

**213 - BACHELOR OF COMPUTER APPLICATIONS (BCA)**

Programme Structure and Scheme of Examination (under CBCS)  
 (Applicable to the candidates admitted in Affiliated Colleges  
 from the academic year 2022 -2023 onwards)

Course Code	Part	Study Components & Course Title	Hours/ Week	Credit	Maximum Marks		
					CIA	ESE	Total
<b>SEMESTER - I</b>							
22UTAML11	I	Language Course - I : Tamil/Other Languages	5	3	25	75	100
22UENGL12	II	English Course - I : Communicative English I	5	3	25	75	100
22UBCAC13	III	Core Course – I: Programming in C	5	4	25	75	100
22UBCAC14		Core Course – II : Digital Computer Fundamentals	5	4	25	75	100
22UBCAP15		Core Practical – I :Programming in C Lab	3	2	40	60	100
22UMAF01		Allied Course - I : Paper -1 Mathematical Foundations-I	5	4	25	75	100
22UENVS18	IV	Environmental Studies	2	2	25	75	100
<b>Total</b>			<b>30</b>	<b>22</b>			<b>700</b>
<b>SEMESTER - II</b>							
22UTAML21	I	Language Course - II : Tamil/Other Languages	5	3	25	75	100
22UENGL22	II	English Course - II : Communicative English II	5	3	25	75	100
22UBCAC23	III	Core Course – III :C++ & Data Structures	5	4	25	75	100
22UBCAC24		Core Practical – II :C++ & Data Structures Lab	3	2	40	60	100
22USMAA02		Allied Course - I : Paper -2 Statistical Methods and their Applications	5	4	25	75	100
22UBCAE26	IV	Internal Elective – I	3	3	25	75	100
22UVALE27		Value Education	2	1	25	75	100
22USOFS28		Soft Skill	2	1	25	75	100
<b>Total</b>			<b>30</b>	<b>21</b>			<b>800</b>

**INTERNAL ELECTIVE COURSES**

22UBCAE26-1	Internal Elective - I	Computer Organization and Architecture
22UBCAE26-2		Computer Graphics
22UBCAE26-3		Internet and its Applications

<b>SEMESTER:I</b> <b>PART: III</b>	<b>22UBCAC13 : PROGRAMMING IN C</b>	<b>CREDIT:4</b> <b>HOURS:5</b>
---------------------------------------	-------------------------------------	-----------------------------------

**COURSE OBJECTIVES**

1. To understand simple algorithms
2. To understand language constructs
3. To understand and develop programming skills in C.
4. To understand the basic concepts of decision making and looping statements.
5. To understand the concepts of arrays , structures, union, pointers and files.

**UNIT I :CONCEPT OF C PROGRAMMING****Hours:15**

History, Introduction of C programming language, Structure of C program, C character set, Data types, Variables, Constants, Keywords and Identifiers, Expression statements in C language, Operators (Arithmetic, Logical, Relational, Assignment etc.).

**UNIT II: CONDITIONAL PROGRAM****Hours:15**

Execution, IF statement, IF.....ELSE statements nested IF.....ELSE and ELSE IF ladder. Program Loops and Iteration, WHILE loop, DO loop and FOR loop, Nested Loops, Use of break, continue and GOTO statements, Switch statement, use of break and default with switch, Storage Class in C language.

**UNIT III: FUNCTIONS****Hours:15**

Built-In and User Defined functions, Function Declaration, Definition and Function Calling, Parameter Passing (Call by Value and Call by Reference), Recursion, Pointers, Macros.

**UNIT IV: ARRAYS****Hours:15**

Definition of array, declaration, Linear Arrays, Multidimensional Arrays, Passing Array to function, String, string handling functions, Dynamic Memory Allocation.

**UNIT V: STRUCTURE AND UNION****Hours:15**

Definition, Programs using Structure and Union, Difference between Structure and Union. File Handling: Opening and Closing data files, Read and Write Functions, different modes of files.

**COURSE OUTCOMES**

After completing the Course successfully, the student will be able to

1. The Student will be able to understand the concepts of Constants, Variables, and Data Types, Operators and Expressions
2. The Student will be able to understand the concepts of Managing Input and Output Operations, Decision Making and Branching, Decision Making and Looping.
3. The Student will be able to understand the concepts of Arrays, Character Arrays and Strings, User Defined Functions.
4. The Student will be able to understand the concepts of Structure and

Unions, Pointers, File Management in C.

5. The Student will be able to understand the concepts of Fundamental Algorithms, Factoring Methods.

**Text Books (In API Style)**

1. BalaguruswamyE., TMH, “Programming in ANSI C”.
2. Kanitkar Yashwant, BPB, “Let Us C”.
3. Kanitkar Yashwant, BPB, “Working With C”.

**Supplementary Readings**

1. Shaum’s Series ,TMH, “Programming in C”.
2. Forouzan, Thomson, Cengase, “Computer Science”

**OUTCOME MAPPING**

<b>CO/PO</b>	<b>PO1</b>	<b>PO2</b>	<b>PO3</b>	<b>PO4</b>	<b>PO5</b>
<b>CO1</b>	<b>2</b>	<b>3</b>	<b>3</b>	<b>3</b>	<b>2</b>
<b>CO2</b>	<b>1</b>	<b>2</b>	<b>2</b>	<b>3</b>	<b>1</b>
<b>CO3</b>	<b>3</b>	<b>3</b>	<b>3</b>	<b>3</b>	<b>2</b>
<b>CO4</b>	<b>1</b>	<b>3</b>	<b>2</b>	<b>2</b>	<b>1</b>
<b>CO5</b>	<b>1</b>	<b>3</b>	<b>3</b>	<b>3</b>	<b>1</b>

1 – Low, 2 – Moderate, 3 – High (Preferably use 2 or 3 levels)

<b>SEMESTER:I</b> <b>PART: III</b>	<b>22UBCAC14 : DIGITAL COMPUTER FUNDAMENTALS</b>	<b>CREDIT: 4</b> <b>HOURS: 5</b>
---------------------------------------	--	-------------------------------------

### **COURSE OBJECTIVES**

1. Develop an understanding of digital circuit design and analysis.
2. Learn design techniques for working with digital electronic devices, and their application to solving problems.
3. Learn analysis skills to effectively report on the design, analysis and data of projects so that others can understand their methodology and results.
4. Become familiar with digital design, analysis and simulation tools.
5. Develop effective written communication skills using various media tools.

### **UNIT-I: NUMBER SYSTEM AND CODES**

**Hours: 12**

Decimal Numbers, Binary Numbers, Decimal to Binary Conversions, Binary Arithmetic, 1's and 2's complements of Binary Numbers, Signed Numbers, Arithmetic Operations with Signed numbers, Hexadecimal Numbers, Octal Numbers, Digital Codes, Error Detection Codes.

### **UNIT-II: LOGIC GATES**

**Hours :12**

The Inverter, The AND gate, The OR gate, The NAND gate, NOR gate, The Exclusive-OR gate and Exclusive OR gate: Boolean Algebra and Logic Simplification-Boolean Operations and Expressions, Laws and Rules, De Morgan's Theorems, Boolean Expressions and Truth Tables, The Karnaugh Map, SOP minimizations.

### **UNIT - III: COMBINATIONAL LOGIC ANALYSIS**

**Hours: 12**

Basic combinational Logic Circuits, Implementing Combinational Logic, The Universal Property of NAND and NOR Gates. Functions of Combinational Logic - Basic Adder, Parallel Binary Adders, Comparators, Decoders, Encoders, Code Converters, Multiplexers, Parity Generator/Checkers.

### **UNIT-IV: LATCHES AND FLIP-FLOPS**

**Hours : 12**

Latches, Edge Triggered Flip-Flops, Flip-Flop Operating characteristics, Flip-Flop Applications, Registers, Counters.

### **UNIT-V: MEMORY AND STORAGE**

**Hours: 12**

Memory Basics, The RAM, The ROM, Programmable ROMs, The Flash Memory, Memory Expansion, Special Types of Memories, Magnetic and Optical Storage.

### **COURSE OUTCOMES**

After completing the Course successfully, the student will be able to

1. Identify the logic gates and their functionality.
2. Perform number conversions from one system to another system.
3. Design basic electronic circuits (combinational circuits).
4. Perform a comparative analysis of the components of different memory units.
5. Perform number conversions.

**Text Books**

1. Floyd, Thomas L,1997, University Book Stall, 10thEdition“Digital Computer Fundamentals”.

**Supplementary Readings:**

1. Malvino, Paul Albert and Leach, Donald P, 2000,TMH, 4thEdition, “Digital Principles and Applications”.
2. Malvino, Paul Albert and Leach, Donald P,1995,TMH, 3<sup>rd</sup> Edition, “Digital Computer Fundamentals”.
3. Bartee, Thomas C,1995, TMH,6<sup>th</sup> Edition, “Digital Computer Fundamentals”.

**OUTCOME MAPPING**

<b>CO/PO</b>	<b>PO1</b>	<b>PO2</b>	<b>PO3</b>	<b>PO4</b>	<b>PO5</b>
<b>CO1</b>	<b>1</b>	<b>3</b>	<b>3</b>	<b>1</b>	<b>3</b>
<b>CO2</b>	<b>1</b>	<b>2</b>	<b>2</b>	<b>2</b>	<b>1</b>
<b>CO3</b>	<b>3</b>	<b>2</b>	<b>3</b>	<b>3</b>	<b>2</b>
<b>CO4</b>	<b>1</b>	<b>3</b>	<b>2</b>	<b>2</b>	<b>1</b>
<b>CO5</b>	<b>1</b>	<b>3</b>	<b>2</b>	<b>3</b>	<b>1</b>

**1 – Low, 2 – Moderate, 3 – High (Preferably use 2 or 3 levels)**

<b>SEMESTER: I</b> <b>PART: III</b>	<b>22UBCACP15: PROGRAMMING IN C LAB</b>	<b>CREDIT: 2</b> <b>HOURS: 3</b>
--	---	-------------------------------------

### **COURSE OBJECTIVES**

1. Apply the specification of syntax rules for numerical constants and variables, data types.
2. Usage of Arithmetic operator, Conditional operator, logical operator and relational operators and other C constructs.
3. Write C programs using decision making, branching, looping constructs
4. Apply and Write C programs to implement one dimensional and two dimensional arrays
5. Writing programs using functions

### **LIST OF EXPERIMENTS**

1. Write a program to find the largest number and smaller number by using if statement
2. Write a program to convert the decimal to binary conversion by using while statement.
3. Write a program to count the positive, negative & zero numbers.
4. Write a program to check whether a given number is a prime or not.
5. Write a program to display the Fibonacci series.
6. Write a program to concatenate two strings without using string library function.
7. Write a program to count the number of vowels, consonants, and digits in a line of Text.
8. Write a program to reverse a String.
9. Write a program to design the calculator functions as
  - a) Addition
  - b) Subtraction &
  - c) Multiplication function.
10. Write a program to find the factorial of a number using recursion.
11. Write a program for ascending order of given N Numbers.
12. Write a program to separate odd and even numbers using file.

### **COURSE OUTCOMES**

After completing the Course successfully, the student will be able to

1. Read, understand and trace the execution of programs written in C language.
2. Write the C code for a given algorithm.
3. Implement Programs with pointers and arrays, perform pointer arithmetic, and use the pre-processor.
4. Write programs that perform operations using derived data types.
5. Know concepts in problem solving

**OUTCOME MAPPING**

<b>CO/PO</b>	<b>PO1</b>	<b>PO2</b>	<b>PO3</b>	<b>PO4</b>	<b>PO5</b>
<b>C01</b>	<b>2</b>	<b>3</b>	<b>3</b>	<b>2</b>	<b>2</b>
<b>C02</b>	<b>2</b>	<b>2</b>	<b>2</b>	<b>3</b>	<b>1</b>
<b>C03</b>	<b>2</b>	<b>3</b>	<b>3</b>	<b>3</b>	<b>2</b>
<b>C04</b>	<b>1</b>	<b>3</b>	<b>2</b>	<b>2</b>	<b>2</b>
<b>C05</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>3</b>	<b>1</b>

1 – Low, 2 – Moderate, 3 – High (Preferably use 2 or 3 levels)

<b>SEMESTER: II</b> <b>PART: III</b>	<b>22UBCAC23: C++ AND DATA STRUCTURES</b>	<b>CREDIT:4</b> <b>HOURS:5</b>
---	---	-----------------------------------

**COURSE OBJECTIVES**

1. To Understand the Principles of Object Oriented Programming
2. To understand the concepts of Classes and Objects
3. To Understand the Concepts of Inheritance
4. To Understand the Concepts of Data Structures
5. To Understand in developing C++ programs

**UNIT I: BASICS OF OOP AND C++****Hours:15**

Object Oriented Programming Concepts – Benefits and Applications of OOP – C++ Program structure, Program Statements, Classes – Creating, Compiling and Linking Source file – Keywords, Identifiers and Constants – Data types: Basic, User defined and derived – Declaration and Dynamic Initialization of Variables, Reference Variables – Operators: Scope resolution, Member Dereferencing, Memory management and Type Cast - Manipulators – Expressions and their Types – Implicit Conversions, Operator Overloading, Operator Precedence.

**UNIT II: CONTROL STRUCTURES & ARRAYS.****Hours:15**

Control Structures: If, If...else, switch, do..while, while, for statements, Functions – The Main function, Function Prototyping, Call by Reference, Return by Reference, Inline functions, Classes and Objects – Specifying a Class, Defining Member Functions, Making an Outside function Inline, Nesting of Member functions, Arrays within a Class, Arrays of Objects, Objects as function Arguments, Friendly functions, Returning Objects.

**UNIT III: CONSTRUCTORS & FILES****Hours:15**

Constructors Parameterized Constructors, Multiple Constructors in a class, Copy constructor and Destructors -Defining Operator Overloading, Overloading Unary and Binary Operators, Overloading, String manipulation using Operators, Rules for overloading operators, Type Conversions. Inheritance: Defining derived classes.-Single, Multilevel , Hierarchical and multiple Inheritance-Pointers, Pointers to derived classes. Files: Opening and closing file, detecting End-of-files.

**UNIT IV: FUNDAMENTALS OF DATA STRUCTURES****Hours:15**

Stack(Array)-Operations-Applications of Stack(Infix and Postfix)- Queue(Array)-operations-Linked list (Singly, circular, Doubly)- Applications of List(Polynomial Addition) Trees: Binary Trees –Binary Search Tree- Operations - Recursive Tree Traversals.

**UNIT V: SORTING & SEARCHING****Hours:15**

Graph - Definition, Types of Graphs, Graph Traversal –Dijkstras shortest path-DFS and BFS-Sorting-Heaps-Quick sort-Merge sort-Bubble sort-searching-Binary search.

**COURSE OUTCOMES**

After completing the Course successfully, the student will be able to

1. To learn the basic concepts Object oriented programming.
2. To learn the control structures and arrays.
3. To implementing the constructors & File opening and closing.
4. To learn the fundamentals of stack & Queue operations.
5. To learn the concepts of graphs, sorting & searching methods.

**TEXT BOOKS (In API Style)**

1. E Balagurusamy , 2014,Tata McGraw Hill,6th Edition, “Object Oriented Programming with C++”.
2. Ellis Horowitz, Sartaj Sahni,2008, Galgotia Publications “ Fundamentals of Data Structure”.

**SUPPLEMENTARY READINGS**

1. Reema Thareja , 2015, Oxford University Press , “Object Oriented Programming with C++”.
2. Balagurusamy, Tata McGraw Hill Edition “C++ programming”.

**OUTCOME MAPPING**

<b>CO/PO</b>	<b>PO1</b>	<b>PO2</b>	<b>PO3</b>	<b>PO4</b>	<b>PO5</b>
<b>CO1</b>	<b>2</b>	<b>3</b>	<b>2</b>	<b>1</b>	<b>2</b>
<b>CO2</b>	<b>1</b>	<b>2</b>	<b>2</b>	<b>3</b>	<b>1</b>
<b>CO3</b>	<b>2</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>2</b>
<b>CO4</b>	<b>1</b>	<b>2</b>	<b>2</b>	<b>2</b>	<b>1</b>
<b>CO5</b>	<b>2</b>	<b>3</b>	<b>3</b>	<b>3</b>	<b>1</b>

**1 – Low, 2 – Moderate, 3 – High (Preferably use 2 or 3 levels)**

<b>SEMESTER:II</b> <b>PART: III</b>	<b>22UBCAP24 : C++ &amp; DATA STRUCTURE LAB</b>	<b>CREDIT: 2</b> <b>HOURS: 3</b>
--	---	-------------------------------------

**COURSE OBJECTIVES**

1. To Impart Practical Training in C++ Programming Language

**LIST OF EXPERIMENTS**

- 1) Implementing classes, object, constructors and member functions for calculating area and perimeter of a circle.
- 2) Implementing function overloading(Find area/volume of rectangle, circle, sphere, cylinder, cone etc).
- 3) Implementing operator over loading( Addition, subtraction, multiplication of matrices)
- 4) Implementing single , multiple , hierarchical inheritance.
- 5) Implementing sequential file operations using error handling functions.
- 6) Implementing PUSH, POP operations of stack using Arrays.
- 7) Implementing add, delete operations of a queue using Arrays.
- 8) Implementing Infix to postfix conversion of an expression using stack.
- 9) Implementing Binary tree recursive traversals (in-order, pre-order, and post-order).
- 10) Implementing Polynomial addition using linked list.

**OUTCOME MAPPING**

	<b>PO1</b>	<b>PO2</b>	<b>PO3</b>	<b>PO4</b>	<b>PO5</b>
<b>CO1</b>	<b>2</b>	<b>3</b>	<b>3</b>	<b>3</b>	<b>3</b>
<b>CO2</b>	<b>2</b>	<b>2</b>	<b>2</b>	<b>3</b>	<b>2</b>
<b>CO3</b>	<b>3</b>	<b>2</b>	<b>3</b>	<b>3</b>	<b>2</b>
<b>CO4</b>	<b>2</b>	<b>3</b>	<b>2</b>	<b>3</b>	<b>2</b>
<b>CO5</b>	<b>2</b>	<b>2</b>	<b>3</b>	<b>3</b>	<b>3</b>

**1 – Low, 2 – Moderate, 3 – High (Preferably use 2 or 3 levels)**

<b>SEMESTER:II</b> <b>PART: III</b>	<b>INTERNAL ELECTIVE-I</b> <b>22UBCAE26 :1</b> <b>COMPUTER ORGANIZATION &amp; ARCHITECTURE</b>	<b>CREDIT: 3</b> <b>HOURS: 3</b>
--	--	-------------------------------------

**COURSE OBJECTIVES**

1. To understand the basic concepts of instruction and its essentials.
2. To Understand the concept of programmed control.
3. To learn how to implement micro operations & instruction formats..
4. To Explain the Input , output controls .
5. To understand the concept of memory concepts.

**UNIT I : Basic Computer Organization and Design****Hours:9**

Instruction Codes: Stored program Organization, Indirect address – Computer Registers: Common bus system – Computer Instructions: Instruction set Completeness – Timing and Control – Instruction Cycle: Fetch and decode, Determine the type of Instruction, Register reference Instructions- Memory reference Instructions – Input Output and Interrupt: I/O Configuration, I/O Instructions, Program Interrupt, Interrupt Cycle.

**UNIT II: Microprogrammed Control****Hours:9**

Control memory – Address sequencing: Conditional Branching, Mapping of Instruction, Subroutines – Microprogram Example: Computer Configuration, Microinstruction formats, Symbolic Microinstructions, The fetch routine, Symbolic Microprogram, Binary Microprogram – Design of Control UNIT: MicroProgram Sequences.

**UNIT III: Central Processing UNIT****Hours:9**

General Register Organization: Control word, Examples of Microoperations – Instructionformats : Three-Address, Two-Address, One-Address and Zero-Address Instructions, RISC Instructions – Addressing Modes: Numerical Example.

**UNIT IV: Input-Output Organization****Hours:9**

Peripheral Devices: ASCII Alphanumeric Characters – Input-Output Interface: I/O Bus and Interface modules, I/O versus Memory Bus, Isolated versus Memory Mapped I/O, Example of I/O Interface – Asynchronous Data Transfer: Strobe Control, Handshaking, Asynchronous Serial Transfer, Asynchronous Communication Interface, First-In, First-Out Buffer – Modes of Transfer – DMA, DMA Controller.

**UNIT V: Memory Organization****Hours:9**

Memory Hierarchy, Main memory: RAM, ROM, Memory Address map, Memory Connection to CPU – Auxiliary Memory - Associative Memory – Cache & virtual Memory.

**COURSE OUTCOMES**

After completing the Course successfully, the student will be able to

1. To learn the computer instructions and bus system.
2. To learn about the addressing modes.
3. To learn about the Input and output controls.
4. To learn about the memory and its types.

**Text Books (In API Style)**

1. Morris Mano M. , 2011, Prentice Hall of India Private Limited,  
“Computer System Architecture” New Delhi .

**Supplementary Readings**

1. William Stallings, 2010, Pearson publications, “Computer Organization and architecture”,

**OUTCOME MAPPING**

<b>CO/PO</b>	<b>PO1</b>	<b>PO2</b>	<b>PO3</b>	<b>PO4</b>	<b>PO5</b>
<b>CO1</b>	<b>1</b>	<b>3</b>	<b>3</b>	<b>2</b>	<b>2</b>
<b>CO2</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>2</b>	<b>1</b>
<b>CO3</b>	<b>3</b>	<b>2</b>	<b>3</b>	<b>3</b>	<b>1</b>
<b>CO4</b>	<b>1</b>	<b>3</b>	<b>2</b>	<b>2</b>	<b>1</b>
<b>CO5</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>2</b>	<b>1</b>

**1 – Low, 2 – Moderate, 3 – High (Preferably use 2 or 3 levels)**

<b>SEMESTER:II</b>	<b>INTERNAL ELECTIVE- I</b>	<b>CREDIT: 3</b>
<b>PART: III</b>	<b>22UBCAE26 : 2 COMPUTER GRAPHICS</b>	<b>HOURS: 3</b>

### **COURSE OBJECTIVES**

1. To understand the basic concepts of drawing algorithms.
2. To understand the concept of Attributes & 2D transformations
3. To understand clipping concepts & its types.
4. To understand the concepts of 3D transformations.
5. To understand the surface detection methods.

#### **UNIT I : Overview of Computer Graphics**

**Hours: 9**

Video Display Devices- Raster Scan System- Random Scan Systems- Hard Copy Deices- Graphic Software- Line Drawing Algorithms: DDA- Bresenham's Line -Circle Generating Algorithms

#### **UNIT II: Attributes & Two Dimensional Transformations**

**Hours: 9**

Line Attributes- Curve Attributes-Color And Gray Scale Level- Area Fill Attributes- Character Attributes- Inquiry Functions- Basic Transformations - Composite Transformation – Other transformation

#### **UNIT III: Two Dimensional Viewing & Clipping**

**Hours: 9**

The Viewing Pipeline- Window To Viewport Transformation –Clipping Operations- Point Clipping- Line Clipping: Cohen Sutherland- Liang Barsky-Sutherland Hodgeman Polygon Clipping- Text Clipping- Exterior Clipping- Logical Classification Of Input Devices- Interactive Picture Construction

#### **UNIT IV: Three Dimensional Transformations & Clipping**

**Hours: 9**

Translation-Rotation-Scaling-Viewing Pipeline- Viewing Coordinates- Projections - View Volumes and General Projection Transformation- Clipping

#### **UNIT V: Visible Surface Detection Methods**

**Hours: 9**

Classification of Visible Surface Detection Algorithms - Back Face Detection - Depth Buffer Method - A Buffer Method - Scan Line Method - Depth Sorting Method- BSP Tree Method -Area Sub Division Method - Octree Methods - Ray Casting Method

### **COURSE OUTCOMES**

After completing the Course successfully, the student will be able to

1. Learn about the basics of graphics drawings
2. To learn about the attributes & its transformations.
3. To learn about the clipping & its types.
4. To learn about the 3D transformations.
5. To learn about the surface detection methods.

**Text Books**

1. Donald Hearn and Hearn and M.Pauline ,2012, Pearson 2<sup>nd</sup> edition  
“Computer Graphics(C version).

**Supplementary Readings:**

1. Edward Angel, Pearson Edition, 5<sup>th</sup> Edition, “Interactive Computer Graphics–A top down approach using Open GL”.
2. Peter Shirley , 2009, Steve Marschner, Cengage Learning, Indian Edition  
“Computer Graphics”.

**OUTCOME MAPPING**

<b>CO/PO</b>	<b>PO1</b>	<b>PO2</b>	<b>PO3</b>	<b>PO4</b>	<b>PO5</b>
<b>CO1</b>	<b>2</b>	<b>3</b>	<b>3</b>	<b>3</b>	<b>2</b>
<b>CO2</b>	<b>1</b>	<b>2</b>	<b>2</b>	<b>3</b>	<b>1</b>
<b>CO3</b>	<b>3</b>	<b>3</b>	<b>3</b>	<b>3</b>	<b>2</b>
<b>CO4</b>	<b>1</b>	<b>3</b>	<b>2</b>	<b>2</b>	<b>1</b>
<b>CO5</b>	<b>1</b>	<b>3</b>	<b>3</b>	<b>3</b>	<b>1</b>

**1 – Low, 2 – Moderate, 3 – High (Preferably use 2 or 3 levels)**

<b>SEMESTER:II</b> <b>PART: III</b>	<b>INTERNAL ELECTIVE-I</b> <b>22UBCACE26:3</b> <b>INTERNET AND ITS APPLICATIONS</b>	<b>CREDIT:3</b> <b>HOURS: 3</b>
--	---	------------------------------------

**COURSE OBJECTIVES**

1. Illustrate basic concepts of Internet.
2. Understand Apply the necessary of Internet Explorer.
3. Analyze, design and implement Email system.
4. Demonstrate the Hyper Text Markup languages
5. To learn the E-marketing & its usage.

**UNIT – I: Fundamentals of Internet****Hours:9**

Introduction to Computers Programming Language types History of Internet Personal Computers, History of World Wide Web- Micro software .NET Java-Web resources.

**UNIT – II: Web Browsers****Hours:9**

Web Browsers- Internet Explorer- connecting to Internet Features of Internet explorer6 Searching the Internet- online help and tutorials- File Transmission Protocol (FTP) Browser settings.

**UNIT – III: E-Mail****Hours:9**

Attaching a file, Electronic mail Creating an E-mail id Sending and Receiving mails- attaching a file- Instance messaging- other web browsers.

**UNIT – IV: HTML****Hours:9**

Introduction to HTML headers - Linking- Images-special characters and line breaks- unordered lists- simple HTML programs.

**UNIT – V: E-Marketing****Hours:9**

E-marketing consumer tracking Electronic advertising search engine-CRM- credit card Payments- Digital cash – e wallets – smart card.

**COURSE OUTCOMES**

After completing the Course successfully, the student will be able to

1. Explain basic usages of internet and its applications.
2. Define and demonstrate the use of Web Browsers.
3. To Explain the E-Mail applications.
4. To demonstrate the HTML & its tags.
5. To Know the E-Marketing and its advertisements.

**Text Books (In API Style)**

1. P.J. Deital and A.B. Goldberg , PHI, third Edition “Internet and World Wide Web”.

**Supplementary Readings**

1. Harley hahn , Tata McGraw hill “The Internet- Complete Reference”.
2. P.Rizwan Ahmed, 2014, , Margham Publication, ” Internet and its Applications”

**OUTCOME MAPPING**

<b>CO/PO</b>	<b>PO1</b>	<b>PO2</b>	<b>PO3</b>	<b>PO4</b>	<b>PO5</b>
<b>CO1</b>	<b>2</b>	<b>3</b>	<b>3</b>	<b>3</b>	<b>2</b>
<b>CO2</b>	<b>1</b>	<b>2</b>	<b>2</b>	<b>2</b>	<b>1</b>
<b>CO3</b>	<b>3</b>	<b>1</b>	<b>3</b>	<b>1</b>	<b>3</b>
<b>CO4</b>	<b>2</b>	<b>3</b>	<b>3</b>	<b>3</b>	<b>3</b>
<b>CO5</b>	<b>1</b>	<b>3</b>	<b>3</b>	<b>1</b>	<b>3</b>

**1 – Low, 2 – Moderate, 3 – High (Preferably use 2 or 3 levels)**