PROGRAMME OBJECTIVES AND OUTCOMES

- **Programme Educational Objectives (PEOs)**

PEO1: Graduates are prepared to be employed in IT industries by providing expected domain Knowledge.

PEO2: Graduates are provided with practical training, hands-on and project experience to meet the industrial needs.

PEO3: Graduates are motivated in career and entrepreneurial skill development to become global leaders.

PEO4:Graduates are trained to demonstrate creativity, to develop innovative ideas and to work in teams to accomplish a common goal.

PEO5: Graduates are trained to address social issues and guided to approach problems with solutions.

- **Programme Specific Outcomes(PSOs)**

After completion of the programme the graduates will be able

PSO1: To understand the fundamental concepts of computer system, including hardware and networking.

PSO2: To Design, and analyze precise specifications of algorithms, procedures, and interaction behavior.

PSO3: To communicate effectively in both verbal and written form in industry and society.

PSO4: To apply the technologies in various fields of Computer Science, including Mobile applications, Web site development and management, databases, and computer networks.

Program Outcomes (POs)	
On successful completion of the BCA program	
PO1	**Disciplinary knowledge:** Capable to apply the knowledge of mathematics, algorithmic principles and computing fundamentals in the modeling and design of computer based systems of varying complexity.
PO2	**Scientific reasoning/ Problem analysis**: Ability to critically analyze, categorizes, formulate and solve the problems that emerges in the field of computer science.
PO3	**Problem solving:** Able to provide software solutions for complex scientific and business related problems or processes that meet the specified needs with

	appropriate consideration for the public health and safety and the cultural, societal and environmental considerations.
PO4	**Environment and sustainability:** Understand the impact of software solutions in environmental and societal context and strive for sustainable development.
PO5	**Modern tool usage:** Use contemporary techniques, skills and tools necessary for integrated solutions.
PO6	**Ethics:** Function effectively with social, cultural and ethical responsibility as an individual or as a team member with positive attitude.
PO7	**Cooperation / Team Work:** Function effectively as member or leader on multidisciplinary teams to accomplish a common objective.
PO8	**Communication Skills:** An ability to communicate effectively with diverse types of audience and also able to prepare and present technical documents to different groups.
PO9	**Self-directed and Life-long Learning:** Graduates will recognize the need for self-motivation to engage in lifelong learning to be in par with changing technology.
PO10	Enhance the research culture and uphold the scientific integrity and objectivity

ALAGAPPA UNIVERSITY, KARAIKUDI
SYLLABUS UNDER CBCS PATTERN FOR AFFILIATED COLLEGES
WITH EFFECT FROM THE ACADEMIC YEAR 2022-23 ONWARDS

B. C. A.
Programme Structure

Sem.	Part	Course Code	Courses	Title of the Paper	T/P	Credits	Hours/ Week	Max. Marks		
								Int.	Ext.	Total
I	I	2211T	T/OL	Tamil /Other Languages -I	T	3	6	25	75	100
	II	712CE	E	Communicative English - I	T	3	6	25	75	100
	III	22BCA1C1	CC	Data Structure & C Programming	T	5	5	25	75	100
		22BCA1P1	CC	Practical -Data Structure & C Programming Lab	P	4	4	40	60	100
		-	AL – IA	IT/Computer Science/ Mathematics/Physics	T	3	3	25	75	100
		-	AL - IA	Practical-Respective Allied Theory Course	P	2	2	40	60	100
	IV	22BVE1	SEC-I	Value Education	T	2	2	25	75	100
		-	-	Library	-	--	2	--	--	--
				Total		**22**	**30**	**205**	**495**	**700**
II	I	2221T	T/OL	Tamil/Other Languages-II	T	3	6	25	75	100
	II	722CE	E	Communicative English - II	T	3	6	25	75	100
	III	22BCA2C1	CC	Object Oriented Programing in C++	T	5	5	25	75	100
		22BCA2P1	CC	Practical-Object Oriented Programing in C++	P	4	4	40	60	100
		-	AL - IB	IT/Computer Science / Mathematics/Physics	T	3	3	25	75	100
		-	AL - IB	Practical-Respective Allied Theory Course	P	2	2	40	60	100
	IV	22BES2	SEC-II	Environmental Studies	T	2	2	25	75	100
		Naan Mudhalvan Course		Language Proficiency for Employability(Effective English)	-	2	2	25	75	100
				Total		**24**	**30**	**230**	**570**	**800**
III	I	2231T	T/OL	Tamil/Other Languages-II	T	3	6	25	75	100
	II	2232E	E	English for Enrichment - I	T	3	6	25	75	100
	III	22BCA3C1	CC	Database Management System	T	3	3	25	75	100
		22BCA3C2	CC	Operating System	T	3	3	25	75	100
		22BCA3P1	CC	Practical-Oracle Lab	P	3	3	40	60	100
		-	AL -IIA	IT/Computer Science / Mathematics/Physics	T	3	3	25	75	100
		-	AL -IIA	Practical-Respective Allied Theory Course	P	2	2	40	60	100
	IV	22BE3	SEC-III	Entrepreneurship		2	2	25	75	100
		-	NME-I	1.Adipadai Tamil (or) 2.Advance Tamil (or) 3.IT Skills for Employment (or) MOOC's	T	2	2	25	75	100
				Total		**24**	**30**	**255**	**645**	**900**
	I	2241T	T/OL	Tamil /Other Languages -IV	T	3	6	25	75	100
	II	2242E	E	English for Enrichment - II	T	3	3	25	75	100

IV	III	22BCA4C1	CC	Java Programming	T	4	4	25	75	100
		22BCA4C2	CC	Computer Networks	T	4	4	25	75	100
		22BCA4P1	CC	Practical–Java Programming	P	3	3	40	60	100
		-	AL – IIB	IT/Computer Science / Mathematics/Physics	T	3	3	25	75	100
		-	AL - IIB	Practical-Respective Allied Theory Course	P	2	2	40	60	100
	IV	-	NME- II	1. Adipadai Tamil(or) 2. Advance Tamil(or) 3. Small Business Management (or) MOOC's	T	2	2	25	75	100
		Naan Mudhalvan Course		Digital Skills for Employability – (Microsoft-Office Fundamentals)	-	2	3	25	75	100
				Total		**26**	**30**	**275**	**645**	**900**
V	III	22BCA5C1	CC	• Net Programming	T	4	4	25	75	100
		22BCA5C2	CC	Python Programming	T	4	4	25	75	100
		22BCA5C3	CC	Web Design Technology	T	4	4	25	75	100
		22BCA5C4	CC	Computer Architecture and Organization	T	4	4	25	75	100
		22BCA5P1	CC	Practical–Python Programming	P	4	6	40	60	100
		22BCA5P2	CC	Practical–Web Design Technology	P	4	6	40	60	100
	IV	-	-	Career Development/ Employability skills	-	-	2	--	--	--
				Total		**24**	**30**	180	**420**	**600**
VI	III	22BEL6I	DSE	Internship		**24**	**30**	**150**	**250**	**400**
	IV	Naan Mudhalvan Course		Emerging Technology for Employability(Course Name: Machine Learning*/Android app**/ Cyber Security***)	-	2	4	25	75	100
				Total		**26**	**30**	**175**	**325**	**500**
		(Or)								
	III	22BCA6E1 22BCA6E2	DSE	(A)Data Mining & Warehousing/ (B)Artificial Intelligence	T	6	6	25	75	100
		22BCA6E3 22BCA6E4		(A)Software Engineering / (B)Internet of Things	T	6	6	25	75	100
		22BCA6E5 22BCA6E6		(A)Cloud Computing / (B) Mobile Application Development	T	6	6	25	75	100
		22BCA6E7 22BCA6E8		(A)Fundamentals of Digital Image Processing / (B) Computer Graphics	T	6	6	25	75	100
	IV	-	Others	Library/Yoga etc.	-	--	2	--	--	--
		Naan Mudhalvan Course		Emerging Technology for Employability(Course Name: Machine Learning*/Android app**/ Cyber Security***)	-	2	4	25	75	100
				Total		**26**	**30**	**125**	**375**	**500**
	III	**(Or)**								
		22BCA6PR	DSE	Project		6	8	25	75	100
		22BCA6E1 22BCA6E2		(A)Data Mining & Warehousing /(B)Artificial	T	6	6	25	75	100

				Intelligence						
		22BCA6E3 22BCA6E4		(A)Software Engineering / (B)Internet of Things	T	6	6	25	75	100
		22BCA6E5 22BCA6E6		(A)Cloud Computing / (B) Mobile Application Development	T	6	6	25	75	100
	IV	Naan Mudhalvan Course		Emerging Technology for Employability(Course Name: Machine Learning*/Android app**/ Cyber Security***)	-	2	4	25	75	100
				Total		**26**	**30**	**125**	**375**	**500**
				Grand Total		**146**	**--**	**--**	**--**	**4400**

*Machine Learning - All Computer Science programmes for Government Colleges
** Android App - All Computer Science programmes for Government Aided College
***Cyber Security - All Computer Science programmes for Self financing College

Sem.	Part	Course Code	Title of the Paper	Credits	Hours/ Week	Marks		
						Int.	Ext.	Total
I	III	71BEPP	Professional English for Physical Science -I	4	5	25	75	100
II		72BEPP	Professional English for Physical Science -II	4	5	25	75	100
III		*	Professional English for Physical Science -III	4	5	25	75	100
IV			Professional English for Physical Science -IV	4	5	25	75	100

*The Syllabus of Professional English for III & IV Semester will be provided after Receiving the syllabus from TANSCHE.

As per TANSCHE, the Professional English book will be taught to all four streams apart from the existing hours of teaching/additional hours of teaching (1hour/day) as a 4 credit paper as an add on course on par with Major paper and completion of the paper is a must to continue his/her studies further.

- T/OL-Tamil or Other Language,
- E – English
- CC-Core course –Core competency, critical thinking, analytical reasoning, research skill & team work
- Allied / GEC -Exposure beyond the discipline
- AECC- -Ability Enhancement Compulsory Course (Professional English & Environmental Studies) - Additional academic knowledge, psychology and problem solving etc.,
- SEC-Skill Enhancement Course - Exposure beyond the discipline (Value Education, Entrepreneurship Course, Computer application for Science, etc.,
- NME -Non Major Elective – Exposure beyond the discipline
- DSE – Discipline specific elective –Additional academic knowledge, critical thinking, and analytical reasoning-Student choice - either Internship or Theory papers or Project + 2 theory paper.
    - If internship – Marks = Internal- 150 (75+75) two midterm evaluation through Viva voce + Report- 150+ External Viva voce- 100 = 400.
    - If Project – Marks = Internal- 50 +Thesis- 100 + Viva voce- 50 = 200 + 2 theory paper- 200 = 400
- MOOCs – Massive Open Online Courses

*T-Theory, P-Practical

Semester - I				
Course code 22BCA1C1	**Core course- I**	**T/P**	**C**	**H/W**
	Data Structures & C Programming	**T**	**5**	**5**
Objectives	➢ To understand and develop well-structured programs using C language. ➢ To learn the basic data structures implementing through C language. ➢ To deal with different memory allocation & input/output methods. ➢ Problem solving through computer programming using C Language.			
Unit -I	**Data Structure:-** Classification of Data Structures, Data Structure Operations, Abstract Data Type. **Stack:**- definition, Stack as ADT. **Queue:**- Definition, Queue as ADT. **Linked List:**- Insertion into Linked List, Deletion into Linked List. **Trees:**- Basic Terminology.			
Unit - II	**Overview of C:-** History of C, Importance of C, Sample C Programs, Structure of a C Programs, Constants, Variables and Data Types, Operators and Expressions, Input and Output Operations.			
Unit - III	**Decision Making – Branching – Looping - Arrays:-** One and Two Dimensional Arrays. **Character Strings:**- Declaring and Initializing String Variables, Reading Strings From Terminal, Writing Strings to Screen, Arithmetic Operations on Characters, String Handling Functions.			
Unit - IV	**User Defined Functions:**- Introduction, Need for User Defined Functions, The Form of C Functions, Return values and their types, Calling a Function, Categories of Functions, Nesting of Functions, Recursion, Functions With Arrays, The Scope and Lifetime of Variables. **Structures and Unions:**- Structure Definition, Giving Values to Members, Structure Initialization, Arrays of Structures, Arrays Within Structures, Structures Within Structures, Structures And Functions, Unions.			
Unit - V	**Pointers:-** Introduction, Understanding Pointers, Accessing the Address of a Variable, Declaring and Initializing Pointers, Accessing a Variable through its Pointer. **File Handling:-** Defining and Opening a File, Closing a File, I/O Operations on Files, Error Handling During I/O Operation.			

Reference and Textbooks:

Text Books:

Balagurusamy, E. (2017). *Programming in ANSI C* (8th ed.). New Delhi: TATA McGraw-Hill Publishing Company Ltd.

Seymour Lipschutz. (2010). *Data Structures* (3rd ed.). New Delhi: TATA McGraw-Hill, Publishing Company Ltd.

Books for Reference:

Byron Gottfried, S. (1996). Schaum's outline series. *Theory and problems of programming with C.* New Delhi: TATA McGraw-Hill Publishing Company Ltd.

Ravichandran, D. (2009). *Programming in C.* New Age International publisher.

Venugopal, K.R. & Sudeep Prasad, R. (1997). *Programming with C.* New Delhi: TATA McGraw-Hill Publishing Company Ltd.

WEB RESOURCES:

https://www.unf.edu/~wkloster/2220/ppts/cprogramming_tutorial.pdf

https://www.tutorialspoint.com/cprogramming/cprogramming_pdf_version.htm **Related Online Contents [MOOC, SWAYAM, NPTEL, Websites etc.]** Introduction to Programming in C – NPTEL Problem solving through Programming in C – SWAYAM C for Everyone : Programming Fundamentals – Coursera	
Outcomes	On Completion of this Course, the students can able to, ➢ Understand and apply the basic Concepts of Data Structures. ➢ Describe the fundamental concepts of C Programming. ➢ Implement the Decision making and Looping Statements, Arrays and Strings. ➢ Define the User Defined functions, Structures and Unions. ➢ Put into Practice the Pointers and File Management in C.

Semester - I				
Course code: 22BCA1P1	**Core Practical - I**	**T/P**	**C**	**H/W**
	Data Structures & C Programming Lab	**P**	**4**	**4**

Objectives

- To introduce the basic knowledge of C programming fundamentals.
- To impart writing skill of C programming to the students and solving problems.
- To implement the basic concepts of Data Structures.

Lab Programs

1. Write a Program to initialize, assignment & printing variables of different data types.
2. Write a Program to demonstrate all the operators
3. Write a Program to read marks of a student in six subjects and print whether pass or fail (using if-else).
4. Write a Program to perform arithmetic operations using switch case.

Do the Following Programs Using for, while, do-while loops.

5. Write a program to calculate sum of individual digits of a given number.
6. Write a program to check whether given number is palindrome or not.
7. Write a program to print prime numbers in the given range.
8. Write a program to store 10 elements in the 1-D array and print sum of the array.
9. Write a program to print minimum and maximum elements in the 1-D array.
10. Write a program to count no. of positive numbers, negative numbers and zeros in the array.
11. Write a program to perform matrix addition and matrix subtraction.
12. Write a program to perform various string manipulations
13. Write a program to print the given strings in ascending order.
14. Write a program to verify the given string is palindrome or not (without built-in functions, with using built-in functions).
15. Write a program to concatenate two strings using arrays.
16. Write a program to swap two numbers using a) Call By Value B) Call By Reference.
17. Write a program to find total marks of individual student and average marks for 10 students using structures.
18. Write a program which copies the contents of one file to another file using command line arguments.
19. Program to Implement the Stack Operations
20. Program to Implement the Queue Operations
21. Program to implement the Linked list

Reference and Textbooks:

AL Kelly & Ira phol (1998). *Programming in C* (4th ed.). Addison-Wesley–Professional.

Balaguruswamy, E. (2019). *Programming in ANSI C* (8th ed.) TATA Mc Graw-Hill.

Brain Kernighan, W., & Dennis Ritchie (1988) *C Programming Language* (2nd ed.). PHI.

Gray Brosin, J. (2006). *A first book of ANSI C* (3rd ed.). Cengage Learning India P. Ltd.

Jeri Hanly, R., & Elli Koffman, B. (2013). *Problem Solving and Program Design in C* (7th ed.). Pearson. ISBN-13: 978-0-13-293649-1, ISBN-10: 0-13-293649-6.

Pradip Dey & Manas Ghosh (2013). *Programming in C* (2nd ed.) Oxford University Press.

Outcomes	On Completion of this Course, the students can able to, ➢ Read, understand and trace the execution of programs written in C language. ➢ Write the C code for a given algorithm and Implement programs with pointers and arrays, perform pointer arithmetic, use the pre-processor. ➢ Write programs that perform operations using derived data types. ➢ Develop the programs to implement the concepts of Data Structure.

Semester - II				
Course code 22BCA2C1	**Core Course- II**	**T/P**	**C**	**H/W**
	Object Oriented Programming in C++	T	5	5
Objectives	➢ To understand how C++ improves C with object-oriented features. ➢ To learn how to write inline functions for efficiency and performance. ➢ To learn the syntax and semantics of the C++ programming language. ➢ To learn how to design C++ classes for code reuse.			
Unit -I	**Principles of Object-Oriented Programming:-** Basic Concepts of Object Oriented Programming, Benefits of OOP, Applications of OPP. **Beginning with C++:-** What is C++? Applications of C++, A Simple C++ Program, More C++ Statements, An Example with Class, Structure of C++ Program. **Tokens, Expressions and Control Structures:-** Introduction, Tokens, Keywords, Identifiers and Constants, Basic Data Types, User-defined Data Types, Derived Data Types, Operators in C++, Expressions and their types, Implicit Conversions, Operator Overloading, Operator Precedence, Control Structures.			
Unit-II	**Function in C++:-** Introduction, The Main Function, Function Prototyping, Call by Reference, Return by Reference, Inline Function, Default Arguments, Const Arguments, Function Overloading, Friend and Virtual Functions, Math Library Functions. **Classes and Objects:-** Introduction, Specifying a Class, Defining Member Function, C++ Program with Class, Making an Outside Function Inline, Nesting of Member Functions, Arrays within a Class, Memory Allocation for Objects, Static Data Members, Static Member Functions, Arrays of Objects, Objects as Function Arguments, Friendly Functions, Returning Objects.			
Unit-III	**Constructors and Destructors:-** Introduction, Constructors, Parameterized Constructors, Multiple Constructors in Class, Constructors with Default Arguments, Dynamic Initialization of Objects, Copy Constructor, Dynamic Constructors, Constructing Two Dimensional Arrays, Destructors. **Inheritance:-** Introduction, Defining Derived Classes, Single Inheritance, Making a Private Member Inheritable, Multilevel Inheritance, Multiple Inheritance, Hierarchical Inheritance, Virtual Base classes, Abstract Classes, Constructors in Derived Classes, **Member Classes:-** Nesting of Classes.			
Unit-IV	**Pointers Virtual Functions and Polymorphism:-** Introduction, Pointers to Objects, This Pointer, Pointers to Derived Classes, Virtual Functions, Pure Virtual Functions. **Managing Console I/O Operations:-** C++ Streams, C++ Stream Classes, Unformatted I/O Operations, Formatted Console I/O Operations, Managing Output with Manipulators. **Working with Files:-** Introduction, Classes for File Stream Operations, Opening and Closing a File, Detecting End of File, More About Open (): File Modes, File Pointers and their Manipulations, Sequential Input and output Operations, Updating a File, Random Access, Error handling During File Operations, Command Line Arguments.			
Unit-V	**Templates:-** Introduction, Function Templates, Overloaded Function Templates,			

	Nesting of Function Calls, Multiple Arguments Function Template, User Defined Templates. **Exception Handling:-** Introduction, Error Handling, Exception Handling Model, Exception handling Constructs, Handler Throwing the Same Exception Again, List of Exceptions, Catch All Exceptions, Exceptions in Constructors and Destructors, Handling Uncaught Exceptions, Ten Rules for Handing Exceptions Successfully.

Reference and Textbooks:

TEXT BOOKS:

Balagurusamy, E. (2019). *Object Oriented Programming with C++* (7th ed.). New Delhi: Tata McGraw-Hill.

REFERENCE BOOKS:

Nabajyoti Barkakati . (1997). Object Oriented Program in C++. New Delhi: PHI P. Ltd.

Venugopal, K. R., Ravishankar, T., & RajKumar (2006). *Mastering C++*. New Delhi : Tata Mc Graw-Hill Publishing Company Limited .

Related Online Contents [MOOC, SWAYAM, NPTEL, Websites etc.]

https://www.spoken-tutorial.org

https://www.tutorialspoint.com/cplusplus/index.htm

https://www.w3schools.com/cpp/

Outcomes	Completion of this Course, the students can able to, ➢ Understanding of the concepts of inheritance, polymorphism and bility to overload operators in C++. ➢ Understanding the difference between function overloading & function overriding. ➢ Ability to incorporate exception handling in object-oriented programs and to use template classes and the STL library in C++.

Semester - II				
Course code: 22BCA2P1	**Core Practical - II**	**T/P**	**C**	**H/W**
	Object Oriented Programming in C++ Lab	**P**	**4**	**4**

Objectives	➢ To implement the various object oriented programming concepts using C++.
Unit -I	1. Write a C++ program to demonstrate function overloading, Default Arguments and Inline function. 2. Write a C++ program to demonstrate Class and Objects 3. Write a C++ program to demonstrate the concept of Passing Objects to Functions 4. Write a C++ program to demonstrate the Friend Functions. 5. Write a C++ program to demonstrate the concept of Passing Objects to Functions 6. Write a C++ program to demonstrate Constructor and Destructor 7. Write a C++ program to demonstrate Unary Operator Overloading 8. Write a C++ program to demonstrate Binary Operator Overloading 9. Write a C++ program to demonstrate Single Inheritance 10. Write a C++ program to demonstrate Multilevel Inheritance 11. Write a C++ program to demonstrate Multiple Inheritance 12. Write a C++ program to demonstrate Hierarchical Inheritance 13. Write a C++ program to demonstrate Hybrid Inheritance 14. Write a C++ program to demonstrate Virtual Functions. 15. Write a C++ program to manipulate a Text File. 16. Write a C++ program to perform Sequential I/O Operations on a file. 17. Write a C++ program to find the Biggest Number using Command Line Arguments 18. Write a C++ program to demonstrate Class Template 19. Write a C++ program to demonstrate Function Template. 20. Write a C++ program to demonstrate Exception Handling.

Reference and Textbooks:

TEXT BOOK:

Balagurusamy, E. (2013). *Object-Oriented Programming with C++* (7th ed.). TATA McGraw-Hill.

REFERENCE BOOKS:

Ashok Kamthane, N. (2003). *Object-Oriented Programming with ANSI and Turbo C++*. Pearson Edu.

Maria Litvin & Gray Litvin. (2002). *C++ for you*. Vikas publication.

Outcomes	On Completion of this Course, the students can able to ➢ Understand the structure and model of the C++ programming language. ➢ Solve problems in C++ demonstrating Object Oriented Concepts.

Semester - III				
Course code	**Core Course - III**	**T/P**	**C**	**H/W**
22BCA3C1	**Database Management System**	**T**	**3**	**3**

Objectives	➢ Gain a good understanding of the architecture and functioning of Database Management Systems ➢ Apply Normalization techniques to normalize a database. ➢ Understand the need of transaction processing and learn techniques for controlling the consequences of concurrent data access. ➢ Understand the use of Structured Query Language (SQL) and its syntax.
Unit -I	**Introduction:-** Database System Applications, Purpose of Database Systems, View of Data, Database Languages, Relational Databases, Database Design, Object based and Semi Structured Databases, Data Storage and Querying, Database Users and Administrators,–Transaction Management, Database users and Architectures, History of Database System. **Entity-Relationship Model:**- E-R model, Constraints, E-R diagrams, E-R Design Issues, Weak Entity Sets, Extended E-R features.
Unit-II	**Relational Database Design:-** Features of good Relational Designs, Atomic Domains and First Normal Form, Decomposition using Functional Dependencies, Functional Dependency Theory, Decomposition using Functional, Decomposition using Multivalued Dependencies, more Normal forms, Database Design Process, Modeling Temporal Data.
Unit-III	**Database System Architecture:-** Centralized and Client-Server architecture, Server System Architecture, Parallel Systems, Distributed Systems, Network Types. **Parallel Databases:-** I/O parallelism, Interquery Parallelism, Intraquery Parallelism. **Distributed Databases:**- Homogeneous and Heterogeneous Databases, Distributed Data Storage, Distributed Transactions, Distributed Query Processing.
Unit-IV	**Schema Objects:-** Data Integrity, Creating and Maintaining Tables, Indexes, Sequences, Views, Users Privileges and Roles, Synonyms.
Unit-V	**PL/SQL:-** PL/SQL, Triggers, Stored Procedures and Functions, Package, Cursors, Transaction.

Reference and Textbooks:

TEXT BOOKS:

Sumathi, S., & Esakkirajan, S. (2007). *Fundamentals of Relational Database Management System.* Springer International Edition.

Silberchatz, A., Henry Korth, F., & Sudarshan, S. (2019). *Database System Concepts* (7th ed.). Tata McGraw Hill.

REFERENCE BOOKS:

Alexis Leon & Mathews Leon (2014). *Fundamentals of DBMS* (2nd ed.). Vijay Nicole Publications

WEB REFERENCES:

NPTEL & MOOC courses titled Relational Database Management Systems
https://nptel.ac.in/courses/106106093/
https://nptel.ac.in/courses/106106095/

Outcomes	On Completion of this Course, the students can able to ➢ Have a broad understanding of database concepts and database management system software ➢ Have a high-level understanding of major DBMS components and their function. ➢ Model an application's data requirements using conceptual modeling tools like ER diagrams and design database schemas based on the conceptual model. ➢ Write SQL commands to create tables and indexes, insert/update/delete data, and query data in a relational DBMS.

Semester - III				
Course code	**Core Course - IV**	**T/P**	**C**	**H/W**
22BCA3C2	**Operating System**	**T**	**3**	**3**
Objectives	➢ To understand the services provided by and the design of an operating system. ➢ To understand the structure and organization of the file system. ➢ To understand what a process is and how processes are synchronized and scheduled. ➢ To understand different approaches to memory management.			
Unit -I	**Introduction:-** Views, Goals, Types of System, OS Structure, Components, Services, System Structure, Layered Approach, Virtual Machines, System Design and Implementation. **Process Management:-** Process, Process Scheduling, Cooperating Process, Treads, Inter-process Communication. **CPU Scheduling:-** CPU Schedulers, Scheduling Criteria, Scheduling Algorithms.			
Unit-II	**Processor Management:-** Process Synchronization, Critical-Section Problem, Synchronization Hardware, Semaphores, Classical Problems of Synchronization, Critical Region, Monitors. **Deadlocks:-** Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Avoidance, Detection, Recovery.			
Unit-III	**Memory Management:-** Address Binding, Dynamic Loading and Linking, Overlays, Logical and Physical Address Space, Contiguous Allocation, Internal & External Fragmentation. **Non-Contiguous Allocation:-** Paging and Segmentation Schemes, Implementation, Hardware-Protection, Sharing, Fragmentation.			
Unit-IV	**Virtual Memory:-** Demand Paging, Page Replacement, Page Replacement Algorithms, Thrashing. **File System:-** File Concepts, Access Methods, Directory Structures, Protection Consistency, Semantics, File System Structures, Allocation Methods, Free Space Management.			
Unit-V	**I/O System:-** Overview, I/O Hardware, Application I/O Interface, Kernel I/O Subsystem, Transforming I/O Requests to Hardware Operations, Performance. **Secondary Storage Structures:-** Protection, Goals, Domain, Access matrix, The Security Problem, Authentication, Threats, Threat Monitoring, Encryption.			

Reference and Textbooks:

TEXT BOOK:

Silberschatz, A., Peter Galvin, B., & Gagne, G. (2018). *Operating System Concepts* (9th ed.). Wiley India Pvt. Ltd.

REFERENCES:

Andrew Tanenbaum, S., & Herbert Bos. (2018). *Modern Operating Systems* (4th ed.). Pearson Edu..

William Stallings (2018). *Operating Systems Internals and Design Principles* (9th ed.). Pearson.

WEB RESOURCES

https://examsdaily.in/wp-content/uploads/2018/08/Operating-System.pdf

http://crsgphathnikund.ac.in/wp-content/uploads/2018/09/operating-system.pdf

https://www.tutorialspoint.com/operating_system/index.htm	
Outcomes	On Completion of this Course, the students can able to, ➢ Identify the role of Operating System. To understand the design of control unit. ➢ Understanding CPU Scheduling, Synchronization, Deadlock Handling and Comparing CPU Scheduling Algorithms. Solve Deadlock Detection Problems. ➢ Idea in the role of paging, segmentation and virtual memory in operating systems. ➢ Knowledge in Protection, security, Comparison of UNIX and Windows based OS. ➢ Defining I/O systems, Device Management Policies and Secondary Storage Structure and Evaluation of various Disk Scheduling Algorithms.

Semester - III				
Course code 22BCA3P1	**Core Practical - III**	**T/P**	**C**	**H/W**
	Oracle Lab	**P**	**3**	**3**

Objectives

- Learn the various DDL and DML commands
- Understand queries in SQL to retrieve information from data base
- Understand PL/SQL statements: Exception Handling, Cursors, and Triggers.
- Develop the database applications using front-end and back-end tools.

Lab Programs

1. (Exercise on retrieving records from the table) EMPLOYEES (Employee_Id, First_Name, Last_Name, Email, Phone_Number, Hire_Date, Job_Id, Salary, Commission_Pct, Manager_Id, Department_Id)

(a) Find out the employee id, names, salaries of all the employees
(b) List out the employees who works under manager 100
(c) Find the names of the employees who have a salary greater than or equal to 4800
(d) List out the employees whose last name is 'AUSTIN'
(e) Find the names of the employees who works in departments 60,70 and 80
(f) Display the unique Manager_Id.

2. (Exercise on updating records in table) Create Client_master with the following fields(ClientNO, Name, Address, City, State, bal_due)

(a) Insert five records
(b) Find the names of clients whose bal_due> 5000 .
(c) Change the bal_due of ClientNO " C123" to Rs. 5100
(d) Change the name of Client_master to Client12 .
(e) Display the bal_due heading as "BALANCE"

3. Rollback and Commit commands Create Teacher table with the following fields(Name, DeptNo, Date of joining, DeptName, Location, Salary)

(a) Insert five records
(b) Give Increment of 25% salary for Mathematics Department .
(c) Perform Rollback command
(d) Give Increment of 15% salary for Commerce Department
(e) Perform commit command

4 . (Exercise on order by and group by clauses) Create Sales table with the following fields(Sales No, Salesname, Branch, Salesamount, DOB)

(a) Insert five records
(b) Calculate total salesamount in each branch
(c) Calculate average salesamount in each branch .
(d) Display all the salesmen, DOB who are born in the month of December as day in character format i.e. 21-Dec-09
(e) Display the name and DOB of salesman in alphabetical order of the month.

5. Create an Emp table with the following fields: (EmpNo, EmpName, Job,Basic, DA, HRA,PF, GrossPay, NetPay) (Calculate DA as 30% of Basic and HRA as 40% of Basic)

(a) Insert Five Records and calculate GrossPay and NetPay.

(b) Display the employees whose Basic is lowest in each department .
(c) If NetPay is less than Rs. 10,000 add Rs. 1200 as special allowances .
(d) Display the employees whose GrossPay lies between 10,000 & 20,000
(e) Display all the employees who earn maximum salary .

6. Employee Database An Enterprise wishes to maintain a database to automate its operations. Enterprise is divided into certain departments and each department consists of employees. The following two tables describes the automation schemas Dept (deptno, dname, loc) Emp (empno, ename, job, mgr, hiredate, sal, comm, deptno)

(a) Update the employee salary by 15%, whose experience is greater than 10 years.
(b) Delete the employees, who completed 30 years of service.
(c) Display the manager who is having maximum number of employees working under him?
(d) Create a view, which contain employee names and their manager

7. Using Employee Database perform the following queries

(a) Determine the names of employee, who earn more than their managers.
(b) Determine the names of employees, who take highest salary in their departments.
(c) Determine the employees, who are located at the same place.
(d) Determine the employees, whose total salary is like the minimum Salary of any department.
(e) Determine the department which does not contain any employees.

8. Consider the following tables namely "DEPARTMENTS" and "EMPLOYEES" Their schemas are as follows, Departments (dept _no , dept_ name , dept_location); Employees (emp_id , emp_name , emp_salary,dept_no);

(a) Develop a query to grant all privileges of employees table into departments table
(b) Develop a query to grant some privileges of employees table into departments table
(c) Develop a query to revoke all privileges of employees table from departments table
(d) Develop a query to revoke some privileges of employees table from departments table
(e) Write a query to implement the save point.

9. Using the tables "DEPARTMENTS" and "EMPLOYEES" perform the following queries

(a) Display the employee details, departments that the departments are same in both the emp and dept.
(b) Display the employee name and Department name by implementing a left outer join.
(c) Display the employee name and Department name by implementing a right outer join.
(d) Display the details of those who draw the salary greater than the average

	salary. 10. PL/SQL programs with control structures. 11. PL/SQL programs with Cursors. 12. PL/SQL programs with Exception Handling. 13. PL/SQL program for Creating and Calling Procedures. 14. PL/SQL program for Creating and Calling Functions. 15. PL/SQL program for creating and Calling Packages. 16. PL/SQL program for Overloading Packages. 17. PL/SQL program for Working with Triggers.
Outcomes	On Completion of this Course, the students can able to ➢ Implement the DDL , DML Commands and Constraints ➢ Create, Update and query on the database. ➢ Design and Implement simple project with Front End and Back End.

Semester - IV				
Course code	**Core Course - V**	**T/P**	**C**	**H/W**
22BCA4C1	**Java Programming**	**T**	**4**	**4**
Objectives	➢ To expose the students with the introduction to OOPs and advantages of object oriented programming. ➢ To describe the concepts of OOPs make it easy to represent real world entities. ➢ To summarize the concepts of converting the real time problems into objects and methods and their interaction with one another to attain a solution. ➢ To observe the syntax of programming language Java for solving the real world problems.			
Unit -I	**Fundamentals of Object Oriented Programming:-** Introduction, Object Oriented Paradigm, Basic Concepts of OOP, Benefits of OOP, Applications of OOP. **Java Evolution:-** Java History, Java Features, Java and Internet, World Wide Web, Web Browsers, H/W and S/W requirements, Java Support Systems, Java Environment. **Overview of Java language:-** Introduction, Simple Java Program, Comments, Java Program Structure, Tokens, Java Statements, Implementing a Java Program, JVM, Command Line Arguments, Constants, Variables, Data Types, Type Casting.			
Unit-II	**Operators and Expressions:-** Arithmetic Operators, Relational, Logical, Assignment, Increment and Decrement, Conditional, Bitwise, Special Operators, Arithmetic Expressions, Evaluation of Expression, Precedence of Arithmetic Operators, Type Conversions, Operator Precedence and associativity, Mathematical Functions. **Decision Making and Branching:-** If –if…..else –Nesting of if……. Else – else if–switch. **Decision Making and Looping:-** While – do – for – jump in loops – labeled loops.			
Unit-III	**Classes, Objects and Methods:-** Defining a Class, Adding Variables, Methods, Creating objects, Accessing Class Members, Constructors, Methods overloading, Static Members, Nesting of Methods, Inheritance, Overriding Methods, Final Variables and methods, Final classes, Finalizer methods, Abstract Methods And Classes, Visibility Control. **Arrays, Strings and Vectors:-** Arrays, One Dimensional Arrays, Creating an array, Two Dimensional Arrays, Strings, Vectors, Wrapper Classes. **Interfaces: Multiple Inheritance:-** Defining Interfaces, Extending Interfaces, Implementing Interfaces, Accessing Interface Variables.			
Unit-IV	**Packages:-** Java API Packages, Using system packages, Naming conventions, Creating Packages, Accessing a Package, Using a Package, Adding a Class to a Package, Hiding Classes. **Multithreaded Programming:-** Creating Threads, Extending the Thread Class, Stopping and Blocking a Thread, Life Cycle of a Thread, Using Thread Methods, Thread Exceptions, Thread Priority, Synchronization, Implementing the 'Runnable' Interface. **Managing Errors and Exceptions:-** Types of Errors, Exceptions, Syntax of			

	Exception Handling Code, Multiple Catch Statements, Using Finally Statement, Throwing Our Own Exceptions, Using Exceptions for Debugging.
Unit-V	**Applet Programming:-** How applets differ from Applications, Preparing to Write Applets, Building Applet Code, Applet life Cycle, Creating an Executable Applet, Designing a Web Page, Applet Tag, Adding Applet to HTML File, Running the Applet, Passing Parameters to Applets, Displaying Numerical Values, Getting input from the user. **Graphics Programming:-** The Graphics Class, Lines and Rectangles, Circles and Ellipses, Drawing Arcs, Drawing Polygons, Line Graphs, Using Control Loops in Applets, Drawing Bar Charts.

Reference and Textbooks:

TEXT BOOK:

Balagurusamy, E. (2010). *Programming with JAVA* (3rd ed.). TATA McGraw-Hill Publishing Company Limited.

REFERENCES:

Deitel, H.M., & Deitel, P.J. (2005). *Java – How to Program* (6th ed.). Pearson Education Pvt. Ltd.

Herbert Schildt (2006). *Java 2 – The Complete Reference* (5th ed.). New Delhi: TATA Mc Graw Hill Publishing Company Limited.

WEB RESOURCES

www.spoken-tutorial.org

www.nptel.ac.in

https://www.w3schools.in/java-tutorial/

Outcomes	On Completion of this Course, the students can able to ➢ Competence and the development of small to medium sized application programs that demonstrate professionally acceptable coding. ➢ Demonstrate the concept of object oriented programming through Java. ➢ Apply the concept of Inheritance, Modularity, Concurrency, Exceptions handling and data persistence to develop java program. ➢ Develop java programs for applets and graphics programming. ➢ Understand the fundamental concepts of AWT controls, layouts and events.

Semester - IV				
Course code 22BCA4C2	**Core Course -VI**	**T/P**	**C**	**H/W**
	Computer Networks	**T**	**4**	**4**
Objectives	➢ To develop an understanding of computer networking basics. ➢ To study different components of computer networks, various protocols, modern technologies and their applications.			
Unit -I	**Introduction:-** Uses of Computer Networks, Network Hardware and network software, Reference models, Example Networks, Network Standardization. **Physical Layer:**- Transmission Media, Telephone System, ISDN, Broadband and Narrowband ISDN, ISDN and ATM, Communication Satellites.			
Unit-II	**Data Link Layer:-** Design Issues, Error Detection and Correction Codes, Elementary data link Protocols, Sliding Window Protocols. **Protocol Specification and Verification:-** Finite state models, Petri net models. **Example Drink Protocols:-** HDLC, SLIP, PPP. **Media access Sub layer:-** Multiple Access Protocols, ALOHA, Carrier Sense, multiple Access protocols, Collision free Protocols.			
Unit-III	**Network Layer:-** Design Issues, Routing Algorithms, Congestion Control Algorithms. **Internetworking:-** Tunneling, Fragmentation, Firewalls, Network Layer in the Internet, IP, Subnets. **Internet Control Protocols:**- Address Resolution Protocol, ICMP, RARP, Internet multicasting. **Network layer in ATM networks:**- Cell Format, Connection setup, Routing and switching, Services Categories, ATM LANs.			
Unit-IV	**Transport Layer:-** Transport Service, **Elements of Transport Protocols:-** Addressing, Floe Control and Buffering, Multiplexing, Crash Recovery, Performance issues, Measuring Network performance, Internet Transport Protocols, TCP, UDP, Protocols for Gigabit Networks.			
Unit-V	**Application Layer:-** Network Security, Cryptography, Secret and Public Key Algorithms, DNS, SNMP, Electronic Mail, Electronic Mail Privacy. **World Wide Web:-** Client Side, Server Side, Multimedia, Audio, Video, Data compression, JPEG, MPEG Standards.			

Reference and Textbooks:

Text Book:

Andrew Tenenbaum, S. (2010). *Computer Networks* (5th ed.). Prentice Hall of India.

Books for Reference:

Behrouz Forouzen, A. (2017). *Data Communication and Networking*. Tata Mc Graw-Hill Edition.

Stallings, W. (2013). *Data and Computer Communications*. PHI.

Outcomes	On Completion of this Course, the students can able to ➢ Recognize the technological trends of Computer Networking. ➢ Discuss the key technological components of the Network. ➢ Evaluate the challenges in building networks and solutions..

Semester - IV				
Course code 22BCA4P1	**Core Practical - IV**	**T/P**	**C**	**H/W**
	Java Programming Lab	**P**	**3**	**3**

Objectives

- To impart Practical Training in JAVA Programming Language.
- Familiarize the different control and decision making statements in JAVA.
- Build programs using Packages and working with Exception handling functions.

Lab Programs

1. Write a JAVA program to display default value of all primitive data type of JAVA.
2. Write a JAVA program to implement class mechanism. – Create a class, methods and invoke them inside main method.
3. Write a JAVA program to implement constructor.
4. Write a JAVA program to implement constructor overloading.
5. Write a JAVA program implement method overloading.
6. Write a JAVA program to implement Single Inheritance
7. Write a JAVA program to implement multi level Inheritance
8. Write a java program for abstract class to find areas of different shapes
9. Write a JAVA program to implement Interface.
10. Write a JAVA program that describes exception handling mechanism
11. Write a JAVA program Illustrating Multiple catch clauses
12. Write a JAVA program that implements Runtime polymorphism.
13. Write a JAVA program for creation of Illustrating throw
14. Write a JAVA program for creation of Illustrating finally
15. Write a JAVA program for creation of Java Built-in Exceptions
16. Write a JAVA program for creation of User Defined Exception
17. Write a JAVA program that import and use the defined your package.

Applet

18. Write a JAVA program to paint like paint brush in applet.
19. Write a JAVA program to display analog clock using Applet.
20. Write a JAVA program to create different shapes and fill colors using Applet.
21. Write a program to draw House.
22. Write a program to draw our National Flag.
23. Write a program to Draw Bar Charts.
24. Write a JAVA program that identifies key-up key-down event user entering text in a Applet.

Outcomes

On Completion of this Course, the students can able to

- Study all the Basic Statements in java Programming.
- Practice the usage of branching and looping statements.
- Apply Packages, Interfaces, Analysis the use of graphics tools in JAVA.

Semester - V				
Course code 22BCA5C1	**Core Course - VII**	**T/P**	**C**	**H/W**
	. NET Programming	**T**	**4**	**4**
Objectives	➢ To explain how to create dynamic Web pages by using ASP.NET. ➢ To configure an ASP.NET application. ➢ To create a user interface on an ASP.NET page by using standard Web server controls. ➢ To create a user control and a custom server control and add them to an ASP.NET page.			
Unit -I	**Introduction:** Overview of Microsoft .NET Framework, The .NET Framework components, The Common Language Runtime (CLR) Environment, The .NET Framework class Library. **Getting Started with Visual Basic .net IDE:-** Set up of work environment, start page, the menu system, toolbars, the new project dialog box, graphical designers, code designers, the object explorer, the toolbox, the solution explorer, the class view window, the properties window, the dynamic help window, the server explorer, the output window, the command window. **Visual basic Language Concept:-** variables, Constants, Data Types, Operators, Control Structures and loops, **Arrays:-** single and multidimensional array, declaring, dynamic array.			
Unit-II	**Introduction to Windows Common Controls:-** Working with Form, Properties: appearance, behaviour, layout, windows style etc, methods and events - Differentiate procedure oriented, object oriented and event driven programming – Input box- Message box. **Working with Common Tool Box Controls:**- Label, button, Textbox, NumericUpDown, Check Box, Radio Button, Group Box, control and all important methods and events.			
Unit-III	**Additional Controls and Menus of Windows:-** Working with other controls of toolbox: Date Time Picker, List Box, Combo box, Picture Box, Rich Text Box, Progress bar, Masked Text box, Link Label, Checked List box. **Working with Menus:**- creating menu, inserting, deleting, assigning short cut keys, popup menu.			
Unit-IV	**Inbuilt Functions and Dialog Box:- Inbuilt Functions:** Mathematical Functions, String manipulation. **Dialog Boxes:**- OpenFileDialog, SaveFileDialog, FontDialog, ColorDialog, PrintDialog. **Sub Procedures and Functions:-** declaring, passing and returning arguments, exiting from it, pass by value and pass by ref. **Exception Handling:-** Structured Error Handling (Try ....Catch ....finally), Unstructured Error Handling (On error go to line, goto 0, goto -1, resume next) - Multiple document interface (MDI) : MDI Parent form and child form.			
Unit-V	**Database Access using Ado.Net:-** ADO .NET Object Model, Dataprovider, Dataset, **ADO .NET Programming:**- Creating a Database Application, Creating Connection to a Database using ADO.NET, Populating Data in ADO.NET, Browsing Records, Datagrid view, Editing, Saving, Adding and Deleting Records using bounded and unbounded.			

Reference and Textbooks:

Text Books:

Julia Bradley, C., & Anita Millspaugh, C. (2002). *Programming in Visual Basic .NET*. Tata Mc Graw- Hill. Higher Education.

Shelly, Cashman, & Quasney (2012). *Microsoft Visual Basic .NET : Comprehensive Concepts and Techniques*. Cengage learning.

Steven Holzner. *Visual Basic .NET Programming*. New Delhi: Black Book. Dreamtech Press Publications.

Outcomes	On Completion of this Course, the students can able to ➢ Understand the Microsoft .NET Framework and ASP.NET page structure. ➢ Design web application with variety of controls. ➢ Access the data using inbuilt data access tools. ➢ Use Microsoft ADO.NET to access data in web Application..

Semester - V				
Course code 22BCA5C**2**	**Core Course - VIII**	**T/P**	**C**	**H/W**
	Python Programming	**T**	**4**	**4**

Objectives	➢ Describe the core syntax and semantics of Python programming language. ➢ Discover the need for working with the strings and functions. ➢ Illustrate the process of structuring the data using lists, dictionaries, tuples and sets. ➢ Understand the usage of packages and Dictionaries.
Unit -I	**Introduction Data, Expressions, Statements:-** Introduction to Python, Features of Python, Installation of Python, Python Indentation, Variables and Identifiers, Keywords, Data types, Python operators, Expressions, Input/Output functions, Create your First Python Program.
Unit-II	**Control Flow, Loops, Functions:-** Conditional statement-if, if-else, elif, Nested if-Pass statement- **Iteration:**- While, For, Break, Continue, Function, Defining a Function, Calling A Function, Function Arguments, Recursive Function, Function Returning More Than One Value, Lambda functions.
Unit-III	**Arrays, Modules and Package:-** Python arrays, Access the Elements of an Array, array methods, Numpy. **Modules Overview**:- Modules Search Path, Import Statement, dir() Function, Executing A Module, Renaming A Module, Python Packages, Packages initialization, Importing modules from a package, Sub Packages.
Unit-IV	**Dictionaries, Sets Lists, Tuples: -** Dictionary type in Python, Set Data type, Lists type in Python, Tuple type in Python. **Object Oriented Programming using Python:**- Encapsulation, Inheritance, Polymorphism
Unit-V	**Errors and Exception Handling, Files:-** Errors, Exception Handling, try block, except block and finally block. **Files:**- Opening a File, Closing a File, Reading And Writing a File, File Methods, Renaming and Deleting A File, Built-in file directories in Python.

Reference and Textbooks:

TEXT BOOKS:

Charles Dierbach (2015). *Introduction to Computer Science using Python - A Computational Problem Solving Focus.* Wiley India Edition.

REFERENCE BOOKS:

Satyanarayana, Ch., Radhika Mani, M., & Jagadesh, B.N. (2018). *Python programming*. Universities Press.

Timothy Budd, A. (2011). *Exploring Python* (1st ed.). Tata MC Graw-Hill Education Pvt. Ltd.

WEB RESOURCES

https://www.w3schools.com/python/default.asp

https://www.tutorialspoint.com/python3/python_tutorial.pdf

Outcomes	On Completion of this Course, the students can able to ➢ Develop Packages by importing appropriate modules. ➢ Develop the emerging applications of relevant field using Python. ➢ Interpret the fundamental Python syntax and semantics and be fluent in the use of Python control flow statements. ➢ Apply the concept of Sets , dictionaries & tuples in Python. ➢ Understand the principles of Python and acquire skills in programming in python.

Semester - V				
Course code 22BCA5C3	**Core Course - IX**	**T/P**	**C**	**H/W**
	Web Design Technology	**T**	**4**	**4**
Objectives	➢ To introduce the fundamentals of Internet, and the principles of web design. ➢ To construct basic websites using HTML and Cascading Style Sheets. ➢ To build dynamic web pages with validation using Java Script objects and by applying different event handling mechanisms. ➢ To develop modern interactive web applications using PHP, XML and MySQL.			
Unit -I	**Introduction:** Concept of WWW, Internet and WWW, HTTP Protocol: Request and Response, Web browser and Web servers, Features of latest version of Web. **Web Design:** Concepts of effective web design, Web design issues including Browser, Bandwidth and Cache, Display resolution, Look and Feel of the Website, Page Layout and linking, User centric design, Sitemap, Planning and publishing website, Designing effective navigation.			
Unit-II	**HTML:-** Basics of HTML, formatting and fonts, commenting code, color, hyperlink, lists, tables, images, forms, XHTML, Meta tags, Character entities, frames and frame sets, Browser architecture and Web site structure. Overview and features of latest version of HTML. **Style sheets:-** Need for CSS, introduction to CSS, basic syntax and structure, using CSS, background images, colors and properties, manipulating texts, using fonts, borders and boxes, margins, padding lists, positioning using CSS, CSS2, Overview and features of of latest version of CSS.			
Unit-III	**JavaScript:-** Client side scripting with JavaScript, variables, functions, conditions, loops and repetition, Pop up boxes. **Advance JavaScript:-** Javascript and objects, JavaScript own objects, the DOM and web browser environments, Manipulation using DOM, forms and validations, **DHTML:-** Combining HTML, CSS and Javascript, Events and buttons.			
Unit-IV	**XML:-** Introduction to XML, uses of XML, simple XML, XML key components, DTD and Schemas, Using XML with application, Transforming XML using XSL and XSLT. **PHP:-** Introduction and basic syntax of PHP, decision and looping with examples, PHP and HTML, Arrays, Functions, Browser control and detection, string, Form processing, Files, Advance Features: Cookies and Sessions			
Unit-V	**PHP and MySQL:-** Basic commands with PHP examples, Connection to server, creating database, selecting a database, listing database, listing table names, creating a table, inserting data, altering tables, queries, deleting database, deleting data and tables, PHP myadmin and database bugs.			

Reference and Textbooks:

Text Books

Ralph Moseley & Savaliya, M. T. (2011). *Developing Web Applications*. Wiley-India Pvt. Ltd.

Robert Sebesta, W. (2013). *Programming the World Wide Web* (7th ed.). Pearson Education.

REFERENCES

<table>
<tr><td colspan="2">Harwani, B. M. (2010). Developing Web Applications in PHP and AJAX. Tata McGraw-Hill.
Joel Sklar. (2015). Principles of Web Design (6th ed.). Cengage Learning.
Paul Deitel, J., Harvey Deitel, M., & Deitel, A. (2011). Internet and World Wide Web How to program (5th ed.). Pearson Education.</td></tr>
<tr><td>Outcomes</td><td>On Completion of this Course, the students can able to
Describe the concepts of World Wide Web & requirements of effective web design.
Develop web pages using HTML and CSS features with different layouts as per need of applications.
Use the JavaScript to develop the dynamic web pages.
Construct simple web pages in PHP and to represent data in XML format.
Use server side scripting with PHP to generate the web pages dynamically using the database connectivity.</td></tr>
</table>

Semester - V				
Course code: 22BCA5C4	**Core Course - X**	**T/P**	**C**	**H/W**
	Computer Architecture and Organization	**T**	**4**	**4**
Objectives	➢ Discuss the basic concepts and structure of computers. ➢ Understand concepts of register transfer logic and arithmetic operations. ➢ Explain different types of addressing modes and memory organization. ➢ Learn the various types of serial communication techniques.			
Unit -I	**Data Representation:-** Data Types, Complements, Fixed Point Representation, Floating Point Representation, Other Binary Codes, rror Detection Codes. **Register Transfer and Micro operations:-** Register Transfer Language, Register Transfer, Bus and Memory Transfers, Arithmetic Microoperations, Logic Microoperations, Shift Microoperations.			
Unit-II	**Basic Computer Organization and Design:-** Instruction Codes, Computer Registers, Computer Instructions, Instruction Cycle, Memory Reference Instructions, Input-Output and Interrupt.			
Unit-III	**Programming the Basic Computer:-** Introduction, Machine Language, Assembly Language, The Assembler, Program Loops, Programming Arithmetic and Logic Operations.			
Unit-IV	**Microprogrammed Control:-** Control Memory, Address Sequencing, Microprogram Example, Design of Control Unit.			
Unit-V	**Central Processing Unit:-** Introduction, General Register Organization, Stack Organization, Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Program Control, Reduced Instruction Set Computer (RISC).			

Reference and Textbooks:

TEXT BOOKS:

Morris Mano, M. (2017). *Computer System Architecture* (3rd ed.). PHI Pvt. Ltd.

REFERENCE BOOKS:

Smruti Ranjan Sarangi (2015). *Computer Organisation and Architecture*. TATA Mc Graw-Hill Education Pvt. Ltd.

WEB RESOURCES

https://byjusexamprep.com/computer-science-engineering-exams/computer-organization-and-architecture

https://www.geektonight.com/computer-organization-and-architecture-notes/

https://mu.ac.in/wp-content/uploads/2021/03/COA_Full.pdf

Outcomes	On Completion of this Course, the students can able to ➢ Understand the theory and architecture of central processing unit. ➢ Design a simple CPU with applying the theory concepts. ➢ Understand the architecture and functionality of central processing unit. ➢ Exemplify in a better way the I/O and memory organization. ➢ Define different number systems, binary addition and subtraction, 2's complement representation and operations with this representation.

Semester - V				
Course code 22BCA5P1	**Core Practical - V**	**T/P**	**C**	**H/W**
	Python Programming Lab	**P**	**4**	**6**

Objectives

- To implement the python programming features in practical applications.
- To write, test, and debug simple Python programs.
- To implement Python programs with conditionals and loops.
- Use functions for structuring Python programs and represent compound data using Python lists, tuples, dictionaries, turtles, Files and modules.

Lab Programs

1. Program to convert the given temperature from Fahrenheit to Celsius and vice versa depending upon user's choice.
2. Program to calculate total marks, percentage and grade of a student. Marks obtained
   in each of the five subjects are to be input by user. Assign grades according to the following criteria:
   Grade A: Percentage >=80 Grade B: Percentage >=70 and <80
   Grade C: Percentage >=60 and <70 Grade D: Percentage >=40 and <60
   Grade E: Percentage <40
3. Program, to find the area of rectangle, square, circle and triangle by accepting suitable input parameters from user.
4. Program to display the first n terms of Fibonacci series.
5. Program to find factorial of the given number using recursive function.
6. Write a Python program to count the number of even and odd numbers from array of N numbers.
7. Python function that accepts a string and calculate the number of upper case letters and lower case letters.
8. Python program to reverse a given string and check whether the give string is palindrome or not.
9. Write a program to find sum of all items in a dictionary.
10. Write a Python program to construct the following pattern, using a nested loop
    1
    22
    333
    4444
    55555
    666666
    7777777
    88888888
    999999999
11. Read a file content and copy only the contents at odd lines into a new file.
12. Create a Turtle graphics window with specific size.
13. Write a Python program for Towers of Hanoi using recursion
14. Create a menu driven Python program with a dictionary for words and their meanings.
15. Devise a Python program to implement the Hangman Game.

Outcomes

On Completion of this Course, the students can able to

- Understand the numeric or real life application problems and solve them.
- Apply a solution clearly and accurately in a program using Python.
- Apply the best features available in Python to solve the situational problems.

Semester - V				
Course code 22BCA5P2	**Core Practical - VI**	**T/P**	**C**	**H/W**
	Web Design Technology Lab	**P**	**4**	**6**

Objectives

- To impart Practical Training in Control panel tools & familiarize with HTML Tags.
- To build programs using Java script and to provide knowledge on working with events and methods.

Lab Programs

1. HTML

a. Create a table to show your class time table.
b. Use tables to provide layout to your HTML page describing your college infrastructure.
c. Use and and tags to provide a layout to the above page instead of a table layout.

2. HTML

a. Use frames such that page is divided into 3 frames 20% on left to show contents of pages, 60% in center to show body of page, remaining on right to show remarks.
b. Embed Audio and Video into your HTML web page.

3. HTML

a. Create a webpage with HTML describing your department use paragraph and list tags.
b. Apply various colors to suitably distinguish key words, also apply font styling like italics, underline and two other fonts to words you find appropriate , also use header tags.
c. Create links on the words e.g. ―Wi-Fi and ―LAN‖ to link them to Wikipedia pages.
d. Insert an image and create a link such that clicking on image takes user to other page.
e. Change the background color of the page; At the bottom create a link to take user to the top of the page.

4. CASCADING STYLE SHEET

Write an HTML page that contains a selection box with a list of 5 countries, when the user selects a country, its capital should be printed next to the list; Add CSS to customize the properties of the font of the capital (color, bold and font size).

5. JAVASCRIPT

a. Write a java script program to test the first character of a string is uppercase or not.
b. Write a pattern that matches e-mail addresses.
c. Write a java script function to print an integer with commas as thousands separators.

6. JAVASCRIPT

a. Write a java script program which compute, the average marks of the following students then this average is used to determine the corresponding grade.
b. Write a java script program to sum the multiple s of 3 and 5 under 1000.

	c. To design the scientific calculator and make event for each button using java script **7. PHP** a. A simple calculator web application that takes two numbers and an operator (+, ,/,*and %) from an HTML page and returns the result page with the operation performed on the operands. b. Write PHP program how to send mail using PHP. **8. PHP** a. Write PHP program to convert a string, lower to upper case and upper case to lower case or capital case. b. Write PHP program to change image automatically using switch case. c. Write PHP program to calculate current age without using any pre-define function. d. Write PHP program to upload image to the server using html and PHP. **9.PHP** a. Write PHP program to upload registration form into database. b. Write PHP program to display the registration form from the database **10.PHP** a. Write PHP program to update the registration form present in database. b. Write PHP program to delete the registration form from database.
Outcomes	➢ Demonstrate the ability to retrieve data from a database and present it in a web page. ➢ Use FTP to transfer web pages to a server, Construct pages that meet guidelines for efficient download and cater to the needs of an identified audience. ➢ Evaluate the functions of specific types of web pages in relationship to an entire web site. ➢ Create web pages that meet accessibility needs of those with physical disabilities and apply the effects of CSS in web page creation..

Semester - VI				
Course code: 22BCA6E1	**Discipline Specific Elective - I**	**T/P**	**C**	**H/W**
	(A) Data Mining and Warehousing	**T**	**6**	**6**
Objectives	➢ Teach the basic principles, concepts and applications of data mining & warehousing. ➢ Introduce the task of data mining as an important phase of knowledge recovery process. ➢ To familiarize Conceptual, Logical, and Physical design of Data Warehouses OLAP applications and OLAP deployment.			
Unit -I	**Introduction to Data Mining:-** Motivation, Importance, Definition of Data Mining, Kind of Data,Data Mining Functionalities, Kinds of Patterns, Classification of Data Mining Systems, Data Mining Task Primitives, Integration of A Data Mining System With A Database or Data Warehouse System, Major Issues In Data Mining, Types of Data Sets and Attribute Values, Basic Statistical Descriptions of Data, Data Visualization, Measuring Data Similarity. **Preprocessing:-** Data Quality, Major Tasks in Data Preprocessing, Data Reduction, DataTransformation and Data Discretization, Data Cleaning and Data Integration.			
Unit-II	**Data Warehousing and On-Line Analytical Processing:-** Data Warehouse basicconcepts, Data Warehouse Modeling - Data Cube and OLAP, Data Warehouse Design and Usage, Data Warehouse Implementation, Data Generalization by Attribute-Oriented Induction. **Data Cube Technology:-** Efficient Methods for Data Cube Computation, Exploration andDiscovery in Multidimensional Databases.			
Unit-III	**Mining Frequent Patterns, Associations and Correlations:-** Basic Concepts,Efficient and Scalable Frequent Item set Mining Methods, Are All the Pattern Interesting, Pattern Evaluation Methods, Applications of frequent pattern and associations. **Frequent Pattern and Association Mining:-** A Road Map, Mining Various Kinds ofAssociation Rules, Constraint-Based Frequent Pattern Mining, Extended Applications of Frequent Patterns.			
Unit-IV	**Classification:-** Basic Concepts, Decision Tree Induction, Bayesian Classification Methods, Rule-Based Classification, Model Evaluation and Selection. **Techniques to Improve Classification Accuracy:-** Ensemble Methods, Handling Different Kinds of Cases in Classification, Bayesian Belief Networks, Classification by Neural Networks, Support Vector Machines, Pattern-Based Classification, Lazy Learners (or Learning from Your Neighbors), Other Classification Methods.			
Unit-V	**Cluster Analysis:-** Basic Concepts of Cluster Analysis, Clustering structures, Major ClusteringApproaches, Partitioning Methods, Hierarchical Methods, Density-Based Methods, ModelBased Clustering, The Expectation-Maximization Method, Other Clustering Techniques, Clustering High-Dimensional Data, Constraint-Based and User-Guided Cluster Analysis, Link-Based Cluster Analysis, Semi-Supervised Clustering and Classification, Bi-Clustering, Collaborative Clustering. **Outlier Analysis:-** Why outlier analysis, Identifying and handling of outliers, DistributionBasedOutlier Detection: A Statistics-Based Approach, Classification-Based Outlier Detection, Clustering-			

	Based Outlier Detection, Deviation-Based Outlier Detection, Isolation-Based Method: From Isolation Tree to Isolation Forest.

Reference and Textbooks:

Text Book :

Amitesh Sinha (2007). *Data Warehousing.* India: Thomson Learning.

Jiawei Han, MichelineKamber, & Jian Pei (2012). *Data Mining: Concepts and Techniques* (3rd ed.). USA: Elsevier.

References:

Margaret Dunham, H. (2006). *Data Mining Introductory and Advanced Topics* (2nd ed.). New Delhi: Pearson Education.

Xingdong Wu & Vipin Kumar (2009). *The Top Ten Algorithms in Data Mining*. UK: CRC Press.

Outcomes	After undergoing the course, Students will be able to understand ➢ Design a data mart or data warehouse for any organization. ➢ Skill to write queries using DMQL & Extract knowledge using data mining techniques. ➢ Adapt to new data mining tools, Apply the techniques of clustering, classification, association finding, feature selection and visualization to real world data.

Semester - VI				
Course code 22BCA6E2	**Discipline Specific Elective - I**	**T/P**	**C**	**H/W**
	(B) Artificial Intelligence	**T**	**6**	**6**
Objectives	➢ To learn the concepts of Artificial Intelligence. ➢ Create awareness of informed search and exploration methods. ➢ To demonstrate AI techniques for knowledge representation, planning & uncertainty.			
Unit -I	**Logic:- Logical Agents:-** Knowledge-based agents, The Wumpus world. **Logic Propositional logic:-** A very simple logic, **First order logic:-** Representation revisited, Syntax and semantics for first order logic, Using first order logic, Knowledge engineering in first order logic. **Inference in First order logic:-** propositional versus first order logic, forward chaining, backward chaining.			
Unit-II	**Decision Making and Learning: - Making Simple Decisions:** The basis of Utility theory, Utility and multi-attribute utility functions, decision networks, The value of information, Decision theoretic expert systems. **Learning from Observations:-** Forms of learning - Inductive learning - Learning decision trees. **Knowledge in Learning:-** Logical formulation of learning, Explanation based learning, Learning using relevant information, Inductive logic programming.			
Unit-III	**Planning and Uncertainty:- Planning:** The planning problem, planning with state, space search, partial order planning, graphs. **Uncertainty:-** Overview of probability concepts, Representing knowledge in an Uncertain Domain, Semantics of Bayesian Networks, Exact Inference in Bayesian Networks.			
Unit-IV	**Decision Making and Learning:- Making Simple Decisions:** The basis of Utility theory, Utility and multi-attribute utility functions, decision networks, The value of information, Decision theoretic expert systems. **Learning from Observations:-** Forms of learning, Inductive learning, Learning decision trees. **Knowledge in Learning:-** Logical formulation of learning, Explanation based learning, Learning using relevant information, Inductive logic programming.			
Unit-V	**Learning and Communication:- Statistical Learning Methods:** Introduction to neural networks, Perceptron's, Multi-layer feed forward network, Application of ANN. **Reinforcement Learning:-** Passive reinforcement learning, Active reinforcement learning, Generalization in reinforcement learning. **Communication:-** Communication as action, Formal grammar for a fragment of English, Syntactic analysis, Augmented grammars, Semantic interpretation, Ambiguity and disambiguation.			

Reference and Textbooks:

Text Book :

Stuart Russell & Peter Norvig (2009). *Artificial Intelligence – A Modern Approach* (3rd ed.). Pearson Education / Prentice Hall of India.

References:

Elaine Rich, Kevin Knight, & Shivashankar Nair, B. (2009). *Artificial Intelligence* (3rd ed.). Tata Mc Graw-Hill Publishing Co. Ltd.

George Luger, F. (2002). *Artificial Intelligence-Structures and Strategies for Complex Problem Solving*. Pearson Education / PHI.

Nils Nilsson, J. (2000). *Artificial Intelligence: A new Synthesis*. Harcourt Asia Pvt. Ltd.

Outcomes	On Completion of this Course, the students can able to ➢ Solve basic AI based problems. ➢ Define the concept of Artificial Intelligence. ➢ Apply AI techniques to real-world problems to develop intelligent systems..

Semester - VI				
Course code: 22BCA6E3	**Discipline Specific Elective - II**	**T/P**	**C**	**H/W**
	(A) Software Engineering	**T**	**6**	**6**
Objectives	➢ To provide an understanding and working knowledge of the techniques for estimation, design, testing and quality management of large software development projects.			
Unit -I	**Introduction to Software Engineering:-** The evolving role of software, changing nature of software, software myths. **A Generic view of process:-** Software engineering- a layered technology, a process framework, the capability maturity model integration (CMMI), process patterns, process assessment, personal and team process models. **Process models:-** The waterfall model, incremental process models, evolutionary process models, the unified process.			
Unit-II	**Software Requirements:-** Functional and non-functional requirements, user requirements, system requirements, interface specification, the software requirements document. **Requirements engineering process:-** Feasibility studies, requirements elicitation and analysis, requirements validation, requirements management. **System models:-** Context models, behavioral models, data models, object models, structured methods.			
Unit-III	**Design Engineering:-** Design process and design quality, design concepts, the design model. **Creating an architectural design:-** software architecture, data design, architectural styles and patterns, architectural design, conceptual model of UML, basic structural modeling, class diagrams, sequence diagrams, collaboration diagrams, use case diagrams, component diag			
Unit-IV	**Testing Strategies:-** A strategic approach to software testing, test strategies for conventional software, black-box and white-box testing, validation testing, system testing, the art of debugging. **Product metrics:-** Software quality, metrics for analysis model, metrics for design model, metrics for source code, metrics for testing, metrics for maintenance.			
Unit-V	**Metrics for Process and Products:-** Software measurement, metrics for software quality. Risk management: Reactive Vs proactive risk strategies, software risks, risk identification, risk projection, risk refinement, RMMM, RMMM plan. **Quality Management:-** Quality concepts, software quality assurance, software reviews, formal technical reviews, statistical software quality assurance, software reliability, the ISO 9000 quality standards.			

Reference and Textbooks:

Text Book :

Booch, G., Rambaugh, J., & Jacobson, I. (2013). *The unified modeling language user guide.* Pearson Education.

Roger Pressman, S. (2004). *Software Engineering, A practitioner's Approach* (6th ed.). TATA Mc Graw-Hill International Edition.

Sommerville, I. (2004). *Software Engineering* (7th ed.). Pearson Education.

References:

James Peters, F., & Witold Pedrycz. *Software Engineering - an Engineering approach*. John Wiley.

Jones. *Fundamentals of object-oriented design using UML*. Pearson Education.

Waman Jawadekar, S. *Software Engineering principles and practice*. TATA Mc Graw-Hill.

Outcomes	On Completion of this Course, the students can able to ➢ Ability to translate end-user requirements into system and software requirements. ➢ Identify and apply appropriate software architectures and patterns to carry out high level design of a system and be able to critically compare alternative choices. ➢ Will have experience and/or awareness of testing problems and will be able to develop a simple testing report.

Semester - VI				
Course code 22BCA6E4	**Discipline Specific Elective - II**	**T/P**	**C**	**H/W**
	(B)Internet of Things	**T**	**6**	**6**
Objectives	➢ To learn the concepts of IOT and its protocols. ➢ To learn how to analysis the data in IOT. ➢ To develop IOT infrastructure for popular applications.			
Unit -I	**Introduction to Io:-** Genesis of IoT, IoT and Digitization, IoT Impact, Convergence of IT and OT, IoT Challenges, IoT Network Architecture and Design, Drivers Behind New Network Architectures, Comparing IoT Architectures, Additional IoT Reference Models.			
Unit-II	**The Core IoT Functional Stack:-** IoT Data Management and Compute Stack, Fog Computing, Edge Computing, The Hierarchy of Edge, Fog and Cloud-Smart Objects, The Things in IoT-Sensors, Actuators and Smart Objects, Sensor Networks, Wireless Sensor Networks, Communication Protocols for Wireless Sensor Networks.			
Unit-III	**Connecting Smart Objects**:- Communications Criteria, IoT Access Technologies, Standardization and Alliances. **Competitive Technologies:-** IEEE 802.15.4, IEEE 802.15.4g and 802.15.4e, IEEE 1901.2a, IEEE 802.11ah, LoRaWAN- NB-IoT and Other LTE Variations UCA90.			
Unit-IV	**IP as the IoT Network Layer:**- The Business Case for IP, Optimizing IP for IoTAuthentication and Encryption on Constrained Nodes, ACE, DICE, Application Protocols for IoT. **The Transport Layer:**- IoT Application Transport Methods, SCADA, Generic WebBased Protocols, IoT Application Layer Protocols, CoAP.			
Unit-V	**IoT in Industry:-** Transportation, Transportation Challenges, IoT Use Cases for Transportation, An IoT Architecture for Transportation, Extending the Roadways IoT, Architecture to Bus Mass Transit, Extending Bus IoT Architecture to Railways, Public Safety, Public and Private Partnership for Public Safety IoT, An IoT Blueprint for Public Safety Emergency Response IoT Architecture, School Bus Safety, School Bus Safety Network Architecture.			

Reference and Textbooks:

Text Book

Hanes, D., Salgueiro, G., Grossetete, P., Barton, R., & Henry, J. (2017). *IoT Fundamentals: Networking Technologies, Protocols and Use Cases for Internet of Things.* Cisco Press.

Reference Books

Arshdeep, B., & Vijay, M. (2015). *Internet of Things – A hands-on approach*. Universities Press.

Honbo Zhou (2012). *The Internet of Things in the Cloud: A Middleware Perspective*. CRC Press.

Olivier Hersent, David Boswarthick, & Omar Elloumi (2012). *The Internet of Things. Key Applications and Protocols.* Wiley.

Outcomes	➢ On Completion of this Course, the students can able to build and test a complete, working IoT system involving prototyping, programming and data analysis.

Semester - VI				
Course code 22BCA6E5	**Discipline Specific Elective - III**	**T/P**	**C**	**H/W**
	(A)Cloud Computing	**T**	**6**	**6**
Objectives	➢ To learn the concept of Cloud Computing basics, cloud storage and Standards. ➢ To learn the concepts Azure and Azure documentation.			
Unit -I	**Cloud Computing Basics:-** Cloud Computing Overview, Applications, Intranets and the Cloud. **Hardware and Infrastructure:-** Clients, Security, Network, Services.			
Unit-II	**Accessing the Cloud :** Platforms – Web Applications – Web Browsers. Cloud Storage: Overview – Cloud Storage Providers. Standards: Application – Client – Infrastructure – Service.			
Unit-III	**Getting started with Microsoft Azure:-** What is Azure? Azure Resource Manager, PowerShell changes for the Resource Manager and classic deployment models, Role, Based Access Control, The Azure portal, Subscription Management and Billing, Azure Documentation and Samples.			
Unit-IV	**App Service and App Service plans**:- Creating and Deploying Web Apps. Configuring, scaling and monitoring Web Apps. What is Azure Virtual Machines? Virtual machine models, Virtual machine components, Create virtual machines, Connecting to a virtual machine, Configuring and managing a virtual machine, Scaling Azure Virtual Machines.			
Unit-V	**Azure Storage:-** Storage accounts, Storage services, Security and Azure Storage. **Creating and managing storage:-** Create a storage account using the Azure portal, Create a file share and upload files using the Azure portal, Create a table and add records using the Visual Studio Cloud Explorer, Create a storage account using PowerShell, Create a container and upload blobs using PowerShell. **AzCopy:-** A very useful tool , The Azure Data Movement Library.			

Reference and Textbooks:

Text Book

Anthony Velte, T., Toby Velte, J., Elsenpeter, R. (2010). *Cloud Computing – A Practical Approach* (Unit I & II). TMH.

Michael Collier & Robin Shahan (2015). *Fundamentals of Azure* (2nd ed.). Microsoft Press.

Reference Book

Haley Beard (2008). *Cloud Computing Best Practices for measuring processes for on demand computing. Applications and data centers in the cloud with SLA's.*

Hash Bai, Steve Maier, Dan Stolts Architecting Microsoft Azure Solutions. Eastern Economy Edition.

Michael Miller (2009). *Cloud Computing – Web based Application.* Pearson Edu. Inc.

Rajkumar Buyya, Christian Vecchiola, & Thamarai Selvi (2013). *Mastering Cloud computing* (Unit II & IV). Mc Gram Hill Edu.

Outcomes	On Completion of this Course, the students can able to ➢ Idea in cloud computing concepts, characteristics, delivery models and benefits. ➢ Understand the key security and compliance challenges of cloud computing. ➢ Understand the key technical and organisational challenges. ➢ Understand the different characteristics of public, private and hybrid cloud deployment models..

Semester - VI				
Course code 22BCA6E6	**Discipline Specific Elective - III**	**T/P**	**C**	**H/W**
	(B)Mobile Application Development	**T**	**6**	**6**
Objectives	➢ To provide an overall knowledge about Mobile Devices, Communication methodologies and its application development.			
Unit -I	**Introduction:**- The Mobile Ecosystem, Operators, Networks, Devices, Platforms, Operating Systems, Application Frameworks, Applications, Services.			
Unit-II	**Mobile Devices Profiles:-** Options for development, **Categories of Mobile Applications:**- SMS, Mobile Websites, Mobile Web Widgets, Native Applications, Games, Utility Apps, Location Based Services (LBS) Apps, Informative Apps, Enterprise Apps.			
Unit-III	**Mobile Information Architecture:**- Introduction, Sitemaps, Click Streams, Wireframes, Prototyping, Architecture for Different Devices. **Mobile Design:**- Interpreting Design, Elements of Mobile Design, Mobile Design Tools, Designing for Different Device Screens.			
Unit-IV	**J2ME Overview:**- -J2ME Architecture and Development Environment, Small Computing Device Requirements, Run-Time Environment, MIDlet Programming, Java Language for J2ME, J2ME SDK, J2ME Wireless Toolkit.			
Unit-V	**Case Study**: **Google Android:-** Introduction, Android Development Environment. Development Framework, SDK, Eclipse, Emulator, Android AVD, Project Framework. **Apple IOS:-** RIM Blackberry, Samsung Bada, Nokia Symbian, Microsoft Windows Phone.			

Reference and Textbooks:

Text Books:

Fling, B. (2009). *Mobile Design and Development*. OReilly Media, Inc.

Keogh, J. (2003). *J2ME: The Complete Reference*. Tata McGraw-Hill.

References Books:

Mark Murphy, L. (2009). *Beginning Android*. Apress.

Zheng, P., & Ni, L. (2006). *Smart Phone and Next-Generation Mobile Computing*. Elseveir.

Outcomes	On Completion of this Course, the students can able to ➢ Install and configure Android application development tools. ➢ Design and develop user Interfaces for the Android platform..

Semester - VI				
Course code	**Discipline Specific Elective - IV**	**T/P**	**C**	**H/W**
22BCA6E7	**(A)Fundamentals of Digital Image Processing**	**T**	**6**	**6**
Objectives	➢ To learn digital image fundamentals. ➢ To be exposed to simple image processing techniques. ➢ To be familiar with image compression and segmentation techniques. ➢ To learn to represent image in form of features.			
Unit -I	**Digital Image Fundamentals: -** Introduction, Origin, Steps in Digital Image Processing, Components, Elements of Visual Perception, Image Sensing and Acquisition, Image Sampling and Quantization, Relationships between pixels, Color models.			
Unit-II	**Image Enhancement:- Spatial Domain:-** Gray level transformations, Histogram processing, Basics of Spatial Filtering, Smoothing and Sharpening Spatial Filtering. **Frequency Domain:** - Introduction to Fourier Transform, Smoothing and Sharpening, Frequency Domain Filters, Ideal, Butterworth and Gaussian filters.			
Unit-III	**Image Restoration and Segmentation:-** Noise models, Mean Filters, Order Statistics, Adaptive filters, Band reject Filters, Band pass Filters, Notch Filters, Optimum Notch Filtering, Inverse Filtering. **Wiener Filtering Segmentation:-** Detection of Discontinuities–Edge Linking and Boundary detection – Region based segmentation-Morphological processing- erosion and dilation.			
Unit-IV	**Wavelets and Image Compression:-** Wavelets, Subband coding, Multiresolution expansions. **Compression:-** Fundamentals, Image Compression models, Error Free Compression, Variable Length Coding, Bit-Plane Coding, Lossless Predictive Coding, Lossy Compression, Lossy Predictive Coding, Compression Standards.			
Unit-V	**Image Representation and Recognition:-** Boundary Representation, Chain Code, Polygonal approximation, signature, boundary segments, Boundary description – Shape number – Fourier Descriptor, moments- Regional Descriptors – Topological feature, Texture - Patterns and Pattern classes - Recognition based on matching.			

Reference and Textbooks:

TEXT BOOK:

Rafael Gonzales, C., & Richard Woods, E. (2010). *Digital Image Processing* (3rd ed.). Pearson Education.

REFERENCES:

Anil Jain, K. (2011). *Fundamentals of Digital Image Processing*. PHI Pvt. Ltd.

Malay Pakhira, K. (2011). *Digital Image Processing and Pattern Recognition* (1st ed.). PHI Pvt. Ltd.

Rafael Gonzales, C., Richard Woods, E., & Steven Eddins, L. (2011). *Digital Image Processing using MATLAB* (3rd ed.). Tata Mc Graw-Hill Pvt. Ltd.

Willliam Pratt, K. (2002). *Digital Image Processing*. John Willey.

http://eeweb.poly.edu/~onur/lectures/lectures.html.

http://www.caen.uiowa.edu/~dip/LECTURE/lecture.html

Outcomes	On Completion of this Course, the students can able to ➢ Discuss digital image fundamentals and apply image enhancement and restoration techniques. ➢ Use image compression and segmentation techniques. ➢ Represent features of images.

Semester - VI				
Course code: 22BCA6E8	**Discipline Specific Elective - IV**	**T/P**	**C**	**H/W**
	(B)Computer Graphics	**T**	**6**	**6**
Objectives	➢ This course prepares students for activities involving the design, development, and testing of modeling, rendering, and animation solutions to a broad variety of problems found in entertainment and sciences. ➢ Students will learn how to develop interactive programs that use effectively the graphics functionalities available in contemporary personal computers, ➢ The fundamental principles, technologies and possibly their future evolutions.			
Unit -I	**Introductory Concepts**:- Introduction of Coordinate representation and Pixel Graphics output devices: CRT, Raster Scan & Random Scan systems; Color CRT monitors, DVST, flat-panel displays, video controller and raster scan display processor. **Graphics Input Devices:-** Keyboard, Mouse, Track-ball, space ball, Joysticks, data Glove, Light Pen, Digitizer, Image scanners, touch panels, voice systems; Graphics software			
Unit-II	**Graphics Output Primitives:-** Point and Lines, Line Drawing Algorithms: Simple, DDA, Bresenham's Line Drawing algorithm, Circle and Ellipse drawing algorithm. **Polygon drawing:**- Representation of polygon; Conventional methods for drawing polygons; Real time Scan Conversion and Run length encoding; Filled area primitives, character generation, Antialiasing.			
Unit-III	**2D Viewing:-** Viewing pipeline, Window-to-viewport transformation, 2-D Clipping, Chen-Sutherland Line Clipping, Mid-point subdivision algorithm, Liang-Barsky clipping, Cyrus-Beck line clipping. **Polygon Clipping:**- Sutherland-Hodgeman and Weiler-Atherton polygon clipping; Character Clipping.			
Unit-IV	**2D-3D Transformations:**- Scaling, Rotation, Translation, Shearing, Reflection; Homogeneous coordinates, Composite Transformations, Affine transformation; 3-D concepts and representation, Solid Body transformations. **Projections:**- Perspective, Orthographic, Axonometric, Oblique projections			
Unit-V	**Advanced Topics: Curves and Surfaces:-** Spline representations, Bezier curves and surfaces, B-spline curves and surfaces. **Visible Surface Detection Methods:**- Back-face detection, depthbuffer, A-buffer, Z- buffer, scan-line. **Illumination Models and Surface Rendering:-** Basic illumination models, Half-toning and dithering techniques, Polygon Rendering, Color models.			

Reference and Textbooks:

TEXT BOOK:

Foley, & van Dam. (2013). *Computer Graphics.* Person Education

Hearn, D., & Baker, P. (2002). *Computer Graphics C Version*. Pearson Education.

REFERENCES:

Foley, & van Dam. (2013). *Computer Graphics.* Person Education

Hearn, & Baker. (2013). *Computer Graphics with OpenGL*. Pearson

Maurya, R. K. (2018). *Computer Graphics with virtual reality systems*. Wiley-India

Rogers, D. (1997). *Procedural Methods for computer graphics*. TMH

Sinha, A., & Udai, A. (2007). *Computer Graphics*. McGraw Hill Education.	
Outcomes	➢ Know and be able to discuss hardware system architecture for computer graphics and be able to design and implement model and viewing transformations, the graphics pipeline and an interactive render loop with a 3D graphics API. ➢ Know and be able to use the underlying algorithms, mathematical concepts, supporting computer graphics, be able to select and use among models for lighting/shading. ➢ Know and be able to use and select among current models for surfaces..