Teaching Plan
Department of Computer Science & BCA
Session (2022-23)
Even Semesters

Term I From commencement of class to 1st Internal Assessment
Term II 1st Internal Assessment to 2nd Internal Assessment
Term III 2nd Internal Assessment to End Semester Examination

Teaching plan : 2022-23 (Even Semesters)

Alok Haldar
Dept. of Computer Science & BCA

Semester - IV		
Syllabus allotted		**BCA2296 : C++ Lab** **BCA2297 : Gr.-A : OS Lab** **Gr.-B : Networking Lab**
C++ Lab	Lec No	**Based on C++ Theory.**
		Term I
	01	Write a C++ program to design a class cylinder and the operations on cylinder as follows : a) Calculate the volume. B) Calculate the surface area. c) Find the area of cylinder base. d) Set the radius , height and the center of the base.
	02	Write a C++ program to find the **area** and **perimeter** of **square** and **triangle** by creating the class **shape**, **square**, **triangle** and required data members and functions like input **values()**, **area()**, **perimeter().**
	03	Write a C++ program to find the **area** and **perimeter** of triangles by creating the class **triangles, isosceles, equilateral** and required data members and functions like **values()**, **area()**, **perimeter().**
	04	Write a C++ program that uses inline function to count the number of vowels,consonant,integers in a string.
	05	Write a C++ program to design a class **Complex** with data members for real and imaginary part. Provide default and parameterized constructors and member functions to **get(), set(), display(), add(), subtract(), multiply(), and divide() two complex numbers.**
	06	Write a C++ program to read two times and add them .To create a class to represent time (Hour,Minute,Second) in 24 Hour Format.
	07	Write a C++ program to create a class **point** that consists of x and y co-ordinates and using this design a class **Linesegment** which consists of two points**.** Use appropriate constructors and destructors. Write a function **compare-lines** to check whether two line segments are parallel or perpendicular or not.
	08	Write a C++ program to check the following operations on the class **STACK** as : a) Current Status. B) Push an item. C) pop an item, d) Empty stack or not.
	09	Write a C++ program to check the following operations on the class **QUEUE** as : a) insert element to the queue. B) Delete an element from the queue. C) Destroy the

		queue.
	10	Write a C++ program to use of copy-constructor in a user defined class.
		Term II
	11	Write a C++ program to overload ' + , - , * ' operator to addition, subtraction and multiplication of two matrices same order
	12	Write a C++ program to overload input and output **operators >> and <<** to take complex numbers from user. Now overload addition and multiplication operators **+ and *** to demonstrate **addition** and **multiplication** of two complex numbers.
	13	Write a C++ Program to overloads an assignment(=) operator for user-defined class.
	14	Write a C++ Program overloads the pre-increment and post-increment operators for user-defined objects.
	15	Write a C++ program to demonstrate single inheritance that uses both public and private access specifier.
	16	Write a C++ program to demonstrate multiple inheritance that uses both public and private access specifier.
	17	Write a C++ program to demonstrate hybrid inheritance that uses both public and private access specifier.
		Term III
	18	Write a C++ program to make **function template** program which can **swap two variables** which may be int, float or character.
	19	Write a C++ program to make **function template** program which can Sort n numbers using **bubble sort/selection sort/insertion sort/merge sort/quicksort method.**
	20	Write a C++ program to write a function **matmul()** using **function template** and **multiply two matrices**. Invoke the functions to operate on **two integer matrices and two float matrices**. Write a display method to show the result.
	21	Write a C++ program to create a **template class for stack** and suitable member functions to show the operation of stack.
	22	Write a C++ program to implement runtime Polymorphism. Use proper constructor data member and functions.
OS Lab & Socket Programming	**Lab**	**Gr.-A** Shell programming : creating a script, making a script executable, shell syntax. Process : starting new process, replacing a process image, duplicating a process image , waiting for a process. Signal : signal handling, sending signals, signal interface, signal sets. Semaphore : programming with semaphores (use functions semctl, semget, semop, set_semvalue, del_semvalue,semaphore_p, semaphore_v). POSIX Threads : programming with pthread functions(viz. pthread_create, pthread_join, pthread_exit, pthread_attr_init, pthread_cancel) Inter-process Communication : pipes(use functions pipe, popen, pclose), named pipes(FIFOs, accessing FIFO. **Gr.-B** Socket Programming : Simple Application using elementary socket system calls in client/server model in unix/linux using c language. TCP/UDP example using only the elementary socket system calls.
		Term I
	01	Write a shell script to evaluate the arithmetic expression M=[(a-b)+(c+d)*e/f] where a,b,c,d,e and f are integer number which is supplied by the user.
	02	Write a shell script to sum of natural number i.e 1+2+3+...+n.
	03	Write a shell script to calculate the factorial of a given number.

Networking Lab		
	04	Write a shell script to check if a number is prime or not.
	05	Write a shell script to display first n fibonacci numbers where n is read from keyboard.
	06	Write a shell script to check whether a number is armstrong or not.
	07	Write a shell script to sum of the digits of a given integer.
	08	Write a shell script to print the multiplication table of any number.
	09	Write a shell script to check whether a string is palindrome or not.
	10	Write a shell script to find out the roots of the quadratic equation i.e $ax^2+bx+c=0$.
	11	Write a shell script to reverse of a given number and the number is supplied from user.
	12	Write a shell script to display the message "Good Morning" or "Good afternoon" or "Good evening according to system time.
	13	Write a shell script to check whether a number is palindrome or not.
	14	Write a shell script to check whether a number is perfect number or not.
	15	Write a shell script to draw the following pattern : 1 1 2 1 2 3
	16	Write a shell script to find out the HCF and LCM of two given numbers.
		Term II
	17	Write a shell script to generate all prime numbers up to a given numbers.
	18	Write a shell script to to generate all perfect numbers up to a given numbers.
	19	Write a shell script / program that takes a file name as command line argument and searches the output whether it exists. If exists its RWX permission and displayed also.
	20	Write a shell script to calculate the number files and directories at your current directory.
	21	Write a shell script to generate all non-fibonacci numbers up to a given range.
	22	Write a shell script / program to find out the values of the series 1!+2!+3!+....+n!
	23	Write a shell script / program to reverse a string. e.g RAM IS A GOOD BOY is BOY GOOD A IS RAM. (Note that a word is not reversed.)
	24	Write a shell script to convert a decimal number to its equivalent binary number.
	25	Write a shell script to calculate 1! +2! +3! .... +n!
	26	Write a shell script to search a number from a given set of numbers.
	27	Write a shell script / program to find out the maximum number from the given set of numbers.
	28	Write a shell script to search a number using Binary Search.
	29	Write a shell script to sort of n numbers using Bubble Sort.
	30	Write a shell script / program to calculate the sum of natural number and the number supplied by command line argument.
		Term III
	31	Write a Program to create a process in unix.
	32	Write a Program to create a child process in unix.
	33	Write a c program to implement both client and server with exchange string using TCP
	34	Write a program that receives an IP address and determines the class of the IP address.
	35	Write a socket program in TCP method for bidirectional data transfer between client and server.
	36	Write a program to detect the IP address.
	37	Write s socket program to send a data from server to client as user like to input using UDP
	38	Write a program in C to transfer file from server to client using TCP socket.

Semester - IV

Syllabus allotted	BCA3202 : Advanced DBMS BCA3294 : Graphics & Multimedia Lab BCA3295 : Industrial Project

BCA3202 (Elective-2): Advanced Database Management System

UNIT-I : Database Design: Multivalued dependencies, theory of normalisation-4NF, 5NF, 6NF DKNF

UNIT-II : ANSI SQL2: DDL, DML, constraints and assertions, views, database security.

UNIT-III : Transaction processing, concurrency control, Recovery management. Transaction model properties, lock base protocols, Two-phase locking, Live – Lock, Time-Stamp Protocol.

UNIT-IV : Brief introduction to distributed database, temporal database and object-oriented database.

UNIT-V : Embedded SQL & Applications.

BCA3294 : Graphics & Multimedia Lab

1. Point plotting, line & regular figure algorithms
2. Raster scan line & circle drawing algorithms
3. Clipping & Windowing algorithms for points, lines & polygons
4. 2-D / 3-D transformations
5. Simple fractals representation, Demonstrate the properties of the Bezier curves.
6. Filling algorithms , Clip line segments against windows
7. Web document creation using Dreamweaver.
8. Creating Animation using Flash.

BCA 3295 : Project(Industrial)

Term I	
01	Database Design, Functional dependencies, Multivalued dependencies definition with examples. Fourth Normal Form(4NF)
02	Join dependencies with Fifth Normal Form(5NF) definition with example
03	Inferences rule for Functional dependencies with examples.
04	Determine closure under Functional dependencies(F+)
05	Equivalence of Sets of Functional dependencies with examples. To find out the minimal sets of Functional dependencies with examples.
06	Dependencies Preservation of Functional dependencies with examples. Non-additive (Lossless) Join Property of a Decomposition.
07	Testing for Non-additive Join property.
08	Domain key constraints , Key constraints, DKNF,6NF with examples.
Term II	
09	DDL,DML,DCL,Constraints and assertions, Views, Database Security.
10	Introduction to Transaction Processing Concepts, Properties of transaction, Serializability,
11	Concurrency Control, Why Concurrency Control is Needed, Two-Phase Locking Techniques for Concurrency Control,
12	Why recovery is needed, Desirable Property of Transaction(ACID Property), Serializability, Locking Mechanisms, Two Phase Commit Protocol.
13	Concurrency Control based on Time-stamp ordering.
Term III	
12	Basic Concepts of Distributed Database, Reliability and Availability.
13	Types of distributed database, Distributed Database architecture, Advantages of Distributed Database.

<table>
<tr><td></td><td>14</td><td>Query Processing and Optimization in Distributed Databases.</td></tr>
<tr><td></td><td>15</td><td>Temporal and Object-Oriented database, Embedding SQL & applications.</td></tr>
<tr><td rowspan="16">Graphics & Multimedia Lab</td><td colspan="2">Term I</td></tr>
<tr><td>01</td><td>Write a Program to draw basic graphics construction like line, circle, arc, ellipse and rectangle.</td></tr>
<tr><td>02</td><td>Write a Program to draw a line using DDA algorithm.</td></tr>
<tr><td>03</td><td>Write a Program to draw a line using Bresenham’s Line Drawing algorithm.</td></tr>
<tr><td>04</td><td>Write a Program to draw a circle using mid-point algorithm.</td></tr>
<tr><td>05</td><td>Write a Program to draw a circle using Bresenham’s circle drawing algorithm.</td></tr>
<tr><td>06</td><td>Write a Program to draw an ellipse using mid-point algorithm.</td></tr>
<tr><td>07</td><td>Write a Program to draw an equilateral triangle without using any inbuilt functions.</td></tr>
<tr><td>08</td><td>Write a program to draw three concentric circle of different color using any circle drawing algorithm without using any inbuilt functions.</td></tr>
<tr><td colspan="2">Term II</td></tr>
<tr><td>09</td><td>Write a Program to perform the following 2D transformation on a triangle(menu driven program)
i) Translation w.r.t an origin.
ii) Rotation w.r.t an origin.
iii) Scaling w.r.t an origin.</td></tr>
<tr><td>10</td><td>Write a Program to rotate a line about 45 with respect to origin.</td></tr>
<tr><td>11</td><td>Write a Program to fill a rectangle using any standard filling algorithm.</td></tr>
<tr><td>12</td><td>Write a Program to implement Cohen-Sutherland line clipping algorithm.</td></tr>
<tr><td colspan="2">Term III</td></tr>
<tr><td>13</td><td>Write a Program to fill the figure with appropriate color
<table><tr><td>RED</td><td>BLUE</td></tr><tr><td>GREEN</td><td>CYAN</td></tr></table></td></tr>
<tr><td>14</td><td>Write a Program to draw Bar Chart of student’s result of last 5 years.</td></tr>
<tr><td>15</td><td>Write a Program to display a moving ball.</td></tr>
<tr><td rowspan="5">Project (Industrial)</td><td></td><td>Term I</td></tr>
<tr><td>1</td><td>Choose the name of Project, Problem Definition, Gathering the information, Prepare E-R diagram and DFD. Total project divided with modules.</td></tr>
<tr><td></td><td>Term II</td></tr>
<tr><td>2</td><td>Implementation of the Project using Latest Software. Finally Testing of the Whole Project through valid data.</td></tr>
<tr><td></td><td>Term III</td></tr>
<tr><td>3</td><td>How to prepare the documentation of Project.</td></tr>
</table>

Teaching plan : 2022-23 (Even Semester)
Dr. Samiran Acharyya
Dept. of BCA

Semester II		
Syllabus allotted	**Lec No.**	**Term I**
Paper GE-02 (T)		**Introduction to Accounting**
	01	Definition,
	01	scope of accounting
	02	Accounting as financial information system
	03-05	Accounting Standards
	06-07	Accounting Principles
		Accounting procedure
	08	Transaction/event, Classification of accounts
	09-12	voucher
	13-15	Preparation of vouchers
	16-19	Journal/ subsidiary books
	20-22	Types of subsidiary books Ledger accounts and trial balance
		Term II
		Depreciation accounting, Capital & Revenue
	23-25	Expenditure & receipts
	26	Methods of depreciations
	27	-Straight-line method
	28	- Reducing method
	29	- Sinking fund method
	30	- Annuity Method
	31	- Machine hour rate method
	32	-Depletion method
		Term III
		Company Final Accounts
	33-35	Preparation of trading a/c
	36-37	Profit & Loss a/c
	38	Balance sheet
	39-40	Accounting for issue of shares

Semester IV		
Syllabus allotted	**Lec No.**	**Term I**
Game Theory (Paper -)	01	Introduction
	02	Analytical discussion of the theorem
	03	Formation of the pay-off matrix
	04-05	Mixed strategy
	06-07	Graphical Solution of 2xn or mx2 games
		Term II
	08-09	Dominance property
	10-11	General rule for dominance
	12	Modified dominance property
	13	Fundamental theorem of a rectangular game
	14	Solution of a game problem by matrix method
	15-16	Solution of the game problem by iterative method

Department of Computer Science and BCA

Syllabus Distribution and Teaching Plan
Even Semester, Session: 2022-2023

Term I: *Commencement of classes to 1st internal*; Term II: *1st internal to 2nd internal*;
Term III: *2nd internal to ESE preparatory break.*

Semester II

FACULTY NAME	ALLOTATED PAPERS	TEACHING PLAN
Sakhi Bandyopadhyay	CC-04(T): DataStructure(3L, 1T) CC-04(P): Data Structure Lab(2P) SL/AL: Special Classes for 2nd Sem (1T)	**Term I** **CC-04 (T):** Lecture 1: Basics of data structures, Array, Sparse Matrix etc. Lecture 2: Linked List, Singly linked lists and their implementations. Lecture 3: Double linked lists and their implementations. Lecture 4: Circular linked lists and their implementations. Lecture 5: Implementing Stack, Infix, Prefix and Postfix expressions. Lecture 6:Utility and conversion of Infix, Prefix and Postfix expressions from one to another,Applications of stack,Limitations of Array representation of stack Lecture 7: Normal andCircular representation of Stack in Lists Lecture 8: Queue, Different types of queues. Lecture 9: Tutorial Lecture 10: Tutorial Lecture 11: Tutorial **CC-04 (P):**Practical on Array, Linked List, Stack and Queue. Program 3 and 4. **Term II** **CC-04 (T):** Lecture 12: Recursion Lecture 13: Trees: Basics, Binary Trees Lecture 14: Binary Search Trees Lecture 15: Threaded Binary Trees Lecture 16: AVL Tree Lecture 17: Tree traversaltechniques, Heap Sort Lecture 18: Tutorial Lecture 19: Tutorial **CC-04 (P):**Practical on Recursion and Tree. Program 8, 9, 10, and 11. **Term III** **CC-04 (T):** Lecture 20: Different Searching Algorithms Lecture 21: Selection Sort, Insertion Sort, Bubble Sort Lecture 22: Quick Sort and Merge sort, Comparison of SortingTechniques. Lecture 23: Hashing Lecture 24: Tutorial

		CC-04 (P): Practical on Searching and Sorting. Program 1 and 2.

Semester IV

FACULTY NAME	ALLOTATED PAPERS	TEACHING PLAN
Sakhi Bandyopadhyay	BCA 2202: Operating System (1L) BCA 2204: Software Engineering (3L)	**Term I** **BCA 2202:** **(System Structure)** Lecture 1:System Structure: Computer system operation, I/O structure, storage structure Lecture 2: Storage hierarchy,different types of protections Lecture 3: Operating system structure (simple, layered, virtual machine), O/Sservices, system calls. **BCA 2204:** Lecture 1: Introduction Lecture 2:System Development Life Cycle Lecture 3:Waterfall Model Lecture 4:Spiral Model Lecture 5:Feasibility Analysis Lecture 6:Cost- Benefit Analysis Lecture 8:COCOMO model Lecture 9: SRS Lecture 10: DFD Lecture 11: Data Dictionary Lecture 12: ERD Lecture 13: System Design Lecture 14: Decision Tree & Table Lecture 15: Object Oriented Approach Lecture 16: Coding Lecture 17: Documentation **Term II** **BCA 2202:** **(Deadlocks: Properties, Conditions, Detection, Prevention, Avoidance, and Recovery).** Lecture 4: System model Lecture 5: Deadlock characterization, Methods for handling deadlocks Lecture 6: Deadlockprevention, Deadlock avoidance Lecture 7:Deadlock detection, Recovery from deadlock **BCA 2204:** Lecture 4: Structured Programming Lecture 5: OO Programming Lecture 6: System Testing Lecture 7:Different types of Testing Lecture 8: Reliability Assessment Lecture 9:Validation &Verification Metrics Lecture 10: Cohesion & Coupling Lecture 11:Monitoring & Control **Term III** **BCA 2202:** **(Protection & Security: security problem, authentication,**

		system threats). Lecture 8:Goals of protection, domain of protection Lecture 9: Security problem, Authentication, one time password, Lecture 10: Program threats, System threats Lecture 11: Threat monitoring, Encryption **BCA: 2204:** **Software Project Management** Lecture 12: Project Scheduling Lecture 13: Staffing Lecture 14: Software Configuration Management Lecture 15: Quality Assurance Lecture 16: Project Monitoring Lecture 17:CASE TOOLS Lecture 18:CASE TOOLS

Semester VI

FACULTY NAME	ALLOTATED PAPERS	TEACHING PLAN
Sakhi Bandyopadhyay	BCA 3295: Project (1L)	**Term I** BCA 3295: Project Selection: Project Title, Group formation, System and Technology Requirement Analysis, etc., Project Blueprint: DFD, ERD, and Design. **Term II** BCA 3295: Project Coding and Implementation: Front End and Back End Development. Project Testing. **Term III** BCA 3295: Documentation of the Project (DOCX, PDF, and PPT) and Preparation for Final Examination.

Instructor Name- Anudyuti Ghorai	
Subject	Teaching Plan
Semester II	
Data Structure Lab (BCACC4P)	1. WAP to calculate factorial and to compute the factors of a given no. using iteration 2. WAP to calculate factorial and to compute the factors of a given no. using recursion 3. WAP to display Fibonacci series using iteration. 4. WAP to display Fibonacci series using recursion. 5. WAP to calculate GCD of 2 number without recursion. 6. WAP to calculate GCD of 2 number with recursion. 7. WAP to implement Diagonal Matrix using one-dimensional array. 8. WAP to implement Lower Triangular Matrix using one-dimensional array. 9. WAP to implement Upper Triangular Matrix using one-dimensional array. 10. Perform Stack operations using Array implementation. 11. Perform Queues operations using Circular Array implementation. 12. Perform Stack operations using Linked List implementation.
Semester IV	
COMPUTER NETWORK (BCA-2205)	**Term I:** Introduction: Data communication- fundamental characteristics, components. Types of data flow. Types of connection. Protocol. Protocol Stack OSI model- Physical Layer, Data link Layer, Network Layer, Transport Layer, Session Layer, Presentation Layer, Application Layer. Physical Layer: Physical Topologies- Mesh, Star, Bus, Ring, Hybrid. Categories of Network- LAN, WAN, MAN, WAN. Analog Signals & Digital Signals- Data Transmission- Bandwidth, Transmission of signals, Attenuation, Nyquist bit rate, Shannon capacity, Latency, Transmission time, Bandwidth delay product. Digital to Digital conversion, Analog to Digital conversion, Digital to Analog conversion, Analog to Analog conversion Multiplexing: FDM, WDM & TDM. Transmission Media: Guided Media, Unguided Media(Wireless). Circuit Switching. **Term II:** Data Link Layer: Error detection and correction: - Type of Errors, Detection, Error Correction, CRC, hamming distance, Hamming code, parity check code. Framing. Data Link Control and Protocols: - Flow and Error control, Stop-and Wait ARQ, Go-Back, N ARQ, Selective Repeat ARQ, HDLC. Multiple Access: Random Access, Controlled Access, Random access Protocol-ALOHA, CSMA, CSMA/CD. Controlled Access Protocol- Reservation Polling, Token passing

	Area Network: Ethernet, Wireless LANS: IEEE802-11, Frame Relay, ATM **Term-III:** Network Layer: IP Addressing - PV4, IPV6 Routing- Interdomain: Distance vector (RIP), Link state (OSPF), Interdomain: Path vector (BGP). Gateway. Protocols:ARP, RARP, ICMP Transport Layer: Process-to-Process Delivery. UDP, TCP Congestion Control& Quality of Service. **Term-IV:** Application Layer: Client Server Model, Domain Name System (DNS), E-mail (SMTP), File Transfer (FTP) HTTP, WWW.
Semester VI	
COMPUTER GRAPHICS AND MULTIMEDIA (BCA-3203)	**Term-I**: Development of Computer Graphics: Basic graphics system and standards, Raster scan and random scan, graphics; Continual refresh and storages display, display processors and character generator, Colour display techniques, Frame buffer and bit operations, concepts in raster graphics. **Term -II**: Points, Line and Curves; Scan Conversion; Line drawing algorithms; circle and ellipse generation; Polygon filling; Conic-section generation, Ant-aliasing. **Term -III**: Two-dimensional viewing: Basic transformations; Co-ordinate systems; Windowing and Clipping; Segments; Interactive picture-construction techniques; interactive input-output device. **Term -IV:** Three-dimensional Concepts: 3-D representation and transformations; 3-D viewing; Algorithm for 3-D volumes, spline curves ad surface; Fractals; Quad tree and oct-tree datastructures; Hidden line and surface rendering, and animation. **Term -V:** An Introduction – Multimedia applications – Multimedia System Architecture – Evolving technologies for Multimedia – Defining objects for Multimedia systems – Multimedia Data interface standards – Multimedia Databases. **Term -VI:** Compression & Decompression – Data & File Format standards – Multimedia I/O technologies - Digital voice and audio – Video image and animation – Full motion video – Storage and retrieval Technologies.
GRAPHICS AND MULTIMEDIA LAB (BCA-3294)	1. Write a program to implement DDA line drawing algorithm 2. Write a program to implement Bresenham‘s line drawing algorithm. 3. Write a program to implement mid-point circle drawing algorithm. 4. Write a program to implement 2D scaling of a rectangle with respect to origin. 5. Write a program to implement 2D scaling of a rectangle with respect to arbitrary point.

	6. Write a menu driven program to show all the standards of 2D reflections. 7. Write a program to rotate a line by an angle 45 degree with respect to the centre position of the axis. 8. Write a program to rotate a line by an angle 45 degree with respect to a arbitrary point.

Department of Computer Science and BCA

Syllabus Distribution and Teaching Plan
Even Semester, Session: 2022-2023

Term I: *Commencement of classes to 1st internal*; Term II: *1st internal to 2nd internal*;
Term III: *2nd internal to ESE preparatory break.*

Semester IV

FACULTY NAME	ALLOTATED PAPERS	TEACHING PLAN
Subhadip Mukherjee	BCA-2201: Object Oriented Programming Using C++ (3L) BCA-2202: Operating System (2L) GE4T: Programming in Python (3L) GE4P: Programming in Python Practical (2P) SL/AL: Special Classes for 4th Sem (1L)	**Term I** **BCA-2201:** **(Introduction to OOPs, Features & Advantages of OOPs, Different elements of C++, Program Control Statements, Loop, Array, Function, Structures, Union and Enum).** Lecture 1:Introduction to OOPs and C++ Element Lecture 2:Features & Advantages of OOPs Lecture 3:Different elements of C++ Lecture 4:Program Control Statements Lecture 5:Loop Lecture 6:Array Lecture 7:Function Lecture 8:Structures Lecture 9:Union and Enum **BCA-2202:** **(Introduction to OS and types of OS, Process Management, Threads, CPU scheduling, Process Synchronization).** Lecture 1:Introduction to OS Lecture 2: Operating system functions Lecture 3: Different types of O.S. Lecture 4: Different types of O.S (Cont..) Lecture 5: Concept of processes Lecture 6: Process scheduling Lecture 7: Operations on processes Lecture 8: Threads Lecture 9: Threads (Cont…) Lecture 10: CPU scheduling Lecture 11: Scheduling algorithms (FCFS, SJF, RR, priority) Lecture 12: Scheduling algorithms (FCFS, SJF, RR, priority) (Cont..) Lecture 13: Introduction Of Process Synchronization Lecture 14: Critical section problem Lecture 15: Synchronization hardware Lecture 16: Classical problems of synchronization Lecture 17: Semaphores **GE4T:** **Planning the Computer Program, Techniques of Problem Solving, Overview of Programming, Introduction to Python, Creating Python Programs.** Lecture 1:Planning the Computer Program Lecture 2: Algorithms

		Lecture 3: Flowcharts Lecture 4:Techniques of Problem Solving Lecture 5:Overview of Programming Lecture 6:Introduction to Python Lecture 7:Operators Lecture 8:Input and Output Statements Lecture 9:Control statements Lecture 10:Functions **GE4P:** Practical on Structure of a Python Program, Python Interpreter, Using Python as calculator, Input and Output Statements, Control statements. Program 1 to 8. **Term II** **BCA-2201:** **(Class, Object, Constructor & Destructor, Static, Friend Function, Pointer, Polymorphism & Inheritance, Virtual Function).** Lecture 10:Class and Object Lecture 11:Constructor Lecture 12: Destructor Lecture 13: DataMember Lecture 14: Member Function Lecture 15:Static Data Member Lecture 16:Static Member Function Lecture 17:Friend Function Lecture 18:Pointer Lecture 19:Implementation of Pointer Lecture 20:Inheritance Lecture 21:Polymorphism Lecture 22: Virtual Function Lecture 23:OperatorOverloading Lecture 24:Function Overloading Lecture 23: Problem Solved **BCA 2202:** **(Storage Management: Memory Management, VirtualMemory, File Systems).** Lecture 18: Introduction of Memory Management Lecture 19:Logical vs. physical address space, Swapping Lecture 20:Contiguous memory allocation Lecture 21:Paging Lecture 22:Segmentation Lecture 23: Introduction of Virtual Memory, Demand paging, Performance Lecture 24:Page replacementalgorithms (FCFS, LRU) Lecture 25:Allocation of frames, Thrashing Lecture 26:File concept, Access methods, Directory structure, File system structure Lecture 27:Allocation methods (contiguous, linked, indexed) Lecture 28:Free-space management (bit vector, linked list, grouping) Lecture 29:Directory implementation (linear list, hashtable), efficiency & performance. **GE4T**: **Iteration and Recursion, Strings and Lists, Object Oriented**

		Programming. Lecture 11: Iteration Lecture 12: Iteration (Cont..) Lecture 13: Recursion Lecture 14: Recursion (Cont..) Lecture 15: Tables Lecture 16: Strings Lecture 17: Lists Lecture 18:Introduction to Classes Lecture 19:Objects and Methods Lecture 20:Standard Libraries **GE4P:** Practical on Looping and Recursion, Strings and Lists, Object Oriented Programming. Program 9 to 13. **Term III** **BCA 2201:** **(File Handling, Exception Handling).** Lecture 24: File Handling Lecture 25: Exception Handling Lecture 26: Files I/O Lecture 27: Tutorial Lecture 28: Tutorial Lecture 29: Tutorial Lecture 30: Tutorial **BCA 2202:** **(I/O Management and Disk Management).** Lecture 30:: I/O hardware, polling, interrupts Lecture 31:DMA Lecture 32:Application I/O interface (block andcharacter devices, network devices, clocks and timers, blocking and nonblocking I/O) Lecture 33:Kernel I/Osubsystem (scheduling, buffering, caching, spooling and device reservation, error handling), Performance Lecture 34:Disk structure Lecture 35: Disk scheduling (FCFS, SSTF, SCAN,C-SCAN) Lecture 36: Disk reliability, Disk formatting, boot block, bad blocks. **GE4T**: **Data Structures, Searching and Sorting.** Lecture 21: Array Lecture 22: List Lecture 23: Set Lecture 24: Stack and Queue Lecture 25: Searching Lecture 26: Sorting Lecture 27: Tutorial Lecture 28: Tutorial Lecture 29: Tutorial **GE4P:** Practical on Data Structures, Searching and Sorting. Program 14 to 17.

Semester VI

FACULTY NAME	ALLOTATED PAPERS	TEACHING PLAN
Subhadip Mukherjee	BCA 3295: Project (1L)	**Term I** BCA 3295: Project Selection: Project Title, Group formation, System and Technology Requirement Analysis, etc., Project Blueprint: DFD, ERD, and Design. **Term II** BCA 3295: Project Coding and Implementation: Front End and Back End Development. Project Testing. **Term III** BCA 3295: Documentation of the Project (DOCX, PDF, and PPT) and Preparation for Final Examination.

Teaching Plan
DepartmentofComputer Science & BCA
Session(2022-23)
EvenSemester

TermI	**Fromcommencement ofclassto1st InternalAssessment**
Term II	**1st Internal Assessment to 2nd Internal Assessment**

Teachingplan:2022-23(EvenSemester)
BISWAJIT LAYA
Dept.ofComputer Science & BCA

Semester II		
Syllabus allotted		BCACC3T: Digital Logic Design
	Lec No	**UNIT-I Number systems:** Positional number systems; Binary, Octal, Hexadecimal and Decimal number systems; conversion of a number in one system to the other; Representation of signed numbers-signed magnitude, one's complement, 2's complement representation techniques, Merits of 2's complement representation scheme; Various binary codes- BCD, excess -3, Gray code, ASCII, EBCDIC, Parity bits; Binary arithmetic- addition, subtraction, multiplication and division of unsigned binary numbers. **UNIT-II Boolean Algebra:** Fundamental of Boolean Expression: Definition of Switching Algebra, Basic properties of Switching Algebra, Huntington's Postulates, Basic Logic gates: (OR, AND, NOT); Universal Logic Gates: (NAND & NOR); Basic logic operations: logical sum (OR), logical product (AND), complementation (NOT), Anti coincidence (EX-OR) and coincidence (EX-NOR) operations: Truth tables of Basic gates; Boolean Variables and Expressions; Demorgan's theorem; Boolean expressions Simplification-Algebraic technique, Karnaugh map technique, 3 variable and 4 variable Karnaugh map. **UNIT-III Combinational Circuits:** Half Adder, Full Adder (3-bit), Half Subtractor, Full Subtractor (3-bit) and construction using Basic Logic Gates (OR, AND, NOT) and Universal Logic Gates (NAND & NOR), Multiplexer, Encoders, Demultiplexer and Decoder circuits, Seven Segment Display. BCD adder/ subtractor comparator; parity generators, code converters, priority encoders. **UNIT-IV Sequential circuits:** Latch, RS, D, JK, T Flip Flops; Race condition, Master Slave JK Flip Flop; Registers: Serial Input Serial Output (SISO), Serial Input Parallel Output (SIPO), Parallel input Serial Output (PISO),Parallel Input parallel Output (PIPO), Universal Shift Registers; Counters: Asynchronous Counter, Synchronous Counter.
	Term I	
	01	Positional number systems; Binary, Octal, Hexadecimal and Decimal number systems;
	02	conversion of a number in one system to the other;
	03	Various binary codes- BCD, excess -3, Gray code, ASCII, EBCDIC, Parity bits; Binary.
	04	Representation of signed numbers-signed magnitude, one's complement, 2's complement

		representation techniques, Merits of 2's complement representation scheme;
	05	arithmetic- addition, subtraction, multiplication and division of unsigned binary numbers
	06	Fundamental of Boolean Expression: Definition of Switching Algebra, Basic properties of Switching Algebra, Huntington's Postulates
	07	Basic Logic gates: (OR, AND, NOT); Universal Logic Gates: (NAND & NOR); Basic logic operations: logical sum (OR), logical product (AND), complementation (NOT),
	08	Anti coincidence (EX-OR) and coincidence (EX-NOR) operations: Truth tables of Basic gates; Boolean Variables and Expressions; Demorgan's theorem;
	09	Boolean expressions Simplification-Algebraic technique
	10	Karnaugh map technique, 3 variable and 4 variable Karnaugh map.
	Term II	
	01	Half Adder, Full Adder (3-bit), Half Subtractor, Full Subtractor (3-bit) and construction using Basic Logic Gates (OR, AND, NOT) and Universal Logic Gates (NAND & NOR).
	02	Multiplexer, BCD adder/ subtractor, Demultiplexer,Encoders
	03	Decoder circuits, Seven Segment Display, parity generators, code converters, priority encoders
	04	Latch, RS, D, JK, T Flip Flops,
	05	Race condition, Master Slave JK Flip Flop
	06	Registers: Serial Input Serial Output (SISO), Serial Input Parallel Output (SIPO), Parallel input Serial Output (PISO)
	07	Parallel Input parallel Output (PIPO),
	08	Counters: Asynchronous Counter,
	09	Synchronous Counter.
	10	Universal Shift Registers
Semester II		
Syllabus Allotted	BCACC3P: Digital Logic Lab **Combinational Circuits & Sequential Circuits:** 1. Implementation of different functions using Basic and Universal Logic gates, SOP, POS 2. Study and prove De-Morgan's Theorem. 3. Implementation of Basic gates using NAND and NOR gates 4. Implementation of half and Full Adder (3-bit) using basic logic gates and Universal logic gates (NAND & NOR). 5. Implementation of half and Full Subtractor (3-bit) using basic logic gates and Universal logic gates (NAND & NOR). 6. Design 2 to 4 decoder using basic / universal logic gates. 7. Design and implement a 8:1 multiplexer. 8. Design and implement a 3×8 decoder. 9. Design and implement a 8 bit parity generator. 10. Design and implement a D flip-flop. 11. Design and implement a J. K. flip-flop. 12. Design and implement a 4 bit synchronous counter.	
	Term I	
	01	Implementation of different functions using Basic and Universal Logic gates, SOP, POS
	02	Study and prove De-Morgan's Theorem.
	03	Implementation of Basic gates using NAND and NOR gates
	04	Implementation of half and Full Adder (3-bit) using basic logic gates and Universal logic gates (NAND & NOR).
	05	Implementation of half and Full Subtractor (3-bit) using basic logic gates and Universal logic gates (NAND & NOR).
	06	Design 2 to 4 decoder using basic / universal logic gates.
	07	Design and implement a 8:1 multiplexer.
	Term II	

	08	Design and implement a 3×8 decoder.
	09	Design and implement a 8 bit parity generator.
	10	Design and implement a D flip-flop.
	11	Design and implement a J. K. flip-flop.
	12	Design and implement a 4 bit synchronous counter

Semester VI

Syllabus Allotted

BCA-3201: OBJECT ORIENTED ANALYSIS AND DESIGN (Using UML)

UNIT-I: Introduction to UML: Importance of modeling, principles of modeling, object oriented modeling,conceptual model of the UML, Architecture, Software Development Lifecycle.

UNIT-II: Basic Structural Modeling: Classes, Relationships, common Mechanisms, and diagrams. Advanced Structural Modeling: Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages.

UNIT-III: Class & Object Diagrams: Terms, concepts, modeling techniques for Class & ObjectDiagrams.

UNIT-IV: Basic BehavioralModeling-I: Interactions, Interaction diagrams.

UNIT-V: Basic BehavioralModeling-II: Use cases, Use case Diagrams, Activity Diagrams.

UNIT-VI: Advanced BehavioralModeling: Events and signals, state machines, processes and Threads, time and space, state chart diagrams.

UNIT-VII: Architectural Modeling: Component, Deployment, Component diagrams and Deployment diagrams.

UNIT-VIII: Case Study: The Unified Library application

	Term I
01	Introduction to UML: Importance of modeling, principles of modeling,
02	object oriented modelling, conceptual model of the UML, Architecture
03	Software Development Lifecycle.
04	Classes, Relationships, common Mechanisms, anddiagrams.
05	Advanced classes, advancedrelationships, Interfaces, Types and Roles, Packages
06	Interfaces, Types and Roles, Packages
07	Class & Object Diagrams
08	Terms, conceptsmodeling techniques for Class & Object
09	Interactions
10	Interaction diagrams
11	Use cases, Use case Diagrams
12	Activity Diagrams
	Term II
13	Events and signals
14	state machinesprocesses andThreads,
15	time and space, state chart diagrams.
16	Architectural Modeling
17	Component diagrams and
18	Deployment diagrams.
19	Case Study: The Unified Library application
20	Case Study: The Unified Library application
21	Case Study: The Unified Library application