UNIVERSITY OF KERALA

Course Structure and Syllabus for Career Related First Degree Programme in

COMPUTER APPLICATION (BCA)

Under Choice based Credit and Semester System (CBCS) System 2 (b)

(2021 Admission onwards)

SCHEME

Semester 1

			Hrs per week			
Course code	**Credits**	**Course Name**	**Lecture**	**Tutorial**	**Lab**	**Total**
EN1111.4	2	Speaking and listening Skills	3	-	-	3
MM1131.9	3	Mathematics I	3	1	-	4
CP1121	3	Computer Fundamentals and Organization	3	-	-	3
CP1131	3	Digital Electronics	3	-	-	3
CP1141	3	C Programming	3	1	-	4
CP1142	3	C Programming Lab	-	-	4	4
CP1122	3	Open Office Lab	-	-	4	4
TOTAL	**20**		**15**	**2**	**8**	**25**

Semester 2

			Hrs per week			
Course code	**Credits**	**Course Name**	**Lecture**	**Tutorial**	**Lab**	**Total**
EN 1211.4	2	Writing and Presentation Skills	3	-	-	3
MM1231.9	3	Mathematics II	3	1	-	4
CP1241	3	Environmental Studies	3	-	-	3
CP1242	3	Object Oriented Programming	3	1	-	4
CP1243	3	Data Structures in C	3	-	-	3
CP1244	3	Object Oriented programming Lab	-	-	4	4
CP1245	3	Data Structures Lab	-	-	4	4
TOTAL	**20**		**15**	**2**	**8**	**25**

Semester 3

			Hrs per week			
Course code	**Credits**	**Course Name**	**Lecture**	**Tutorial**	**Lab**	**Total**
CP1331	3	Computer Graphics	3	-	-	3
CP1341	3	Computer Networks	3	1	-	3
CP1342	3	Operating Systems	3	-	-	3
CP1343	3	Database Management Systems	3	-	-	4
CP1344	3	Programming in Java	3	1	-	4
CP1345	3	DBMS Lab	-	-	4	4
CP1346	3	Java Programming Lab	-	-	4	4
TOTAL	**21**		**15**	**2**	**8**	**25**

Semester 4

			Hrs per week			
Course code	**Credits**	**Course Name**	**Lecture**	**Tutorial**	**Lab**	**Total**
CP1441	3	Software Engineering	3	-	-	3
CP1442	3	Python Programming	3	1	-	4
CP1443	3	Web Programming	3	.	-	3
CP1444	3	Data Mining & Warehousing	3	-	-	3
CP1445	2	Mini Project	-	1	3	4
CP1446	3	Python Lab	-	-	4	4
CP1447	3	Web Programming Lab	-	-	4	4
TOTAL	**20**		**12**	**2**	**11**	**25**

Semester 5

Course code	Credits	Course Name	Hrs per week			
			Lecture	Tutorial	Lab	Total
CP1541	3	PHP and MYSQL	3	1	-	4
CP1542	3	Cloud Computing	3	-	-	3
CP1543	3	Visual Programming	3	-	-	3
CP1544	3	Design and Analysis of Algorithms	3	1	-	4
CP1551.1 CP1551.2 CP1551.3	2	**Open Course** Digital Marketing Internet and WWW Impact of Social Media Networks	3	-	-	3
CP1545	3	PHP and MYSQL Lab	-	-	4	4
CP1546	3	Visual Programming Lab	-	-	4	4
TOTAL	**20**		**15**	**2**	**8**	**25**

Semester 6

Course code	Credits	Course Name	Hrs per week			
			Lecture	Tutorial	Lab	Total
CP1641	3	Artificial Intelligence	3	-	-	3
CP1642	4	Software Testing	3	-	-	3
CP1643	3	Object Oriented Analysis and Design	3	1	-	4
CP1661.1 CP1661.2 CP1661.3	3	**Elective** Internet of Things (IoT) Digital Marketing Big Data Analysis	3	-	-	3
CP1644	3	Information Security			4	4
CP1645	4	Major Project	-	-	8	8
TOTAL	**20**		**12**	**1**	**12**	**25**

Division of Marks (Lab Examination)

1. Program in Part A should be sufficiently simple– 25 marks
   (Logic – 10 marks, Successful compilation – 10 marks, Result – 5 marks)
2. Program in Part B should be based on advanced concepts - 30 marks
   (Logic – 15 marks, Successful compilation – 10 marks, result – 5 marks)
3. Viva Voce - 15 marks
4. Lab Record - 10 marks

Total Marks - 80 marks

Semester One

Course code	Credits	Course Name	Hrs per week			
			Lecture	Tutorial	Lab	Total
EN1111.4	2	Speaking and listening Skills	3	-	-	3
MM1131.9	3	Mathematics I	3	1	-	4
CP1121	3	Computer Fundamentals and Organization	3	-	-	3
CP1131	3	Digital Electronics	3	-	-	3
CP1141	3	C Programming	3	1	-	4
CP1142	3	C Programming Lab	-	-	4	4
CP1122	3	Open Office Lab	-	-	4	4
TOTAL	**20**		**15**	**2**	**8**	**25**

CP1121 COMPUTER FUNDAMENTALS AND ORGANIZATION

COURSE OUTCOMES: At the end of the Course, the Student will be able to

CO1	Understand the basics of computer
CO2	To get the functional knowledge about PC hardware, operations and concepts.
CO3	To understand the functional units of a standard PC and it's working.
CO4	To understand the memory organization in a computer

COURSE CONTENT

Module I: Characteristics of Computer: Von Neumann model; Inside a Computer: SMPS, Motherboard, BIOS, CMOS, Ports and Interfaces, Expansion Cards, Ribbon Cables, ASCII; Types of Input Devices, Types of Output Devices, Software, Types of Softwares.

Module II: Memory Representation, Hierarchy, Memory Units: RAM (SRAM, DRAM); ROM; Secondary Storage Devices: Magnetic Tape, Magnetic Disk, Types of Magnetic Disks, Optical Disk, Types of Optical Disks; USB: Pen drive, External Hard Disk; Memory Stick; CPU Registers, Cache Memory, Operations in Cache memory, hit ratio; Virtual Memory.

Module III: Instruction Format; Instruction Cycle: Fetch Cycle, Execution Cycle; Instruction Set: CISC Architecture, RISC Architecture, Comparison; Memory Chips; Pipelining and Parallel Processing; Micro-programmed Control and Hardwired Control.

Module IV: Input/Output Organization: Asynchronous Data Transfer, Programmed I/O (concepts only); Interrupts: Types of interrupts, processing interrupts, interrupt hardware and priority, DMA: DMA Controller, DMA Transfer Modes; I/O Processor.

CORE TEXTS

1. Introduction to Information Technology, 2nd Edition, ITL Education Solutions Limited, Pearson.
2. John D. Carpinelli, Computer systems Organization & Architecture, Pearson Education.

ADDITIONAL REFERENCES

1. E. Balaguruswamy, Fundamentals of Computers, McGraw hill, 2014
2. Carl Hamacher, Vranesic, Zaky, Computer Organization 4th Edition, McGraw-Hill

CP1131 DIGITAL ELECTRONICS

COURSE OUTCOMES: At the end of the course the students should be able to:

CO1	Remember the basic concepts of electronics
CO2	Familarise the concept of different number systems
CO3	Understanding the properties of logic gates
CO4	Apply different techniques and theorems to simplify the sop forms
CO5	Analyse the characteristics of different combinational logic circuits.

COURSE CONTENT

Module I: Review of Basic Electronics: PN Junction Diode, Volt –ampere characteristics of PN junction diodes –Half wave and Full wave rectifiers, Ripple factor, Diode clippers and types, Transistors-types, terminals, NPN and PNP transistors, symbols, transistor as an amplifier, Introduction to oscillator, Barkhausen criteria, RC phase shift oscillator, Astable and Monostable multivibrator (555 timer).

Module II: Data Representation: Concept of number system bases –binary, octal, decimal and hexadecimal number systems and conversion between each, Binary arithmetic: Addition, Subtraction- 1s and 2s complement method, Gray code, BCD &BCD addition, Error detection codes-parity method & Hamming code

Module III: Logic gates- AND, OR and NOT, NAND, NOR, XOR and X-NOR. Universal property of NAND and NOR gates, Laws and rules of Boolean algebra, DeMorgan's theorems. Standard forms-SOP and POS Karnaugh map. Flip flops –SR flip flop, JK flip flop, D flip flop

Module IV: Digital Circuits: Full and Half adders, Comparators –1 bit and 2 bit, Encoder and keyboard Encoder, Decoder, Multiplexer, De-multiplexer, Counters(2-bit,4 bit and decade counters).Shift registers-SI-SO, SI-PO, PI-SO, PI-PO

CORE TEXTS

1. Thomas L Floyd –Digital Fundamentals-Pearson, 11th edition
2. V K Mehta, Rohit Mehta -Principles of Electronics, S. Chand & Company 12/e

ADDITIONAL REFERENCES

1. M Morris Mano –Digital Logic and Computer Design-Pearson, 2013
2. B L Theraja –Basic Electronics-Chand Publications

CP1141 C PROGRAMMING

COURSE OUTCOMES: At the end of the Course, the Student will be able to

CO1	Expose students to algorithmic thinking and algorithmic representations
CO2	Understand the structure of program writing in C
CO3	Apply control structures and pointers
CO4	Analyze built-in and user defined functions
CO5	Understand dynamic memory allocation
CO6	Understand string handling functions
CO7	Understand standard library functions in C language

Module I: Introduction to programming: Algorithm & Flow charts, Program Writing – Structure of the Program, Source code, Object code, Executable file, Extensions of different files, Program Compilation, Running of a Program; Header file concept. Variables and Constants, Rules for naming the Variables/Identifiers; Basic data types of C, int, char, float, double; storage capacity – range of all the data types; Storage classes, Type casting.

Module II: Basic Elements: Operators and Expressions: Assignment Operator, Arithmetic Operator and Arithmetic expression, Relational Operator and Relational expression. Logical Operators, Expression Evaluation (Precedence of Operators); simple I/O statements, Control structures, if, if else, switch-case, for, while, do-while, break, continue. Arrays, Defining simple arrays, Multi-dimensional arrays, declaration, initialization and processing;

Module III: Functions & Pointers: Concept of modular programming, Library, User defined functions, declaration, definition & scope, recursion, Pointers: The & and * Operators, pointer declaration, assignment and arithmetic, visualizing pointers, call by value, call by reference, dynamic memory allocation.

Module IV: Advanced features: Array & pointer relationship, pointer to arrays, array of pointers. Strings: String handling functions; Structures and unions; File handling: text and binary files, file operations, Library functions for file handling, Modes of files.

CORE TEXTS

1. E. Balaguruswamy, *Programming in ANSI C*, McGrawhill, Sixth Edition
2. Computer Fundamentals and Programming in C by Reema Thareja, 2nd Edition, Oxford publication

ADDITIONAL REFERENCES

1. Ashok N. Kamthene, *Programming in C*, Pearson Education, Second edition
2. Yashavant Kanitkar, Let us C Authentic Guide to C programming Language, 17th edition

WEB REFERENCES

1. https://www.tutorialspoint.com/cprogramming/index.htm
2. https://www.programiz.com/c-programming
3. https://www.w3schools.in/c-tutorial/intro/

CP1142 C PROGRAMMING LAB

Part A

The C laboratory work will consist of 25-30 Experiments

1-15.Testing out and interpreting a variety of simple programs to demonstrate the syntax and use of the following features of the language: basic data types, operators and control structures.

Part B

16. 1-D Arrays: A variety of programs to declare, initialise, read, print and process 1-D arrays of various basic data types. Processing to include, selection, sum, counting, selective sum, selective counting, reversing etc.
17. Pointers: A large number of trivial programs involving all possible data types to familiarize the syntax of pointers in a variety of situations and to draw memory diagrams based on the observations.
18. Structures: A variety of programs to declare, initialise, read, print and process structures made up of a variety of data types and structures.
19. 2-D Arrays: A variety of programs to declare, initialise, read, print and process 2-D arrays of various basic data types. Processing to include, selection, sum, counting, selective sum, selective counting, reversing etc.
20. Array of Structures and Structure of Arrays: Programs to demonstrate declaration and processing of structure of arrays and array of structures.
21. Pointers to Arrays: A number of programs to demonstrate handling of 1-D and 2-D arrays using pointers and to draw memory diagrams based on the observations.
22. Pointers to Structures: A number of programs to demonstrate use of pointers to structures and to draw memory diagrams based on the observations.
23. Functions–I: Simple Examples of declaring and using functions of the following categories (i) no argument, no return, (ii) argument, no return, (iii) no argument, return, (iv) argument, return, all pass by value
24. Functions–II: Declaring and using functions with pass by reference, Passing and Returning structures, Recursive functions.
25. Files: Simple Example involving use of multiple files: declaring, opening, closing, reading from and writing to text files.
26. Files: Example involving use of multiple files: declaring, opening, closing, reading from and writing to binary files.
27. Library functions: A variety of Examples demonstrating (i) string processing functions (ii) a variety of selected library functions
28. Debugging programs involving syntactic and/or logical errors

29-30. Developing programming solutions to problems including program design, algorithm development and data structure selection.

CP 1122 OPEN OFFICE LAB

Part A.

1. To experience the features of Linux Operating System
2. Working with Linux commands

Part B.

1. Working with word processor
2. Working with worksheet
3. Working with presentation

Semester Two

			Hrs per week			
Course code	**Credits**	**Course Name**	**Lecture**	**Tutorial**	**Lab**	**Total**
EN 1211.4	2	Writing and Presentation Skills	3	-	-	3
MM1231.9	3	Mathematics II	3	1	-	4
CP1241	3	Environmental Studies	3	-	-	3
CP1242	3	Object Oriented Programming	3	1	-	4
CP1243	3	Data Structures in C	3	-	-	3

CP1244	3	Object Oriented programming Lab	-	-	4	4
CP1245	3	Data Structures Lab	-	-	4	4
TOTAL	**20**		**15**	**2**	**8**	**25**

CP1241 ENVIRONMENTAL STUDIES

COURSE OUTCOMES

CO1	Improve the knowledge on the environmental systems
CO2	Understand the concepts of biodiversity and conservations
CO3	Educate the knowledge on the impact of human communities on the environments
CO4	Apply pollution management techniques
CO5	Understand natural systems and resources

Module I: Environmental Studies– Introduction, Multidisciplinary nature, Scope and importance, Concept of sustainability and sustainable development. **Ecosystems –** Structure, function, Energy flow, food chains, food webs and ecological succession, Forest ecosystem, Grassland ecosystem, Desert ecosystem, Aquatic ecosystems, **Natural Resources** -Renewable and Non---renewable Resources, Land resources and use, land degradation, soil erosion and desertification, Deforestation - Causes and impacts due to mining, dam building on environment, forests, biodiversity and tribal populations, Water: Use and over---exploitation of surface and ground water, floods, droughts, conflict over water - international & interstate, Energy resources- renewable and non-renewable, use of alternate energy sources, growing energy needs.

Module II: Biodiversity and Conservation- Levels of biological diversity: genetic, species and ecosystem diversity; Bio geographic zones of India; Biodiversity patterns and global biodiversity hot spots, endangered and endemic species of India, Threats - habitat loss, poaching of wildlife, man-wildlife conflicts, biological invasions; Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity, Eco-system and bio-diversity services: Ecological, economic, social, ethical, aesthetic and Informational value.

Module III: Environmental Pollution- Environmental pollution: types, causes, effects and controls; Air, water, soil and noise pollution, Nuclear hazards and human health risks, Solid waste management: Control measures of urban and industrial waste, Pollution case studies. Environmental Policies & Practices - Climate change, global warming, ozone layer depletion, acid rain and impacts on human communities and agriculture, Environment Laws: Environment Protection Act; Air (Prevention & Control of Pollution) Act; Water (Prevention and control of Pollution) Act; Wildlife Protection Act; Forest Conservation Act. International agreements: Montreal and Kyoto protocols and Convention on Biological Diversity (CBD).Nature reserves, tribal populations and rights, and human wild life conflicts in Indian context. **Policies & Practices**-Climate change, global warming, ozone layer depletion, acid rain and impacts on human communities and agriculture, Environment Laws: Environment Protection Act; Air (Prevention & Control of Pollution) Act; Water (Prevention and control of Pollution) Act.

Module IV: Human Communities and the Environment – Human population growth: Impacts on environment, human health and welfare, Resettlement and rehabilitation of project affected persons; case studies, Disaster management: floods, earthquake, cyclones and land-slides, Environmental movements: Chipko Silent valley, Bishno is of Rajasthan, Environmental ethics: Role of Indian and other religions and cultures in environmental conservation.

CORE TEXT

1. "Text book for Environmental Studies for undergraduate courses of all branches of Higher Education", Erach Bharucha for University Grants Commission.

ADDITIONAL REFERENCE

1. N Arumuganand V Kumaresan, "Environmental Studies", Saras Publication, 2014.

CP1242 OBJECT ORIENTED PROGRAMMING

COURSE OUTCOMES: At the end of the course, the student will be able to

CO1	Understand the concepts of classes and object
CO2	Analyze classes for a given situation and instantiate objects for specific problem solving
CO3	Apply the object initialization and destroy concept using constructors and destructors

CO4	Apply the concept of polymorphism to implement compile time polymorphism in programs by using overloading methods and operators.
CO5	Apply the concept of inheritance to reduce the length of code and evaluate the usefulness
CO6	Understand standard template library for faster development.

COURSE CONTENT

Module I: Introduction to OOP, Procedural vs Object oriented Programming, Characteristics and applications of OOP, Concepts: Object, class, data abstraction, data encapsulation, inheritance and Polymorphism. Static and dynamic binding, message passing, Advantages of object orientation- reusability, maintenance, security. Basic C++ program structure, Preprocessor directive, data types, operators, variables and constant declarations. Input and Output statements, type conversion, control statements- if, if..else, switch..case, jump statements- break, continue, goto, exit- looping statements- for, while, do..while.

Module II: Introduction to objects-classes- Declaration of classes in C++- Abstraction and Encapsulation, Member function definition, Creating Objects- Calling member functions- Array of objects-Objects as Function arguments- Scope resolution operator-static data members, access modifiers,

Module III: Polymorphism- function overloading-inline functions-, friend functions- Member functions of a class as friends of another class, Friend function as a bridge between two classes. Friend classes, early binding and late binding, Constructors – default and parameterized constructor, constructor overloading, copy constructor -destructors,

Module IV: Inheritance-Introduction to code reuse, containership-Parent and Derived classes, public, private and protected inheritance, single, multilevel, multiple, hierarchical, inheritances- function overriding- virtual functions, Objects and pointers, this pointer, pointers to derived class, object slicing, C++ Exception Handling: Try Throw, Catch, Throwing an Exception, Catching an Exception, Templates-Introduction to code sharing-function templates and class templates

CORE TEXT

Object oriented Programming with C++- E Balagurusamy – Sixth Edition

ADDITIONAL REFERENCES

1. Object-oriented Programming with C++ - A. K. Sharma – Second edition
2. Object-oriented Programming in C++- Robert Lafore - Fourth Edition

CP1243 DATA STRUCTURES in C

COURSE OUTCOMES: At the end of the Course, the student will be able to

CO1	Remember purpose of Data Structures
CO2	Understand different Data Structures
CO3	Able to implement the above data structures in C
CO4	Analyze working of different data structures
CO5	Evaluate expressions
CO6	Create different Data Structures

COURSE CONTENT

Module I: **Introduction:** Concept of Data Structures, Types of Data Structures, Linear versus Non Linear Data Structures, Data Structure Operations, Time-Space complexity of algorithms. **Array:** Linear Array-Memory representation, insertion and deletion operation, Multidimensional Arrays-memory representation, Sparse Matrices. **Linked List:** Concept of Linked List, Memory representation, Single Linked List - Traversing, Searching, Insertion, Deletion, Circular Header Linked List , Doubly Linked List - Insertion, Deletion, Difference of Linked List and Array.

Module II: Stack: Representation and operations on Stack using arrays and linked list, application of Stack - Polish Notation- Conversions to Infix, postfix and prefix notations, Infix to postfix conversion using stack, Evaluation of postfix expression using stack **Queue:** Implementation and operations on Queue using arrays and linked list, Deque- Types Input and output restricted, Priority Queues-Array and Linked list representation

Module III: Trees: Concept of Trees, Tree terminologies, **Binary tree:** Complete and Extended Binary tree, Expression trees, Representation of Binary Tree, Traversing Binary Trees – Preorder, Inorder, Postorder.

Binary Search Tree (BST): Search, Insertion and Deletion operations, creating a Binary Search Tree. **Graph:** Concept of Graph, Graph terminologies, Graph Traversal – BFS, DFS.
Module IV: Sorting: Bubble Sort, Selection Sort, and Insertion Sort. **Searching**: Sequential searching, Binary searching. Hashing- hash table, types of hash functions, Collision Resolution Techniques-linear probing, quadratic probing, double hashing, chaining.

CORE TEXTS

1. Seymour Lipschutz, Data Structures, Schaum's outline Series. The McGraw Hill
2. S.K.Srivastava, Deepali Srivastava. Data Structures Through C in Depth. BPB Publications. Second Revised & Updated Edition.

ADDITIONAL REFERENCES

1. K Sharma. Data Structures using C. Pearson, Second Edition.
2. Ashok N. Kamthane, Introduction to Data Structures in C, Pearson

CP1244 OBJECT ORIENTED PROGRAMMING LAB

The laboratory work will consist of 15-20 experiments, only by using class concept

Lab Exercise

Part –A

1. Testing out and interpreting a variety of simple programs to demonstrate the syntax and use of the following features of the language : basic data types, operators and control structures
2. Solving problems using classes, array of objects and objects as function arguments
3. Class definitions and usage involving variety of constructors and destructors

Part – B

4. Programs involving various kinds of inheritances
5. Programs involving function overloading and operator overloading
6. Programs involving virtual base classes, friend functions
7. Programs to demonstrate early binding and late binding
8. Programs to demonstrate Exception handling
9. Programs to demonstrate class and function templates

CP1245 DATA STRUCTURES LAB

The laboratory work will consist of 15-20 experiments that should be implemented in C language

Part A

- Implementation of different searching techniques.
- Implementation of different sorting technique.
- Stack and Queue Array Implementation.

Part B

1. Circular Queue Implementation.
2. Single Linked List and Double Linked List Operations.
3. Stack and Queue Linked List Implementation.
4. Evaluation of expression using Stack.
5. Tree traversal.
6. Graph Implementation.

Semester 3

			Hrs per week			
Course code	**Credits**	**Course Name**	**Lecture**	**Tutorial**	**Lab**	**Total**
CP1331	3	Computer Graphics	3	-	-	3
CP1341	3	Computer Networks	3	1	-	3
CP1342	3	Operating Systems	3	-	-	3
CP1343	3	Database Management Systems	3	-	-	4
CP1344	3	Programming in Java	3	1	-	4
CP1345	3	DBMS Lab	-	-	4	4
CP1346	3	Java Programming Lab	-	-	4	4

TOTAL	**21**		**15**	**2**	**8**	**25**

CP 1331 COMPUTER GRAPHICS

COURSE OUTCOMES: At the end of the course, the student will be able to

CO1	Compare various graphics devices
CO2	Apply various transformations to 2D and 3D graphics objects
CO3	Analyze algorithms for clipping
CO4	Classify various projections of 3D objects
CO5	Explain current trends in computer graphics
CO6	derive various projections of 3D objects

Module I : Introduction: Computer graphics, application of computer graphics, pixel, resolution, aspect ratio, frame buffer, Raster scan , horizontal and vertical retrace, Random scan, video adapter, video controller, various input devices- keyboard, mouse, trackball, joystick, dataglove, digitizers, image scanners, touch panels, light pens, voice systems, display devices- CRT, LCD, LED, DVST, beam penetration method, shadow mask CRT, printer – impact and non-impact printer, plotter, coordinate systems- modeling coordinates, world coordinates, device coordinates, output primitives: Straight line, DDA algorithm (m<1 left to right), Bresenham's Line drawing algorithm, midpoint circle algorithm, polygon filling algorithms-boundary fill, flood fill, scan line algorithm.

Module II: Two dimensional transformations: Basic transformations : Translation, Rotation, Scaling, composite transformation, Other transformations: reflection, shearing, transformations with respect to arbitrary points, matrix formulation and concatenation of transformations, 2D viewing pipeline: window point, view port, window to viewport transformation, workstation transformation, 2D clipping; clip window, Point clipping, Line clipping, Cohen-Sutherland Line Clipping algorithms, Midpoint subdivision algorithm, Polygon clipping- Sutherland Hodgeman algorithm, text clipping, exterior clipping.

Module III: 3D concepts and techniques: 3D display techniques, 3D object representations, basic 3D transformations, Projections: parallel and perspective projections, vanishing points, visible surface detection algorithms-scan line method, Z buffer algorithm, A- buffer algorithm, depth sorting.

Module IV : color models – RGB, HIS, CMYK, Animation, morphing, tweening, warping, zooming, panning, rubber band methods, light sources, ambient light, Polygon rendering : Gourad Shading, Phong Shading

CORE TEXT

- Donald D. Hearn, M. Pauline Baker, Computer Graphics (C Version) 2/e , Pearson

ADDITIONAL REFERENCES

- Amarendra N Sinha and Arun D Udai, Computer Graphics, McGraw Hill Publications

CP1341: COMPUTER NETWORKS.

COURSE OUTCOMES: At the end of the Course, the student will be able to

CO1	Remember the purpose of computer networks and its developments
CO2	Understand various network technologies, design issues and characteristics
CO3	Apply the use of layer architecture for networking systems
CO4	Analyze the working of different models of network and data communication
CO5	Evaluate data link controls
CO6	Create different networking protocols

COURSE CONTENT

Module I: **Introduction**: Use of computer networks-applications, network topologies, Network hardware-LAN, WAN, MAN, VPN, PAN, broadcast, point-to-point, unicasting, multicasting. Network software-connection-oriented, connectionless, protocol hierarchies, layering and design issues. Reference model-OSI, TCP/IP- comparison. Internet, mobile phone network, wireless LAN, RFID and sensor networks. Guided transmission media-magnetic, twisted, coaxial, fiberoptics. Wireless transmission- radio waves, microwaves, infrared, lightwave. Communication satellites.

Module II: Data communication:- Data flow, data transmission mode-simplex, half-duplex, full-duplex. Bandwidth- bit rate, baud rate, Digital modulation and multiplexing-FDM, TDM, WDM. Switching - circuit, packet, message. Mobile system-1G, 2G, 3G, GSM. **Data link controls:** Framing, error control, flow control- feedback-based, rate-based, Error detection and correction-hamming code, parity, checksum, CRC, Stop & wait ARQ, Stop and wait protocol, sliding window protocol-Piggybacking, pipelining, Go-back-N, Selective repeat

Module III: Access Control: Multiple access protocols-ALOHA, pure, slotted, CSMA, CSMA/CD. LAN transmission equipment- Network Interface Card (NIC), repeaters, hubs, bridges, routers, switches, gateway, Ethernet

Module IV: Internetworking- Datagram, virtual circuit network, routing algorithm- adaptive, non-adaptive, centralized, distributed, static, dynamic, distance vector, link state, Dijikstra algorithm (shortest path), tunneling, packet fragmentation,. Internet protocol (IP)- services, IP address, IPv4,IPv6, TCP, UDP. Network applications - client-server model, DNS, Remote login, FTP, email-SMTP, MIME, WWW.

CORE TEXT

1. Andrew S. Tanenbaum, *"Computer Networks"*, Fourth/Fifth edition, Pearson
2. Brijendra Singh, "*Data Communication and Computer Networks"*, Fourth edition, PHI

ADDITIONAL REFERENCES

1. Behrouz A Forouzan, "*Data Communication and Computer networks"*, 4th edition, McGraw Hill
2. Achyut S Godbole, "*Data communications and networks"*, Second edition McGrawHill

CP1342 OPERATING SYSTEMS

COURSE OUTCOMES: At the end of the Course, the student will be able to

CO1	Understand working of various Operating Systems
CO2	apply constrained resource allocation, process scheduling and memory management techniques
CO3	Evaluate synchronization of processes and protection of an Operating System
CO4	Analyze salient features available to various Operating Systems

COURSE CONTENT

Module I: Operating System Overview: Introduction - Structure of Operating System, the Evolution of Operating System, Operating System Functions, System Calls. **Distributed Systems**: introduction, Trends in Distributed System, challenges.

Module II: Process Management: The Process, Process State, PCB, Threads, Process Scheduling - Basic Concepts, Scheduling Criteria, Scheduling Algorithms. **Process Coordination**: Critical Section problems, Semaphores, Synchronization – Inter-process Communication Problems. Deadlock – Definition, Resource Allocation Graph, Conditions of deadlock, deadlock prevention, deadlock avoidance, deadlock detection, deadlock recovery.

Module III: Memory Management: Basic Hardware, Address binding, Logical vs. physical address space, Dynamic Loading and Linking, Swapping, Memory Allocation Methods, Paging, Structure of Page Table, Segmentation, Virtual Memory- Background Demand Paging, Page Replacement- Basic Page Replacement, FIFO Page Replacement, Optimal Page Replacement, LRU Page Replacement, Thrashing.

Module IV: Storage Management: File Concept, Access Methods, Protection, Implementation- File System Structure, Allocation Methods, Recovery, Secondary Storage- Overview, Disk Scheduling, Disk Management, RAID. **I/O Systems**- I/O Hardware, Application I/O Interface, Kernel I/O Subsystem. **Case Study Analysis**: Comparison of different OS using above functionalities-DOS, WINDOWS, UNIX, LINUX, etc.

CORE TEXT

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne-Operating System Concepts, 10th Edition.

ADDITIONAL REFERENCES

1. P. Balakrishna Prasad- Operating Systems and Systems Programming, 5th Edition.

2. Achyut S Godbole and Atul Kahate - Operating systems, McGrawhill
3. George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair – Distributed Systems, Concepts and Designs, 5th Edition

CP1343 DATABASE MANAGEMENT SYSTEMS

COURSE OUTCOMES: At the end of the Course, the student will be able to

CO1	Understand the concept of database.
CO2	Develop skills to design an ER diagram.
CO3	Create database using SQL and perform operations in SQL.
CO4	Familiarize the management of concurrent transactions.
CO5	Apply the design concepts and normalization in database easily.

COURSE CONTENT

Module I: Introduction: Database system applications, Purpose of database systems, View of data, Database languages, Database design, Database and application architecture. **Data models:** Hierarchical model, Network model, Entity Relationship model, Object oriented data model, Relational model. **Introduction to relational model:** Structure of relational database, Database schema, Keys, Relational algebra and calculus.

Module II: Database design using ER model: Overview of the design process, Entity relational model, Complex attribute, Mapping cardinalities, Primary key, removing redundant attributes in entity sets, Reducing ER diagram to relational schema, Entity relationship design issues. **Relational database design:** Features of good relational design, Decomposition using Functional Dependencies, Normal forms (1NF, 2NF, 3NF, BCNF, 4NF)

Module III: Introduction to SQL: Overview of the SQL query language, SQL data definition, Basic structure of SQL queries, Additional basic operations, Set operations, Null values, Aggregate functions, Nested subqueries, Modification of the database, **Intermediate SQL:** Join expressions, Views, Integrity constraints, Authorization.

Module IV: Transactions: Transaction concept, A simple transaction model, ACID property, Serializability, **Concurrency control:** Lock based protocol, Deadlock handling, Multiple granularity, insert operations, delete operations and predicate Reads, Timestamp based protocols, Validation based protocols. **Basic security issues:** The need for security, Physical and Logical security, Design issues, Maintenance issues, Operating system issues and availability, Accountability.

CORE TEXT

- Avi Silberschatz, Hentry F. Korth, S. Sudarshan, Database System Concepts, Seventh Edition.

ADDITIONAL REFERENCES

- Ramon AM and Pauline K. Cushman, Database Management Systems, MC Graw Hill Edn.
- Atul Kahate, Introduction to Database Management Systems.

CP1344: PROGRAMMING IN JAVA

COURSE OUTCOMES: On the completion of the course students will be able to

CO1: Understand the java programming and oops concepts.
CO2: Understand the concepts of Interface, exception handling, threading, and package
CO3: Understand the basic concepts of Applet, Networking.
CO4: Idea to approach and use a new package.

COURSE CONTENT

Module I: Java Introduction: Object Oriented Programming concepts, Features of Java language, Types of Java programs, Java architecture, Program Structure, Literals, Data Types and variables, Operators, Control Statements, Arrays, Classes and objects: Class, Objects, Defining a class, Method declaration Constructor, Method overloading.

Module II: **Inheritance:** Creating subclasses, Method Over-riding, Super keyword, Final keyword, Abstract Classes. Packages and Interfaces: Package, Import statement, Access Modifiers, Interfaces. IO Packages,

Java Input Stream Classes, Java Output Stream Classes, File Class.

Module III: **Exception:** Introduction, exception handling techniques, Creating your own exceptions. Threads: Multitasking, Creation of new Threads, State of a thread, Multithreaded programming, Thread Priorities.
Module IV: Applets- Introduction, Applet class, Applet Structure, Example Applet Program, Applet Life Cycle, **Graphics**, Stand-alone GUI applications with Awt/swing components, Event handling- Event delegation model, different Events and corresponding Listeners/Adapters. **JDBC**, **Socket Programming**- Socket class, Server Socket class, example client/server program.

CORE TEXT

- Dr. K. Somasundaram, Programming in Java 2, Jaico publishing House

ADDITIONAL REFERENCES

- E Balagurusamy, "Programming with Java – A Primer", McGraw Hill, 2017
- Deitel, Java: How to Program, Pearson Education
- Java Programming, Schaum Outline Series

CP1345 DBMS Lab

Lab Exercises: The laboratory work will consist of 15-20 experiments.

Part A

1. SQL statements for creating, dropping and updating tables.
2. Record manipulation using insert, delete and update.
3. Experiments that clarify the importance of keys.
4. Practice all constraints of attributes.
5. Queries with substring comparison.
6. Usage of BETWEEN.
7. Aggregate functions.
8. Finding values with a certain range.
9. Queries with string comparison and ordering.
10. Usage of GROUPBY clause.

Part B

1. Join between two tables.
2. Foreign key.
3. Nested queries.
4. Views.
5. The EXISTS and UNIQUE in SQL.
6. Renaming attributes and joining tables.

CP1346: JAVA PROGRAMMING LAB

The laboratory work will consist of 15-20 Experiments

Part A

1. Testing out and interpreting a variety of simple programs to demonstrate the syntax and use of the following features of the language: basic data types, operators and control structures.
2. Class definitions and usage involving variety of constructors and finalizers
3. Programs involving various kinds of inheritances,
4. Program involving Method Over-riding, Method Over-loading
5. Program involving Abstract Class and Methods

Part B

6. Program involving Interface,
7. Program to demonstrate creation and handling of packages, their imports and Class Path.
8. Programs involving a variety of Exception Handling situations
9. Program to define a class that generates Exceptions and using objects of the class.
10. Program involving creating and handling threads in applications and applets.

11-12: Programs to demonstrate methods of various i/o classes

13. Programs to demonstrate methods of string class
14. Program to demonstrate AWT/Swing graphic methods

15. Program for Loading and Viewing Images, Loading and Playing Sound
16. Programs to demonstrate various Layouts
17. Programs to demonstrate event handling
18. Program to demonstrate simple server-client (using a single m/c both as client and server)
19. Debugging programs involving syntactic and/or logical errors

Semester 4

			Hrs per week			
Course code	**Credits**	**Course Name**	**Lecture**	**Tutorial**	**Lab**	**Total**
CP1441	3	Software Engineering	3	-	-	3
CP1442	3	Python Programming	3	1	-	4
CP1443	3	Web Programming	3	.	-	3
CP1444	3	Data Mining & Warehousing	3	-	-	3
CP1445	2	Mini Project	-	1	3	4
CP1446	3	Python Lab	-	-	4	4
CP1447	3	Web Programming Lab	-	-	4	4
TOTAL	**20**		**12**	**2**	**11**	**25**

CP1441 SOFTWARE ENGINEERING

COURSE OUTCOMES: At the end of the course, the students should be able to:-

CO1	Understand the importance of having a process for software development.
CO2	Familiarize with various software testing techniques and tools.
CO3	Apply various models in the software development projects.
CO4	Analyze the process of software development

COURSE CONTENT

Module I: Introduction: Evolution, Software Development Projects-Program versus product, Types of Software Development Projects Software life cycle models: A few basic concepts, Waterfall model and its extensions, Rapid Application Development (RAD), Agile development models, Spiral model, Comparison of different life cycle models

Module II: Software Project Management, Project Planning, Metrics for project size estimations, Project Estimation Techniques, Basic COCOMO model, Scheduling-Work break down structure, Activity Network, Basic concepts of CPM, PERT and Gantt Chart. Software Requirements Analysis and Specification: Requirements gathering and analysis, Software Requirements Specification.

Module III: Software Design: overview of the design process, How to characterize a good software design, Cohesion and Coupling, Approaches to software design, Function oriented design: Overview of SA/SD Methodology, Structured analysis, Developing the DFD model of a system, Structured Design, Object modelling Using UML, Unified Modelling Language(UML), UML diagrams-Class, Interaction, Activity and State chart diagram .

Module IV: Coding and Testing: Coding, Code review, Testing, Unit testing, Black box testing, white box testing: Basic concepts, Debugging, Integration testing, system testing, Software Reliability and quality management: Software reliability, Software maintenance: Characteristics of software maintenance, Software reverse engineering, Emerging Trends: Client Server Software, Client Server architectures, CORBA, Service Oriented Architectures (SOA), Software as a Service.

CORE TEXT

- Rajib Mall, Fundamentals of Software Engineering, Fifth Edition, PHI

ADDITIONAL REFERENCES

- Pankaj Jalote, An Integrated Approach to Software Engineering, Narosa Publishing House
- Roger S Pressman, Software Engineering - A practitioner’s approach (Sixth Edition), McGraw Hill Companies, Inc

CP1442 PYTHON PROGRAMMING

COURSE OUTCOMES: At the end of the course, the student will be able to

CO1	Remember the concepts of python programming
CO2	Understand data types and differences
CO3	Apply CGI programming
CO4	Analyze the concepts of database programming in python
CO5	Evaluate the usage of Python package installer PIP
CO6	Create programs using libraries such as Flask, SQL Alchemy, Pandas, numpy etc.

COURSE CONTENT

Module I: Introduction to Python - Features of Python, Identifiers, Reserved Keywords, Variables, Comments in Python, Input, Output and Import Functions, Operators; **Data Types and Operations** - int, float, complex, Strings, List, Tuple, Set, Dictionary, Mutable and Immutable Objects, Data Type Conversion; **Flow control** - Decision Making, Loops-for, range() while, break, continue, pass;

Module II: Functions- Definition, calling, arguments, anonymous function, recursion, return; **Modules & Packages** - Built-in Modules, Creating Modules, import statement, Locating, modules ,Namespaces and Scope, dir (), reload (), Packages in Python; **File Handling**- open, close, write, read, methods, rename, delete, directories;

Module III: Object oriented programming- class, object, method, attribute, destructor, encapsulation, data hiding; **Exception handling**- built in exceptions, Handling, Exception with arguments, Raising and User defined exceptions, Assertions in Python**; Regular expressions –** match, search, replace, patterns.

Module IV: Database Programming- Connection, Create, insert, update, delete, commit, rollback, disconnection, exceptions; **Iterators**- Data type supports iterators; **CGI Programming**- HTTP Header, Env variables, Forms, Radio button, Dropdown box, check box, text area, cookies, uploading file.

CORE TEXT

- Jeeva Jose, “Taming PYTHON By Programming”, Khanna Publications, 2017

ADDITIONAL REFERENCES

1. Allen B. Downey, Think Python- How to think like a computer scientist, 2nd Edition, O’Reilly, 2016.
2. Paul Gries, Jennifer Campbell and Jason Montojo, “Practical Programming: An Introduction to Computer Science using Python 3”, Second edition, Pragmatic Programmers, LLC, 2013.

CP1443 WEB PROGRAMMING

COURSE OUTCOMES: At the end of the Course, the Student will be able to

CO1	**Remember** basic concept of web technology.
CO2	**Understand** the different Web Programming languages**.**
CO3	**Apply** the styles to the web pages at various levels using CSS.
CO4	**Analyse** the looping structure and functions of JavaScript
CO5	**Understand** the basic concepts of XML.
CO6	**Create** the modern Web applications using these languages.

COURSE CONTENT

Module I: Introduction to the Internet: The World Wide Web, Web Basics, Client-side scripting verses server-side scripting, The Hypertext Transfer Protocol, Common Gateway Interface (CGI). **Introduction to HTML:** Origins and Evolution of HTML, Basic Syntax of HTML, Standard HTML Document Structure, Basic Text Markup, Lists, Tables, Images, Hypertext Links, Forms, Frames.

Module II: Advanced HTML: HTML DOM, Difference between HTML and HTML5, semantic elements in HTML5, basic HTML5 tags: The article tag, figure tag, header tag, hgroup tag, footer tag, navigation tag, section tag, aside tag, address tag**,**HTML5 Canvas**. Introduction to Cascading Style Sheets:** Types of Style Sheets, Specification Formats, Selector Forms, Property-Value Forms, Font Properties, List Properties, Alignment of Text, Color, The Box Model, Background Images, The span and div Tags.

Module III: **Introduction to JavaScript:** The Basics of JavaScript**,** variables and datatypes, Operators , Control structure, conditional, loop statements, Functions, pop up Boxes, Advanced JavaScript: JavaScript Arrays, Objects: Number, Boolean, string, date, math Event handling, JavaScript DOM, JavaScript RegExp, JavaScript Validation.
Module IV: **Introduction to XML:** The Syntax of XML, XML Document Structure, Namespaces, XML Schemas, Displaying Raw XML Documents, Displaying XML Documents with CSS, XSLT Style Sheets, XML Applications. DHTML, Components of DHTML, Difference between HTML and DHTML, DHTML JavaScript, DHTML CSS.

CORE TEXT

- Paul Deitel, Harvey Deitel & Abbey Deitel, **Internet and World Wide Web: How to Program**, Pearson, Fifth Edition, 2016.

ADDITIONAL REFERENCES

1. HTML 5 Black Book, Covers CSS 3, JavaScript, XML, XHTML, AJAX, PHP and jQuery, 2ed Kindle Edition
2. Gopalan & Akilandeswari, Web Technology: A Developer's Perspective, 2nd ed

WEB REFERENCES:

1. https://www.w3schools.com/html/
2. https://www.tutorialspoint.com/css/index.htm
3. https://www.javatpoint.com/javascript-tutorial
4. https://www.tutorialspoint.com/xml/xml_overview.htm
5. http://www.academictutorials.com/dhtml/

CP1444: DATA MINING & DATA WARE HOUSING

COURSE OUTCOMES: At the end of the course, the student will be able to

CO1	Recognize data warehouse concepts, architecture, business analysis and tools
CO2	Understand data pre-processing and data visualization techniques
CO3	Evaluate algorithms for finding hidden and interesting patterns in data
CO4	Understand and apply various classification and clustering techniques using tools
CO5	Analyze a variety of real-world applications that require data mining
CO6	Formulate useful patterns and associations in huge quantities of data

COURSE CONTENT

Module I: Introduction-: Introduction: -Data, Information, Knowledge, KDD, types of data for mining, technologies for mining, issues in data mining, data mining functionalities/tasks. Data pre-processing-overview, Data cleaning, Data integration, Data reduction, Data transformation and discretization. **Data Warehouses**-basic concepts, Data Mart, Databases Vs Data warehouses, Data ware houses Vs Data mart, OLTP Vs OLAP, OLAP operations/functions, OLAP Multi-Dimensional Models- Data cubes, Star, Snow Flakes, Fact constellation data models.

Module II: Association rules- Market Basket Analysis, Frequent Item sets, Closed Item sets, and Association Rules, **Frequent Item sets Mining Methods**- Apriori Algorithm: Finding Frequent Itemset by Confined Candidate Generation, Generating Association Rules from Frequent item sets, Improving the Efficiency of Apriori.

Module III: Classification– Basic Concepts, Decision Tree Induction, Bayesian Classification, Rule Based Classification, Classification by Back propagation, Support Vector Machines, Associative Classification, Lazy Learners

Module IV: Clustering- Cluster analysis: definition and Requirements, Characteristics of clustering techniques, Types of data in cluster analysis, Overview of Basic Clustering Methods, Partitioning methods-K-Means and K -medoid methods, **Outlier detection**- definition and types of outliers, Outlier Detection

Methods-Supervised, Semi-Supervised, and Unsupervised Methods, Statistical Methods, Proximity-Based Methods, and Clustering-Based Methods (basic concepts only)

CORE TEXT

- Jiawei Han & Micheline Kamber& Jian Pei Data Mining Concepts & Techniques

ADDITIONAL REFERENCES

1. Sunitha Tiwari & Neha Chaudary, Data Mining and Warehousing, Dhanpat Rai & Co.
2. Margaret H Dunham Data Mining-Introductory& Advanced Topics; Pearson

CP1445: MINI PROJECT

COURSE OUTCOMES: At the end of the course, the student will be able to

CO1	Opportunity for structured team work and project management.
CO2	Practice the various phases in the SDLC
CO3	Plan And Estimate A Project
CO4	Practic Time, resource and person management
CO5	Coding And Implementation Of a Software

1. GUIDELINES FOR MINOR PROJECT

The minor project is considered as a stepping stone in implementing Major projects. Hence students should plan and organize their minor projects meticulously and necessary discussions and planning should be done so as to achieve this objective. The following guidelines should be adhered to:

1. Team size should preferably be three with a maximum limit of 4 members.
2. Individual projects may be permitted in exceptional cases, for valid reasons
3. Minor Projects should be purely internal in nature.
4. No restriction on tools/platform/language chosen should be made.
5. Internal guide(s) should be assigned to each team.
6. Two interim reports(one after analysis and another after design) should be submitted to internal guides.
7. The number of records to be submitted is limited to team size + one (Departmental copy). Hard binding of reports is optional.
8. The report format guidelines used to document Major Projects should be followed for making the final report and evaluation will be made on the same grounds.

2. EVALUATION

Criteria for evaluation (External evaluation only)

External evaluation is done by an external examiner appointed by the University. The following components are to be assessed for the End Semester Evaluation of the Minor Project:

- Quality of work & report - 50 marks
- Presentation of work - 30 marks
- Viva - 20 marks
- Total - 100 marks

CP1446: PYTHON LAB

The laboratory work will consist of 10-15 Experiments

Part A

- To write, test, and debug simple Python programs.
- To implement Python programs with conditionals and loops. (square root, gcd, exponentiation, sum of an array of numbers, linear search, binary search, bubble sort, insertion sort, selection sort etc.)
- Use functions for structuring Python programs.
- Represent compound data using Python lists, tuples, dictionaries.

Part B

- Read and write data from/to files in Python.

- Programs to demonstrate creating and handling of modules and packages
- Programs involving a variety of Exception Handling situations
- Programs involving Database manipulation
- CGI programming

CP1447: WEB PROGRAMMING LAB

The laboratory work will consist of 20-25 Experiments

PART A

- Programs to practice basic html tags, formatting tags, heading, lists
- Programs to create tables, frames, link and anchor tags, Marquee
- Programs to include image audio, video and sound
- Programs to understand the different styles in CSS and various types of selectors in CSS
- Programs to understand the basics of XML

PART B

- Program to demonstrate FORM tags and its various controls.
- Program covering the basic html5 tags.
- Programs covering JavaScript basics, popup boxes, loops and conditions
- Programs to implement function and events
- Create a web site using HTML,CSS and JavaScript

SEMESTER V

			Hrs per week			
Course code	**Credits**	**Course Name**	**Lecture**	**Tutorial**	**Lab**	**Total**
CP1541	3	PHP and MYSQL	3	1	-	4
CP1542	3	Cloud Computing	3	-	-	3
CP1543	3	Visual Programming	3	-	-	3
CP1544	3	Design and Analysis of Algorithms	3	1	-	4
CP1551.1 CP1551.2 CP1551.3	2	**Open Course** Digital Marketing Internet and WWW Impact of Social Media Networks	3	-	-	3
CP1545	3	PHP AND MYSQL Lab	-	-	4	4
CP1546	3	Visual Programming Lab	-	-	4	4
TOTAL	**20**		**15**	**2**	**8**	**25**

CP1541: PHP AND MYSQL

COURSE OUTCOMES: At the end of the course, the student will be able to

CO1	Remember features, operators
CO2	Understand the control structures/object-oriented terminologies
CO3	Develop skills to write database queries
CO4	Analyze current trends and styles in web design and applications
CO5	Create state-of-the-art websites

COURSE CONTENT

Module I: Overview of PHP, Benefits, and drawbacks in running PHP as a Server Side Script, PHP Language Basics: The building blocks of PHP: variables, globals & super globals Data types: Set type, type casting, test type, Operators & Expressions, Flow control functions in PHP, Functions: Defining a function variable scope, calling a function returning values, setting default values for arguments, passing variable reference, built-in functions.

Module II: Arrays: Creating arrays (associative & multidimensional), Array related functions. Working with string functions: Formatting strings, Using Date and Time functions. working with files and directories. PHP OOP: Concept, Classes, and Objects, constructor, destructor, access modifiers, Inheritance, Constants, Abstract classes, Interfaces, Traits, Static methods, Static properties, Namespaces, iterable.

Module III: Forms in PHP: Form elements, adding elements to a form, creating a simple input form, combining HTML & PHP code on a single page, redirecting the user, creating a send mail form, File upload form. Cookies: Introduction, different types of cookies, setting a cookie with PHP, deleting a cookie, session function overview: starting a session, working with session variables, passing session IDs in the query string, destroying sessions &unsetting variables, Working with images.
Module IV: Database concepts: Open-source database software: MySQL features MySQL data types: Numeric, date & time, string Table creation in MySQL: insert, select, where clause, ordering the result, like operator Selecting Multiple tables: using join, using queries Modifying records: update command, replace command, delete command date & time functions in MySQL Interacting with MySQL using PHP: connecting to MYSQL, Executing queries, Retrieving error messages, inserting data with PHP, retrieving data with PHP.

CORE TEXT

- Meloni, J. C. *Sams teach yourself PHP, MySQL and Apache all in one*.

ADDITIONAL REFERENCES

- Holzner, S. *Complete Reference PHP*.
- W3schools.com
- https://www.w3schools.com/php/php_oop_what_is.asp.
- Vaswani, V. *MySQL(LM): The complete reference*. McGraw-Hill Education, Indian Edition.

CP1542: CLOUD COMPUTING

COURSE OUTCOMES: At the end of the course, the student will be able to

CO1	Remember the basics of cloud computing
CO2	Understand the main concepts and key technologies of cloud computing.
CO3	Apply the concept of virtualization in the cloud computing
CO4	Analyze the evolution of cloud from the existing technologies.
CO5	Evaluate and choose the technologies for implementation and use of cloud.
CO6	Create services using cloud computing

COURSE CONTENT

Module I: Cloud Computing Foundation: Introduction to Cloud Computing- Basics, History, importance, Characteristics, Pros and Cons of Cloud computing. Types of Cloud – Public and Private Cloud. Cloud Computing infrastructure.

Module II: Cloud Computing Architecture: Cloud Computing Technology- Cloud Life Cycle Model. Cloud Architecture – Cloud Computing Logical Architecture, Developing Holistic Cloud computing Reference Model. Cloud System Architecture, Cloud Development Model.

Module III: Virtualization: Definition, Adopting Virtualization, Types, Virtualization and Software, Virtual Clustering, Virtualization Application, Pitfalls of Virtualization.

Module IV: Data Storage & Security: Data Storage- Introduction to Enterprise Data Storage, Data Storage Management, File Systems, Cloud Data Stores. Cloud Computing and Security – Risks in Cloud Computing, Types of Risks, Risk Management Process.

CORE TEXT

A.Srinivasan and J.Suresh, Cloud Computing – A Practical Approach for Learning and Implementation, Pearson India Publications, 2014

ADDITIONAL REFERENCES

1. Rajkumar Buyya, James Broberg, Andrzej, Cloud Computing: Principles and Paradigms, Wiley India Publications, 2011
2. Barrie Sosinsky, “Cloud Computing Bible”, 1st Edition, Wiley India Pvt. Ltd., New Delhi, 2011.

CP1543: VISUAL PROGRAMMING

COURSE OUTCOMES: At the end of the course, the student will be able to

CO1	Understand basic information about the features of visual studio tools
CO2	to use cookies very effectively
CO3	Get the idea of using SSL with GUI

SYLLABUS

Module I: An Introduction to ASP.NET web programming – An introduction to web programming, an introduction to ASP.NET application development, quick preview of how an ASP.NET application works. Visual Studio features for working with CSS. Introduction to server controls, how to work with button controls, text boxes, labels, check boxes, radio button, list controls, and other web server controls like image, hyperlink, file upload, and calendar controls.

Module II: Introduction to validation controls, basic validation controls, validation techniques and advanced validation controls. How to manage state – how to use view state, session state and application state. How to use cookies.

Module III: An introduction to database programming – Introduction to relational database, how to use SQL to work with data in database. Introduction to ADO.NET 4. How to use SQL data source, how to use custom statements and stored procedures, Data list controls, Data binding, advanced features of a SQL data source.

Module IV: Customise the GridView control, Update GridView data, DataListView controls, FormView Control, ListView control and update ListView data. Introduction to SSL, how to get and use digital source certificate, how to use a secure connection, Introduction to authentication, how to setup authentication and authorization, how to use login controls, how to configure ASP.NET application, how to deploy an ASP.NET application.

CORE TEXT

- **Anne Boehm**, Murach's ASP.NET 4 web programming with VB 2010, Shroffs publishers and Distributors Pvt. Ltd

ADDITIONAL REFERENCES

1. **Imar Spaanjaars**, Beginning ASP.NET 4.0 in C# and VB, Wiley publishers
2. **Simon Smart**, learn ASP.NET 4.0, C# and VB 2010, publishers Smart Method

CP1544: DESIGN AND ANALYSIS OF ALGORITHMS

COURSE OUTCOMES At the end of the Course the Student will be able to

CO1	Develop and analyze new algorithms by themselves
CO2	Analyze the complexity of algorithms
CO3	Understand good algorithms among multiple solutions for a problem
CO4	Have better knowledge on fundamental strategies of algorithm design and awareness on complex algorithm design strategies
CO5	Implement some typical algorithms

COURSE CONTENT

Module I : Algorithm Analysis: Algorithm, Properties of a good algorithm, efficiency considerations, Complexity: Time and Space complexity, Asymptotic notations: Big O notations, best case, worst case, average case, simple examples, recursion and its elimination- recursive and no-recursive algorithms for binary search.

Module II : Algorithm design techniques – Divide and conquer method : binary search as a divide-and-conquer algorithm, finding maximum and minimum, Strassen's matrix multiplication, Greedy method: Knapsack problem, minimum cost spanning trees, Prim's algorithm, Kruskal's algorithm.

Module III: Dynamic programming: principle of optimality, all pair shortest paths, single source shortest paths, travelling sales person's problem, Backtracking: implicit constraints and explicit constraints, N queen problem, Branch and bound: LC search.

Module IV: Standard Algorithms: sorting-quicksort, merge sort, complexity of sorting algorithms, Deterministic and non-deterministic algorithms, NP-hard and NP complete-basic concepts.

CORE TEXT

- Anany Levitin, *Introduction to design and analysis of algorithms*, Pearson, Second Edition

ADDITIONAL REFERENCES

- Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekhara *Computer Algorithms/ C++*, Second Edition-Universities Press.

Assignments and Activities: *Studies on complexities of various algorithms, best case, average case, worst case analysis.*

CP1551: OPEN COURSE

CP1551.1: DIGITAL MARKETING

COURSE OUTCOMES: At the end of the course, the student will be able to

CO1	Understand different digital marketing types
CO2	Understand the main concepts and key technologies of digital marketing.
CO3	Remember the concept of e-banking, cyber security
CO4	Analyze the evolution of digital marketing from the existing technologies.
CO5	Analyze services using digital marketing

SYLLABUS

Module I: Introduction: Nature, Scope and Importance of Digital Marketing; Evolution of Digital Marketing; Core Concepts-Inbound Marketing, Content Marketing, Email Marketing, Influential Marketing; Holistic Digital Marketing Concept, 10Ps of digital marketing; Digital Marketing Environment: Macro and Micro Environment.

Module II: E-banking: approaches, devices, services, benefits, drawbacks, Electronic payment systems-credit cards, debit cards, smart cards, credit accounts, cyber security, encryption, secret key cryptography, public key cryptography, digital signatures, firewalls

Module III: Digital Marketing: Search Engine Optimization (SEO), Social Media, Content Marketing; Email Marketing, Mobile Marketing. Challenges for Digital Marketing: Increased Security Risk, Cluttered Market, Less Focus on Keywords, More Ad Blockers, Increased Ad Costs**.**

Module IV: Digital Marketing: Pay per Click-Search Engine Advertising, Advantages, Factors, Conversion Rate Optimization (CRO); Digital Marketing- Web Analytic. Social Media Marketing: Face book, Pinterest, Twitter, LinkedIn, YouTube, Google Adwords, Google Analytics; Issues and Future enhancement of Digital Marketing.

CORE TEXT

Ian Dodson-*The art of Digital Marketing*, Wiley

ADDITIONAL REFERENCES

Puneet Singh Bhatia- *Fundamentals of Digital Marketing*, Pearson Education

CP1551.2 INTERNET and WWW

COURSE OUTCOMES: At the end of the course, the student will be able to

CO1	To **understand** the basic concepts of Networks.
CO2	To **learn** the working of Internet.
CO3	To **analyse** different search engines and its working

CO4	To **familiarise** Network Protocols and WWW.

3. SYLLABUS

Module I: Introduction to Network-Types of Network-Network Topologies, Basic communication technology, Intranet, Extranet-Advantages, Internet-History, Modes of Connecting to Internet-Dialup Access-Direct to dedicated connections, Internet Service Providers, Domain Name Service, Internet Addresses-AddressingScheme-IPV6, Modems, Routers, Network cards Communication Software, Internet Tool-File Transfer Protocols, Search Tools, Telnet.

Module II: Introduction to WWW-WWW and HTTP, Webpage, Introduction to Web Browser-Book Marks-Comparison, Directories, Search Engines-Working and features-Search Strategies – Search Generalization-Search Specialization-Working.

Module III: Uniform Resource Locator (URL), OSI reference model, Introduction to TCP/IP-TCP/IP Model, Email-Working with Email-Sending Mail-Reading Mail-Replying to Mail-Deleting Mail-Advantages and Disadvantages of Email, Basics of Chat Rooms, SMTP.

Module IV: Introduction to Web Server-Personal Web Server (PWS)-Internet Information Server (IIS)-Apache Webserver-Benefits of Web Server, Introduction to Security-Internet Security-Identifying Network Stations, Network Protocols-Internet Security Threats.

CORE TEXT

Dr.Surender Jangra, “Basics of Internet and Web”, Vayu Education of India. New Delhi, 110002

ADDITIONAL REFERENCES

Raymond Greenlaw, Ellen Hepp, “Fundamentals of Internet and the World Wide Web, McGraw-Hill.

CP1551.3 IMPACT OF SOCIAL MEDIA NETWORKS

COURSE OUTCOMES: At the end of the course, the student will be able to

CO 1	To understand the types of social media networks and its uses.
CO 2	To learn the impact of social media on society& commerce
CO 3	To analyse the impact of social media on work, training & development and on relationships
CO 4	To familiarize challenges of social media in terms of privacy ,security & health

SYLLABUS

Module I: Introduction to Social Media networks: Types of Social Media-Uses of Social Media Networks-Popular Social Media Websites-Mobile social media- Types of Users- Growth of social media networks. Impact of Social Media on Society: Sharing of Information and its Need – Entertainment – Communication Tool -Influence-Social media activism-societal issues-social cause-Impacts on Politics-Pros and Cons-Positive and negative effects of Social media on Society.

Module II: Impact of Social Media on Commerce: Social media Marketing-Promotion of Business-Digital Marketing& SMM -Advantages to Business-Knowledge sharing and Collaborative Work Management-Customer Benefits-Impacts-pros and cons. Impact of Social Media on the World of Work: Job Recruitment and Hiring-Benefits- Researching Job Candidates-Impact of professional social media networks.

Module III: Impact of Social Media on the Training & Development: Social media in Learning-Online-Long Distance Learning-Impact of Blogs, Wikis, LinkedIn, Twitter, Facebook and Podcast- Privacy and Frauds. Impact of Social Media on relationship: Bonding and Friendships –Pros and Cons- Issues Arising-Impact of Social Media on Kids, Teens & Youth.

Module IV: Challenges of Social Media- Criticism on Social Media-Cyberbullying-Lack of Privacy-

Security Issues. Addictions to Social media –Games- Impact on Education: Positive and negative effects– Impact on Physical and Mental Health-Emotional Insecurities- Depression-Anxiety-Behavioural Issues-Wastage of Time etc. Future of Social Media Networks.

CORE TEXT

Hana S.Noor Al-Deen, John Allen Hendricks, "Social Media-Usage & Impact", Rowman & Little field Publishing Group

ADDITIONAL REFERENCES

1. Dedria Bryfonski, "The Global Impact of Social Media"-Green haven Press
2. Dr.Sanjay Singh Baghel, Dr Uma s singh, 'Social Media and Indian Youth" Apple Books Publishers

CP1545: PHP and MYSQL LAB

Setup WAMP/XAMPP Server or Setup Apache, MySQL, and PHP separately in your PHP Lab. The laboratory work will consist of 15-20 Experiments.

PART A

- Write, test, and debug simple PHP programs.
- Familiarize the use of Conditional Statements.
- Programs with Loops.
- Programs to handle Strings.

PART B

- Implement programs with Functions, Arrays & Images.
- Read and write data from/to files in PHP.
- Programs to demonstrate OOP concepts.
- Programs to handle forms in PHP.
- Programs to interact with MySQL using PHP.

CP1546: VISUAL PROGRAMMING LAB

On completion of this course, student should able to:

1. Get basic information about the features of visual studio tools
2. Introduction to ASP.NET Web Application
3. Get the idea of using SSL with GUI

Part –A

1. Testing out and interpreting a variety of simple programs to demonstrate and use of different controls in ASP.NET
2. CSS selectors-id and class
3. Session Management and Cookies

Part – B

4. Testing out and interpreting programs to demonstrate and use of different validation controls.
5. Develop a C#/VB program for Email Registration
6. Design a feedback form which will allow to enter some simple data and store in db.
7. Design a form to count the number of records using Ms SQL
8. Design a form to retrieve data from a table and use GridView control.
9. Programs to demonstrate ListView Control.
10. Programs to demonstrate Login Control.

SEMESTER VI

			Hrs per week			
Course code	**Credits**	**Course Name**	**Lecture**	**Tutorial**	**Lab**	**Total**
CP1641	3	Artificial Intelligence	3	-	-	3

CP1642	4	Software Testing	3	-	-	3
CP1643	3	Object Oriented Analysis and Design	3	1	-	4
CP1661.1 CP1661.2 CP1661.3	3	**Elective** Internet of Things (IoT) Digital Marketing Big Data Analysis	3	-	-	3
CP1644	3	Information Security			4	4
CP1645	4	Major Project	-	-	8	8
TOTAL	**20**		**12**	**1**	**12**	**25**

CP1641 ARTIFICIAL INTELLIGENCE

COURSE OUTCOMES: At the end of the Course, the Student will be able to

CO1	Remember features of AI and knowledge-based systems
CO2	Understand basic parsing techniques
CO3	Apply search and control strategies
CO4	Analyse different matching techniques
CO5	Evaluate the performance of various searching algorithms
CO6	Create AND-OR graphs

COURSE CONTENT

Module I: Overview of Artificial Intelligence: What is AI, The importance of AI; Knowledge: Introduction, Definition and Importance of knowledge, Knowledge–Based Systems, Representation of Knowledge, Knowledge Organization, Knowledge Manipulation, Acquisition of Knowledge.

Module II: Formalized Symbolic Logics: Introduction, Syntax and Semantics for Propositional Logic and FOPL, Properties of WFFs, Conversion to Clausal Form, Inference Rules, The Resolution Principle; Structured Knowledge: Associative Networks, Frame Structures, Conceptual Dependencies and Scripts.

Module III: Search and Control Strategies: Preliminary concepts, Examples of Search Problems, Uniformed or blind Search, Informed Search, Searching And-Or graphs; Matching Techniques: Introduction, Structures Used in Matching, Measures for Matching, Partial Matching, The RETE Matching Algorithm.

Module IV: Natural Language Processing: Introduction, Overview of Linguistics, Grammars and Languages, Basic Parsing Techniques, Semantic Analysis and Representation Structures, Natural Language Generation, Natural Language Systems

CORE TEXT

Dan W. Patterson, Introduction to Artificial Intelligence And Expert Systems, PHI Learning 2014

ADDITIONAL REFERENCES

Elaine Rich, Kevin Knight, Shivashankar B Nair, Artificial Intelligence, Third Edition, McGraw Hill Education (India) PVT LTD

CP1642: SOFTWARE TESTING

At the end of this course, the student will able to

CO1	Remember the terms used in testing.
CO2	Understand the basic concepts of testing.
CO3	Apply different testing techniques.
CO4	Evaluate the testing methodologies.
CO5	Create different testing tools.

COURSE CONTENT

Module I: Introduction: humans and errors, bugs, faults, failures, purpose of software testing, Approaches to software testing, testing techniques, types of testing, characteristics of testing, psychology of testing, limitations of software testing, basic concepts and definitions. Verification and Validation, software verification techniques, software validation techniques, verification and Validation in Software Development Life Cycle.

Module II: White Box Testing:-static testing, static testing by Humans, static analysis tools. Structural testing, code coverage testing, code complexity testing, challenges in white box testing.

Module III: Black Testing:-Requirements based testing, positive and negative testing, Boundary value analysis, Decision Table, equivalence partitioning, state based or Graph based testing, compatibility testing, User Documentation testing, Domain testing.

Module IV: Test Management and Automation:-Introduction, Test planning, Test management, Test process, Test reporting, Test automation, Test tools

CORE TEXT

Srinivasan Descikan, Gopalaswamy Ramesh, Software testing principles and practices, Pearson

ADDITIONAL REFERENCES

Desai & Srivastava, Software Testing Practical Approach, 2e/d

CP1643: OBJECT ORIENTED ANALYSIS AND DESIGN

COURSE OUTCOME: At the end of this course the students will be able to

CO1	**Remember** object oriented features
CO2	**Understand** Object Oriented System Development
CO3	**Apply** Unified Approach
CO4	**Analyse** various UML diagrams
CO5	**Evaluate** objects static and dynamic model
CO6	**Create** UML diagrams for any system

COURSE CONTENT:-

MODULE I: Object Oriented concepts: class, object, defining a class, comparison between algorithmic decomposition and object oriented decomposition, Object Oriented System Development Life cycle(use-case driven).

MODULE II: Object Oriented Methodologies, The Unified Approach, Concepts of UML,OOAD using UML, UML diagrams, static model, UML class diagrams, representing various features of a class, messages, types of associations; object diagram ;use case diagram, identifying use cases, elements of use case diagram, uses and extends associations.

MODULE III: UML dynamic modeling: UML interaction diagram, Sequence Diagram, elements of sequence diagrams, benefits of sequence diagrams, collaboration diagram, elements; Activity diagram, elements of Activity diagram; State chart diagram ,examples.

MODULE IV: UML Implementation diagrams: Component diagram, elements, examples, Deployment diagram, elements examples; UML meta model. Object Oriented Analysis: Classification theory, Approaches for identifying classes; Object Oriented Design process, Design axioms and corollaries, an example UML class diagram.

CORE TEXT

- Ali Bahrami-Object Oriented Sytem Development, McGraw Hill

ADDITIONAL REFERENCES

1. Grady Booch, Object Oriented Analysis and Design, Addison Wesley, Pearson
2. Edward Yourdon, Carl Argila, Case studies in Object Oriented Analysis and Design
3. Joey F.George, Dinesh Batra, Joseph S.Valacich, Jeffrey A.Hoffer, Object Oriented System Analysis and Design

Elective

CP1661. 1 Internet of Things (IoT)

COURSE OUTCOMES: At the end of the Course, the student will be able to

CO1	**Remember** the purpose of computer networks and its developments
CO2	**Understand** various network technologies, design issues and characteristics
CO3	**Apply** the use of layer architecture for networking systems
CO4	**Analyze** the working of different models of network and data communication
CO5	**Evaluate** data link controls
CO6	**Create** different networking protocols

COURSE CONTENT:-

Module I: Introduction IoT - An Architectural Overview - Building an architecture, Main design principles and needed capabilities, An IoT architecture outline, standards considerations. M2M and IoT Technology Fundamentals - Devices and gateways, Local and wide area networking, Data management, Business processes in IoT, Everything as a Service (XaaS), M2M and IoT Analytics, Knowledge Management

Module II: Reference Architecture IoT Architecture-State of the Art – Introduction, State of the art, Reference Model and architecture, IoT reference Model - IoT Reference Architecture- Introduction, Functional View, Information View, Deployment and Operational View, Other Relevant architectural views. Real-World Design Constraints- Introduction, Technical Design constraints-hardware is popular again, Data representation and visualization, Interaction and remote control.

Module III: IoT Data Link Layer & Network Layer Protocols PHY / MAC Layer (3GPP MTC, IEEE 802.11, IEEE 802.15), Wireless HART, ZWave, Bluetooth Low Energy, Zigbee Smart Energy, DASH7 - Network Layer-IPv4, IPv6, 6LoWPAN, 6TiSCH,ND, DHCP, ICMP, RPL, CORPL, CARP

Module IV: Transport & Session Layer Protocols: Transport Layer (TCP, MPTCP, UDP, DCCP, SCTP)- (TLS, DTLS) – Session Layer HTTP, CoAP, XMPP, AMQP, MQTT. **SERVICE LAYER PROTOCOLS & SECURITY,** Service Layer - oneM2M, ETSI M2M, OMA, BBF - Security in IoT Protocols - MAC 802.15.4, 6LoWPAN, RPL, Application Layer

CORE TEXT

1. Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stefan Aves and, Stamatis Karnouskos, David Boyle, **"**From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence**",** 1st Edition, Academic Press, 2014.

ADDITIONAL REFERENCES

1. Peter Waher**, "**Learning Internet of Things", PACKT publishing, BIRMINGHAM – MUMBAI
2. Bernd Scholz-Reiter, Florian Michahelles, "Architecting the Internet of Things", ISBN 978-3-642-19156-5 e-ISBN 978-3-642-19157-2, Springer
3. Daniel Minoli, "Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications", ISBN: 978-1-118-47347-4, Willy Publications
4. Vijay Madisetti and Arshdeep Bahga, **"**Internet of Things (A Hands-on Approach)**",** 1st Edition, VPT, 2014.
5. http://www.cse.wustl.edu/~jain/cse570-15/ftp/iot_prot/index.html

CP1661.2 DIGITAL MARKETING

COURSE OUTCOMES: At the end of the course, the student will be able to

CO1	Understand different digital marketing types
CO2	Understand the main concepts and key technologies of digital marketing.
CO3	Remember the concept of e-banking, cyber security
CO4	Analyze the evolution of digital marketing from the existing technologies.
CO5	Analyze services using digital marketing

SYLLABUS

Module I: Introduction: Nature, Scope and Importance of Digital Marketing; Evolution of Digital Marketing; Core Concepts-Inbound Marketing, Content Marketing, Email Marketing, Influential

Marketing; Holistic Digital Marketing Concept, 10Ps of digital marketing; Digital Marketing Environment: Macro and Micro Environment.

Module II: E-banking: approaches, devices, services, benefits, drawbacks, Electronic payment systems-credit cards, debit cards, smart cards, credit accounts, cyber security, encryption, secret key cryptography, public key cryptography, digital signatures, firewalls

Module III: Digital Marketing: Search Engine Optimization (SEO), Social Media, Content Marketing; Email Marketing, Mobile Marketing. Challenges for Digital Marketing: Increased Security Risk, Cluttered Market, Less Focus on Keywords, More Ad Blockers, Increased Ad Costs**.**

Module IV: Digital Marketing: Pay per Click-Search Engine Advertising, Advantages, Factors, Conversion Rate Optimization (CRO); Digital Marketing- Web Analytic. Social Media Marketing: Face book, Pinterest, Twitter, LinkedIn, YouTube, Google Adwords, Google Analytics; Issues and Future enhancement of Digital Marketing.

CORE TEXT

- Ian Dodson-*The art of Digital Marketing*, Wiley

ADDITIONAL REFERENCES

- Puneet Singh Bhatia- *Fundamentals of Digital Marketing*, Pearson Education

CP1661.3 BIG DATA ANALYTICS

COURSE OUTCOMES: At the end of the course, the student will be able to

CO1	Learn tips and tricks for Big Data use cases and solutions.
CO2	Learn to build and maintain reliable, scalable, distributed systems with Apache Hadoop
CO3	Able to apply Hadoop ecosystem components
CO4	Analyzing data security and event logging

COURSE CONTENT:-

Module I: Introduction to Big Data: Types of Digital Data-Characteristics of Data – Evolution of Big Data - Definition of Big Data - Challenges with Big Data - 3Vs of Big Data - Non Definitional traits of Big Data - Business Intelligence vs. Big Data - Data warehouse and Hadoop environment - Coexistence. Big Data Analytics: Classification of analytics - Data Science - Terminologies in Big Data - CAP Theorem - BASE Concept. NoSQL: Types of Databases – Advantages – NewSQL - SQL vs. NOSQL vs NewSQL. Introduction to Hadoop: Features – Advantages – Versions - Overview of Hadoop Eco systems - Hadoop distributions - Hadoop vs. SQL – RDBMS vs. Hadoop - Hadoop Components – Architecture – HDFS - Map Reduce: Mapper – Reducer – Combiner – Partitioner – Searching – Sorting - Compression. Hadoop 2 (YARN): Architecture - Interacting with Hadoop Eco systems.

Module II: No SQL databases: Mongo DB: Introduction – Features - Data types - Mongo DB Query language - CRUD operations – Arrays - Functions: Count – Sort – Limit – Skip – Aggregate - Map Reduce. Cursors – Indexes - Mongo Import – Mongo Export. Cassandra: Introduction – Features - Data types – CQLSH - Key spaces - CRUD operations – Collections – Counter – TTL - Alter commands - Import and Export - Querying System tables.

Module III: Hadoop Eco systems: Hive – Architecture - data type - File format – HQL – SerDe - User defined functions - Pig: Features – Anatomy - Pig on Hadoop - Pig Philosophy - Pig Latin overview - Data types - Running pig - Execution modes of Pig - HDFS commands - Relational operators - Eval Functions - Complex data type - Piggy Bank - User defined Functions - Parameter substitution - Diagnostic operator. Jasper Report: Introduction - Connecting to Mongo DB - Connecting to Cassandra - Introduction to Machine learning: Linear Regression – Clustering - Collaborative filtering - Association rule mining - Decision tree.

Module IV: BIG DATA PRIVACY, ETHICS AND SECURITY Privacy – Re identification of Anonymous People – Why Big Data Privacy is self-regulating? – Ethics – Ownership – Ethical Guidelines – Big Data Security – Organizational Security.

CORE TEXT

1. Seema Acharya, Subhashini Chellappan, “Big Data and Analytics”, Wiley Publication, 2015.
2. Michael Minelli, Michelle Chambers, and Ambiga Dhiraj, “Big Data, Big Data Analytics Trends for Today’s Businesses. Wiley ,2013
3. Tom White, “Hadoop: The Definitive Guide”, O’Reilly Publications, 2012.

ADDITIONAL REFERENCES

4. Judith Hurwitz, Alan Nugent, Dr. Fern Halper, Marcia Kaufman, “Big Data for Dummies”, Joh Wiley & Sons, Inc., 2013.
5. Kyle Banker, “Mongo DB in Action”, Manning Publications Company, 2012.
6. Russell Bradberry, Eric Blow, “Practical Cassandra A developers Approach“, Pearson Education, 2014.

CP1644 INFORMATION SECURITY

COURSE OUTCOMES: At the end of the Course, the student will be able to

CO1	Remember the principles and protocols of internetworks
CO2	Understand the basic issues in information security
CO3	Apply information security measures
CO4	Analyze working of various ciphers
CO5	Evaluate the concept of digital signatures and e-mail security policies
CO6	Create awareness on information security

COURSE CONTENT

Module I: **Introduction**: Computer Security- objectives, security attacks, services, security techniques, network security model, basic network security terminologies, **Cryptography**: Symmetric cipher model, cryptanalysis and Brute force attack, Classical Encryption Techniques:- substitution techniques- caesar, monoalphabetic, playfair, hill cipher, polyalphbetic, One-time Pad, transposition techniques, Steganography, block and stream ciphers, block cipher modes of operations, Feistel cipher, Data Encryption Standard(DES)-strength , Public-key cryptosystem-applications, RSA algorithm-working

Module II: Cryptographic Integrity and Authentication: cryptographic hash function-application, message digest, message authentication code-requirements, functions, authentication methods, Digital signature:- Digital signature model and implementation, properties, Digital Signature Standard(DSS)-approach, Digital Signature Algorithm (DSA)-working, Key management- Public key distribution approaches

Module III: Network and Internet Security: Web Security:- threats, Secure Socket Layer(SSL)-architecture- session and connection, SSL protocol- Record, Alert, Handshake, **E-Mail security:**- Pretty Good Privacy, operations of PGP, MIME, S/MIME, **IP Security:**- Applications of IPsec, benefits, IPsec services-transport and tunnel mode, IPsec components –AH, ESP, IKE, IPv4 and IPv6- comparison

Module IV: System Security: Intrusion detection, malicious software-Virus-types, working of antivirus, Worms, Trojans, Spyware, Firewall-types, characteristics and benefits, **Security and Law:**-Regulations in India- Information Technology Act 2000/2008, Cyber Crime- cyber law, Indian Copyright Act, Indian Contract Act, Consumer Protection Act, Future Trends –The Law of Convergence.

CORE TEXT

- William Stallings, “*Cryptography and Network Security: Principles and Practice”*, Fifth edition Pearson,

ADDITIONAL REFERENCES

1. V K Pachghare, “*Cryptography and Information Security”*, Third edition, PHI
2. Atul Kahate,“*Cryptography and Network Security”*, Mc Graw Hill
3. Behrouz A. Forouzan, Debdeep Mukhopadhyay, *"Cryptography and Network Security"* , 2nd edition, Paperback
4. Mohammad Amjad, *"Cryptography and Network Security"*, Paperback

CP1645: MAJOR PROJECT

COURSE OUTCOMES: At the end of the course, the student will be able to

CO1	CREATE an industry-standard project through a real-life project work under time and deliverable constraints, applying the knowledge acquired through various courses
CO2	provide an opportunity to apply the knowledge gained through various courses in solving a real life problem
CO3	provide an opportunity to practice different phases of software/system development lifecycle
CO4	introduce the student to a professional environment and/or style typical of a global IT industry
CO5	provide an opportunity for structured team work and project management
CO6	To provide an opportunity for effective, real-life, technical documentation
CO7	provide an opportunity to practice time, resource and person management

PROJECT GUIDELINES

- Group Size – Maximum 3
- No. of records – No. of group members + 1 (Department copy)
- Certificate should include the names of all members

The minimal phases for the project are:

1. Project search,
2. Finalization and allocation,
3. Investigation of system requirements,
4. Data and process modelling,
5. System Design,
6. Program design,
7. Program coding and unit testing,
8. System integration,
9. System implementation and
10. Acceptance testing.

The **first 6 steps** listed above has to be completed and an analysis and design report should be submitted at the end of this course. This report should be valued internally and marks should be awarded. Up to coding should be finished in 5th semester. Coding should begin in the 6th semester.

1. **Planning the Project:** The Major Project is an involved exercise which has to be planned well in advance. The topic should be chosen in Semester 4 itself; based on the project topic, though on Exceptional cases, for valid reasons, the project guide may waive this condition. Related reading, training and discussions should start from semester 5 itself.
2. **Selection of project work:** Project work could be of 3 types:
   a. **Developing solution for a real-life problem:** In this case, a requirement for developing a computer-based solution already exists and the different stages of system development life cycle is to be implemented successfully. Examples are Accounting Software Package for a particular organization, Computerisation of administrative functions of an organization, Web Based Commerce, etc. The scope for creativity and Exploration in such projects is limited, but if done meticulously, valuable Experience in the industrial context can be gained.
   b. **Innovative Product development:** These are projects where a clear-cut requirement for developing a computer-based solution may not be existing, but a possible utility for the same is conceived by the proposer. An Example is a Malayalam Language Editor with Spell Checker, Computer Music Software for Indian Music, Heat Engines Simulation Software for Learning, Digital Water Marking Software.
   c. **Research level project:** These are projects which involve research and development and may not be as structured and clear cut as in the above case. Examples are Malayalam Character Recognition, Neural Net Based Speech Recogniser, Biometric Systems, Machine Translation System etc. These projects provide more challenging opportunities to students, but at EX level is a difficult choice. If any student identifies proper support in terms of guidance, technology and references from external organizations and also the supervisors are convinced of the ability of the student(s) to take up the project, it shall be permitted. The methodology and reporting of such projects could be markedly different from type (a) and is left to the proposer/external supervisor of the projects.
3. **Selection of Team:** To meet the stated objectives, it is imperative that Major Project is done through a team effort**.** Though it would be ideal to select the team members at random (drawing lots) and this should be strongly recommended, due to practical considerations, students may also be given the

choice of forming themselves into teams with 3 to 5 members (teams less than 3 members may be permitted in Exceptional cases, for valid reasons). A gender mix should also be strongly suggested. A team leader shall be elected through drawing lots. Teams shall maintain team meeting minutes and ensure that every team member has tasks assigned in writing. Team meeting minutes shall form a part of the Project Report. Even if students are doing projects as groups, each one must independently take up different modules of the work and must submit the reports also independently (though, in such cases, some common materials is permissible). Evaluation will also be done independently.

4. **Selection of Tools:** No restrictions shall be placed on the students in the choice of platforms/tools/languages to be utilized for their project work, though open source is strongly recommended, wherever possible. No value shall be placed on the use of tools in the evaluation of the project.
5. **Selection of Organisation & Guide:** No restrictions shall be placed on the students in the choice of organization where project work may be done, in terms of locality, type (public/private) etc. It is the duty of the Head of Institute/Principal of College to ensure that the Aim, Objectives and full project guidelines are communicated to the external organization.

   The guide should ideally be a post-graduate with minimum 2 years of work experience. Students may also choose to do project in the college/institute (or partially in the college and partially in an external organization), especially product-based work, but in such cases the supervisors must ensure that
   a. industry practices are followed
   b. the students undertake a planned visit to an IT industry with international operations to make up for the loss of experience and
   c. The services of an external guide with industry experience is obtained.
6. **Project Management:** Head of Institute/Principal of College should publish a list of students, projects topics, internal guide and external organization (if any) and teams agreed, before the end of semester 5. Changes in this list may be permitted for valid reasons and shall be considered favourably by Head of Institute/Principal of College any time before commencement of the project. Any request for change after commencement should considered by a committee of 3 teachers and their recommendation shall be accepted by Head of Institute/Principal of College.
   - **Gantt-chart** of proposed activities and a draft statement of project deliverables (which may subsequently be altered if justified) should be prepared before the commencement of the project. The actual completion of each phase should be noted on the chart in the course of the project work. Students should submit a fortnightly report of progress which could be indication of percentage of completion marked on the original Gantt-chart, with any notes attached. Students should ideally keep a daily activity log sheet. Team meetings should be documented in the format given at the end. Changes in the submitted documents are possible, as project development is essentially an evolutionary process. The project guide must ensure that changes are necessary due to the knowledge gained in succeeding phases of the project. The date of completion of a phase should be brought forward if the changes made are deemed to be errors and not due to additional knowledge gained from a succeeding phase.
7. **Documentation:** The following are the major guidelines: The final outer dimensions of the report shall be 21 cm X 30 cm. The colour of the flap cover shall be light green. Only hard binding should be done, with title of the thesis and the words “<BRIEF TITLE> BCA Project Report 200…” displayed on the spine in 20 point, Bold, Times New Roman, as in example below. In case the title is too long, a shorter version of it may be used (Like “Image Pro” instead of ”Image Pro – An Interactive Image Processing package”). It is highly recommended that Latex be used for documentation.
   a. The text of the report should be set in 12 pt, Times New Roman, Single Spaced.
   b. Headings should be set as follows: CHAPTER HEADINGS 20 pt, Times New Roman, Bold, All Caps, Centered.

   For Example:

WEB BASED BILLING SOFTWARE: BCA PROJECT 2024

1. SECTION HEADINGS 12 pt, Times New Roman, Bold, All Caps, Left Adjusted.

1.1 Section Sub-headings 12 pt, Times New Roman, Bold, Left Adjusted.

Titles of Figures, Tables etc are done in 12 point, times New Roman, Italics, Centered.

<PROJECT TITLE>

<STUDENT’S NAME>

<COLLEGE NAME>

PROJECT REPORT

Submitted in partial fulfilment of the

Requirements for the award of
Bachelor of Computer Applications (BCA) of
University of Kerala
2024

Some general guidelines on documentation stylistics are:

- Double quotes and single quotes (“ ”, ‘ ’) should be used only when essential. In most cases words put in quotes are better highlighted by setting them in italics.
    - Eg: This process is known as “morphing”. This process is known as *morphing.*
- Page numbers shall be set at right hand top corner, paragraph indent shall be set as 3.
- Only single space need be left above a section or sub-section heading and no space may be left after them.
- Certificate should be in the format: “Certified that this report titled..... is a bonafide record of the project work done by Sri/Kum..... under our supervision and guidance, towards partial fulfilment of the requirements for the award of the Degree of BCA degree in of the University of Kerala” with dated signatures of Internal; Guide, external guide and also Head of Institute/College.
  If the project is done in an external organization, another certificates on the letterhead of the organization is required: “Certified that his report titled..... is a bonafide record of the project work done by Sri/Kum..... under any supervision and guidance, at the .....Department of..... (Organization) towards partial fulfilment of the requirements for the award of the BCA degree of the University of Kerala”.
- References shall be IEEE format (see any IEEE magazine or transaction). Take care in use of italics and punctuation. While doing the project, keep note of all books you refer, in the correct format, and include them in alphabetical order in your reference list.
    - Eg: A book is cited as: Kartalopoulos, S V Understanding Neural Networks and Fuzzy Logic, BPB Publishers, 1996, pp. 21-27. (pp.21-27 indicates that pages 21-27 have been referred. If the whole book is being referred, this may be omitted. If a single page is referred, say 7, it may be cited as p.7

  Report writing is NOT a hasty activity done after finishing the project. Students must try to develop the report along with the work, so as to give it flesh and blood. Drafts should be read, modified, spell checked and grammar checked at least thrice during the course of the project and before a final printout is taken, the same may be got approved from the internal guide. The students should send two interim reports to internal guides. This will also help the students in their report writing.
- The Gantt chart, fortnightly progress reports, and team meeting should appear as appendix to the project report. Regarding the body of the report, as an indicative
    - *Example, the following is given (though students should not attempt to fit every kind of project report into this format):*
        a. Organizational overview (of the client organization, where applicable)
        b. Description of the present system
        c. Limitations of the present system
        d. The Proposed system- Its advantages and features
        e. Context diagram of the proposed system.
        f. Top level DFD of the proposed system with at least one additional level of Expansion
        g. Structure Chart of the System
        h. System flowchart
        i. Menu Tree
        j. Program List
        k. Files or tables (for DBMS projects) list. Class names to be entered for each file in OO systems.
        l. List of fields or attributes (for DBMS projects) in each file or table.
        m. Program – File table that shows the files/tables used by each program and the files are read, written to, updated, queried or reports were produced from them.
        n. Reports List with column headings and summary information for each report.
        o. System Coding and variable/file/table naming conventions
        p. System controls and standards
        q. Screen layouts for each data entry screen.
        r. Report formats for each report.
- Program documentation is suggested on the following lines:
    i. Program id

ii. Program level run chart
iii. Program function Explanation
iv. Data entry screen (reproduced from system documentation).
v. Report layout (reproduced from system documentations)
vi. Program level pseudo code or flowchart.
vii. Decision tables, decision trees, with English Explanation where necessary.
viii. Program listing
ix. Test data
x. Test results.

8. **Methodology:** Wherever applicable, object oriented approach should be used for software development. The project report should generally contain details of the following steps *(though students should not attempt to fit every kind of project into this format):*
    a. Analysis
        (a) Study of existing systems and its drawbacks (general)
        (b) Understanding the functionalities of the system (detailed)
        (c) Preparation of requirement
        (d) Conduct of Feasibility study
        (e) Identification of relevant Objects
        (f) Abstraction of each object (attributed and methods)
        (g) Relationship between objects
    b. Design:
        (a) Design of each subsystems
        (b) Design of each classes
        (c) Design of communications between objects
        (d) Design of Algorithms for problem solving
        (e) User interface Design
        (f) Any other steps if necessary
    c. Coding and Impletion
    d. Testing
    e. Security, Backup and Recovery Mechanisms
    f. On line help and User Manuals
    g. Upgradability Possibilities

9. **Project IPR & Utilisation:** The intellectual property rights in all project work done by the students shall vest with the University of Kerala, except in cases where some external organizations seek undertaking from students to concede IPR in all work done in their organization or under their guidance. Where possible, students should attempt to obtain at least a joint IPR for the University. In cases where project works are of public utility, students shall be asked to publish their work including source code and documentation, in so far as their rights are clear.

10.

Evaluation of Project

Documentation	: 30 marks
Content & Methodology	: 30 marks
Presentation	: 20 marks
Viva Voce	: 20 marks
TOTAL	**: 100 marks**

INDUSTRIAL VISIT - CUM STUDY TOUR

Study tour to be mandatory for the BSc Computer Science programme. In view of the current Covid-19 situation, considering the safety of all, it is applicable only after the lockdown period. It should cover an organization where functional applications of concepts/ theories covered in the programme are being practiced. The visit should be pre- planned with an objective to learn identified applications like computation, algorithms, programming languages, program design, computer software, computer hardware etc.

The total time to be devoted in the organization is one day out of five days set aside for the tour. The outcome of the visit to be documented in a report with the following form a Student Tour Dairy

1. Name of the College
2. Name of the Student

3. B.Sc Programme .....Semester
4. Name of the Organisation
5. Date of Visit
6. Learning Objectives
7. Interactions held .....
8. Outcome in the form of learnings .....
9. Observations (along with pictures, citations, illustrations)

Counter Signed by HoD