Weeks**
- 1 credit** is equivalent to **30 Notional hours**.
- \* Self-learning hours shall not be reflected in the Timetable.
- \* Self-learning includes micro-projects/assignments/other activities.

**II. RATIONALE:**

In today's world computers have become an integral part of the business sector for professional activities not only for professional activities but for personal activities also. As technologies have evolved networking came into the picture and slowly from initial wired network technology we moved to this wireless network technology. For diploma students, it is important to understand the function of computer networks and obtain requisite knowledge about hardware and software requirements of networks and acquire skills to establish a network using necessary hardware & software tools and configure various services over it. The objectives of this course are to make students learn the technology of designing, building and maintaining computer networks.

**III. COURSE-LEVEL LEARNING OUTCOMES (CO's)**

Students will be able to achieve & demonstrate the foLLoWing CO's on completion of course-based learning:

- CO1. Classify various computer networks.
- CO2 Identify relevant transmission media and switching techniques.
- CO3 Recognize TCP/IP protocols in the standard network model.

- CO4. Configure the network using connecting devices and IP addressing  
 CO5. Perform error detection and correction in network communication.  
 CO6. Build a network using wireless technology

#### IV. THEORY LEARNING OUTCOMES AND ALIGNED COURSE CONTENT

Sr. No	Theory Learning Outcomes (TLO's) aligned to CO's.	Learning content mapped with TLO's.	Suggested Learning Pedagogies	Relevant COs
<b>UNIT I FUNDAMENTALS OF COMPUTER NETWORK (CL Hrs-06, Marks-12)</b>				
1.	<p><b>TLO1.1</b> List the benefits and applications of Computer Networks.</p> <p><b>TLO1.2</b> Classify computer networks according to their geography</p> <p><b>TLO1.3</b> Select appropriate network architecture according to the requirements.</p>	<p><b>1.1</b> Fundamentals of Computer Network- Definition Need of Computer Network, Applications, Component of Computer Network.</p> <p><b>1.2</b> Network Benefits: Sharing Information (File Sharing, Email), Sharing Resources (Printer Sharing, Application Services), Maintaining the Network and backing up data.</p> <p><b>1.3</b> Computer Network Classifications- Classification of Networks by their Geography.- LAN, MAN, WAN.</p>	Hands-on Demonstration Presentations	<b>CO1</b>
<b>UNIT II TRANSMISSION MEDIA (CL Hrs-07, Marks-12)</b>				
2	<p><b>TLO2.1</b> Explain guided and unguided media.</p> <p><b>TLO2.2</b> Select appropriate media for network implementation</p> <p><b>TLO2.3</b> Differentiate between circuit-switched and datagram networks.</p>	<p><b>2.1</b> Guided Media: Types of Cable-Twisted Pair Cable, Co-axial Cable, Fiber Optic Cable.</p> <p><b>2.2</b> Unguided media: Microwave Communication, Radio wave Communication, Satellite Communication.</p> <p><b>2.3</b> Switching Basics: Circuit Switched Networks and Datagram Networks.</p>	Hands-on Demonstration Presentations	<b>CO2</b>
<b>UNIT III NETWORK COMMUNICATION MODELS(CL Hrs-10, Marks-14)</b>				
3	<p><b>TLO 3.1</b> Identify functions and features of the given layer of the OSI Reference model.</p> <p><b>TLO 3.2</b> Compare the specified service on the given parameters.</p> <p><b>TLO 3.3</b> Describe with sketches the procedure to configure the given TCP/IP service.</p>	<p><b>3.1</b> THE OSI MODEL: Layered Architecture, Encapsulation</p> <p><b>3.2</b> Layers in OSI Model(Functions of each layer)-Physical Layer, Data-Link Layer, Network Layer, Transport Layer, Session Layer, Presentation Layer, Application Layer</p> <p><b>3.3</b> TCP/IP Layers and their functions: Host Network Layer, Internet Layer, Transport Layer, Application Layer</p> <p><b>3.4</b> Protocols: Host To Network Layer-SLIP, PPP, Internet Layer-IP, ARP, RARP, ICMP, Transport Layer-TCP and UDP, Application Layer-FTP, HTTP, SMTP, TELNET, BOOTP, DHCP</p>	Hands-on Demonstration Presentations	<b>CO5</b>

UNIT IV NETWORK TOPOLOGIES, DEVICES AND IP ADDRESSING (CL Hrs-10, Marks-14)				
4	<p><b>TLO 4.1</b> Compare different computing models on the given parameter.</p> <p><b>TLO 4.2</b> Identify relevant network topology for the given situation.</p> <p><b>TLO 3.3</b> Compare different topologies on the given parameter.</p> <p><b>TLO 4.4</b> Select the network connecting device for the given situation.</p> <p><b>TLO 4.5</b> Describe with sketches the procedure to configure the given networking device.</p> <p><b>TLO 4.6</b> Classifies IP Addresses based on their class from the given set of addresses.</p> <p><b>TLO 4.7</b> Distinguish between IPv4 and IPv6 on the given parameters.</p>	<p><b>4.1</b> Network Computing Model: Peer To Peer, Client Server</p> <p><b>4.2</b> Network Topologies: Introduction, Definition, Selection criteria, Types of Topology- Star, Mesh, Tree, Hybrid CSMA/CA, CAMS/CD, TOKEN RING, TOKEN BUS</p> <p><b>4.3</b> Network Connecting Devices: Switch, Router, Repeater, Bridge, Gateways and Modem</p> <p><b>4.4</b> Addressing: Physical Address, Logical Address, Port Address</p> <p><b>4.5</b> IP Address-Concept, Notation, Address Space</p> <p><b>4.6</b> IPv4 Addressing: Classful and Classless Addressing, subnet mask, super netting, subnetting</p> <p><b>4.7</b> IPV6 Addressing Scheme and basic structure</p>	Hands-on Demonstration Presentations	CO6
UNIT V ERROR DETECTION AND CORRECTION (CL Hrs-06, Marks-08)				
5	<p><b>TLO 5.1</b> Explain the working of the given error detection and correction method</p> <p><b>TLO 5.2</b> Classify framing types</p> <p><b>TLO 5.3</b> Describe error detection</p> <p><b>TLO 5.4</b> Perform error correction</p>	<p><b>5.1</b> Types of Errors, Forward Error Correction Versus Retransmission</p> <p><b>5.2</b> Framing: Fixed-Sized and Variable-Sized Framing</p> <p><b>5.3</b> Error Detection: Repetition codes, Parity bits, Checksums, CRC</p> <p><b>5.4</b> Error Correction: Automatic Repeat, Request (ARQ), Hamming Code</p>		
UNIT VI WIRELESS COMMUNICATION (CL Hrs-06, Marks-10)				
6	<p><b>TLO 6.1</b> Explain the characteristics of the given layer in IEEE 802.11 architecture.</p> <p><b>TLO 6.2</b> Explain with sketches the process of creating a Bluetooth environment using the given architecture.</p> <p><b>TLO 6.3</b> Compare the specified generations of mobile telephone systems on the given parameter.</p>	<p><b>6.1</b> Wireless LAN IEEE 802.11 standard Architecture, Features of IEEE 802.11 versions: 802.11, 802.11a, 802.11b, 802.11g, 802.11n, 802.11p</p> <p><b>6.2</b> Bluetooth Architecture: Piconet, Scatternet</p> <p><b>6.3</b> Mobile Generations: 3G, 4G and 5G</p>	Hands-on Demonstration Presentations	CO4

**V. LABORATORY LEARNING OUTCOME AND ALIGNED PRACTICAL/ TUTORIAL EXPERIENCES.**

Sr. No	Practical/Tutorial/Laboratory Learning Outcome (LLO)	Laboratory Experiment/ Practical Titles /Tutorial Titles	Number of hrs.	Relevant COs
1	LLO1: Survey existing network infrastructure	*Visit your computer laboratory Identify the type of topology	1	1

		ii) Identify types of connecting devices with specifications iii) Identify types of cables with specifications iv) List the type of network applications commonly used in the laboratory v) Draw the layout of the installed network		
2	<b>LLO2:</b> Create a standard network straight cable by using a cable tester.	*Create and Test standard straight network cable(Universal Colour Code) using crimping tool	1	2
3	<b>LLO3:</b> Create standard cross-network cable by using a cable tester.	Create and Test standard Cross network cable(Universal Colour Code) using a crimping tool	1	2
4	<b>LLO4:</b> Create a Small LAN	To connect and understand different network control devices used in LAN	1	1
5	<b>LLO5:</b> Create a star topology network	Connect the computer in a star topology using wired and wireless medium.	2	4
6	<b>LLO6:</b> Use the IP address and appropriate subnet mask for the given problem statement.	*Configure static IP address in the operating system along with appropriate subnet mask for a given problem	2	4
7	<b>LLO7:</b> Troubleshoot computer network using commands	*Run network diagnostic commands: ipconfig, ping, tracert, netstat, route, arp	1	3
8	<b>LLO8:</b> Implement IPv6 addressing scheme on a network	*Create a network based on IPV6 address using a simulator.	2	4
9	<b>LLO9:</b> Use Wireshark packet sniffer software.	*Install Wireshark and configure as packet sniffer- i) Capture IP, TELNET, and FTP packets using Wireshark	2	3
10	<b>LLO10:</b> Implement IP addresses for the intranet in Class A, Class B, and Class C.	* Implement Classful Address in a given network node i) Identify a range of IP Addresses in various classes ii) Justify the reason for choosing various IP address classes for creating a given network	2	4
11	<b>LLO11:</b> Identify different error detection methods. <b>LLO 8.2</b> Detect errors using Checksum	Locate the error bit in the given data string by applying the checksum error detection method	2	5
12	<b>LLO12.1:</b> Draw a block diagram for parity check. <b>LLO12.2:</b> Implement parity check with examples.	Write a 'C' program for parity check error detection	2	5
13	<b>LLO13:</b> Transfer data using Bluetooth.	*Create a Hybrid Network Using Bluetooth	1	6
14	<b>LLO14:</b> Use Network simulator CISCO packet tracer.	Create Tree topology using CISCO packet tracer software	1	6

**Note:**

1. Out of the above suggestive **LLOs** – '\*' Marked Practicals (**LLOs**) Are mandatory.
2. Judicial mix of **LLOs** is to be performed to achieve desired outcomes.

## VI. SUGGESTED MICRO PROJECT/ASSIGNMENT/ACTIVITIES FOR SPECIFIC LEARNING/SKILLS DEVELOPMENT (SELF-LEARNING)

**Self-Learning** NA



**Micro project**

Suggested list of case studies for Micro Project:

1. Planning Network-based Firewalls
2. Case study for creation of network with IPV6/ARP/RARP
3. Analyse the UDP/TCP packet of GPP network using wireshark,analyze it and write your observation.
4. Configuring a Cisco Router as a DHCP Server
5. Install printer in GPP NETWORK also share the folder of one of the computers in network
6. Write steps required to troubleshoot Network as an administrator if any website is not responding.
7. If students laptop's wireless connection is not working then how to add laptops to the organisations LAN/intranet.
8. Visit to GPP website and observe ARP,ICMP,TCP,UDP packet /frame structure using any simulator ex:wireshark
9. Create a smart 2/3 computer network having IPV6 addresses using simulator,All computers should be able to communicate and able to share data
10. Observing Static and Dynamic Routing
11. Examining Network Address Translation (NAT)
12. Configure router (Home or Small office home office Device)
13. Execute TCP/IP network commands:ipconfig,ping,tracert
14. Execute TCP/IP network commands:, pathping, route
15. Capture TCP and UDP packet using Wireshark
16. Capture ARP and ICMP packet Traffic using Wireshark
17. Performing an Initial Switch Configuration
18. Write a 'C' program for Cyclic Redundancy Check(CRC) error detection
19. Configuring and Troubleshooting a Switched Network
20. Configuring WEP on a Wireless Router
21. Using the Cisco IOS Show Commands
22. Exploring Different LAN Switch Options
23. Configuring Ethernet and Serial Interfaces
24. Install and configure NIC and find MAC Address of Device Design a network using any topology and d
25. Create a tool that monitors network bandwidth usage in real-time

**Assignment**

Prepare a journal of practicals performed in the laboratory.

**VII. LABORATORY EQUIPMENT/INSTRUMENTS/TOOLS/SOFTWARE REQUIRED**

Sr.No	Equipment Name with Broad Specifications	Relevant LLO Number
1	Computer Systems (NIC Installed)	All
2	Network control devices and transmission media	All
3	Crimping Tool, RJ45 Connector	2
4	Wireshark sniffer software	10

VIII. SUGGESTED FOR WEIGHTAGE TO LEARNING EFFORTS & ASSESSMENT PURPOSE

(Specification Table)

Sr. No	Unit	Unit Title	Aligned COs	Learning Hours	R-Level	U-Level	A-Level	Total Marks
1	I	Fundamentals of computer network	1	6	6	4	2	12
2	II	Transmission Media	2	7	4	6	2	12
3	III	Network Communication Models	3	10	4	6	4	14
4	IV	Network Topologies, Devices And IP Addressing	4	10	4	4	6	14
5	V	Error Detection/Correction	5	6	—	4	4	08
6	VI	Wireless Communication	6	6	4	4	2	10
<b>Grand Total</b>				<b>30</b>	<b>22</b>	<b>36</b>	<b>24</b>	<b>70</b>

IX. ASSESSMENT METHODOLOGIES/TOOLS

Formative assessment (Assessment for Learning)	Summative Assessment (Assessment of Learning)
Lab performance, Assignment, Self-learning and Seminar/Presentation	Lab. Performance, viva voce

X. SUGGESTED COS- POS MATRIX FORM

Course Outcomes (Cos)	Programme Outcomes(Pos)							Programme Specific Outcomes *(PSOs)	
	PO-1 Basic and Discipline-Specific Knowledge	PO-2 Problem Analysis	PO-3 Design/ Development of Solutions	PO-4 Engineering Tools	PO-5 Engineering Practices for Society, Sustainability and Environment	PO-6 Project Management	PO-7 Life Long Learning	PSO-1	PSO-2
CO1	1	1	1	1	1	1	1	2	-
CO2	2	—	2	2	1	1	1	2	-
CO3	1	2	2	2	2	2	1	1	2
CO4	1	2	2	2	2	3	1	1	2
CO5	1	3	3	2	1	2	1	1	3
CO6	1	2	1	2	2	1	1	1	3

**Legends:- High:03, Medium:02, Low:01, No Mapping: -**  
 \*PSOs are to be formulated at the institute level





XI. SUGGESTED LEARNING MATERIALS/BOOKS

Sr.No	Author	Title	Publisher
1	Behrouz A. Forouzan	Data Communication and Networking	McGraw-Hill Higher Education ISBN-13 978-0-07- 296775-3
2	Behrouz A. Forouzan:	TCP/IP Protocol Suit	McGraw Hill Education ISBN-13 978-0073376042

3	A.S. Tanenbaum	Computer Networks	PRENTICE HALL ISBN-10: 0-13-212695-8 , ISBN 13:978-0-13-21269
4	Godbole Achyut	Data Communication and Networks	McGraw Hill Education ISBN-10 9780071077705,ISBN-13 978-0071077705
5	Comer Douglas E.	TCP/IP Principles, Protocols and Architectures	PEARSON ISBN 10: 0-13-608530-X ISBN 13: 978- 0-13-608530-0

## XII. LEARNING WEBSITES &amp; PORTALS

Sr.No	Link / Portal	Description
1	<a href="https://www.geeksforgeeks.org/data-communication-definition-components-types-channels/">https://www.geeksforgeeks.org/data-communication-definition-components-types-channels/</a>	All practicals
2	<a href="https://www.tutorialspoint.com/data_communication_computer_n">https://www.tutorialspoint.com/data_communication_computer_n</a>	All practicals
3	<a href="https://nptel.ac.in/courses/106105081">https://nptel.ac.in/courses/106105081</a>	All practicals
4	<a href="https://nptel.ac.in/courses/106105183">https://nptel.ac.in/courses/106105183</a>	All practicals
5	Introduction To Computer Networks   Study tonight	All practicals

Name & Signature:  <b>Smt. Khushboo S. Sathawane</b> Lecturer in Computer Engineering		Name & Signature:  <b>Smt. Aafiya A. Shaikh</b> Lecturer in Computer Engineering	
<b>(Course Experts)</b>			
Name & Signature:  <b>Smt. J.R. Hange</b> (Programme Head)		Name & Signature:  <b>Shri. S.B. Kulkarni</b> (CDC In-charge)	

**GOVERNMENT POLYTECHNIC, PUNE**  
**'120 – NEP' SCHEME**

<b>PROGRAMME</b>	<b>DIPLOMA IN CM/IT</b>
<b>PROGRAMME CODE</b>	<b>06/07</b>
<b>COURSE TITLE</b>	<b>DIGITAL TECHNIQUES AND MICROPROCESSOR PROGRAMMING</b>
<b>COURSE CODE</b>	<b>CM31204</b>
<b>PREREQUISITE COURSE CODE &amp; TITLE</b>	<b>NA</b>
<b>CLASS DECLARATION COURSE</b>	<b>NO</b>

**I. LEARNING & ASSESSMENT SCHEME**

Course Code	Course Title	Course Type	Learning Scheme						Credits	Paper Duration	Assessment Scheme										Total Marks
			Actual Contact Hrs./Week			SLH	NLH	Theory			Based on LL & TSL				Based on SL						
			CL	TL	LL			Total			Practical		SLA								
											FA-TH	SA-TH	FA-PR	SA-PR	Max	Min	Max	Min			
CM31204	DIGITAL TECHNIQUES AND MICROPROCESSOR PROGRAMMING	AEC	3	1	2	2	8	4	3	30	70	100	40	25	10	25@	10	25	10	175	

**Total IKS Hrs for Term:** 01 Hr

**Abbreviations:** CL-Classroom Learning, TL-Tutorial Learning, LL-Laboratory Learning, SLH-Self Learning Hours, NLH-Notional Learning Hours, FA - Formative Assessment, SA -Summative assessment, IKS – Indian Knowledge System, SLA- Self Learning Assessment

**Legends:** @-Internal Assessment, # - External Assessment, \*# - Online Examination, @\$ - Internal Online Examination

**Note:**

**FA-TH** represents an average of two class tests of 30 marks each conducted during the semester.

- If a candidate is not securing minimum passing marks in **FA-PR** (Formative Assessment - Practical) of any course, then the candidate shall be declared as '**Detained**' in that course.
- If a candidate does not secure minimum passing marks in SLA (Self Learning Assessment) of any course, then the candidate shall be declared as '**fail**' and will have to repeat and resubmit SLA work.
- Notional learning hours** for the semester are **(CL + LL + TL + SL) hrs. \* 15 Weeks**
- 1 credit** is equivalent to **30 Notional hours**.
- \* Self-learning hours shall not be reflected in the Timetable.
- \* Self-learning includes micro-projects/assignments/other activities.

**II. RATIONALE:**

As a computer engineering student, it is essential to know the fundamentals of digital electronics to understand the concept of microprocessor and its application. The microprocessor is challenging, to meet the challenges of growing advanced microprocessor technology. The student should be conversant with microprocessor programming

**III. COURSE-LEVEL LEARNING OUTCOMES (CO's)**

Students will be able to achieve & demonstrate the following CO's on completion of course-based learning

- CO1: Use the number system and codes of the digital system.
- CO2: Simplify Boolean expressions for logic circuit.
- CO3: Analyze 8086 microprocessor functionality.
- CO4: Develop assembly language programs.
- CO5: Use procedure and macro in assembly language programs.



## IV. THEORY LEARNING OUTCOMES AND ALIGNED COURSE CONTENT

Sr. No	Theory Learning Outcomes (TLO's) aligned to CO's.	Learning content mapped with TLO's.	Suggested Learning Pedagogies	Relevant COs
<b>UNIT-I NUMBER SYSTEM, CODES &amp; LOGIC GATES AND BOOLEAN ALGEBRA (CL Hrs-10, Marks-18)</b>				
1	<p><b>TLO1.1</b> Convert codes from one number system to another.</p> <p><b>TLO1.2-</b> Perform arithmetic operations with different number systems.</p> <p><b>TLO1.3</b> Differentiate various logic gates and apply the logic on Boolean algebra.</p> <p><b>TLO1.4</b> Explain theorems for Boolean algebra.</p> <p><b>TLO1.5</b> Create simplified logic circuits</p>	<p><b>1.1</b> Introduction to Number systems: Decimal, Binary, Octal, hexadecimal</p> <p><b>1.2</b> Binary arithmetic: Addition, subtraction, multiplication, Division</p> <p><b>1.3</b> One's complement, Two's Complement, Signed Numbers, Codes, Error code.</p> <p><b>1.4</b> LogicGates: Introduction, Working principles and Truth of AND, OR, NOT, NOR, NAND, EX-OR, EX-NOR Gates, Universal Gates.</p> <p><b>1.5</b> Boolean Algebra: Basic Boolean Operations, Basic Laws of Boolean Algebra, Duality Theorem, De-Morgan's Theorems</p>	<p>Classroom Learning/ Flipped Classroom/ Collaborative Learning/ Use of logic simulator like Virtual Labs/online converters etc</p>	CO1
<b>UNIT-II COMBINATIONAL AND SEQUENTIAL LOGIC CIRCUITS (CL Hrs-10, Marks-16)</b>				
2	<p><b>TLO2.1</b> Construct K-MAP using logic functions and vice versa.</p> <p><b>TLO2.2</b> Simplify equations in the minterms/maxterms.</p> <p><b>TLO2.3</b> Design Multiplexer and De-Multiplexer.</p> <p><b>TLO2.4:</b>Implement combinational logic design with multiplexers.</p> <p><b>TLO2.5:</b>Implement combinational logic design with demultiplexers.</p>	<p><b>2.1</b> KARNAUGH map representation, Simplification of logic function using K-MAP.</p> <p><b>2.2</b> Minimization of logical function specified in minterms/maxterms or truth table Don't care conditions.</p> <p><b>2.3</b> Multiplexers and their use in combinational, logic design</p> <p><b>2.4</b> De-multiplexer/decoders and their use in combinational logic design</p> <p><b>2.5</b> De-multiplexer: 4 to 16-line DEMUX. Demux design using the sop method. 1:4, 1:8, 1:16 DEMUX.</p> <p><b>2.6</b> Clock signal, flipflop, latches, counter, buffer and tri-state buffer (only concept)</p>	<p>Lecture Using Chalk-Board Flipped Classroom Collaborative Learning Virtual Lab</p>	CO2
<b>UNIT- III MICROPROCESSOR ARCHITECTURE &amp; MICROCOMPUTER SYSTEMS(CL Hrs-08, Marks-10)</b>				
3	<p><b>TLO3.1:</b> Describe Microprocessor architecture.</p> <p><b>TLO 3.2:</b> Understand 8086 registers and instruction format.</p> <p><b>TLO 3.3:</b> Draw a timing diagram for the read/write memory cycle.</p>	<p><b>3.1</b> Microprocessor – Introduction, Features, and its Operations</p> <p><b>3.2</b> 8086 Microprocessor - Introduction, Architecture, and Working, Pin configuration, Memory segmentation in 8086.</p> <p><b>3.3</b> Minimum mode and Maximum mode configuration of 8086, Timing diagram Minimum mode and Maximum mode 8086.</p>	<p>Classroom Learning Flipped Classroom Cooperative Learning</p>	CO3

UNIT –IV 8086 ASSEMBLY LANGUAGE PROGRAMMING (CL Hrs-10 Marks-16)				
4	<p><b>TLO 4.1</b> Write and execute 8086 programs for addition and subtraction.</p> <p><b>TLO 4.2</b> Write programs implementing branching.</p>	<p><b>4.1</b> Instruction format and Addressing modes in 8086, Assembler and Directives.</p> <p><b>4.2</b> 8086 Instructions set and classification of instructions - Arithmetic, Logical, Data transfer, String, Bit manipulation, Flag manipulation, Branching, Machine Control.</p>	<p>Classroom Learning</p> <p>Collaborative Learning</p> <p>Flipped Classroom</p> <p>Program development tools and simulators</p>	CO4
UNIT V –PROCEDURE AND MACRO IN ASSEMBLY LANGUAGE PROGRAM (CL Hrs-07, Marks-10)				
5	<p><b>TLO 5.1</b> Write and execute assembly language programs using procedures.</p> <p><b>TLO 5.2</b> Write and execute assembly language programs using macros.</p>	<p><b>5.1</b> Procedures - Defining Procedure, Directives used, FAR and NEAR, CALL and RET instructions, Assembly Language Programs using Procedure.</p> <p><b>5.2</b> Macros - Defining Macros, Assembly Language Programs using Macros, Directives used.</p>	<p>Classroom Learning</p> <p>Collaborative Learning</p> <p>Flipped Classroom</p> <p>Program development tools and simulators</p>	CO5

#### V. LABORATORY LEARNING OUTCOME AND ALIGNED PRACTICAL/ TUTORIAL EXPERIENCES.

Sr. No	Practical/Tutorial/Laboratory Learning Outcome (LLO)	Laboratory Experiment/ Practical Titles /Tutorial Titles	Number of hrs.	Relevant COs
1	<b>LLO 1.1:</b> Describe the basic component of digital lab.	*Know the Digital Lab IC Tester, Multimeter, Bread Board, Trainer Kit.	2	CO1
2	<b>LLO 2.1:</b> Implement the basic Gate	*Study of Basic Gates ICs (7400, 7404, 7408, 7486, 7432) and verification of Truth tables by monitoring the output of ICs on BreadBoard.	2	CO1
3	<b>LLO 3.1:</b> Implement the Derived Gate	*To derive AND, OR, NOT gates using universal gates by forming circuits on the Breadboard.	2	CO1
4	<b>LLO 4.1:</b> Verify De-Morgan's Theorem using the basic gate.	*Verify De-Morgan's Theorem by forming the circuit on BreadBoard.	2	CO1
5	<b>LLO 5.1:</b> Desing K map for SOP and POS forms, minimized it and designed circuit.	*Minimization and realization of function using K-maps and its implementation by constructing the circuit on the breadboard.	2	CO2
6	<b>LLO 6.1:</b> Develop an assembly language program to addition and subtraction 8 bit/16-bit signed/ unsigned numbers	* Write an Assembly language Program(ALP) for Addition and subtraction of two 16-bit numbers.	2	CO4

Sr. No	Practical/Tutorial/Laboratory Learning Outcome (LLO)	Laboratory Experiment/ Practical Titles /Tutorial Titles	Number of hrs.	Relevant COs
7	<b>LLO 7.1:</b> Develop an assembly language program to divide and Multiplication two 8-bit /16-bit signed/ unsigned numbers	*Write an Assembly language Program(ALP) to Multiplication and division of 8-bit/16-bit/32-bit signed/ unsigned numbers.	2	CO4
8	<b>LLO 8.1:</b> Develop assembly language programming for finding the Sum of a given series of numbers.	* Write an Assembly language Program(ALP) to Sum of given series of numbers (8-bit / 16-bit).	2	CO4
9	<b>LLO 9.1:</b> Develop an assembly language program to Sort numbers of given arrays in greatest order	*Write an Assembly language Program(ALP) to Find the smallest and greatest number from the given series	2	CO4
10	<b>LLO 10.1:</b> Develop an assembly language program to Sort numbers of given arrays in ascending and descending order	*Write an Assembly language Program(ALP) to Arrange the given numbers in ascending and descending order.	2	CO4
11	<b>LLO 11.1:</b> Develop an assembly language program using String	*Write an Assembly language Program(ALP) to String related programs (any 5)	2	CO4
12	<b>LLO 12.1</b> Develop an assembly language program using Procedure.	*Write an Assembly language Program(ALP) to Programs using Procedure, Macros – 2 of each	2	CO5

**Note: Out of the above suggestive LLOs –**

1. '\*' Marked Practicals (LLOs) Are mandatory.
2. A judicious mix of LLOs is to be performed to achieve the desired outcomes

## VI. SUGGESTED MICRO PROJECT/ASSIGNMENT/ACTIVITIES FOR SPECIFIC LEARNING/SKILLS DEVELOPMENT (SELF-LEARNING)

### Self-Learning

Following are some suggestive case studies:

#### 1) Designing a Digital Combination Lock System

A company wants to enhance security measures by implementing a digital combination lock system for access control to sensitive areas. The system needs to accept a predefined numeric code and grant access if the correct code is entered.

#### 2) Converting Temperatures: A Case Study on Number Systems

A weather forecasting agency needs to convert temperatures between different scales for accurate reporting. They want to develop a simple system to automate these conversions.

#### 3) Barcode System Implementation for Inventory Management

A small retail store wants to streamline its inventory management process by implementing a barcode system. They aim to track product information efficiently and improve inventory accuracy.

**4) Automatic Light Control System Using Logic Gates**

A homeowner wants to install an automatic light control system to save energy and enhance convenience. They aim to develop a simple system that turns on lights in specific areas of the house based on motion detection and ambient light levels.

**5) Digital Door Lock System Using Boolean Algebra**

A small office wants to enhance security by installing a digital door lock system. They aim to design a system that grants access only to authorized personnel by using a combination of key codes.

**6) Simplifying Boolean Functions Using Karnaugh Maps**

A small electronics company is designing a circuit for a simple electronic device. They need to simplify the Boolean functions representing the device's logic to minimize the number of logic gates used and optimize circuit performance.

**7) Data Selection Using Multiplexers in a Computer Peripheral**

A computer peripheral manufacturer is designing a data input/output system for their new product. They need to incorporate a multiplexer to efficiently select data from multiple sources and route it to the main processing unit.

**8) LED Display Control Using Demultiplexers**

A small electronics hobbyist is designing a LED display system for a project. They need to incorporate demultiplexers to efficiently control the display of different patterns or characters on the LEDs.

**9) Implementing a Real-Time Monitoring System Using the Intel 8086 Microprocessor**

A manufacturing plant requires a real-time monitoring system to track various parameters such as temperature, pressure, and humidity in critical areas of the facility. The system needs to collect data continuously, process it in real-time, and provide alerts in case of anomalies.

**10) Developing a Simple Calculator Using 8086 Assembly Language Programming**

A computer science student is learning 8086 assembly language programming and wants to develop a simple application to reinforce their understanding of the language. They decide to create a basic calculator program that can perform addition, subtraction, multiplication, and division operations on two operands entered by the user.

**11) Automated Report Generation Using Macro Assembly Language Programming**

A small business wants to automate the process of generating monthly sales reports for their products. They have a database containing sales data for each product, including sales quantity and revenue. To streamline the report generation process, they decide to develop a macro assembly language program that can read the sales data from the database, perform calculations, and generate the monthly sales report automatically.

**12) File Processing Utility Using Procedure Assembly Language Programming**

A software development company needs to create a file processing utility to perform various operations on text files, such as counting the number of lines, searching for specific strings, and extracting data based on user-defined criteria. To achieve this, they decide to develop a procedure-based assembly language program that can handle file input/output operations and perform the required file processing tasks efficiently.

**13) Binary to Gray Code Converter and Display**

Design a digital system that converts a 4-bit binary number to its Gray code equivalent and displays the Gray code using a seven-segment display.

**Note:**

1. The above is suggestive list of case studies for SLA
2. The faculty must allocate any 1 case study in group of 2 students. Considering the students technical skills.



**Activities**

- Students are encouraged to use online tools to improve their learning, such as the e-Kumbh from AICTE and the virtual Labs from IIT.
- Students should be encouraged to participate in various coding competitions, such as hackathons, and online coding contests on websites like Hackerrank, Codechef etc.
- At the department level, encourage students to start a coding club
- Students are encouraged to register themselves in various MOOCs such as Infosys Springboard, Swayam etc. to further enhance their learning.

**Assignment**

Prepare a journal of practicals performed in the laboratory.

**IKS: Invention of Zero**

<https://sites.tufts.edu/tquinto/files/2021/01/HistoryOfZero.pdf>

**VII. LABORATORY EQUIPMENT/INSTRUMENTS/TOOLS/SOFTWARE REQUIRED**

Sr.No	Equipment Name with Broad Specifications	Relevant LLO Number
1	1) Digital Multimeter: 3 and 1/2 digit 2) Pulse Generator/Function Generator: TTL Pulse Generator 20mA per Channel(max), 0 to 5.0 V (max) 3) DC Regulated Power Supply: 2 x 0-30 V; 0-2 AA Automatic Overload (Current Protection) Constant Voltage and Constant Current Operation Digital Display for Voltage and Current Adjustable Current Limiter Excellent Line and Load Regulation 4) Basic logic gates (AND-7408, OR- 7432, NOT- 7404), Universal gates (NAND7400, NOR-7402) EX-OR-7486, EX-NOR-74266 5) 4:1 Multiplexer IC-74LS153 6) Demultiplexer IC -74139 7) Breadboards, connecting wires, Stripper, Soldering Gun, Soldering Metal, Flux, IC Tester, LEDs, Digital ICs, Data sheets of ICs used in Lab.	1,2,3,4,5
2	1) Personal Computer Intel Pentium Onwards Minimum 2GB RAM. 500Gbyte HDD) installed with Windows 2000 onwards 2) Any Editor to write/edit programs 3) Turbo/Macro Assembler (TASM / MASM) 4) Turbo Linker (LINK/LINK 5) Turbo Debugger (ID/Debug), (DOSBOX utility for higher-end operating systems)	6,7,8,9,10,11,12

**VIII. SUGGESTED FOR WEIGHTAGE TO LEARNING EFFORTS & ASSESSMENT PURPOSE****(Specification Table)**

Sr. No	Unit	Unit Title	Aligned COs	Learning Hours	R-Level	U-Level	A-Level	Total Marks
1	I	Number System, Codes & Logic Gates and Boolean Algebra	CO1	10	04	04	10	18
2	II	Combinational and Sequential Logic Circuits	CO2	10	03	03	10	16
3	III	Microprocessor Architecture & Microprocessor	CO3	8	02	02	06	10

		System						
4	IV	8086 Assembly Language Programming	CO4	10	04	04	08	16
5	V	Procedure and Macro in Assembly Language Program	CO5	7	02	04	04	10
<b>Grand Total</b>					<b>15</b>	<b>17</b>	<b>38</b>	<b>70</b>

**IX. ASSESSMENT METHODOLOGIES/TOOLS**

Formative assessment (Assessment for Learning)	Summative Assessment (Assessment of Learning)
Lab performance, Assignment, Self-learning, and Seminar/Presentation	Lab. Performance, viva voce

**X. SUGGESTED COS- POS MATRIX FORM**

Course Outcomes (Cos)	Programme Outcomes(Pos)							Programme Specific Outcomes *(PSOs)	
	PO-1 Basic and Discipline-Specific Knowledge	PO-2 Problem Analysis	PO-3 Design/ Development of Solutions	PO-4 Engineering Tools	PO-5 Engineering Practices for Society, Sustainability and Environment	PO-6 Project Management	PO-7 Life Long Learning	PSO-1	PSO-2
CO1	2	2	1	1	-	1	1	1	-
CO2	2	2	2	2	-	1	1	2	-
CO3	2	2		1	-	1	1	1	1
CO4	2	2	2	2	-	1	1	-	2
CO5	2	2	2	2	-	1	1	-	2

**Legends:- High:03, Medium:02, Low:01, No Mapping: -**


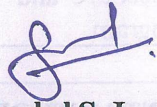


\*PSOs are to be formulated at the institute level

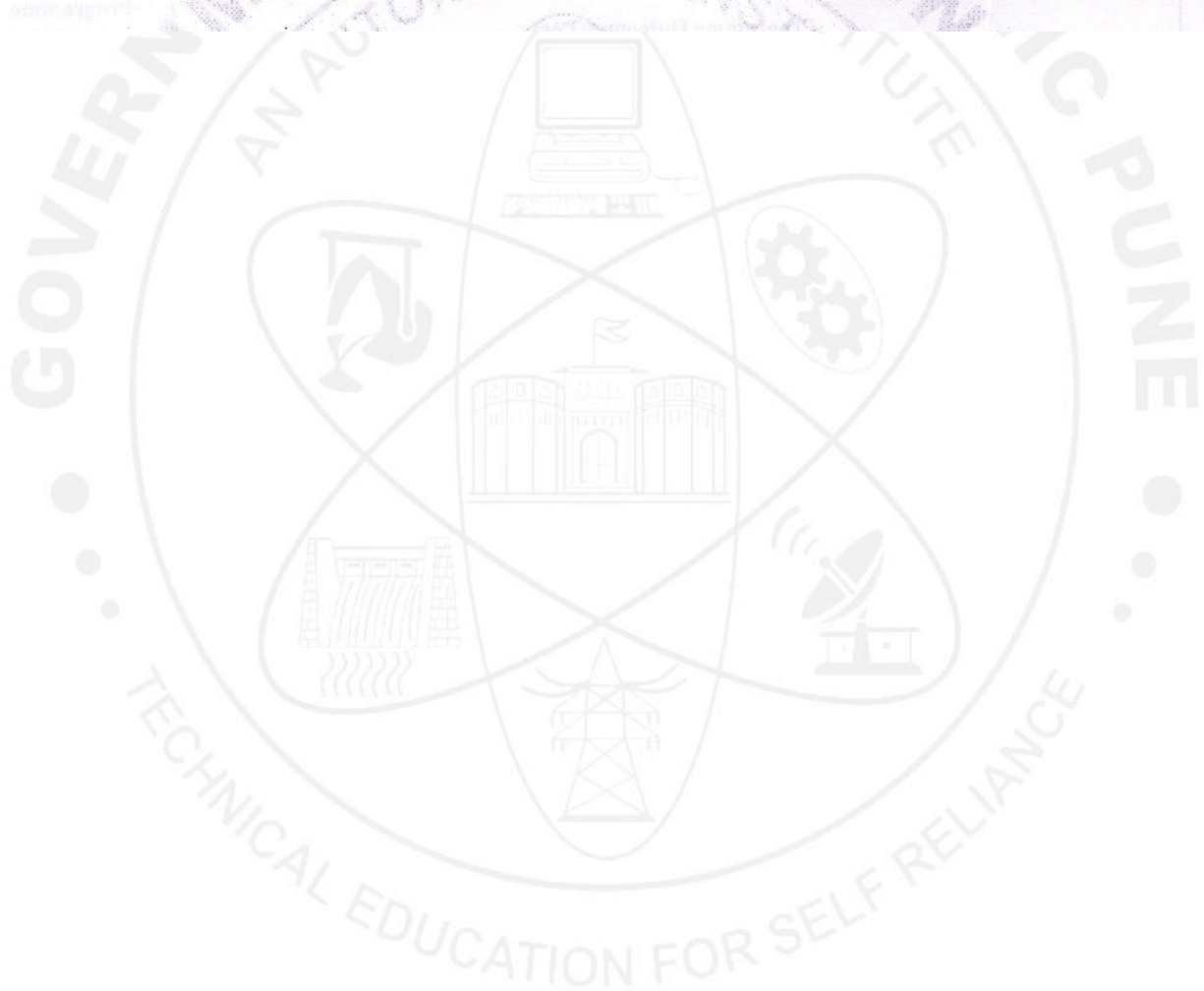
**XI. SUGGESTED LEARNING MATERIALS/BOOKS**

Sr.No	Author	Title	Publisher
1	R P Jain	Modern Digital Electronics	McGraw Hill Education; 4 <sup>th</sup> edition
2	Douglas Hall	Microprocessors and Interfacing: Programming and Hardware, Intel Version	McGraw-Hill Education; 2 <sup>nd</sup> edition
3	Bhurchandi K. M., Roy A. K	Advanced microprocessors and peripherals 3/E	Tata McGraw Hill Education, New Delhi, 2016, ISBN:9781259006135

**XII. LEARNING WEBSITES & PORTALS**

Sr.No	Link / Portal	Description
1	<a href="http://www.nptel.ac.in">http://www.nptel.ac.in</a>	All practicals
2	<a href="https://www.falstad.com/circuit/">https://www.falstad.com/circuit/</a>	All practicals
3	<a href="https://logic.ly/">https://logic.ly/</a>	All practicals

Name & Signature:  <b>Mrs. Archana S. Paik</b> Lecturer in Computer Engineering		Name & Signature:  <b>Mrs. Snehal S. Ingavale</b> Lecturer in Computer Engineering	
<b>(Course Experts)</b>			
Name & Signature:  <b>Smt. J.R. Hange</b> (Programme Head)		Name & Signature:  <b>Shri. S.B. Kulkarni</b> (CDC In-charge)	



## GOVERNMENT POLYTECHNIC, PUNE

'120 – NEP' SCHEME

PROGRAMME	DIPLOMA IN CM/IT
PROGRAMME CODE	06/07
COURSE TITLE	OBJECT ORIENTED PROGRAMMING
COURSE CODE	CM31203
PREREQUISITE COURSE CODE & TITLE	NA
CLASS DECLARATION COURSE	NO

## I. LEARNING &amp; ASSESSMENT SCHEME

Course Code	Course Title	Course Type	Learning Scheme						Credits	Paper Duration	Assessment Scheme										Total Marks
			Actual Contact Hrs./Week			SLH	NLH	Theory			Based on LL & TSL				Based on SL						
			CL	TL	LL						Practical				SLA						
			FA-TH	SA-TH	Total		FA-PR				SA-PR	SLA									
Max	Max	Max	Min	Max	Min	Max	Min	Max	Min												
CM31203	OBJECT ORIENTED PROGRAMMING	SEC	3	--	4	1	8	4	3	30	70	100	40	25	10	25@	10	25	10	175	

Total IKS Hrs for Term: 0 Hrs

**Abbreviations:** CL-Classroom Learning, TL-Tutorial Learning, LL-Laboratory Learning, SLH-Self Learning Hours, NLH-Notional Learning Hours, FA - Formative Assessment, SA -Summative assessment, IKS – Indian Knowledge System, SLA- Self Learning Assessment

**Legends:** @-Internal Assessment, # - External Assessment, \*# - Online Examination, @\$ - Internal Online Examination

**Note:**

FA-TH represents an average of two class tests of 30 marks each conducted during the semester.

- If a candidate is not securing minimum passing marks in FA-PR (Formative Assessment - Practical) of any course, then the candidate shall be declared as '**Detained**' in that course.
- If a candidate does not secure minimum passing marks in SLA (Self Learning Assessment) of any course, then the candidate shall be declared as '**fail**' and will have to repeat and resubmit SLA work.
- Notional learning hours** for the semester are (CL + LL + TL + SL) hrs. \* 15 Weeks
- 1 credit** is equivalent to **30 Notional hours**.
- \* Self-learning hours shall not be reflected in the Timetable.
- \* Self-learning includes micro-projects/assignments/other activities.

## II. RATIONALE:

This course provides students with an introduction to entry-level fundamentals of Object Oriented Programming. The goals of the course are to develop the programming ability of students and to improve their proficiency in applying the fundamentals of Object Oriented Programming. To achieve this goal high-level programming language used is C++. The topics include different programming paradigms in computer programming, limitations of procedural approaches and solutions given by object-oriented programming.

## III. COURSE-LEVEL LEARNING OUTCOMES (CO's)

Students will be able to achieve & demonstrate the following CO's on completion of course-based learning

- CO1 Understand procedural and object-oriented paradigms.
- CO2 Implement different functions in OOP.
- CO3 Develop programs using classes and objects.
- CO4 Implement programs on inheritance.
- CO5 Apply concepts of polymorphism and type conversion.
- CO6 Develop applications for file handling.



## THEORY LEARNING OUTCOMES AND ALIGNED COURSE CONTENT

Sr. No	Theory Learning Outcomes (TLO'S) aligned to CO's.	Learning content mapped with TLO's.	Suggested Learning Pedagogies	Relevant COs
<b>UNIT-I FUNDAMENTALS OF OBJECT ORIENTED PROGRAMMING (CL Hrs-4, Marks-8 )</b>				
1	<p><b>TLO1.1</b> Differentiate between OOP and POP</p> <p><b>TLO1.2</b> Explain the Features of OOP</p> <p><b>TLO1.3</b> Use of control Structures, Arrays, Functions, Structures</p>	<p><b>1.1</b> Different programming paradigms</p> <p><b>1.2</b> Limitations of Procedural Programming and Need of OOP</p> <p><b>1.3</b> Features of OOP</p> <p><b>1.4</b> Beginning with C++: Tokens, Expressions, Control Structures, Array, Functions, Structures</p>	Hands-on Demonstration Presentations	<b>CO1, CO2</b>
<b>UNIT-II FUNCTIONS IN C++ (CL Hrs-6, Marks-12 )</b>				
2	<p><b>TLO2.1</b> Structure of C++ Program</p> <p><b>TLO2.2</b> functions using different function approaches.</p> <p><b>TLO2.3</b> Use of Call by value and Call by reference</p>	<p><b>2.1</b> Functions in C++</p> <p><b>2.2</b> The main function</p> <p><b>2.3</b> Function Prototyping</p> <p><b>2.4</b> Call by Reference, Return by Reference</p> <p><b>2.5</b> Inline Functions</p> <p><b>2.6</b> Default Argument and const Arguments</p>	Hands-on Demonstration Presentations	<b>CO2</b>
<b>UNIT-III CLASSES AND OBJECTS (CL Hrs-10, Marks-14)</b>				
3	<p><b>TLO3.1:</b> Define Class and object</p> <p><b>TLO3.2:</b> Understand memory allocation concepts.</p> <p><b>TLO3.3:</b> Differentiate between constructors and destructors.</p>	<p><b>Classes &amp; Objects</b></p> <p><b>3.1</b> Specifying a class, Defining member functions, Nesting of Member Functions, Private Member Functions</p> <p><b>3.2</b> Creating objects, Memory allocation for objects, Static data and member function, Array of objects and Objects as function arguments</p> <p><b>3.3</b> Constructors and their types, Constructor Overloading, Constructors with Default Arguments, Dynamic Initialization Of Objects</p> <p><b>3.4</b> Destructors</p> <p><b>3.5</b> String Class and objects, manipulating string objects, Relational Operations, string characteristics, accessing characters in strings, Comparing and swapping strings</p>	Hands-on Demonstration Presentations	<b>CO3</b>
<b>UNIT- IV INHERITANCE (CL Hrs-08, Marks-12)</b>				
4	<p><b>TLO4.1:</b> Define inheritance.</p> <p><b>TLO4.2:</b> Explain the need for inheritance.</p> <p><b>TLO4.3:</b> Implement various types of inheritances.</p>	<p><b>4.1</b> Introduction Base Classes, Derived classes Member declaration: Public, Private, protected</p> <p><b>4.2</b> Types Of Inheritance Single, Multilevel, Multiple, Hierarchical, Hybrid</p> <p><b>4.3</b> Virtual base classes</p> <p><b>4.4</b> Abstract classes</p> <p><b>4.5</b> Constructors in derived classes</p>	Hands-on Demonstration Presentations	<b>CO4</b>

UNIT –V POLYMORPHISM AND TYPE CONVERSION (CL Hrs-10, Marks-14)				
5	<p><b>TLO 5.1:</b> Explain the concept of operator overloading.</p> <p><b>TLO 5.2:</b> Understand and implement object-oriented programming language key features like polymorphism.</p> <p><b>TLO 5.3:</b> Describe pointers in C++.</p> <p><b>TLO 5.4:</b> Implement type conversion for various data types.</p>	<p><b>5.1</b> Compile Time Polymorphism Functions overloading Operator Overloading (unary and binary ) Overloading Vs Overriding</p> <p><b>5.2</b> Run Time Polymorphism Pointers in C++, Pointers to Objects ‘This’ Pointer, Pointers to Derived Classes, Virtual functions, Static and dynamic binding</p> <p><b>5.3</b> Type Conversion: Introduction, basic to class type, class to basic type, one class to another type, data conversion example</p>	Hands-on Demonstration Presentations	CO5
UNIT –VI FILES AND EXCEPTION HANDLING (CL Hrs-7, Marks-10)				
6.	<p><b>TLO 6.1:</b> Define files in C++.</p> <p><b>TLO 6.2:</b> Implement various operations that can be performed on files.</p> <p><b>TLO 6.3:</b> 6c. Execute a program to handle exceptions in the programs</p>	<p>Files:</p> <p><b>6.1.</b> C++ Streams and Stream Classes</p> <p><b>6.2.</b> Unformatted IO Operations</p> <p><b>6.3.</b> File Stream Classes</p> <p><b>6.4.</b> Opening and Closing a File</p> <p><b>6.5.</b> Deleting a File</p> <p><b>6.6.</b> File Modes</p> <p>Exception Handling:</p> <p><b>6.7.</b> Introduction, basics of exception handling, types of exceptions,</p> <p><b>6.8.</b> Structure to handle an exception, exception handling mechanism</p> <p><b>6.9.</b> Throwing mechanism, catching mechanism, re-throwing an exception, specifying exceptions.</p>	Hands-on Demonstration Presentations	CO6

#### IV. LABORATORY LEARNING OUTCOME AND ALIGNED PRACTICAL/ TUTORIAL EXPERIENCES

Sr. No	Practical/Tutorial/Laboratory Learning Outcome (LLO)	Laboratory Experiment/ Practical Titles /Tutorial Titles	Number of hrs.	Relevant COs
1	<p><b>LLO 1.1:</b> Write Simple C program using constant and variables.</p> <p><b>LLO1.2:</b> Use of different operators.</p> <p><b>LLO1.3:</b> Use the various expressions in the C Program.</p>	*Write a Program using Input and Output Statements.	2	CO1
2	<p><b>LLO 2.1:</b> Write a C program based on arrays and structure.</p> <p><b>LLO 2.1:</b> Write a C program using an array of Structure.</p>	*Write a Program using structure and array of structure.	2	CO1
3	<b>LLO 3.1:</b> Write user defined functions in C++.	*Write a Program using call by value.	2	CO2
4	<b>LLO 4.1:</b> Write user defined functions in C++.	*Write a Program using call by reference.	2	CO2

5	<b>LLO 5.1:</b> Understand the concept of polymorphism. <b>LLO 5.2:</b> Write a programs to implement the concept of function overloading.	*Write a Program using Function Overloading.	2	CO3
6	<b>LLO 6.1:</b> Define Class and create objects. <b>LLO 6.2:</b> Write a program using class and objects.	*Write a Program using Class and Objects.	4	CO3
7	<b>LLO 7.1:</b> Write a program using class and array of objects.	Write a Program using array of Objects.	2	CO3
8	<b>LLO 8.1:</b> Implement the concept of object as function argument.	*Write a Program using object as function argument.	2	CO3
9	<b>LLO 9.1:</b> Use of static data members and member functions.	*Write a Program using static members. (variables and functions)	2	CO3
10	<b>LLO 10.1:</b> Write a Program using friend functions.	*Write a Program using Friend Function.	2	CO3
11	<b>LLO 11.1:</b> Use of constructor to initialize objects. <b>LLO11.2:</b> write a Program using constructors and destructors.	*Write a Program using Constructor and Destructor.	4	CO3
12	<b>LLO 12.1:</b> Apply the logic to implement different types of constructor in single program.	Write a Program using Constructor Overloading.	2	CO3
13	<b>LLO 13.1:</b> Understand various predefined string functions. <b>LLO 13.2:</b> Implement program using predefined string functions.	Write a program to perform following string operations using pre-defined string functions:- a) String concatenation b) String Comparison c) Find position of an character in a given string d) String reversing	4	CO3
14	<b>LLO 14.1:</b> Understand various predefined string functions. <b>LLO 14.2:</b> Implement program using predefined string functions.	*Write a program to perform following string operations without using pre-defined string functions :- a) String concatenation b) String Comparison c) Find position of an character in a given string d) String reversing	4	CO3
15	<b>LLO 15.1:</b> Understand the concept of Inheritance. <b>LLO 15.2:</b> Implement single inheritance.	*Write a Program using single Inheritance.	2	CO4
16	<b>LLO 16.1:</b> Understand the concept of Inheritance. <b>LLO 16.2:</b> Implement multilevel inheritance.	*Write a Program using multilevel Inheritance.	2	CO4

17	<b>LLO 17.1:</b> Understand the concept of Inheritance. <b>LLO17.2:</b> Implement multiple inheritance.	*Write a Program using Multiple Inheritance.	2	CO4
18	<b>LLO 18.1:</b> Understand the concept of diamond problem. <b>LLO17.2:</b> Implement hybrid inheritance.	*Write a Program using Virtual Base Class.	2	CO4
19	<b>LLO 19.1:</b> Understand the concept of polymorphism. <b>LLO 19.2:</b> Write a programs to implement the concept of operator overloading.	*Write a Program for Operator Overloading. (Unary and Binary operator)	4	CO5
20	<b>LLO 20.1:</b> Understand the concept of polymorphism. <b>LLO 20.2:</b> Write a programs to implement the concept of operator overloading using friend function.	Write a Program for Operator Overloading using friend function. (Unary and Binary operator)	4	CO5
21	<b>LLO 21.1:</b> Understand the concept of Pointer. <b>LLO 21.2:</b> Implement this pointer.	*Write a Program using 'this' Pointer.	2	CO5
22	<b>LLO 22.1:</b> Understand the concept of function Overriding. <b>LLO 22.2:</b> Implement virtual functions.	*Write a Program using Virtual Function.	2	CO5
23	<b>LLO 23.1:</b> Understand conversion of basic to class type, class type to basic type, class type to class type.	*Write a program to implement type conversion concept.	2	CO5
24	<b>LLO 24.1:</b> Understand the concept of file processing. <b>LLO 24.2:</b> Implement various file operations. <b>LLO 24.3:</b> Write a program for exception handling.	*Write a Program For File Processing.	2	CO6

**Note: Out of the above suggestive LLOs –**

1. '\*' Marked Practicals (LLOs) Are mandatory.
2. A judicious mix of LLOs is to be performed to achieve the desired outcomes

#### V. SUGGESTED MICRO PROJECT/ASSIGNMENT/CASE STUDIES /ACTIVITIES FOR SPECIFIC LEARNING/SKILLS

##### Self Learning Assessment- Yes

Suggestive list of Case studies for SLA:

##### 1. Expense Tracker and Savings Calculator

- i. A boy gets the same amount of pocket money every month. I use the pocket money for bought some college necessities and snacks. However, he also wanted to save some of the remaining money pocket that he has, although not necessarily every month. He asked his friend who studied Informatics to make an application to calculate it all by displaying monthly expenses,



total expenses, and current savings this.

ii. Notes:

- Display data iteratively
- Using Class

## 2. Gas Station Price Calculation System

- i. At a gas station, you want to make a program that can calculate the total price which will be issued for two types of gasoline, namely ABC and XYZ. ABC has a price of Rs. 7,000/liter, while XYZ is Rs. 9,000/liter.
- Output from the program is expected to be in the form of a receipt that has buyer details
- i. (Total liters, type gasoline, customer name, total price)

## 3. Restaurant Menu and Delivery System

- i. Develop a system where a restaurant has a menu and provides delivery order services for customers whose homes are more than 4 KM will be charged a delivery fees of Rs.500, if it is less than that distance, it will be charged delivery fees of Rs.100. If the total purchase is more than Rs.4000 will get a discounted fee Rs.400. If the total purchase exceeds Rs.6000, discount will be given 25%.
- ii. -The output that comes out is expected in the form of a payment slip.

**4. Library Management System:** Develop a program to handle basic banking operations such as account creation, deposits, withdrawals, and balance inquiries.

**5. Bank Management System:** Develop a system for creating and managing customers, accounts and transactions as well as performing banking services such as withdrawals, deposits, and transfers. It also allows customer to view their account information including balances, recent transaction.

**6. Student Management System:** Create a system to manage student information, including adding new students, updating records, and generating reports.

**7. Hospital Management System:** Design a program to manage patient information, doctor schedules, and appointment bookings.

**8. Inventory Management System:** Implement a system to track information about products, including their quantity, price and other details and generate reports.

**9. Hotel Booking System:** Create a program to handle hotel room bookings, cancellations, and availability checks, make different packages including activities for kids and adults, apply discount charges on activities and generate final bill report.

**10. Payroll System:** Develop a system to manage employee records, tracking hours worked and calculating tax and generate reports.

**11. Online Quiz System:** Design an application to conduct online quizzes, store results, and provide instant feedback.

**12. Car Rental System:** Implement a program to manage car rentals, including booking, returning cars, and calculating rental fees.

**13. Bus Reservation System:** Create a system to manage bus reservations, including seat selection, booking, and cancellations.

**14. Shopping Cart System:** Develop an e-commerce shopping cart system that allows users to add items to their cart, view the cart, and proceed to checkout.

**15. Ticket Booking System:** Implement a system for booking tickets for events such as movies, concerts, or travel.

**16. Flight Reservation System:** Develop a program to handle flight bookings, cancellations, and check-ins.

**17. Medical Information System:** Implement a system to store and organize patient medical information such as clinical data, lab results.

**18. Tic Tac Toe game:** Design a game for layers take turns putting their marks in empty squares. The first player to get 3 of her marks in a row (up, down, across, or diagonally) is the winner.

- 19. E-Learning Management System:** The University of ABC, a large public institution, faced challenges related to providing quality education to a diverse student population spread across different geographical locations. Traditional classroom-based teaching methods struggled to accommodate the needs of working professionals and students with varied schedules. The primary objectives included providing tools for course creation, student enrollment, assessments, and progress tracking to facilitate a seamless transition to online education.
- 20. Unit Converter:** Design a system to convert different physical quantities like Mass, Length, Area, Temperature, Time, Currency etc. to be converted one unit into another.
- 21. Supermarket Billing System:** Design a system in a place where customer come to purchase their daily products and pay for that. So, there is need to calculate how many products are sold and generate the bill for the customer. The system will be able to generate the bill, Store how many products are sold, store products and their prices with other information, and see the rates of discounts on the products.
- 22. Food Bank Management System:** Develop a system to manage and organize data including adding donors, receivers, and staff profiles, update the pantry capacity, and generate report.
- 23. Error detection and correction code System:** Design a system for different error detection and correction methods.
- 24. Calculator for Scientific Operations:** Design a code to perform different scientific calculations.
- 25. Book Donation System:** Develop a system to manage and organize data including adding details of donors and details of book, receivers, and staff profiles, update the book capacity, and generate report.

**Note:**

1. The above is suggestive list of case studies for SLA
2. The faculty must allocate any 1 Case study to individual student. Considering the students technical skills.

**Activities**

- Students are encouraged to use online tools to improve their learning, such as the e-Kumbh from AICTE and the virtual Labs from IIT.
- Students should be encouraged to participate in various coding competitions, such as hackathons, and online coding contests on websites like Hackerrank, Codechef etc.
- At the department level, encourage students to start a coding club
- Students are encouraged to register themselves in various MOOCs such as Infosys Springboard, Swayam etc. to further enhance their learning.

**Assignment**

Prepare a journal of practicals performed in the laboratory.

**VI. LABORATORY EQUIPMENT/INSTRUMENTS/TOOLS/SOFTWARE REQUIRED**

Sr. No	Equipment Name with Broad Specifications	Relevant LLO Number
1	Basic configuration systems with editor supporting C++ language program execution.	ALL

**VII. SUGGESTED FOR WEIGHTAGE TO LEARNING EFFORTS & ASSESSMENT PURPOSE  
(Specification Table)**

Sr. No	Unit	Unit Title	Aligned COs	Learning Hours	R-Level	U-Level	A-Level	Total Marks
1	I	Fundamentals Of Object Oriented Programming	CO1	4	-	4	4	8
2	II	Functions In C++	CO2	6	4	4	4	12
3	III	Classes And Objects	CO3	10	4	2	8	14
4	IV	Inheritance	CO4	8	6	4	2	12
5	V	Polymorphism And Type Conversion	CO5	10	4	4	6	14
6	VI	Files And Exception Handling	CO6	7	2	4	4	10
<b>Grand Total</b>				<b>45</b>	<b>20</b>	<b>22</b>	<b>28</b>	<b>70</b>

**VIII. ASSESSMENT METHODOLOGIES/TOOLS**

Formative assessment (Assessment for Learning)	Summative Assessment (Assessment of Learning)
Lab performance, Assignment, Self-learning and Seminar/Presentation	Lab. Performance, viva voce

**IX. SUGGESTED COS- POS MATRIX FORM**

Course Outcomes (Cos)	Programme Outcomes(Pos)							Programme Specific Outcomes*(PS Os)	
	PO-1 Basic and Discipline-Specific Knowledge	PO-2 Problem Analysis	PO-3 Design/ Development of Solutions	PO-4 Engineering Tools	PO-5 Engineering Practices for Society, Sustainability and Environment	PO-6 Project Management	PO-7 Life Long Learning	PSO-1	PSO-2
CO1	3	2	1	2	-	2	3	-	1
CO2	3	2	3	3	2	2	3	-	2
CO3	3	2	3	3	2	-	3	-	3
CO4	3	2	3	3	1	2	3	-	3
CO5	3	2	3	3	1	2	3	-	3
CO6	3	2	3	3	3	2	3	-	3

**Legends:- High:03, Medium:02, Low:01, No Mapping: -**

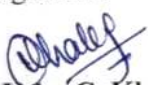




\*PSOs are to be formulated at the institute level

**X. SUGGESTED LEARNING MATERIALS/BOOKS**

Sr.No	Author	Title	Publisher
1	E. Balagurusamy	Object Oriented Programming with C++	McGraw Hill Education (India) Private Limited, New Delhi
2	Herbert Schildt	C++ The Complete Reference, 4th Edition	McGraw Hill/ Oshome, New Delhi
3	Yashwant P. Kanetkar	Let Us C++, 2nd Edition	BPB Publication

**XI. LEARNING WEBSITES & PORTALS**

1. [www.nptel.com](http://www.nptel.com)
2. <https://www.quora.com>
3. <https://www.softwaretestinghelp.com>
4. <https://www.cplusplus.com>
5. <https://www.learncpp.com>

Name & Signature:		
 <b>Mrs. Usha C. Khake</b> Lecturer in Computer Engineering	 <b>Mrs. Lalita S. Korde</b> Lecturer in Computer Engineering <b>(Course Experts)</b>	 <b>Mrs. Heena F. Khan</b> Lecturer in Information Technology
Name & Signature:		Name & Signature:
 <b>Smt. J.R. Hange</b> (Programme Head)		 <b>Mr. S. B. Kulkarni</b> (CDC In-charge)





**GOVERNMENT POLYTECHNIC, PUNE**  
**'120 – NEP' SCHEME**

<b>PROGRAMME</b>	<b>DIPLOMA IN CM/IT</b>
<b>PROGRAMME CODE</b>	<b>06/07</b>
<b>COURSE TITLE</b>	<b>RELATIONAL DATABASE MANAGEMENT SYSTEM</b>
<b>COURSE CODE</b>	<b>CM41201</b>
<b>PREREQUISITE COURSE CODE &amp; TITLE</b>	<b>NA</b>
<b>CLASS DECLARATION COURSE</b>	<b>NO</b>

**I. LEARNING & ASSESSMENT SCHEME**

Course Code	Course Title	Course Type	Learning Scheme						Credits	Paper Duration	Assessment Scheme										Total Marks
			Actual Contact Hrs./Week			SLH	NLH	Theory			Based on LL & TSL				Based on SL						
			CL	TL	LL			FA-TH			SA-TH	Total	FA-PR		SA-PR		SLA				
			Max	Max	Max	Min	Max						Min	Max	Min	Max		Min			
CM41201	RELATIONAL DATABASE MANAGEMENT SYSTEM	DSC	3	-	4	1	8	4	3Hrs	30	70	100	40	25	10	25@	10	25	10	175	

Total IKS Hrs for Term: 0 Hrs

**Abbreviations:** CL-Classroom Learning, TL-Tutorial Learning, LL-Laboratory Learning, SLH-Self Learning Hours, NLH-Notional Learning Hours, FA - Formative Assessment, SA -Summative assessment, IKS – Indian Knowledge System, SLA- Self Learning Assessment

**Legends:** @-Internal Assessment, #- External Assessment,\*# - Online Examination,@\$ - Internal Online Examination  
**Note:**

**FA-TH** represents an average of two class tests of 30 marks each conducted during the semester.

1. If a candidate is not securing minimum passing marks in **FA-PR** (Formative Assessment - Practical) of any course, then the candidate shall be declared as '**Detained**' in that course.
2. If a candidate does not secure minimum passing marks in SLA (Self Learning Assessment) of any course, then the candidate shall be declared as '**fail**' and will have to repeat and resubmit SLA work.
3. **Notional learning hours** for the semester are **(CL + LL + TL + SL) hrs. \* 15 Weeks**
4. **1 credit** is equivalent to **30 Notional hours**.
5. \* Self-learning hours shall not be reflected in the Timetable.
- 6.\* Self-learning includes micro-projects/assignments/other activities.

**II. RATIONALE:**

The objectives of this course are to provide a strong formal foundation in Database Concepts, technology and practice to the students to enhance them into well-informed application developers. After learning this subject, the students will be able to understand the designing of RDBMS and can use any RDBMS package as a backend for developing database applications

**III. COURSE-LEVEL LEARNING OUTCOMES (CO's)**

Students will be able to achieve & demonstrate the following CO's on completion of course-based learning

- CO1: Understand Database Management System concepts
- CO2: Design a database for a given problem
- CO3: Execute SQL commands on the database
- CO4: Use performance-tuning objects in SQL
- CO5: Implement PL/SQL code on a given database
- CO6: Apply security and backup techniques on the database

**IV. THEORY LEARNING OUTCOMES AND ALIGNED COURSE CONTENT:**

Sr. No	Theory Learning Outcomes (TLO's) aligned to CO's.	Learning content mapped with TLO's.	Suggested Learning Pedagogies	Relevant COs
<b>UNIT-I INTRODUCTION TO DATABASE SYSTEM (CL Hrs-7, Marks-12 )</b>				
1.	<p><b>TLO1.1:</b> State the importance of a database management system.</p> <p><b>TLO1.2:</b> Define data, database, DBMS, data independence, data abstraction, and schema.</p> <p><b>TLO1.2.1:</b> State Codd's laws.</p> <p><b>TLO1.2.2:</b> Describe the Overall structure of DBMS.</p> <p><b>TLO1.3:</b> Describe the architecture of DBMS.</p> <p><b>TLO1.4:</b> Distinguish Hierarchical, networking and relational data model.</p> <p><b>TLO1.5:</b> Describe advanced database concepts</p>	<p><b>1.1 Database concepts:</b> Data, Database, Database management system, File system versus DBMS, Applications of DBMS, Data Abstraction, Data Independence, Database Schema</p> <p><b>1.2 The Codd's rules,</b> the Overall structure of DBMS</p> <p><b>1.3 Architecture:</b> Two-tier and Three-tier architecture of DBMS.</p> <p><b>1.4 Data Models:</b> Hierarchical, Networking, and Relational Data Models</p> <p><b>1.5 Introduction to advanced database concepts:</b> Data mining, Data Warehousing, Big data</p>	<p>Hands-on Demonstration Presentations</p>	<p><b>CO1</b></p>
<b>UNIT-II 2 RELATIONAL DATA MODEL (CL Hrs-6, Marks-10 )</b>				
2.	<p><b>TLO2.1:</b> Define table, row, column, domain, attribute</p> <p><b>TLO2.2:</b> State types of keys and give examples of each.</p> <p><b>TLO2.3:</b> Describe data constraints.</p> <p><b>TLO2.4:</b> Draw an ER diagrams</p> <p><b>TLO2.5:</b> Describe database design in terms of 1NF, 2NF and 3NF</p>	<p><b>2.1 Relational Structure-</b> Tables (Relations), Rows(Tuples), Domains, attributes</p> <p><b>2.2 Keys:</b> Super Keys, Candidate Key, Primary Key, Foreign Key</p> <p><b>2.3 Data Constraints:</b> Not Null, Unique, Primary Key, Foreign Key, Check, Default.</p> <p><b>2.4 Entity Relationship Model,-</b> Strong Entity set, Weak Entity set, Types of Attributes, E-R Diagrams</p> <p><b>2.5 Normalization -</b>Normalization based on functional dependencies, Normal forms: 1NF, 2NF, 3NF</p>	<p>Hands-on Demonstration Presentations</p>	<p><b>CO2</b></p>

<b>UNIT-III INTERACTIVE SQL (CL Hrs-12, Marks-14)</b>				
3.	<p><b>TLO3.1.1:</b> Enlist Oracle data types.</p> <p><b>TLO3.1.2:</b> Compare DDL, DML, DCL and TCL.</p> <p><b>TLO3.1.3:</b> Write SQL queries on DDL, DML, DCL and TCL.</p> <p><b>TLO3.2:</b> Describe and write SQL queries on GROUP BY, ORDER BY, and HAVING clauses</p> <p><b>TLO3.3.1:</b> Enlist operators and compare between Relational, Arithmetic, Logical, and set operators.</p> <p><b>TLO3.3.2:</b> Write SQL queries to evaluate the use of operators.</p> <p><b>TLO3.4.1:</b> Enlist functions and compare Date, time, String functions and Aggregate Functions.</p> <p><b>TLO3.4.2:</b> Write SQL queries to evaluate the use of functions.</p> <p><b>TLO3.5:</b> Describe INNER and OUTER JOINS and Write SQL queries to evaluate the use of Join</p>	<p><b>3.1 SQL:</b> Invoking SQL*PLUS, The Oracle Data- types, Data Definition Language (DDL), Data Manipulation Language (DML), Data Control Language (DCL), Transaction control language (TCL).</p> <p><b>3.2 Clauses:</b> Different types of clauses in SQL</p> <p><b>3.3 Operators:</b> Relational, Arithmetic, Logical, set operators.</p> <p><b>3.4 Functions:</b> Date and time, String functions, Aggregate Functions.</p> <p><b>3.5 Joins:</b> Types of Joins, Nested queries</p>	Hands-on Demonstration Presentations	CO3
<b>UNIT- IV ADVANCED SQL: PERFORMANCE TUNING (CL Hrs-04, Marks-10)</b>				
4.	<p><b>TLO4.1.1:</b> Define view, sequence and index.</p> <p><b>TLO4.1.2:</b> Describe the view with its types.</p> <p><b>TLO4.1.3:</b> Write SQL queries to create a view and perform different operations on it.</p> <p><b>TLO4.2:</b> Write SQL queries to create a sequence and perform different operations on it.</p> <p><b>TLO4.3.1:</b> Describe types of indexes.</p> <p><b>TLO4.3.2:</b> Write SQL queries to create an index and perform different operations on it.</p>	<p><b>4.1 Creating Views, Views:</b> Types of Views: Read Only View and Updatable Views, Dropping Views.</p> <p><b>4.2 Sequences:</b> Creating Sequences, Altering Sequences, Dropping Sequences</p> <p><b>4.3 Indexes:</b> Index Types, Creating of an Index: Simple Unique, and Composite Index, Dropping Indexes.</p>	Hands-on Demonstration Presentations	CO4
<b>UNIT -V PL/SQL PROGRAMMING (CL Hrs-12, Marks-14)</b>				
5.	<p><b>TLO5.1:</b> Enlist PL/SQL data types and State the advantages of PL/SQL.</p> <p><b>TLO5.2.1:</b> Describe the control</p>	<p><b>5.1 Introduction of PL/SQL:</b> The PL/SQL Syntax, The PL/SQL Block Structure, PL/SQL data types, and Advantages of PLSQL.</p>	Hands-on Demonstration Presentations	CO5

	<p>structure with its types.  <b>TLO5.2.2:</b> Write PL/SQL block to evaluate the use of different control structures.  <b>TLO5.3.1:</b> Describe exception handling with its types.  <b>TLO5.3.2:</b> Write PL/SQL block to create different types of Exception.  <b>TLO5.4.1:</b> Describe the working of cursors.  <b>TLO5.4.2:</b> Distinguish between Implicit and Explicit cursors.  <b>TLO5.4.3:</b> Write PL/SQL block to create different types of cursors.  <b>TLO5.5:</b> Define Procedure, Function Trigger and State advantages.  <b>TLO5.6:</b> Write PL/SQL block to create stored procedures and function  <b>TLO5.7:</b> Describe the working of triggers</p>	<p><b>5.2</b> Control Structure: Conditional Control, Iterative Control, Sequential Control.  <b>5.3</b> Exception handling: Predefined Exception, User defined Exception.  <b>5.4</b> Cursors: Implicit and Explicit Cursor  <b>5.5</b> Procedures: Advantages, Creating, Executing and Deleting a Stored Procedure  <b>5.6</b> Functions: Advantages, Creating, Executing and Deleting a Function.  <b>5.7</b> Database Triggers: Use of Database Triggers, Types of Triggers, Syntax for Creating Triggers, Deleting Trigger.</p>		
<b>UNIT –VI NoSQL AND DATABASE ADMINISTRATION OVERVIEW (CL Hrs-04, Marks-10)</b>				
6.	<p><b>TLO6.1.1:</b> Compare SQL with NoSQL  <b>TLO6.1.2:</b> Enlist Benefits of NoSQL  <b>TLO6.2:</b> Write basic NoSQL queries with MongoDB  <b>TLO6.3.1:</b> Explain types of failure and its types  <b>TLO6.3.2:</b> Describe the procedure to take database backup  <b>TLO6.4.:</b> Describe Database Recovery and its types</p>	<p><b>6.1</b> Introduction to NoSQL-comparison between SQL and NoSQL database system, Benefits of NoSQL, Types of NoSQL databases  <b>6.2</b> MongoDB: Installation of MongoDB, Querying with MangoDB  <b>6.3</b> Database Backup-Types of failure, Causes of failure and database backup introduction, Types of database Backup-Physical and logical  <b>6.4</b> Database Recovery-Recovery Concepts, Recovery Techniques- Roll forward and Rollback</p>	Hands-on Demonstration Presentations	CO6

**V. LABORATORY LEARNING OUTCOME AND ALIGNED PRACTICAL/ TUTORIAL EXPERIENCES.**

Sr. No	Practical/Tutorial/ Laboratory Learning Outcome (LLO)	Laboratory Experiment/ Practical Titles /Tutorial Titles	Relevant COs	Number of hrs
1.	<b>LLO 1.1:</b> Create a database schema for a given application	*Draw an ER diagram for a given database.	1	2
2.	<b>LLO 2.1:</b> Execute queries using DDL commands.	*Applying Constraints on relation.	2	2
3.	<b>LLO 3.1:</b> Execute queries using DDL commands.	*Create and execute queries using DDL commands.	3	4



Sr. No	Practical/Tutorial/ Laboratory Learning Outcome (LLO)	Laboratory Experiment/ Practical Titles /Tutorial Titles	Relevant COs	Number of hrs
4.	<b>LLO 4.1:</b> Execute queries using DML commands.	*Create and execute queries using DML commands	3	4
5.	<b>LLO 5.1:</b> Execute queries using DCL commands.	*Create and execute queries using DCL and TCL commands.	3	4
6.	<b>LLO 6.1:</b> Implement queries using causes	*Write Queries using different types of clauses.	3	2
7.	<b>LLO 7.1:</b> Implement queries using different operators	*Write Queries using various types of operators like (Set, Relational, Arithmetic and Logical)	3	4
8.	<b>LLO 8.1:</b> Implement queries using different functions in SQL	*Write Queries using various Functions like (Date, Time, String, and Aggregate).	3	4
9.	<b>LLO 9.1:</b> Execute queries based on inner-outer joins	*Write Queries using different types of Joins.	3	2
10.	<b>LLO 10.1:</b> Implement queries using Views	*Write Queries to Create, Insert, Update and Drop View	4	2
11.	<b>LLO 11.1:</b> Implement queries using Sequence	*Write Queries to Create, Alter and Drop Sequence	4	2
12.	<b>LLO 12.1:</b> Write queries for Index	*Write Queries to Create Simple and composite Indexes and Drop them.	4	4
13.	<b>LLO 13.1:</b> Implement PL/SQL program using Conditional Statement	*Write the PL/SQL Program using 1. IF...THEN...ELSE 2, NESTED IF..THEN ELSE 3. IF THEN ELSEIF	5	2
14.	<b>LLO 14.1:</b> Implement PL/SQL program using Iterative Statement	*Write the PL/SQL Program using 1. FOR LOOP 2. REVERSE FOR LOOP	5	2
15.	<b>LLO 15.1:</b> Implement PL/SQL program using Iterative WHILE Statement	*Write the PL/SQL Program using WHILE LOOP	5	2
16.	<b>LLO 16.1:</b> Implement PL/SQL program using Sequential Statement	*Write the PL/SQL Program using 1. GOTO 2. EXIT 3. CONTINUE	5	2
17.	<b>LLO 17.1:</b> Implement PL/SQL program based on Pre-define Exception	*Write the PL/SQL Program using pre-defined	5	2
18.	<b>LLO 18.1:</b> Implement PL/SQL program based on User defined Exception	*Write the PL/SQL Program using user-defined Exceptions	5	2
19.	<b>LLO 19.1:</b> Create implicit and explicit cursor	*Write the PL/SQL Program to implement Implicit and Explicit Cursor	5	2
20.	<b>LLO 20.1:</b> Implement PL/SQL programs using Procedure	*Write the PL/SQL Program to implement the Stored Procedure	5	2
21.	<b>LLO 21.1:</b> Implement PL/SQL programs using Function	*Write the PL/SQL Program to implement the Function	5	2

Sr. No	Practical/Tutorial/ Laboratory Learning Outcome (LLO)	Laboratory Experiment/ Practical Titles /Tutorial Titles	Relevant COs	Number of hrs
22.	<b>LLO 22.1:</b> Implement trigger for a given database	*Write the PL/SQL Program for Creating Trigger, Deleting Trigger	5	2
23.	<b>LLO 23.1:</b> Install MongoDB	*Installing NoSQL database: MongoDB	6	2
24.	<b>LLO 24.1:</b> Implement basic NoSQL queries on MongoDB	*Perform Basic queries-Create, Insert, Update, and Delete commands on MongoDB	6	2
			<b>Total Hrs</b>	<b>60</b>

**Note: Out of the above suggestive LLOs –**

1. All Practicals (LLOs) are mandatory.
2. A judicious mix of LLOs is to be performed to achieve the desired outcomes

#### **VI. SUGGESTED MICRO PROJECT/ASSIGNMENT/ACTIVITIES FOR SPECIFIC LEARNING/ SKILLS DEVELOPMENT (SELF-LEARNING)**

##### **Self-Learning**

Draw ER Diagram and design database with the help of DDL, DML, DCL, TCL, Index, Sequence, View, PL/SQL, Procedure, Function, Trigger concepts.

Following are some suggestive topics for Self-learning:

1. Library Management System:
2. Student Management System
3. Employee Management system
4. Product Inventory System
5. Hotel Management System
6. Bus reservation Management System
7. Travel agency Management System
8. Bank Management System
9. Airline Management System
10. Bloodbank Management System
11. Hospital Management
12. Payroll Management
13. Hostel Management
14. Movie Ticket Reservation system
15. Electricity Bill Management System
16. Insurance Management System
17. ATM Management System
18. Patient health record
19. Online bookstore management
20. Car rental Management System
21. Student Grade database
22. Food Delivery Order Management System
23. Charity Donation Management
24. Online Exam Management System
25. Train Reservation Management System

**Activities**

- Students are encouraged to use online tools to improve their learning, such as the e-Kumbh from AICTE and the virtual Labs from IIT.
- Students should be encouraged to participate in various coding competitions, such as hackathons, and online coding contests on websites like Hackerrank, Codechef etc.
- The department level, encourage students to start a coding club
- Students are encouraged to register themselves in various MOOCs such as Infosys Springboard, Swayam etc. to further enhance their learning.

**Note:**

1. The above is suggestive list of topics for SLA
2. The faculty must allocate any 1 topic to individual student. Considering the students technical skills.

**Assignment**

Prepare a journal of practicals performed in the laboratory.

**VII. LABORATORY EQUIPMENT/INSTRUMENTS/TOOLS/SOFTWARE REQUIRED**

Sr.No	Equipment Name with Broad Specifications	Relevant LLO Number
1	a) Computer System with all necessary Peripherals and Internet connectivity. b) SQL/Oracle software c) MangoDB software	ALL

**VIII. SUGGESTED FOR WEIGHTAGE TO LEARNING EFFORTS & ASSESSMENT PURPOSE (Specification Table)**

Sr. No	Unit	Unit Title	Aligned COs	Learning Hours	R-Level	U-Level	A-Level	Total Marks
1	I	Introduction to Database System	CO1	7	06	06	00	12
2	II	Relational Data Model	CO2	6	02	04	04	10
3	III	Interactive SQL	CO3	12	04	04	06	14
4	IV	Advanced SQL: Performance Tuning	CO4	4	02	04	04	10
5	V	PL/SQL Programming	CO5	12	04	04	06	14
6	VI	NoSQL and Database Administration Overview	CO6	4	02	02	06	10
<b>Grand Total</b>				<b>45</b>	<b>20</b>	<b>24</b>	<b>26</b>	<b>70</b>

**IX. ASSESSMENT METHODOLOGIES/TOOLS**

Formative assessment (Assessment for Learning)	Summative Assessment (Assessment of Learning)
Lab performance, Assignment, Self-learning and Seminar/Presentation	Lab. Performance, viva voce

## X. SUGGESTED COS- POS MATRIX FORM

Course Outcomes (COs)	Programme Outcomes(POs)							Programme Specific Outcomes *(PSOs)	
	PO-1 Basic and Discipline-Specific Knowledge	PO-2 Problem Analysis	PO-3 Design/ Development of Solutions	PO-4 Engineering Tools	PO-5 Engineering Practices for Society, Sustainability and Environment	PO-6 Project Management	PO-7 Life Long Learning	PSO-1	PSO-2
CO1	3	2	3	3	3	3	3	-	2
CO2	3	2	3	2	2	2	3	-	3
CO3	2	3	3	3	2	2	3	-	3
CO4	2	3	3	3	2	3	3	-	3
CO5	2	3	3	3	2	3	3	-	3
CO6	3	2	2	2	2	3	3	-	2

**Legends: - High:03, Medium:02, Low:01, No Mapping: -**  
\*PSOs are to be formulated at the institute level

## XI. SUGGESTED LEARNING MATERIALS/BOOKS SUGGESTED LEARNING MATERIALS/BOOKS

Sr.No.	Author	Title	Publisher
1	Abraham Silberschtz, Henry Korth and S.Sudharshan	Introduction to Database System	Tata McGraw Hill, 3rd edition,
2	Ivan Bayross	PLSQL	BPB Publication, 3rd edition SQL,
3	Kogent Learning Solutions Inc	Database Management Systems Application	Dreamtech Press 2014

## XII. LEARNING WEBSITES &amp; PORTALS

Sr.No	Link / Portal	Description
1	<a href="https://nptel.ac.in/courses/106105175">https://nptel.ac.in/courses/106105175</a>	All practicals
2	<a href="https://www.w3schools.com/sql">https://www.w3schools.com/sql</a>	All practicals
3	<a href="https://www.tutorialspoint.com/sql">https://www.tutorialspoint.com/sql</a>	All practicals
4	<a href="https://www.studytonight.com/dbms">https://www.studytonight.com/dbms</a>	All practicals
5	<a href="https://docs.mongodb.com/manual/tutorial/install-mongodb-on-windows">https://docs.mongodb.com/manual/tutorial/install-mongodb-on-windows</a>	All practicals

Name &amp; Signature:

Smt. Jyoti P. Dandale

Lecturer in Computer Engineering

Smt. Sonali B. Gosavi

Lecturer in Computer Engineering

Smt. Priyanka. L. Sonawane

Lecturer in Information Technology

(Course Experts)

Name &amp; Signature:

Smt. J.R. Hange  
(Programme Head)

Name &amp; Signature:

Shri. S.B. Kulkarni  
(CDC In-charge)



## GOVERNMENT POLYTECHNIC, PUNE

'120 – NEP' SCHEME

PROGRAMME	DIPLOMA IN CM
PROGRAMME CODE	06
COURSE TITLE	WEB TECHNOLOGIES
COURSE CODE	CM41203
PREREQUISITE COURSE CODE & TITLE	NA
CLASS DECLARATION COURSE	NO

## I. LEARNING &amp; ASSESSMENT SCHEME

Course Code	Course Title	Course Type	Learning Scheme						Credits	Paper Duration	Assessment Scheme								Total Marks
			Actual Contact Hrs./Week			SLH	NLH	Theory			Based on LL & TSL		Based on SL		Total				
			CL	TL	LL						Practical		SLA						
											FA-TH	SA-TH		FA-PR		SA-PR			
Max	Max	Max	Min	Min	Max	Min	Max	Min											
CM41203	WEB TECHNOLOGIES	DSC	2	-	4	2	8	4	--	--	--	--	50	20	50	20	50	20	150

Total IKS Hrs for Term: 0 Hrs

**Abbreviations:** CL-Classroom Learning, TL-Tutorial Learning, LL-Laboratory Learning, SLH-Self Learning Hours, NLH-Notional Learning Hours, FA - Formative Assessment, SA -Summative assessment, IKS – Indian Knowledge System, SLA- Self Learning Assessment

**Legends:** @-Internal Assessment, #- External Assessment,\*# - Online Examination,@\$ - Internal Online Examination

**Note:**

FA-TH represents an average of two class tests of 30 marks each conducted during the semester.

- If a candidate is not securing minimum passing marks in FA-PR (Formative Assessment - Practical) of any course, then the candidate shall be declared as '**Detained**' in that course.
- If a candidate does not secure minimum passing marks in SLA (Self Learning Assessment) of any course, then the candidate shall be declared as '**fail**' and will have to repeat and resubmit SLA work.
- Notional learning hours** for the semester are (CL + LL + TL + SL) hrs. \* **15 Weeks**
- 1 credit** is equivalent to **30 Notional hours**.
- \* Self-learning hours shall not be reflected in the Timetable.
- \* Self-learning includes micro-projects/assignments/other activities.

## II. RATIONALE:

In the current trends of the web world, dynamic and platform-independent web applications are required. Client and Server-Side Scripting are an important scripting technology for computer engineering and Information Technology diploma graduates to develop dynamic and platform-independent web-based applications. Javascript, XML, NodeJS, and MySQL are recently used for web development.

## III. COURSE-LEVEL LEARNING OUTCOMES (CO's)

Students will be able to achieve & demonstrate the following CO's on completion of course-based learning

- CO1: Apply Java Script for web development.  
 CO2: Perform programs on Client Side Scripting  
 CO3: Write scripts on Server-Side programming.  
 CO4: Develop XML programs for web development.  
 CO5: Use NodeJs for server-side programming

#### IV. THEORY LEARNING OUTCOMES AND ALIGNED COURSE CONTENT

Sr. No	Theory Learning Outcomes (TLO's) aligned to CO's.	Learning content mapped with TLO's.	Suggested Learning Pedagogies	Relevant COs
<b>UNIT-I WEB ESSENTIALS (CL Hrs-4 , Marks- Nil )</b>				
1	<b>TLO 1.1:</b> Understand basics <b>TLO 1.2:</b> Differentiate between client and server <b>TLO 1.3:</b> Describe client-server architecture	<b>1.1.</b> Clients, Servers, and Communication. Internet-Basic Internet Protocols, World Wide Web-HTTP request-response message <b>1.2.</b> Web Clients, Web Servers example <b>1.3.</b> Working flow of Client-Side Scripting. Working flow of Server-Side Scripting <b>1.4.</b> Introduction to Client Scripting: JavaScript, VBScript, jQuery, Angular, CSS, React <b>1.5.</b> Introduction to Server Scripting: CGI, Java Server Page	Chalk-Board Demonstration Assignment	<b>CO1</b>
<b>UNIT-II CLIENT SIDE PROGRAMMING (CL Hrs-7, Marks- Nil )</b>				
2	<b>TLO 2.1:</b> Perform simple programs on JavaScript. <b>TLO 2.2:</b> Perform programs on Variables, loops, control statements, arrays, strings, etc <b>TLO 2.3:</b> Perform programs on objects, debuggers, file handling, etc	<b>2.1.</b> Client-Side Programming: The JavaScript Language, History and Versions <b>2.2.</b> Introduction JavaScript in Perspective- Syntax Variables and Data Types, Statements, Operators, Literals <b>2.3.</b> Functions, Objects, Arrays, Built-in Objects, JavaScript Debuggers.	Chalk-Board Demonstration Assignment	<b>CO2</b>
<b>UNIT-III SERVER SIDE PROGRAMMING (CL Hrs- 7, Marks- Nil )</b>				
3	<b>TLO 3.1:</b> Describe Server. <b>TLO 3.2:</b> Describe Servlet. <b>TLO 3.3:</b> Perform programs on the Server side. <b>TLO 3.4:</b> Explore Recent Technologies in Server-Side Programming.	<b>3.1.</b> Server-Side Programming: Java Servlets Architecture <b>3.2.</b> Overview: A Servlet-Generating Dynamic Content Life Cycle Parameter <b>3.3.</b> Data Sessions, Cookies, URL Rewriting, Other Capabilities <b>3.4.</b> Data Storage, Servlets and Concurrency- examples, Related Technologies	Chalk-Board Demonstration Assignment	<b>CO3</b>
<b>UNIT- IV XML (CL Hrs-04, Marks- Nil )</b>				
4	<b>TLO 4.1:</b> Perform programs on XML.	<b>4.1</b> XML: Introduction to XML, Defining XML tags, their attributes and values, Document type definition, <b>4.2</b> XML Schemas, Document Object Model, XHTML	Chalk-Board Demonstration Assignment	<b>CO4</b>

		4.3 Parsing XML Data:DOM and SAX parsers in Java		
<b>UNIT –V NodeJS (CL Hrs-08 , Marks-Nil )</b>				
5	<b>TLO 5.1:</b> Perform programs on NodeJS	5.1 NodeJs Introduction 5.2 NodeJs Express 5.3 NodeJS MySQL 5.4 Nodejs MongoDB	Chalk-Board Demonstration on Assignment	<b>CO5</b>

#### V. LABORATORY LEARNING OUTCOME AND ALIGNED PRACTICAL/ TUTORIAL EXPERIENCES

Sr. No.	Practical/Tutorial/Laboratory Outcome (LLO)	Learning	Laboratory Experiment/Practical Titles/Tutorial Titles	No.of Hrs.	Relevant COs
1	<b>LLO1</b> Write program using write().	JavaScript document.	Write JavaScript program using document.write().	2	CO2
2	<b>LLO2</b> Write program using operators.	JavaScript Arithmetic	*Write a JavaScript program to find the sum of two numbers using var	2	CO2
3	<b>LLO3</b> Write a program on the document object.	JavaScript	*Write a JavaScript program to illustrate the properties and methods of the document object	2	CO2
4	<b>LLO4</b> Write a program on date and time.	JavaScript	*Write a JavaScript program that displays today's date and current time.	2	CO2
5	<b>LLO5</b> Write a program to compute the day of the week from a given date.	JavaScript	*Write a JavaScript program to compute the day of the week (in words) while you input the date within the prompt dialogue box	2	CO2
6	<b>LLO6</b> Write JavaScript program on loop statement.	JavaScript	* Write a JavaScript program to generate the table of numbers using for loop statement	2	CO2
7	<b>LLO7</b> Write program on loop statement.	JavaScript	*Write a JavaScript program to generate the first 10 Fibonacci numbers.	2	CO2
8	<b>LLO8</b> Write program on functions.	JavaScript	*Create an HTML document that illustrates the function definition and calling of the Welcome() function	2	CO2
9	<b>LLO9</b> Write program on decision making.	JavaScript	*Write Javascript programs based on decision-making statements.	2	CO2
10	<b>LLO10</b> Write program on loop statement.	JavaScript	*Write Javascript programs based on looping statements.	2	CO2
11	<b>LLO11</b> Write a program on the array.	JavaScript	*Write Javascript programs based on arrays	2	CO2
12	<b>LLO12</b> Write program on functions.	JavaScript	*Write Javascript programs based on functions	2	CO2
13	<b>LLO13</b> Write program on strings.	JavaScript	*Write Javascript programs based on strings	2	CO21, 2
14	<b>LLO14</b> Write programs based on Objects, Functions, and Classes.	JavaScript	*Write Javascript programs on JS Objects, Functions, Classes	2	CO3



15	<b>LLO15</b> Write a JavaScript program based on HTML DOM.	*Write Javascript programs on JS HTML DOM	2	CO3
16	<b>LLO16</b> Write Javascript programs on JS Browser BOM	*Write Javascript programs on JS Browser BOM	2	CO3
17	<b>LLO17</b> Write Javascript programs on JS Web APIs	*Write Javascript programs on JS Web APIs	2	CO3
18	<b>LLO18</b> Write Javascript programs on AJAX	*Write Javascript programs on AJAX	2	CO3
19	<b>LLO19</b> Write simple XML programs.	*Write simple XML programs.	2	CO4
20	<b>LLO20</b> Write programs on XML AJAX.	*Write programs on XML AJAX.	2	CO4
21	<b>LLO21</b> Write programs on XML DOM.	*Write programs on XML DOM.	2	CO4
22	<b>LLO22</b> Write programs on XML Web Services.	*Write programs on XML Web Services.	2	CO4
23	<b>LLO23.1</b> Install and Create Node Environment.	*A) Install and Create Node environment a. Extending JavaScript b. V8 c. The Process Object	2	CO5
	<b>LLO23.2</b> Write a program on Asynchronous Event-Driven Programming.	B) Write programs on Asynchronous Event Driven Programming a. Broadcasting Events b. Listening for Events c. Timers d. Event Loop e. Callbacks and errors	2	CO5
24	<b>LLO24.1</b> Write programs on Streaming Data Across nodes and clients.	*A) Write programs on Streaming Data Across Node And clients a. Exporting Steams b. Creating an HTTP Server c. The Request object d. Working with Headers e. Handling Post Data	4	CO5
		B) Write programs on File System Operations: Create, Read, and Write files and directories.	4	CO5
	<b>LLO24.2</b> Write programs on Express.	C) Write programs on Using Express a. Installing Express and Making it Easier to Build Web Server b. Routes c. Static File and Middleware d. JSON	4	CO5
25	<b>LLO25</b> Write programs on Databases.	*Write programs on Databases. a. Relational Database and SQL b. Node & MYSQL c. NoSQL and Documents d. MongoDB and Mongoose e. CRUD operation using MongoDB	4	CO5

**Note: Out of the above suggestive LLOs –**

1. '\*' Marked Practicals (LLOs) Are mandatory.
2. A judicious mix of LLOs is to be performed to achieve the desired outcomes

**VI. SUGGESTED MICRO PROJECT/ASSIGNMENT/ACTIVITIES FOR SPECIFIC LEARNING/SKILLS DEVELOPMENT (SELF-LEARNING)****Self-Learning: Yes**



Suggestive list of case studies for self learning:

### 1. Build a book directory

Books Directory is a classic example of a project that a beginner can start. This project can be created by just using NodeJS and ExpressJS. To build this application you only need a collection of books and endpoints using four basic methods: GET, PUT, POST, and DELETE.

### 2. Portfolio Applications

This NodeJS-based application focuses on the collection of data and its management, whether it's for a revenue or economic expansion portfolio. It comprises three parts: the server, which delivers data on demand, the database, which arranges the data, and the application, which channels it. The server must respond to the user's requirements. You can also utilize the CRUD operation to build, interpret, upgrade, and remove entries in this case. It should have a showcase feature that displays the most recent data. The user's profile for managing personal details should be encrypted as well.

### 3. Job Search Application

This application is used for searching for a job related to one's field or area of expertise. Under Index.js, which represents the core file, various components can be generated. These components include programming language for API calls, writing job search details, and error messages in the event of an incorrect entry. **Helmet** can safeguard customers' details such as contact info and emails. It is a **NodeJS package containing 11 security components** that you can use to guarantee the robustness of your JavaScript backend and prevent data loss and unconfident connections.

### 4. Natural Language Processing

This application pushes human interaction to greater heights. The primary function of this application is to change a verbal message to a text-based message. Natural Language Process (NLP.js) is a general natural language software. NodeJS is a programming language for automatic speech identification and entity extraction that endorses 40 languages. This project provides a Graphic user interface-based platform that can be built to evaluate log files, extracting and forming insights to assist best and support customers and boost sales.

### 5. Chat Applications

This introductory project will teach you how to create your own real-time chat application. More importantly, it will showcase an essential concept that NodeJS alone isn't always enough to finish a task. If you're creating a chat application, you'll almost certainly require Express.js and a few other packages and libraries.

### 6. Web Scraper

A web scraper is a good place to start if you need data for your website. Assume you decided to collect film reviews. To obtain that data, you would need to create a web scraper. Perhaps you'd like a content aggregation service. You'd have to create a web scraper for that as well. Create a scraper that can extract particular data from specific websites.

### 7. Task Management

This application will assist you in maintaining your daily tasks and keeping a smooth workflow. This application also necessitates create, read, update, and delete (CRUD) operations. Start creating a primary index.js document to launch the application, routers to give appropriate routing to certain other paths, models for data storage in the database, a database that will collect all the data, middleware for verification, and any other packages that are required to operate the application. All of the tools required by software developers to operate NodeJS applications are contained in Node.exe. It's ideal for software that requires a stable link from the browser to the server.

### 8. Email Sender

This project is very interesting, to begin with. While working on this project, you can learn how to send and schedule emails in NodeJS. You can employ the Nodemailer plugin, which simplifies email sending and includes functions like HTML embedding.

NodeJS is also used to generate the emails you receive during the verification process for the user. The most commonly used network transport method is **Simple Mail Transfer Protocol (SMTP)** technology, which is used to send outgoing emails throughout various networks. It also acts as a transmission provider, allowing emails to be sent from one domain controller to another.

### 9. Discord Website

What do you think is more popular these days for a real-life chat? The answer is of course Discord and Slack. There are numerous bots available on Discord that make chatting more convenient. To enhance the functionality of your servers, you could test by developing a Discord bot with Node.js. As an example, consider a project planning bot for an efficient server.

### 10. Personal Website

This is a project you can undertake to help yourself in the future, create a personal website as well as a personal portfolio. Single-page web applications are the strength of Node.js. To make it perfect, instead of tough coding the data, build a database with your own information in it. This will help you gain more knowledge and experience with database management and processing. It will also make your website more adaptable.

### 11. Todo List Application:

Create a simple todo list application where users can add, edit, and delete tasks.

Implement features such as marking tasks as completed, filtering tasks by status, and clearing completed tasks. This project will help you understand the basics of React state management and handling user interactions.

### 12. Weather App:

- Build a weather application that allows users to search for weather forecasts by city.
- Utilize a weather API (such as OpenWeatherMap) to fetch weather data based on user input.
- Display weather information including temperature, humidity, wind speed, etc.
- This project will help you practice making API calls, handling asynchronous data, and rendering dynamic content in React.

### 13. Recipe Finder:

- Develop a recipe finder application that enables users to search for recipes by ingredients or cuisine.
- Integrate with a recipe API (e.g., Spoonacular, Edamam) to fetch recipe data based on user queries.
- Display recipe details such as ingredients, instructions, and cooking time.
- Implement features like saving favorite recipes or filtering recipes by dietary preferences.
- This project will give you experience working with external APIs, managing application state, and handling user input.

### 14. Movie Catalog:

- Create a movie catalog application where users can browse and search for movies.
- Fetch movie data from a movie database API (e.g., The Movie Database, OMDb API).
- Display movie details such as title, poster, release year, genres, etc.
- Implement features like sorting movies by popularity, filtering movies by genre, and adding movies to a watchlist.
- This project will help you practice working with APIs, managing complex data structures, and building reusable components.

### 15. Blogging Platform:

- Develop a simple blogging platform where users can create, edit, and delete blog posts.
- Implement user authentication and authorization to allow only authenticated users to create or edit posts.

- Use a backend server (e.g., Node.js with Express) to store and retrieve blog post data.
- Display blog posts with features like pagination, comments, and likes.
- This project will give you experience with full-stack development, handling user authentication, and interacting with a backend server.

### **16. Interactive To-Do List**

Create a simple to-do list application that allows users to add, remove, and mark tasks as complete. Firstly, start with a basic HTML structure. Then, use JavaScript to handle user interactions and update the list dynamically. Next, focus on implementing features like adding new tasks, checking off completed tasks, and deleting tasks. Additionally, enhance the project by incorporating features such as storing tasks in the browser's local storage, enabling task prioritization, or adding due dates.

### **17. Simple Weather Application**

Build a weather application that fetches weather data from an Application Programming Interface (API) and displays it to the user. To begin with, design a basic user interface using HTML and CSS. Then, use JavaScript to make an API request and dynamically update the weather information based on the user's location or a selected city. Additionally, one can include features such as displaying current weather conditions, temperature, humidity, and a five-day forecast.

### **18. Interactive Quiz Game**

Another interesting Javascript project idea is an interactive quiz game where users can answer questions and receive immediate feedback on their responses. Start by designing the quiz layout using HTML and CSS. Next, use JavaScript to handle the logic of presenting questions, accepting user answers, and providing feedback on accuracy. Finally, store the quiz questions and answers in an array or object and randomize the order of questions for added variety.

### **19. Memory Game**

Build a memory game where users have to match pairs of cards within a certain time limit. First, use JavaScript to handle game logic, track scores, and control card interactions. Then, add features like difficulty levels, card animations, and a scoreboard to make it more engaging.

### **20. Tip Calculator**

Develop a tip calculator that allows users to calculate the appropriate tip amount based on the total bill and chosen tip percentage. Then, use JavaScript to calculate and display the tip amount and total bill dynamically. One can also enhance the app by adding options to split the bill among multiple people or include tax calculations.

### **21. Note-Taking Application**

Build a simple note-taking application that allows users to create, edit, and delete notes. Javascript can organize the "create, read, update, and delete" operations and store the notes in the browser's local storage.

### **22. BMI Calculator**

Create a Body Mass Index (BMI) calculator that allows users to input their height and weight to calculate their BMI. Then, perform the BMI calculation based on the input values and display the result on the webpage. Additionally, consider including features such as different unit options (metric or imperial) and providing a BMI classification based on the calculated value.

### **23. Random Quote Generator**

Develop a random quote generator that displays a new inspiring quote each time the user clicks a button. Use JavaScript to store an array of quotes or draw them from an API and display them on the webpage. Remember, the emphasis here is on learning and building a strong foundation in JavaScript. Start with smaller, viable projects and gradually increase the complexity as confidence and experience increase.

### **24. E-Commerce Website**



Create an online store where people can browse and buy products. To do this, coders must use tools like React.js for the visual part and Node.js for the behind-the-scenes stuff. Additionally, it's important to make the website look good and easy to use. One can add features like search, filters, and a smooth checkout process. Remember to include user accounts, order history, and inventory management to make it more complete and professional.

### 25. Social Media Application

Another exciting project is to build one's own social media platform. For instance, create a platform where people can connect and share things. First, use frameworks like React.js or Vue.js to make it look good and Node.js with a database to store the data. Then, start with features like user profiles, news feeds, and the ability to post updates. Then, make it interactive by adding features like liking and commenting.

### Activities

- Students are encouraged to use online tools to improve their learning, such as the e-Kumbh from AICTE and the virtual Labs from IIT.
- Students should be encouraged to participate in various coding competitions, such as hackathons, and online coding contests on websites like Hackerrank, Codechef etc.
- At the department level, encourage students to start a coding club
- Students are encouraged to register themselves in various MOOCs such as Infosys Springboard, Swayam etc. to further enhance their learning.

### Note:

1. The above is suggestive list of case studies for SLA
2. The faculty must allocate any 1 Case study to individual student. Considering the students technical skills.

### Assignment

Prepare a journal of practicals performed in the laboratory.

## VII. LABORATORY EQUIPMENT/INSTRUMENTS/TOOLS/SOFTWARE REQUIRED

Sr.No	Equipment Name with Broad Specifications	Relevant LLO Number
1	JavaScript	CO1, CO2, CO3
2	XML	CO4
3	NodeJS	CO5
4	MongoDB	CO5

## VIII. SUGGESTED FOR WEIGHTAGE TO LEARNING EFFORTS & ASSESSMENT PURPOSE (Specification Table)

Sr. No	Unit	Unit Title	Aligned COs	Learning Hours	R-Level	U-Level	A-Level	Total Marks
1	I	Web Essentials	CO1	4	-	-	-	-
2	II	Client-Side Programming	CO2	7	-	-	-	-
3	III	Server-Side Programming	CO3	7	-	-	-	-
4	IV	XML	CO4	4	-	-	-	-
5	V	NodeJS	CO5	8	-	-	-	-
<b>Grand Total</b>				<b>30</b>	-	-	-	-



**IX. ASSESSMENT METHODOLOGIES/TOOLS**

Formative assessment (Assessment for Learning)	Summative Assessment (Assessment of Learning)
Lab performance, Assignment, Self-learning and Seminar/Presentation	Lab. Performance, viva voce, etc

**X. SUGGESTED COS- POS MATRIX FORM**

Course Outcomes (Cos)	Programme Outcomes(Pos)							Programme Specific Outcomes *(PSOs)	
	PO-1 Basic and Discipline-Specific Knowledge	PO-2 Problem Analysis	PO-3 Design/ Development of Solutions	PO-4 Engineering Tools	PO-5 Engineering Practices for Society, Sustainability and Environment	PO-6 Project Management	PO-7 Life Long Learning	PSO-1	PSO-2
CO1	2	2	2	2	-	-	2	-	2
CO2	2	3	3	3	3	2	2	-	3
CO3	2	3	3	3	3	2	2	-	3
CO4	2	1	1	2	2	-	2	-	3
CO5	2	3	3	3	3	2	2	-	3



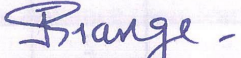

**Legends:- High:03, Medium:02, Low:01, No Mapping: -**  
\*PSOs are to be formulated at the institute level

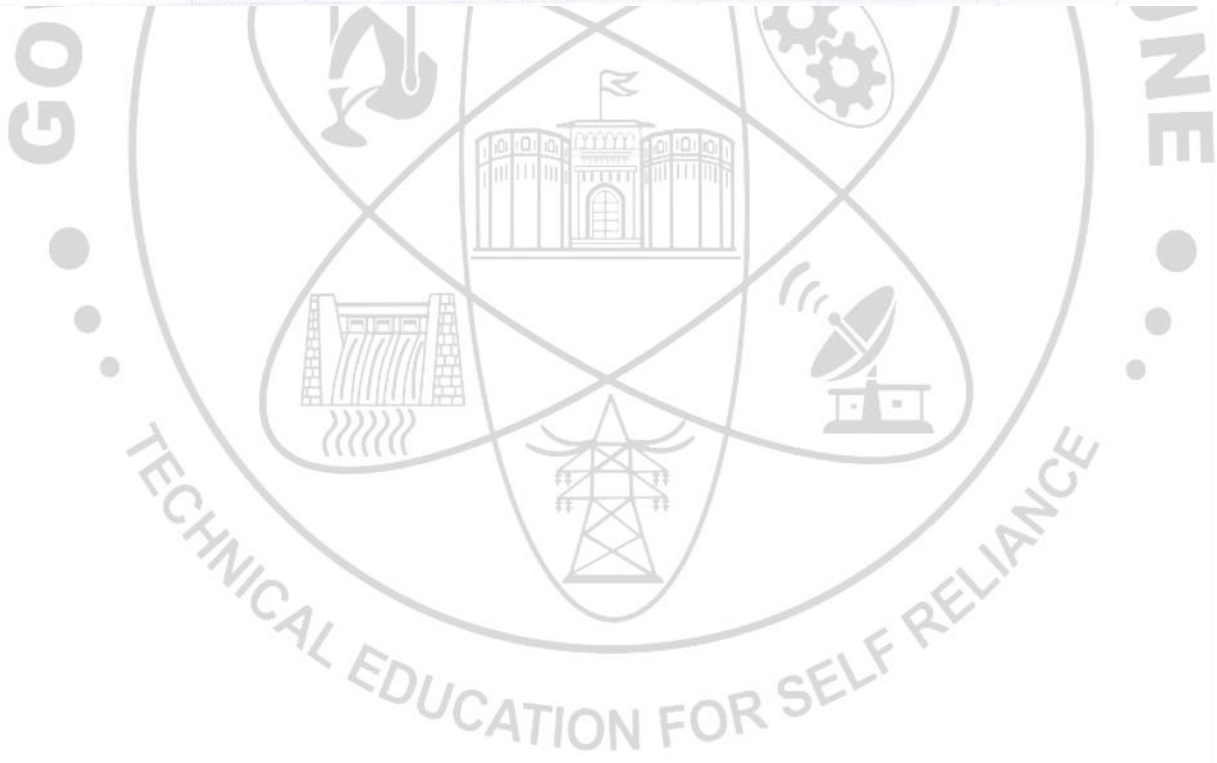
**XI. SUGGESTED LEARNING MATERIALS/BOOKS**

Sr.No	Author	Title	Publisher
1	Jim Keogh	JavaScript Demystified	Tata McGraw Hill, First Edition - June 2005, ISBN: 0072254548
2	Michael Moncur	JavaScript in 24 hours	Sam's Publishing; 7th edition – February 2019, ISBN-10: 0672338092 ISBN-13: 978-0672338090
3	David Herron	Node.js Web Development - Fifth Edition: Server-side web development made easy with Node 14 using practical examples	Packt Publishing
4	Kogent Learning Solutions	Web Technologies: HTML, JAVASCRIPT, PHP, JAVA, JSP, XML and AJAX, Black Book Kindle Edition	Dreamtech Press

**XI. LEARNING WEBSITES & PORTALS**

1. <a href="https://www.w3schools.com/">https://www.w3schools.com/</a>
2. <a href="https://www.tutorialspoint.com/index.html">https://www.tutorialspoint.com/index.html</a>
3. <a href="https://www.javatpoint.com/">https://www.javatpoint.com/</a>
4. <a href="https://nodejs.org/en">https://nodejs.org/en</a>

Name & Signature:  <b>Dr. S.B. Nikam</b> Lecturer in Computer Engineering		Name & Signature:  <b>Smt.N.R. Wagh</b> Lecturer in Computer Engineering	
<b>(Course Experts)</b>			
Name & Signature:  <b>Smt.J.R.Hange</b> (Programme Head)		Name & Signature:  <b>Shri. S.B. Kulkarni</b> (CDC In-charge)	



GOVERNMENT POLYTECHNIC, PUNE

'120 – NEP' SCHEME

PROGRAMME	DIPLOMA IN CE/EE/ET/ME/MT/CM/IT/DDGM
PROGRAMME CODE	01/02/03/04/05/06/07/08
COURSE TITLE	INDIAN CONSTITUTION: CORE CONCEPTS AND VALUES
COURSE CODE	HU21203
PREREQUISITE COURSE CODE & TITLE	NA
CLASS DECLARATION COURSE	NO

I. LEARNING & ASSESSMENT SCHEME

Course Code	Course Title	Course Type	Learning Scheme					Credits	Paper Duration	Assessment Scheme										Total Marks
			Actual Contact Hrs./Week			SLH	NLH			Theory	Practical	Based on LL & TSL				Based on SL				
			CL	TL	LL							Practical				SLA				
			FA-TH	SA-TH	Total		FA-PR			SA-PR	SLA									
Max	Max	Max	Min	Max	Min	Max	Min	Max	Min											
HU21203	INDIAN CONSTITUTION : CORE CONCEPTS AND VALUES	VEC	1	--	--	1	2	1	--	--	--	--	--	--	--	--	50	20	50	

Total IKS Hrs for Term: 0 Hrs

Abbreviations: CL-Classroom Learning, TL-Tutorial Learning, LL-Laboratory Learning, SLH-Self Learning Hours, NLH-Notional Learning Hours, FA - Formative Assessment, SA -Summative assessment, IKS – Indian Knowledge System, SLA- Self Learning Assessment

Legends: @-Internal Assessment, #- External Assessment,\*# - Online Examination,@\$ - Internal Online Examination

Note:

1. FA-TH represents an average of two class tests of 30 marks each conducted during the semester.
2. If a candidate is not securing minimum passing marks in FA-PR (Formative Assessment - Practical) of any course, then the candidate shall be declared as 'Detained' in that course.
3. If a candidate does not secure minimum passing marks in SLA (Self Learning Assessment) of any course, then the candidate shall be declared as 'fail' and will have to repeat and resubmit SLA work.
1. Notional learning hours for the semester are (CL + LL + TL + SL) hrs. \* 15 Weeks
4. 1 credit is equivalent to 30 Notional hours.
5. \* Self-learning hours shall not be reflected in the Timetable.
6. \* Self-learning includes micro-projects/assignments/other activities.

II. RATIONALE:

Introducing a course on the Indian Constitution can provide students with a comprehensive understanding of the country's legal framework and democratic principles. Such a course could cover the historical context of its creation, the structure and functions of the government it establishes, and the fundamental rights and duties of citizens. It could also explore the significant amendments and judicial interpretations that have shaped its evolution over time. This foundational knowledge is not only for fostering informed and engaged citizens who can contribute to the nation's democratic processes but also enriches the educational experience by fostering a sense of national identity and ethical responsibility among future engineers. Furthermore, embedding Electoral Literacy and Voter Education in diploma engineering programs strategically empowers these future professionals with an awareness of their electoral privileges and the workings of democracy.

**III. COURSE-LEVEL LEARNING OUTCOMES (CO's):**

Students will be able to achieve & demonstrate the following CO's on completion of course-based learning

- CO1:** Foster comprehension of the fundamental principles and goals embedded in the Indian constitution.
- CO2:** Elaborate on the core rights and duties conferred upon Indian citizens by the Constitution.
- CO3:** Comprehend the distribution of legislative, executive, and financial powers between the Union and the States.
- CO4:** Understand the functioning of Indian democracy, encompassing its frameworks and mechanisms at local, state, and national levels.
- CO5:** Cultivate the skills and perspectives required for active participation in electoral processes, the conscientious exercise of voting rights, and the promotion of informed democratic participation within society.

**IV. THEORY LEARNING OUTCOMES AND ALIGNED COURSE CONTENT**

Sr. No	Theory Learning Outcomes (TLO's) aligned to CO's.	Learning content mapped with TLO's.	Suggested Learning Pedagogies	Relevant COs
<b>UNIT-I INTRODUCTION TO INDIAN CONSTITUTION (CL Hrs-03, Marks-NIL)</b>				
1.	<p><b>TLO 1.1</b> Understand the historical context and events leading to the drafting of the Indian Constitution.</p> <p><b>TLO 1.2</b> Comprehend the essential features and understand the significance of the Indian Constitution in shaping India's democratic governance and societal ethos.</p> <p><b>TLO 1.3</b> Analyze the vision and ideals articulated in the Preamble and their relevance in contemporary Indian society.</p>	<p>1.1 Historical background and making of the Indian Constitution</p> <p>1.2 Salient features and significance of the Indian Constitution</p> <p>1.3 Preamble: Vision and Ideals of the Indian Constitution</p>	<p>Presentations</p> <p>Case Studies and Analysis</p> <p>Role-Playing and Simulations</p> <p>Project-Based Learning</p>	<b>CO1</b>
<b>UNIT - II FUNDAMENTAL RIGHTS, FUNDAMENTAL DUTIES AND DIRECTIVE PRINCIPLES (CL Hrs-04, Marks-NIL)</b>				
2	<p><b>TLO2.1</b> Understand the introduction and structure of Fundamental Rights in Part III of the Indian Constitution.</p> <p><b>TLO2.2</b> Understand the principles of the Right to</p>	<p><b>2.1</b> Fundamental Rights: Introduction &amp; its Scheme under Part -III</p> <p><b>2.2</b> Right to Equality (Article 14-18)</p> <p><b>2.3</b> Right to Freedom (Article 19-22)</p> <p><b>2.4</b> Right to Life (Article 21)</p> <p><b>2.5</b> Fundamental Duties and their Significance under Part IV-A</p>	<p>Presentations</p> <p>Case Studies and Analysis</p> <p>Role-Playing and Simulations</p> <p>Project-Based Learning</p>	<b>CO2</b>



	<p>Equality, Right to Freedom, and Right to Life.</p> <p><b>TLO2.3</b> Identify fundamental duties in general and in particular with the engineering field.</p> <p><b>TLO2.4:</b> Grasp the significance and practical application of Directive Principles of State Policy outlined in Part IV of the Indian Constitution.</p>	<p><b>2.6</b> Directive Principles of State Policy under Part – IV: importance and implementation.</p>		
<b>UNIT- III UNION AND STATE EXECUTIVE(CL Hrs-03, Marks-NIL)</b>				
3	<p><b>TLO 3.1</b> 3.1: Gain insight into the structure and functions of the Union executives and the jurisdiction of the Supreme Court.</p> <p><b>TLO 3.2</b> 3.2: Understand the organization and responsibilities of the State Executives and the functions of the State Judiciary(High Courts).</p>	<p><b>3.1</b> Union Government, Union Legislature (Parliament), Lok Sabha and Rajya Sabha (with Powers and Functions), Union Executive, President of India (with Powers and Functions), Prime Minister of India (with Powers and Functions), Union Judiciary (Supreme Court), Jurisdiction of the Supreme Court.</p> <p><b>3.2</b> State Government, State Legislature (Legislative Assembly/ Vidhan Sabha, Legislative Council / Vidhan Parishad), Powers and Functions of the State Legislature, State Executive, Governor Of the State (with Powers and Functions), The Chief Minister Of the State (With Powers and Functions) State Judiciary (High Courts).</p>	<p>Presentations Case Studies and Analysis Role-Playing and Simulations Project-Based Learning</p>	CO3
<b>UNIT-IV AMENDMENTS AND EMERGENCY PROVISIONS(CL Hrs-03, Marks-NIL)</b>				
4	<p><b>TLO 4.1</b> Comprehend the meaning and significance of constitutional amendments, as well as the procedural rules detailed in Article 368 of the Indian Constitution.</p> <p><b>TLO 4.2</b> Recognize the roles of various branches of government in the amendment process,</p> <p><b>TLO 4.3</b> Examine the significant procedures and historical context of major constitutional amendments</p>	<p><b>4.1 Introduction to Constitutional Amendments:</b> Definition and significance of constitutional amendments. Constitutional provisions governing the amendment procedure (Article 368).</p> <p><b>4.2 Types of Amendments:</b> Simple majority amendments, Special majority amendments, Amendments requiring ratification by states.</p> <p><b>4.3 Role of the Executives Amendments:</b> Role of Parliament: Lok Sabha and Rajya Sabha, Role of President: Assent to amendments, Role of State Legislatures: Ratification of certain</p>	<p>Presentations Case Studies and Analysis Role-Playing and Simulations Project-Based Learning</p>	CO4

		<p>amendments.</p> <p><b>4.4 Major Constitutional Amendment procedures:</b> Major Constitutional Amendment procedures - 1st, 7th, 42nd, 44th, 73rd &amp; 74th, 76th, 86th, 52nd &amp; 91st, 102nd</p>		
<b>UNIT –V ELECTORAL LITERACY (CL Hrs-02, Marks-NIL)</b>				
5	<p><b>TLO5.</b> Electoral Literacy: Develop understanding and proficiency in electoral processes, voter registration, rights and responsibilities of voters, electoral reforms, and initiatives promoting electoral literacy.</p>	<p><b>5.1 Understanding the Electoral Process :</b> Overview of the electoral process: registration, voting, counting, and declaration of results, Role and functions of the Election Commission of India Types of elections: Lok Sabha, Rajya Sabha, State Legislative Assembly, Local Body elections</p> <p><b>5.2 Voter Registration and Electoral Rolls:</b> Importance of voter registration Eligibility criteria for voter registration Process of voter registration: online, offline, and special drives Checking and updating voter details in electoral rolls</p> <p><b>5.3 Rights and Responsibilities of Voters:</b> Understanding fundamental rights related to elections Responsibilities of voters towards ensuring free and fair elections Consequences of electoral malpractices and non-participation</p> <p><b>5.4 Electoral Reforms and Initiatives:</b> Overview of electoral reforms aimed at enhancing transparency, inclusivity, and integrity of elections Role of technology in improving electoral processes: Voter Verifiable Paper Audit Trail (VVPAT), Online voter registration, e-voting Initiatives by the Election Commission and civil society organizations to promote electoral literacy</p>	<p>Presentations Case Studies and Analysis Role-Playing and Simulations Project-Based Learning</p>	CO5

V. LABORATORY LEARNING OUTCOME AND ALIGNED PRACTICAL/ TUTORIAL EXPERIENCES.

NOT APPLICABLE

VI. SUGGESTED MICRO PROJECT/ASSIGNMENT/ACTIVITIES FOR SPECIFIC LEARNING/SKILLS DEVELOPMENT (SELF-LEARNING)

- i) **Case Study Analysis:** Select a few landmark Supreme Court cases related to Fundamental Rights (e.g., Kesavananda Bharati v. State of Kerala, Maneka Gandhi v. Union of India) and analyze the court's interpretation and impact on these rights.
- ii) **Comparative Analysis:** Compare the provisions of the Right to Equality under Articles 14-18 with similar provisions in the constitutions of other countries. Highlight similarities, differences, and the reasoning behind them.
- iii) **Public Awareness Campaign:** Design a public awareness campaign to educate citizens about their Fundamental Rights and Duties. Create informative posters, social media content, and interactive workshops to engage people in discussions about constitutional rights and responsibilities.
- iv) Write a reflective essay discussing the historical context and debates surrounding the inclusion of Fundamental Rights in the Indian Constitution.
- v) Create a visual timeline depicting the evolution of laws related to equality in India, from independence to the present day. Include major legislative reforms and judicial decisions.
- vi) Conduct a comparative analysis of the implementation of Directive Principles in different states of India, identifying successful initiatives and areas needing improvement.
- vii) **Case Study Analysis:** Choose a recent constitutional or political issue that has been debated in Parliament. Analyze the roles played by the Loksabha and Rajya Sabha in addressing the issue and the impact of their decisions.
- viii) **Case Study Analysis: Analyze a landmark constitutional amendment in India (e.g., the 42nd Amendment) and its impact on governance, fundamental rights, and the balance of power between different branches of government.**
- ix) **Infographic Creation:** Create an infographic illustrating the process of amending the Indian Constitution as outlined in Article 368. Highlight key steps and requirements for different types of amendments.
- x) **Timeline Project:** Create a timeline highlighting major constitutional amendments in India, such as the 1st, 7th, 42nd, 44th, 73rd & 74th, 76th, 86th, 52nd & 91st, and 102nd amendments. Include key provisions and the political context surrounding each amendment.
- xi) **Debate:** Organize a debate on the topic "Should the President have the power to refuse assent to constitutional amendments?" Encourage students to research and present arguments from legal, political, and ethical perspectives.
- xi) **Campaign Design:** Design a social media campaign to raise awareness about the importance of voter participation and responsible voting. Create visually engaging posters, infographics, and videos highlighting the consequences of electoral malpractices and non-participation.
- xii) **Online Tutorial:** Create a step-by-step tutorial video or guide demonstrating the voter registration process, both online and offline. Include instructions for checking and updating voter details in

electoral rolls.

- xiii) **Survey Project:** Conduct a survey to assess the awareness and accessibility of voter registration facilities among different demographic groups in your locality. Analyze the results and propose strategies to improve voter registration rates.

**VII. LABORATORY EQUIPMENT/INSTRUMENTS/TOOLS/SOFTWARE REQUIRED**

NOT APPLICABLE

**VIII. SUGGESTED FOR WEIGHTAGE TO LEARNING EFFORTS & ASSESSMENT PURPOSE (Specification Table)**

NOT APPLICABLE

**IX. ASSESSMENT METHODOLOGIES/TOOLS**

Formative assessment (Assessment for Learning)	Summative Assessment (Assessment of Learning)
Assignment, Self-learning and Terms work Seminar/Presentation	—

**X. SUGGESTED COS- POS MATRIX FORM**

Course Outcomes (Cos)	Programme Outcomes(Pos)							Programme Specific Outcomes *(PSOs)	
	PO-1 Basic and Discipline-Specific Knowledge	PO-2 Problem Analysis	PO-3 Design/ Development of Solutions	PO-4 Engineering Tools	PO-5 Engineering Practices for Society, Sustainability and Environment	PO-6 Project Management	PO-7 Life Long Learning	PSO-1	PSO-2
CO1	--	--	--	--	2	--	2		
CO2	--	--	--	--	3	--	2		
CO3	--	--	--	--	3	--	2		
CO4	--	--	--	--	3	--	2		
CO5	--	--	--	--	3	--	2		

**Legends:- High:03, Medium:02, Low:01, No Mapping: -**  
\*PSOs are to be formulated at the institute level






**XI. SUGGESTED LEARNING MATERIALS/BOOKS**

Sr.No	Author	Title	Publisher
1	M. Laxmikanth	"Indian Polity"	McGraw Hill Education: ISBN-13: 978-9352603633
2	D. D. Basu	Introduction to the Constitution of India	LexisNexis: ISBN-13: 978-8180386477
3	Subhash C. Kashyap	Our Constitution: An Introduction to India's Constitution and Constitutional Law	National Book Trust, India ISBN-13: 78-8123748462
4	Arun K. Thiruvengadam	The Constitution of India: A Contextual Analysis	Oxford University Press ISBN-13: 978-0199467078
5	Oxford University Press	The Making of India's Constitution	Oxford University Press Oxford University Press

**XI. LEARNING WEBSITES & PORTALS**

Sr.No.	Link/Portal	Description
1	<a href="https://prsindia.org/">https://prsindia.org/</a>	In-depth analysis of parliamentary affairs, legislative processes, and policy Issues in India.
2	<a href="https://awmin.gov.in">https://awmin.gov.in</a>	Official repository providing access to the full text of the Indian Constitution.
3	<a href="https://constitution.org.in">https://constitution.org.in</a>	Interactive platform offering the text of the Constitution along with annotations and historical context.
4	<a href="https://indiankanoon.org">https://indiankanoon.org</a>	Legal search engine offering a vast database of Indian case law, including constitutional judgments.
5	<a href="https://nptel.ac.in">https://nptel.ac.in</a>	Offers video lectures and course materials on studies of law and the constitution.

Name & Signature:	
 <b>Mr. S.B. Kulkarni</b> Lecturer in Mechanical Engineering (Course Experts)	
Name & Signature:	Name & Signature:
 <b>Smt. J.R. Hange</b> (Programme Head)	 <b>Shri. S.B. Kulkarni</b> (CDC In-charge)