

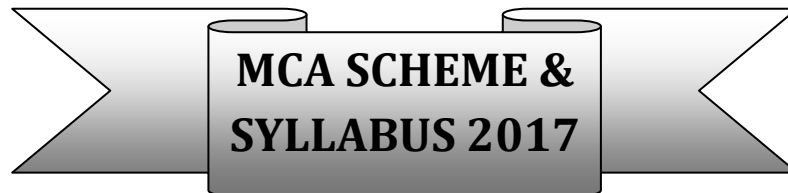
NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY

(Autonomous Institution, Affiliated to VTU, Belgaum, Approved by AICTE & State Govt. of Karnataka)
Yelahanka, Bangalore – 560064

DEPARTMENT OF MCA



KNOWLEDGE ★ CHARACTER ★ UNITY



NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY
DEPARTMENT OF M.C.A
SCHEME OF TEACHING AND EXAMINATION 2017-2019

I SEMESTER		CREDIT BASED						
Subject Code	Name of the Subject	Teaching hours/week		Durati on of Exam in Hours	Marks for		Total marks	Credits
		Lectu re	Practical / Assignm ent		CIE	SEE		
17MCA101	Problem solving using C	4	-	3	50	50	100	4
17MCA102	UNIX Programming	4	-	3	50	50	100	4
17MCA103	Web Programming	4	2	3	50	50	100	5
17MCA104	Computer Organization	4	-	3	50	50	100	4
17MCA105	Professional Communication and Management	4	-	3	50	50	100	4
17MCA106L	C Laboratory	-	3	3	50	50	100	1.5
17MCA107L	UNIX Laboratory	-	3	3	50	50	100	1.5
17MCA108L	Soft skills Activity	-	2	-	50	50	100	1
Total		20	10	24	400	400	800	25

17MCA108L Soft skills Activity

Conduct Activities

i) To improve:

- Oral and written communication
- Interview skills
- Presentation and public speaking
- Leadership and interpersonal skills
- Time and stress management

ii) Develop:

- Positive mental attitude
- Confidence and self esteem

NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY
DEPARTMENT OF M.C.A
SCHEME OF TEACHING AND EXAMINATION 2017-2019

II SEMESTER				CREDIT BASED				
Subject Code	Name of the Subject	Teaching hours/week		Duration of Exam in Hours	Marks for		Total marks	Credits
		Lecture	Practical/Assignment		CIE	SEE		
17MCA201	Object Oriented Programming with C++	4	-	3	50	50	100	4
17MCA202	Data Structures using C	4	2	3	50	50	100	5
17MCA203	Database Management System	4	-	3	50	50	100	4
17MCA204	Operating System	4	-	3	50	50	100	4
17MCA205	System Programming	4	-	3	50	50	100	4
17MCA206L	Object Oriented Programming with C++ Laboratory	-	3	3	50	50	100	1.5
17MCA207L	Database Management System Laboratory	-	3	3	50	50	100	1.5
17MCA208S	Technical Seminar	-	2	-	50	50	100	1
Total		20	10	21	400	400	800	25

17MCA209S Technical Seminar

- Students should present the seminar on cutting edge/emerging/state of the art technologies in the field of Computer Science and Applications.
- Duration of the seminar should be approximately 45 minutes.
- Student should submit the write up on seminar topic containing at least 10 pages

NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY
DEPARTMENT OF M.C.A
SCHEME OF TEACHING AND EXAMINATION 2017-2019

III SEMESTER				CREDIT BASED				
Subject Code	Name of the Subject	Teaching hours/week		Duration of Exam in Hours	Marks for		Total marks	Credits
		Lecture	Practical/Assignment		CIE	SEE		
17MCA301	Computer Networks	4	-	3	50	50	100	4
17MCA302	Programming with Java	4	-	3	50	50	100	4
17MCA303	Python Programming	4	-	3	50	50	100	4
17MCA304	Discrete Mathematical Structures	4	-	3	50	50	100	4
17MCA305	Software Engineering	3	-	3	50	50	100	3
17MCA306E	Elective-1	3	-	3	50	50	100	3
17MCA307L	Java Programming Laboratory	-	3	3	50	50	100	1.5
17MCA308L	Python Programming Laboratory	-	3	3	50	50	100	1.5
Total		22	6	24	400	400	800	25
Elective – 1								
17MCA3061E		Intellectual Property Rights						
17MCA3062E		Enterprise Resource Planning						
17MCA3063E		Advanced Topics in DBMS						
17MCA3064E		Cyber Security						

NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY
DEPARTMENT OF M.C.A
SCHEME OF TEACHING AND EXAMINATION 2017-2019

IV SEMESTER				CREDIT BASED				
Subject Code	Name of the Subject	Teaching hours/week		Duration of Exam in Hours	Marks for		Total marks	Credits
		Lecture	Practical/Assignment		CIE	SEE		
17MCA401	Advanced Java Programming	4	-	3	50	50	100	4
17MCA402	Analysis and Design of Algorithms	4	-	3	50	50	100	4
17MCA403	Software Testing and Practices	4	-	3	50	50	100	4
17MCA404	Principles of User Interface Design	3	-	3	50	50	100	3
17MCA405E	Elective -2	3	-	3	50	50	100	3
17MCA406E	Elective -3	3	-	3	50	50	100	3
17MCA407L	Advanced Java Programming Laboratory	-	3	3	50	50	100	1.5
17MCA408L	Software Testing Laboratory	-	3	3	50	50	100	1.5
17MCA409S	Seminar (web Based)	-	2	-	50	50	100	1
Total		21	8	24	450	450	900	25

Elective - 2		Elective -3	
17MCA4051E	Advanced Computer Networks	17MCA4061E	Mobile Technologies
17MCA4052E	Data Warehousing and Data Mining	17MCA4062E	Big Data Analytics
17MCA4053E	Software Architecture	17MCA4063E	Software Quality Management
17MCA4054E	Cryptography and Network Security	17MCA4064E	MIS & E-Commerce

17MCA409S Seminar:

- Students should present the seminar on cutting edge/emerging/state of the art technologies in the field of Web Technology
- Duration of the seminar should be approximately 45 minutes.
- Student should submit the write up on seminar topic containing at least 10 pages

NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY
DEPARTMENT OF M.C.A
SCHEME OF TEACHING AND EXAMINATION 2017-2019

V SEMESTER					CREDIT BASED			
Subject Code	Name of the Subject	Teaching hours/week		Duration of Exam in Hours	Marks for		Total marks	Credits
		Lecture	Practical/ Assignment		CIE	SEE		
17MCA501	Object Oriented Modelling and Design Patterns	4	2	3	50	50	100	5
17MCA502	Programming Using C# and .NET	4	-	3	50	50	100	4
17MCA503	Mobile Applications	4	-	3	50	50	100	4
17MCA504E	Elective - 4	3	-	3	50	50	100	3
17MCA505E	Elective - 5	3	-	3	50	50	100	3
17MCA506L	.NET Laboratory	-	3	3	50	50	100	1.5
17MCA507L	Mobile Applications in Android Laboratory	-	3	3	50	50	100	1.5
17MCA508P	Internet of Things(IoT)		3	3	50	50	100	3
Total		21	8	24	400	400	800	25

Elective - 4		Elective -5	
17MCA5041E	Rich Internet Applications	17MCA5051E	Software Defined Networks
17MCA5042E	Storage Area Networks	17MCA5052E	Cloud Computing
17MCA5043E	Artificial Intelligence	17MCA5053E	Service Oriented Architecture
17MCA5044E	Multimedia	17MCA5054E	Software Project Management

Internet of Things (IoT): A self-study project based on IoT will be implemented and presented by a group of students (group size not exceeding six)

NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY
DEPARTMENT OF M.C.A
SCHEME OF TEACHING AND EXAMINATION 2017-2019

VI SEMESTER		CREDIT BASED			
Subject Code	Subject Name	Examination			Credits
		CIE	SEE	Total	
17MCA601P	Dissertation Assessment	75	75	150	10
17MCA602P	Internal Evaluation and Viva voce	75	75	150	15
	Total	150	150	300	25
Grand Total (I to VI Semesters):4400 Marks : 150 Credits					

Dissertation Assessment (17MCA601P)

- The project shall be carried out in the same institution or in industry/R&D labs based on software tools and technologies learnt in MCA course/internship for minimum period of 16 weeks.
- The internal examiners (Project Guide with at least 2 years experience) and the external examiners shall be appointed by the College for the Final Evaluation of the project.
- Internal and external examiners shall carry out the evaluation of Dissertation report for 75 marks individually.
- Dissertation Assessment by the internal examiner will be considered as **CIE** and Dissertation Assessment by the external examiner will be considered as **SEE**.

Internal Evaluation and Viva Voce (17MCA602P)

- Internal assessment (CIE) shall be evaluated by both the **internal and external guide** for 75 marks individually. The average of the marks allotted by the internal and external guides shall be the final marks for the Internal Evaluation.
- The project presentation and Viva-voce (SEE) shall be evaluated jointly by both the **internal and external examiners** for 75 marks.

SEMESTER - I

Problem Solving Using C			
[As per Choice Based Credit System (CBCS) scheme]			
SEMESTER – I			
Subject Code	17MCA101	CIE Marks	50
Number of Lecture Hours/Week	04	SEE Marks	50
Total Number of Lecture Hours	50	SEE Hours	03
CREDITS – 04			
Course Outcome (CO): At the end of this course, the students will be able to :			
<p>CO1: Understand basics of C programming language</p> <p>CO2: Acquire knowledge of, Various types of control statements and structures</p> <p>CO3: Analyze the performance of, arrays, functions, pointers, structures, unions, files and preprocessor directives</p> <p>CO4: Implement all the applications of C in a high-level language</p> <p>CO5: Design and apply appropriate C language constraints for solving computing problems.</p>			
Modules			Teaching Hours
Module 1			10 Hours
Algorithms, Flow Charts, C structure, Variables, Data types, Constants, Declarations, Operators, Precedence, Associativity, Order of evaluation, Type conversion, Storage classes, Programming Examples, Input and output statements – scanf, getchar, gets, printf, putchar, puts			
Module 2			10 Hours
Control Statements – if, else-if, switch, Control Structures – while, for, do-while, break and continue, goto, Programming Examples			
Module 3			10 Hours
Arrays – Single dimension, Two dimensional, Multi dimensional Arrays, Strings, Programming Examples, Functions, Categories of functions			
Module 4			10 Hours
Pointers, Pointer arithmetic, Call by value, Pointer Expression, Pointer as function arguments, recursion, Passing arrays to functions, passing strings to functions, Call by reference, Functions returning pointers, Pointers to functions, Programming, Examples, Structures and Unions – defining, declaring, initialization, accessing, comparing, operations on individual members; array of structures, structures within structures, structures and functions, pointers and structures, bit fields, Programming Examples			
Module 5			10 Hours
Files – defining, opening, closing, input and output operations, error handling, random access; Command line arguments; Dynamic Memory Allocation –definition, malloc, calloc, realloc, free, dynamic arrays; Preprocessor – definition, macro substitution, file inclusion, compiler control directives, Programming Examples			
Question paper pattern:			
<ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. 			
The students will have to answer 5 full questions, selecting one full question from each			

module

Text Books

1. Programming with C, Byron Gottfried, Published in 2006 by Tata McGraw-Hill 2nd edition
2. The C Programming Language, Brian W Kernighan, Dennis M Richie, Prentice Hall, 2 edition, published in 1988 ..

Reference Books:

1. Programming with C, Balaguruswamy, published in 2007
2. Let us C, Yashwant Kanetkar, BPB Publications, published in 2016

UNIX Programming [As per Choice Based Credit System (CBCS) scheme] SEMESTER – I			
Subject Code	17MCA102	CIE Marks	50
Number of Lecture Hours/Week	04	SEE Marks	50
Total Number of Lecture Hours	50	SEE Hours	03
CREDITS – 04			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Understand and experience the UNIX environment. Basic commands. CO2: Demonstrate commands to extract, interpret data for further processing. CO3: Apply commands to perform different tasks using AWK filtering for various applications CO4: Analyze the usage of different shell commands, variables and analyzing the working of Source Code Control System Tool-GIT with Example CO5: Evaluate different commands with sample shell scripts</p>			
Modules			Teaching Hours
Module 1			10 Hours
<p>Introduction of UNIX and Shell: Introduction, History, Architecture, Experience the Unix environment, Basic commands ls, cat, cal, date, calendar, who, printf, tty, sty, uname, passwd, echo, tput, bc, script, spell and ispell, Introduction to Shell Scripting, Shell Scripts, read, Command Line Arguments, Exit Status of a Command, The Logical Operators && and , exit, if, and case conditions, expr, sleep and wait, while, until, for, \$, @, redirection. The here document, set, trap, Sample Validation and Data Entry Scripts.</p>			
Module 2			10 Hours
<p>UNIX File System: File, filename, parent-child relationship, pwd, the Home directory, absolute pathnames, using absolute pathnames for a command, cd, mkdir, rmdir, Relative pathnames, The UNIX file system. Basic File Attributes: ls –l, the –d option, File Permissions, chmod, Security and File Permission, users and groups, security level, changing permission, user masks, changing ownership and group, File Attributes, More file attributes: hard link, symbolic link, umask, find.</p>			
Module 3			10 Hours
<p>Filters and Awk Pr, head, tail, cut, paste, sort, uniq, tr commands, Filters using Regular Expression : grep & sed grep, Regular Expression, egrep, fgrep, sed instruction, Line Addressing, Inserting and Changing Text, Context addressing, writing selected lines to a file, the –f option, Substitution, Properties of Regular Expressions Context addressing, writing selected lines to a file, the –f option, Substitution, Properties of Regular Expressions. Awk-Advanced Filters: Simple awk Filtering, Splitting a Line into Fields, printf, the Logical and Relational Operators, Number Processing, Variables, The –f option, BEGIN and END positional</p>			

Parameters, get line, Built-in variables, Arrays, Functions, Interface with the Shell, Control Flow.	
Module 4	10 Hours
<p>Advanced Shell Programming The sh command, export, cd, the Command, expr, Conditional Parameter Substitution, Merging Streams, Shell Functions, eval, Exec Statement and Examples</p> <p>Source Code Control Systems- GIT Version Control Systems, Distributed version Control Systems, Advantages of GIT, DVCS Terminologies, Life Cycle, Create Operation, Create New User, Create a Bare Repository</p>	
Module 5	10 Hours
<p>Process and System Administration Process basics, PS, internal and external commands, running jobs in background, nice, at and batch, cron, time commands, Essential System Administration root, administrator's privileges, startup & shutdown, managing disk space, cpio, tar, Customizing the Environment : System Variables, profile, sty, PWD, Aliases, Command History, On-line Command Editing. Advanced System Administration: Case Study: emacs editor and any one distribution of Linux</p>	
<p>Question paper pattern:</p> <ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. <p>The students will have to answer 5 full questions, selecting one full question from each module</p>	
<p>Text Books:</p> <ol style="list-style-type: none"> 1. Your UNIX-The Ultimate Guide, Sumitabha Das, Tata Mc GrawHill,2001 2. The UNIX Programming Environment by Kernighan and pike, Pearson, 2005 3. “Beginning Shell Scripting”, Eric Foster-Johnson, John C Welch, 2005 Micah Anderson, Wrox publication. 	
<p>Reference Books:</p> <ol style="list-style-type: none"> 1. “Unix Shell Programming”, Yashwant Kanetkar, 2012 2. UNIX: Concepts and Applications, Sumitabha Das, Tata McGrawHill, 2011 	

Web Programming [As per Choice Based Credit System (CBCS) scheme] SEMESTER – I			
Subject Code	17MCA103	CIE Marks	50
Number of Lecture Hours/Week	04	SEE Marks	50
Number of Practical Hours/Week	02		
Total Number of Lecture Hours	50	SEE Hours	03
CREDITS – 05			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Develop Web applications using XHTML and JavaScript applying CSS. CO2: Build dynamic XHTML documents using Document Object Model (DOM). CO3: Design XML document and presentation of XML document using CSS and XSLT. CO4: Create Dynamic Web Applications using PHP and MYSQL CO5: Analyse and model requirements and constraints for the purpose of designing and implementing systems based on collaborating Web Services, with an emphasis on extensible and reusable architecture.</p>			
Modules			Teaching Hours
Module 1			10 Hours
<p>Introduction to XHTML; Basic syntax, Standard structure, Basic text markup, Images, Hypertext Links. Lists, Tables, Forms, Frames. Cascading Style Sheets: Introduction, Levels of style sheets, Style specification formats, Selector forms, Property value forms, Font properties, List properties, Color, Alignment of text, The box model, Background images, The and <div> tags, Conflict resolution.</p>			
Module 2			10 Hours
<p>The Basics of JavaScript: Overview of JavaScript, Object orientation and JavaScript, Syntactic characteristics, Primitives, operations, and expressions, Screen output and keyboard input, Control statements, Object creation and modification, Arrays, Functions, Constructors, Pattern matching using regular expressions, Errors in scripts, Java Script and XHTML Documents The JavaScript Execution Environment, The Document Object Model, Elements Access in Java Script, Events and Event Handling, Handling Events from Body Elements, Handling Events from Text Box and password Elements, The DOM2 Event Model, The navigator Object, Dom Tree Traversal and Modification.</p>			
Module 3			10 Hours
<p>Introduction to XML Introduction, Syntax, Document Structure, Document type definitions, Namespaces, XML schemas, displaying raw XML documents, Displaying XML documents with CSS, XSLT style sheets, XML processors, Web services.</p>			
Module 4			10 Hours
<p>Introduction to PHP Origins and uses of PHP, Overview of PHP, General syntactic</p>			

characteristics, Primitives, operations and expressions, Output, Control statements, Arrays, Functions, Pattern matching, Form handling, Files Building Web applications with PHP Cookies, sessions, Using Databases, Handling XML.	
Module 5	10 Hours
Introduction web 2.0 What is Web 2.0?, Folksonomies and Web 2.0, Software As a Service (SaaS). Web Services: SOAP, RPC Style SOAP, Document style SOAP, WSDL, REST services, JSON format, What is JSON?, Array literals, Object literals, Mixing literals, JSON 0053yntax, JSON Encoding and Decoding, JSON versus XML.	
Question paper pattern: <ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. <p>The students will have to answer 5 full questions, selecting one full question from each module</p>	
Text Books: <ol style="list-style-type: none"> 1. Robert W. Sebesta: Programming the World Wide Web, 4thEdition, Pearson education, 2012. 2. Advanced Web Programming, Hemanth Kumar B.N. & Prassana K.S. 	
Reference Book: <ol style="list-style-type: none"> 1. Chris Bates: Web Programming Building Internet Applications, 3rd Edition, Wiley India, 2006 	
<i>Lab Experiments</i>	
1a) Develop and demonstrate, using JavaScript script, a XHTML document that collects the USN (the valid format is: A digit from 1 to 4 followed by two uppercase characters followed by two digits followed by two uppercase characters followed by three digits; No embedded spaces allowed) of the user. Event handler must be included for the Form element that collects this information to validate the input. Messages in the alert windows must be produced when errors are detected. 1b) Modify the above program to get the current semester also (restricted to be a number from 1 to 6).	
2) Develop using JavaScript script, an XHTML document that use of onload and onfocus events.	
3a) Design an XML document to store information about a student in an engineering college affiliated to VTU. The information must include USN, Name, Name of the College, Brach, Year of Joining, and e-mail id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document. 3b) Create an XSLT style sheet for one student element of the above document and use it to create a display of that element.	
4a) Create XHTML form with Name, address line1, address line2 and email text fields. On submitting, store the values in MySQL table. Retrieve and display the data based on name. 4b) Implement a PHP program to read student data from an XML file and store into the MySQL database. Retrieve and display using SEARCH function.	

Project

Develop a web application project using the languages and concepts learnt in the theory and exercises listed in part A with a good look and feel effects. You can use any web technologies and frameworks and databases.

Note:

1. A team of two or three students must develop the web based project. However during the examination, each student must demonstrate the project individually.
2. The team must submit a brief project report (20-25 pages) that must include the following:
 - a. Introduction
 - b. System Requirement Specification
 - c. System Design
 - d. Implementation
 - e. Testing
 - f. Screen Shots
 - g. Conclusion & Future Enhancement
 - h. Bibliography
3. The Demonstration, Viva and Report Evaluation will be done for 20 Marks.

Instructions:

1. Mini project student group size is limited to two or three students only.
2. The mini project under has to be evaluated for 20 marks.
3. Project report duly signed by the Guide and HOD need to be submitted during examination.

Computer Organization [As per Choice Based Credit System (CBCS) scheme] SEMESTER – I			
Subject Code	17MCA104	CIE Marks	50
Number of Lecture Hours/Week	04	SEE Marks	50
Total Number of Lecture Hours	50	SEE Hours	03
CREDITS – 04			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Understand the Basics of Digital System.</p> <p>CO2: Understand the Basics of Computer System Organization.</p> <p>CO3: Understand the concepts of the number system in Designing Digital System.</p> <p>CO4: Gain knowledge on combinational circuits and sequential circuits.</p> <p>CO5: Analyse the need of Logic circuits in digital system.</p>			
Modules			Teaching Hours
Module 1			10 Hours
<p>Binary Systems and Combinational Logic Digital Computers and Digital Systems, Binary Numbers, Number Base Conversion, Octal and Hexadecimal Numbers, subtraction using r's and (r-1)'s complements, Binary Code. Axiomatic Definition of Boolean Algebra, Basic Theorems and Properties of Boolean Algebra, Boolean Functions, Canonical and Standard Forms, Other Logic Operations, The map Method, Two–and Three– Variable Maps, Four–Variables Map, Don't Care Conditions.</p>			
Module 2			10 Hours
<p>Arithmetic Circuits Digital Logic Gates, NAND and NOR Implementation, Introduction, Adders, Subtractors, Binary Parallel Adder, Decimal Adder, Magnitude Comparator, Decoders, Multiplexers, BOOTH algorithm for signed numbers with example.</p>			
Module 3			10 Hours
<p>Sequential Logic Introduction, different types of Flip– Flops, Triggering of Flip- Flops, Registers, Shift Registers, Ripple counter and Synchronous Counter. Machine Instruction: Introduction to Assemblers and Compilers, Memory Locations and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes.</p>			
Module 4			10 Hours
<p>Assembly Language and Input/ Output Organization Basics of Assembly Language Programme, Examples from Assembly Language Programming. Accessing I/O Devices, Interrupts, DMA, Buses.</p>			
Module 5			10 Hours
<p>The Memory System Basic Concepts, Semiconductor RAM Memories, Read– Only Memories, Speed, Size, and Cost, Cache Memories, Virtual Memories, Memory Management Requirements, Secondary Storage.</p>			
Question paper pattern:			
<ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks 			

- There will be 2 full questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module

Text Books:

1. M. Morris Mano, "Digital Logic and Computer Design", Pearson, 2012.
2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, "Computer Organization", 5th edition, Tata Mc Graw-Hill, 2011

Reference Books:

1. John P. Hayes, "Computer Architecture and Organization", Tata Mc Graw- Hill, Edition, 2012.
2. Soumitra Kumar Mandal, "Digital Electronics Principles and Applications", Tata Mc Graw-Hill, 2010

Professional Communication and Management [As per Choice Based Credit System (CBCS) scheme] SEMESTER – I			
Subject Code	17MCA105	CIE Marks	50
Number of Lecture Hours/Week	04	SEE Marks	50
Total Number of Lecture Hours	50	SEE Hours	03
CREDITS – 04			
Course Outcome (CO): At the end of this course, the students will be able to:			
CO1: Understanding the process of communication and its importance. CO2: Understanding how to write effective reports, Resumes, Letters, etc. CO3: Apply effective presentation strategies and group communication. CO4: Analyze motivation and leadership qualities. CO5: Understand the importance of ethics as an IT professional.			
Modules		Teaching Hours	
Module 1		10 Hours	
Communication in the workplace- Role of communication in Business, Process of Human Communication. Informal Communication- Listening, Non verbal Communication. Correctness Of Communication- Nature of correctness, Standard for Punctuation, Standards for Grammar, Standards for the use of numbers, Spelling, Capitalization.			
Module 2		10 Hours	
Writing for Effect- Business Etiquette, conversational style, view point, positive language, and courtesy. Basics Report Writing- Determine the Factors, Gathering the Information, Organizing the Report, Writing The Report. Physical Presentation Of Letters, Memos, And Reports- document preparation, Form of Business Letters. Strategies in the job search Process- Preparing application, Resume, Cover letter, Facing an Interview.			
Module 3		10 Hours	
Group Communication- Introduction, Group discussion, Organizational group discussion, group discussion as part of selection process meetings. Effective Presentation strategies- Introduction, Defining purpose, Analyzing audience and place, organizing contents, preparing outline, Visual Aids, Understanding Nuances of delivery, Kinesics, Proxemics, Paralinguistics, Chronemics, Sample speech.			
Module 4		10 Hours	
Motivation- Motivation and Motivators, Motivation: The Carrot and the stick, The Hierarchy of needs Theory, The Motivation – Hygiene Approach to Motivation. Leadership- Defining Leadership, Ingredients of Leadership, Trait Approaches to Leadership, Leadership behavior and styles, Situational or contingency, Approaches to Leadership.			
Module 5		10 Hours	
Ethics: An overview of Ethics- What are Ethics? Ethics in the business World, Ethics in Information Technology (IT). Ethics for IT Professionals and IT users- IT professionals, The Ethical behavior of IT professionals, IT Users.			

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 20 marks.
- There will be 2 full questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module

Text Books

1. Meenakshi Raman and Sangeeta Sharma: Technical Communication-Principles and Practices, Oxford University Press, 2004.
2. George Reynolds: Ethics in Information Technology, 2nd Edition, Thomson Course Technology, 2007

Reference Books

1. Lesikar and Flatley: Communication-Basic Business Communication Skills for Empowering the Internet Generation 9th Edition, Tata McGraw-Hill Edition, 2002
2. Harold Koonlz and Heinz Wehrich: Management-Essentials of Management, 9th Edition, McGraw-Hill International Edition, 2012

C Laboratory [As per Choice Based Credit System (CBCS) scheme] SEMESTER – I			
Subject Code	17MCA106L	CIE Marks	50
Number of Lecture Hours/Week	03 Hrs Laboratory	SEE Marks	50
		SEE Hours	03
CREDITS – 1.5			
Course Outcome (CO): At the end of this course, the students will be able to:			
CO1: Apply C concepts to develop interactive applications in C.			
CO2: Be fluent in the use of input output statements, constants, variables ,expressions and functions			
CO3: Be fluent in the use control structures, arrays ,structures and pointers			
CO4: Be fluent in the use of file handling techniques.			
Laboratory Experiments:			
1.a. Convert degrees into Fahrenheit and vice versa			
b. Calculate the salary of an employee given his basic pay, HRA = 10% of basic pay, TA=5% of his basic pay and deductions IT = 2.5% of his basic pay			
2. a. Check whether the given number is perfect number Defn: A perfect number is a positive integer that is equal to the sum of its proper positive divisors, that is, the sum of its positive divisors excluding the number itself. Example - The first perfect number is 6, because 1, 2, and 3 are its proper positive divisors, and $1 + 2 + 3 = 6$.			
b. Solve quadratic equations for the given values of a,b,c.			
3 a. Generate all Armstrong numbers upto n. Defn: An Armstrong number of three digits is an integer such that the sum of the cubes of its digits is equal to the number itself. Example - 371 is an Armstrong number, since $3^3 + 7^3 + 1^3 = 371$			
b. Convert a decimal number to a hexadecimal number			
4. Implement a menu driven C program to			
a. Insert an element into an array			
b. Delete an element from the array (first occurrence)			
5. Implement a Menu Driven C Program to			
a. Accept a string from the user			
b. Encode the string.			
c. Decode the string			
Apply the following procedure to encode it.			
1. Convert each character in a string to its ASCII value.			
2. Add an integer value to it and display the encoded string			
3. Decode the string using reverse procedure and display.			
6. Develop C program to multiply two matrices that satisfy the constraint of matrix Multiplication			
7. Develop a C program to find the saddle point of a matrix. Defn: Given a RxC Matrix, A, i.e. R rows and C columns we define a Saddle-Point as Saddle_Pt (A) for a row I and column j is that A(i,j) that is the minimum of Row i and the maximum of Col j.			
8. Develop a C program to implement a magic square of size n. Defn: A magic square is an arrangement of numbers (usually integers) in a square			

grid, where the numbers in each row, and in each column, and the numbers that run diagonally in both directions, all add up to the same number.

9. Implement a Menu driven C program to

- a. Accept two numbers n and m
- b. Sum of all integers ranging from n to m
- c. Sum of all odd integers ranging from n to m
- d. Sum of all even integers ranging from n to m

Display an error message if $n > m$. Create functions for each of the options

10. Implement a Menu Driven C Program to implement the following using recursion

- a. Factorial of a number
- b. Fibonacci series

11. Create a structure Complex Number having real and imaginary part as properties.

Write functions to add and subtract the two complex numbers.

12. Implement a menu driven C Program

- a. to copy two strings
- b. to compare two strings
- c. to reverse a string

Using pointers and not using any library functions.

13. Define a structure called student having the properties of student_id, student name and branch of the student with a sub structure of marks of 3 subjects. Write a Menu Driven C Program to:

- a. Add new student detail
- b. Delete a student detail
- c. Display all student details
- d. Display the average marks scored by the students

14. a. Implement a C Program to remove all white spaces and newline characters from a file.

b. Find whether a given word exists in the file. If it exists display the location of the word

c. Write a C program to copy one file content to another file without using inbuilt Functions

Note: In the examination each student should pick one program from the above list and another program as per examiner choice.

UNIX Programming Laboratory [As per Choice Based Credit System (CBCS) scheme] SEMESTER – I			
Subject Code	17MCA107L	CIE Marks	50
Number of Lecture Hours/Week	01 Hr Tutorial(Instructions) 02 Hrs Laboratory	SEE Marks	50
		SEE Hours	03
CREDITS – 1.5			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Understand the Unix programming environment.</p> <p>CO2: Be fluent in the use of Vi editor.</p> <p>CO3: Be able to design and implement shell scripts to manage users with different types of permission and file based applications.</p> <p>CO4: Be fluent to write Awk scripts.</p>			
<i>Laboratory Experiments:</i>			
<p>Explore the Unix environment and Explore vi editor with vim tutor. Perform the following operations using vi editor, but not limited to:</p> <ol style="list-style-type: none"> 1. Insert character, delete character, replace character 2. save the file and continue working 3. save the file and exit the editor 4. quit the editor 5. quit without saving the file 6. rename a file 7. insert lines, delete lines, 8. setline numbers 9. search for a pattern 10. move forward and backward 			
<p>1. Develop a shell script that takes a valid directory name as an argument and recursively descend all the subdirectories, finds the maximum length of any file in that hierarchy and writes this maximum value to the standard output.</p>			
<p>2. Develop shell script to implement terminal locking (similar to the lock command). It should prompt the user for a password. After accepting the password entered by the user, it must prompt again for the matching password as confirmation and if match occurs, it must lock the keyword until a matching password is entered again by the user, Note that the script must be written to disregard BREAK, control-D. No time limit need be implemented for the lock duration.</p>			
<p>3. Develop a shell script that displays all the links to a file specified as the first argument to the script. The second argument, which is optional, can be used to specify in which the search is to begin. If this second argument is not present, the search is to begin in current working directory. In either case, the starting directory as well as all its sub directories at all levels must be searched. The script need not include any error checking.</p>			
<p>4. Develop a shell script to display the calendar for current month with current date replaced by * or ** depending on whether the date has one digit or two digits</p>			
<p>5. Implement a shell script to list all the files in a directory whose filename is at least 10 characters. (us expr command to check the length)</p>			
<p>6. Develop a shell script that accept a list of filenames as its argument, count and report occurrence of each word that is present in the first argument file on other Argument files.</p>			
<p>7. Develop a shell script that reports the logging in of a specified user within one minute after</p>			

he/she login. The script automatically terminate if specified user does not login during a specified period of time.

8. Develop a shell script that folds long lines into 40 columns. Thus any line that exceeds 40 characters must be broken after 40th, a “\” is to be appended as the indication of folding and the processing is to be continued with the residue. The input is to be supplied through a text file created by the user.

9. Develop an awk script to delete duplicated line from a text file. The order of The original lines must remain unchanged.

10. Develop an awk script to compute gross salary of an employee accordingly to rule given below. If basic salary is < 10000 then HRA=15% of Basic & DA=45% of basic. If basic salary is >= 10000 then HRA=20% of basic & DA=50% of basic.

Note: In the examination each student should pick one program from the above list and another program as per examiner choice.

17MCA108L Soft skills Activity

Conduct Activities

i) To improve:

- Oral and written communication
- Interview skills
- Presentation and public speaking
- Leadership and interpersonal skills
- Time and stress management

ii) Develop:

- Positive mental attitude
- Confidence and self esteem

SEMESTER - II

<p align="center">Object Oriented Programming with C++ [As per Choice Based Credit System (CBCS) scheme] SEMESTER – II</p>			
Subject Code	17MCA201	CIE Marks	50
Number of Lecture Hours/Week	04	SEE Marks	50
Total Number of Lecture Hours	50	SEE Hours	03
CREDITS – 04			
<p>Course Outcome (CO): At the end of this course, the students will be able to:</p> <p>CO1: Apply C++ features to program design and implementation. CO2: Use C++ to demonstrate practical experience in developing object-oriented solutions. CO3: Analyze a problem description and build object-oriented software using good coding practices and techniques. CO4: Implement an achievable practical application and analyze issues related to object-oriented techniques in the C++ programming language.</p>			
Modules			Teaching Hours
Module 1			10 Hours
<p>Introduction to OOPs, Modular Programming with Functions Object Oriented paradigm, Structured vs. Object Oriented Paradigm. Elements of Object Oriented Programming: Object, Classes, Encapsulation & data abstraction, Inheritance, Polymorphism etc., C++ Overview, different data types, operators, expressions, const& Volatile qualifiers, arrays and strings, reference variables. Modular Programming with Functions: Function Components, argument passing, inline functions, function templates, recursive functions</p>			
Module 2			10 Hours
<p>Classes & Objects and Operator Overloading Introduction, Class Specification, Class Objects access members, defining member functions, Data hiding, constructors, destructors, parameterized constructors, static data members, Functions, scope resolution operator, passing objects as arguments, returning objects, friend Functions & classes, arrays of objects, Dynamic objects – Pointers to objects, Class members. Creating a Member Operator function, Binary operator overloading, Concatenation of strings, strings comparison using operator overloading, overloading the assignment operator ,overloading operators such as [], ->, increment & decrement operators, Operator overloading using friend functions +,-, overloading input stream and output stream, overloading the output operator <<, Overloading the input operator >></p>			
Module 3			10 Hours
<p>Operator Overloading(Contd): Defining Operator Function, Rules for overloading Operators. Overloading unary operators, overloading binary Operators, Overloading Comma, [], (), ->, new, delete Operators, Inheritance: Basic Concepts, Reusability & Extensibility. Defining derived classes, protected access specifier in Base class – public, private & protected inheritance – constructors and Destructors in derived classes – Types of Inheritances. Virtual base class. Virtual functions:</p>			

Normal member functions accessed with pointers, virtual member function access, late binding, pure virtual function, abstract classes. Polymorphism: Overloading Concepts Function Overloading: Functions with different sets of parameter default and constant parameters.	
Module 4	10 Hours
Templates: Generic classes, a class template with more than one generic type, the power of templates. Namespaces and Conversion Functions: Namespace fundamentals, using keyword, unnamed namespaces, const member function and mutable, Volatile member functions Exception Handling: Exception handling model, Exception handling constructs, list of Exceptions catch all exceptions, handling uncaught exceptions.	
Module 5	10 Hours
I/O Streams: IO Stream basics, Output operator <<, input >>, additional I/O operators, file input & output, manipulators Type Conversions: Basic to class, Class to Basic and one Class to another Class type. Advanced Typecasting Run Time Type ID and the Casting Operators: Run Time Type Identification(RTTI), the Casting Operator, dynamic_cast STL STL: An overview, containers, vectors, lists, maps	
Question paper pattern: <ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. The students will have to answer 5 full questions, selecting one full question from each module	
Text Books: <ol style="list-style-type: none"> 1. Herbert Schildt: C++ The Complete Reference, 4th Edition, Tata McGraw Hill, 2007. 2.K.R. Venugopal, RajkumarBuyya, T. Ravishankar: Mastering C++ ,TataMcGraw Hill Publication, 2006 3. Stanley B.Lippmann, JoseeLajore: C++Primer, 4th Edition, Addison Wesley, 2005. 	
Reference Books: <ol style="list-style-type: none"> 1. Stephen Prata : C++ Primer Plus, 6th Edition, Pearson Education. 2. Al Stevens: C++ Programming, 7th Edition, Wiley India Publications 3. Object oriented programming with C++, E. Balaguruswamy, TMH. 	

Data Structures using C [As per Choice Based Credit System (CBCS) scheme] SEMESTER – II			
Subject Code	17MCA202	CIE Marks	50
Number of Lecture Hours/Week	04	SEE Marks	50
Total Number of Lecture Hours	50	SEE Hours	03
Credits – 05			
Course Outcome (CO): At the end of this course, the students will be able to:			
CO1: Understand the concepts of Pointers, structures, strings, arrays and their representations.			
CO2: Acquire the knowledge of various types of data structures as ADT, Sorting and searching operations.			
CO3: Analyze the performance of Stack, Queue, Lists, Trees, Hashing, Searching and Sorting techniques.			
CO4: Implement all the applications of Data structures in a high-level language.			
CO5: Design and apply appropriate data structures for solving computing problems.			
Modules			Teaching Hours
Module 1			10 Hours
Introduction to Data Structures Classification of Data Structures: Primitive and Non- Primitive, Linear and Nonlinear; Data structure Operations Pointers: Pointer Expression, Pointer as function arguments, Functions returning pointers, Pointers to functions. Structures: Declaring and using structure types. ADT: Array as ADT, Operations - Insert, Delete, Search, Sort. String: Definition, Representation, String as ADT, Operations – Insert, Delete, Concatenate, Comparing, Substring. Recursion - Factorial, GCD, Fibonacci Sequence. Algorithm Complexity, Space Complexity & Time Complexity.			
Module 2			10 Hours
Stack: Definition, Representation, Stack as ADT, Operations and applications: Polish and reverse polish expressions, Infix to postfix conversion, evaluation of postfix expression, infix to prefix, postfix to infix conversion; Queue: Definition, Representation, Queue as ADT, Operations, Queue Variants: Circular Queue, Priority Queue, Double Ended Queue; Applications of Queues. Programming Examples.			
Module 3			10 Hours
Linked List: Limitations of array implementation, Memory Management: Static (Stack) and Dynamic (Heap) Memory Allocation, Memory management functions. Definition, Representation, Operations: getnode() and Freenode() operations, Types: Singly Linked List. Linked list as a data Structure, Inserting and removing nodes from a list, Linked implementations of stacks, Example of list operations such as insert and delete an element before a key element, Header nodes, Array implementation of lists. Circular Linked List: Inserting, deleting and searching elements in lists.			
Module 4			10 Hours
Double Linked List: Inserting and Deleting Nodes, Queue as doubly			

linked lists, such as insert into position, Delete a specified element. Application of Linked Lists: Stacks, Queues, Double-ended Queues, Priority Queues, Trees, BST. Application of DLL to memory management. Graph Data Structure: Basics, Operations- Add/Remove Vertex, Add/Remove Edge, Display. Hashing: The Hash Table organizations, Hashing Functions.	
Module 5	10 Hours
Trees: Definitions, Terminologies, Array and linked Representation of Binary Trees, Types- Complete/full, Almost Complete, Strictly, Skewed; Traversal methods - Inorder, postorder, preorder; Binary Search Trees - Creation, Insertion, Deletion, Traversal, Searching. AVL Tree and Operations, Red-Black Tree.	
Question paper pattern: <ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. <p>The students will have to answer 5 full questions, selecting one full question from each module</p>	
Text Books: <ol style="list-style-type: none"> 1. Data Structures Using C and C++ by Yedidyah Langsam and Moshe J. Augenstein and Aaron M Tenanbanum, 2nd Edition, Pearson Education Asia, 2002. 2. Richard F Giberg and Behrouz A Forouzan: Data Structures – A Pseudo code Approach with C, 2nd Edition, Cengage Learning. 2004 	
Reference Books: <ol style="list-style-type: none"> 1. Balaguruswamy: Data Structures Using C, McGraw Hill Education. 2013 2. Data Structures and Algorithm Analysis in C, Mark Allen Weiss, 2nd Edition, Pearson Education Aisa, 1997. 	
Laboratory Experiments:	
1. Develop a C program to implement the following using recursion	
<ol style="list-style-type: none"> a. Sum of n numbers b. Generate Fibonacci sequence. 	
2. Implement a Program in C for Converting an Infix Expression to Postfix Expression	
3. Implement a Program in C for Evaluating an Postfix Expression .	
4. Implement a menu driven Program in C for the following operations on Circular QUEUE of Integers (Array Implementation of Queue with maximum size MAX)	
<ol style="list-style-type: none"> a. Insert an Element on to Circular QUEUE b. Delete an Element from Circular QUEUE c. Demonstrate Overflow and Underflow situations on Circular QUEUE d. Display the status of Circular QUEUE e. Exit 	
5. Develop a C program to Simulate the working of a linked list providing the following	

<p>operations</p> <ol style="list-style-type: none"> a. Insert at the beginning b. Insert at the end c. Delete at the beginning d. Delete at the end e. Display
<p>6. Develop a C Program using Doubly Linked List to Implement Stack operations to store Integers</p> <ol style="list-style-type: none"> a. Push an Element on to Stack b. Pop an Element from Stack c. Demonstrate Overflow and Underflow situations on Stack d. Display the status of Stack
<p>7. Implement a menu driven Program in C for the following operations on Binary Tree of Integers</p> <ol style="list-style-type: none"> a. Create a BST of N Integers b. Traverse the BST in Inorder, Preorder and Post Order c. Delete an element from BST d. Exit
<p>8. Develop a C program to Implement the following searching techniques</p> <ol style="list-style-type: none"> a. Linear Search b. Binary Search
<p>9. Develop a C program to implement the following sorting algorithms using user defined functions:</p> <ol style="list-style-type: none"> a. Bubble sort (Ascending order) b. Selection sort (Descending order)
<p>10. Develop a C program to implement Double ended queue.</p>
<p>11. Develop a C program using dynamic variables and pointers, to construct a singly linked list consisting of the following information in each node: student id (integer), student name (character string) and semester (integer). The operation to be supported are:</p> <ol style="list-style-type: none"> a. The insertion operation <ol style="list-style-type: none"> i. At the front of a list. ii. At the back of the list. iii. At any position in the list. b. Deleting a node based on student id. If the specified node is not present in the list an error message should be displayed. Both the options should be demonstrated. c. Searching a node based on student id and updates the information content. If the specified node is not present in the list an error message should be displayed. Both situations should be displayed. d. Displaying all the nodes in the list.
<p>12. Develop a C program using dynamic variables and pointers, to construct an ordered singly linked list based on the rank of the student, where each node consists of the following information: student id, student name and rank (integer).</p>
<p>13. Implement a C program to sort a given array using Quick sort.</p>
<p>14. Develop a C program to perform following operations on string without using string inbuilt functions.</p> <ol style="list-style-type: none"> a) Copying b) Concatenation

Database Management System [As per Choice Based Credit System (CBCS) scheme] SEMESTER – II			
Subject Code	17MCA203	CIE Marks	50
Number of Lecture Hours/Week	04	SEE Marks	50
Total Number of Lecture Hours	50	SEE Hours	03
Credits – 04			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Demonstrate the fundamentals of data models and conceptualize and depict a database system and Make use of ER diagram in developing ER Model</p> <p>CO2: Summarize the SQL and relational database design.</p> <p>CO3: Illustrate transaction processing, concurrency control techniques and recovery</p> <p>CO4: Inference the database design in the real world entities.</p>			
Modules			Teaching Hours
Module 1			10 Hours
<p>Introduction: Characteristics of Database approach, Actors on the Scene, Workers behind the scene, Advantages of using DBMS approach, Data models, schemas and instances, Three –schema architecture and data independence, Database languages and interfaces, the database system environment, Centralized and client -server architectures, Classification of Database Management systems. Entity-Relationship Model: Conceptual Database using high level conceptual data models for Database Design, A Sample Database Application, Entity types, Entity sets, Attributes and Keys Relationship types, Relationship Sets, Roles and Structural Constraints Weak Entity Types.</p>			
Module 2			10 Hours
<p>Relational Model Relational Model Concepts and Constraints , Relational Database Schema Update Operations, Transactions and Dealing with Constraint violations, Unary Relational operations, Relational Algebra Operations from Set Theory, Binary Relational Operations, JOIN and DIVISION, Additional Relational Operations, Examples of Queries in Relational Algebra Relational Database Design Using ER-to-Relational Mapping</p>			
Module 3			10 Hours
<p>Introduction to SQL: Overview of the SQL Query Language, SQL Data Definition, Basic structure of SQL Queries, Additional Basic Operations, Null values, Aggregate Functions, nested Sub queries, Modification of the Database, Join Expressions, Views, Transactions, Integrity Constraints, SQL Data Types and Schemas, Authorization. Database programming issues and techniques, Embedded SQL.</p>			
Module 4			10 Hours
Database Design:			

<p>Informal Design Guidelines for Relation Schemas, Functional Dependencies, Normal Forms based on Primary Keys, General Definitions of 2nd and 3rd Normal Forms, Boyce Codd Normal Forms, Multivalued Dependencies and IV Normal Forms, Join Dependencies and V Normal Forms, Inference Rules, Equivalence and Minimal Cover.</p>	
<p>Module 5</p>	<p>10 Hours</p>
<p>Transaction Management: Transaction Concept, Transaction properties Atomicity and Durability, Serializability and isolation (ACID properties), Transaction Isolation and Transaction Isolation Levels. Concurrency Control: Lock Based Protocols-Types of Locks, Simplistic Lock Protocol, Pre-claiming Lock Protocol, Two-Phase Locking (2PL) and Strict Two-Phase Locking. Deadlock Handling-deadlock prevention schemes, Deadlock Avoidance,(Wait-for Graph method) Recovery System: Failure Classification, Storage, Recovery and Atomicity, Recovery Algorithm.</p>	
<p>Question paper pattern:</p> <ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. <p>The students will have to answer 5 full questions, selecting one full question from each module</p>	
<p>Text Books:</p> <ol style="list-style-type: none"> 1. Elmasri and Navathe: Fundamentals of Database Systems, 5th Edition, Addison -Wesley, 2011. 2. Silberschatz, Korth and Sudharshan Data base System Concepts,6th Edition, Tata McGraw Hill, 2011. 	
<p>Reference Books:</p> <ol style="list-style-type: none"> 1. Raghu Ramakrishnan and Johannes Gehrke: Database Management Systems, 3rd Edition, McGraw-Hill, 2003 2. Database Principles Fundamentals of Design, Implementation and Management by Coronel, Morris, Rob- Cengage Learning 2012 	

Operating Systems [As per Choice Based Credit System (CBCS) scheme] SEMESTER – II			
Subject Code	17MCA204	CIE Marks	50
Number of Lecture Hours/Week	04	SEE Marks	50
Total Number of Lecture Hours	50	SEE Hours	03
CREDITS – 04			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Understand the Basics of Computer and Operating System Structure CO2: Realize the concept of Process Management and Mutual Exclusion CO3: Understand the concepts of the Deadlock and different approaches to memory management CO4: Learn the concepts of file system CO5: Understand the concepts of Computer Security and Linux</p>			
Modules			Teaching Hours
Module 1			10 Hours
<p>Operating Systems Structure Operating System operations, Registers, Memory Hierarchy, Cache Memory, Introduction to Operating System, Mainframe Systems, Desktop Systems, Multiprocessor Systems, Distributed Systems, Clustered Systems, Real- Time Systems, Handheld Systems, System Components, System Calls, System Programs, System Structure, Virtual Machines</p>			
Module 2			10 Hours
<p>Process Management and Threads Process States, Process Management, IPC, Threads, Overview of Threads, Symmetric Multiprocessing (SMP), Scheduling criteria; Scheduling algorithms. Principles of Concurrency, Mutual Exclusion, Hardware Support: Semaphores, Monitors, Readers/Writes Problem.</p>			
Module 3			10 Hours
<p>Deadlock and Memory Management: Principles of Deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Dining Philosophers Problem, Memory Management: Swapping, Contiguous Memory Allocation, Paging, Segmentation, Segmentation with Paging, Demand Paging, Page Replacement Alg,</p>			
Module 4			10 Hours
<p>File System and Secondary Storage File Concept, Access Methods, Directory Structure, File System Mounting, File Sharing, Protection, File-System Structure, File- System Implementation, Directory Implementation, Allocation Methods, Free-Space Management, Disk Structure, Disk Scheduling, Disk Management.</p>			
Module 5			10 Hours
<p>Computer Security and Linux Security Measurements, Program and System Threats, Fork/Exec Model, Network Layers in Linux System. Linux Components, Buddy Heap Algorithm, Block Buffer cache, ex2fs file system, Case study on the modern operating system.</p>			
Question paper pattern:			

- The question paper will have ten questions.
- Each full question consists of 20 marks.
- There will be 2 full questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module

Text Books:

1. Silberschatz, Galvin, Gagne, “Operating System Concepts” John Wiley, Sixth Edition, 2004
2. William Stallings, “Operating System Internals and Design Principles” Pearson, 6th edition, 2012

Reference Books:

1. Chakraborty, “Operating Systems” Jaico Publishing House, 2011.
2. Dhananjay M. Dhamdhere, “Operating Systems—A Concept—Based Approach”, TataMcGraw—Hill, 3rd Edition, 2012
3. Elmasri, Carrick, Levine, “Operating Systems—Aspiral Approach”, Tata McGraw—Hill, 2012

System Programming [As per Choice Based Credit System (CBCS) scheme] SEMESTER – II			
Subject Code	17MCA205	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	50	SEE Hours	03
CREDITS – 03			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Understand the basics of system programs like editors, compiler, assembler, linker, loader, interpreter and debugger.</p> <p>CO2: Describe the various concepts of assemblers and Macro Processors.</p> <p>CO3: Understand the various phases of compiler and compare its working with assembler.</p> <p>CO4: Understand how linker and loader create an executable program from an object module created by assembler and compiler.</p> <p>CO5: know various editors and debugging techniques.</p>			
Modules			Teaching Hours
Module 1			10 Hours
<p>Machine Architecture: Introduction, System Software and Machine Architecture, Simplified Instructional Computer (SIC) - SIC Machine Architecture, SIC/XE Machine Architecture, SIC Programming Examples.</p> <p>Editors and Debugging Systems: Text Editors - Overview of Editing Process, User Interface, Editor Structure, Interactive Debugging Systems - Debugging Functions and Capabilities, Relationship With Other Parts Of The System, User-Interface Criteria.</p>			
Module 2			10 Hours
<p>Assemblers: Basic Assembler Function - A Simple SIC Assembler, Assembler Algorithm and Data Structures, Machine Dependent Assembler Features - Instruction Formats & Addressing Modes, Program Relocation. Machine Independent Assembler Features – Literals, Symbol-Definition Statements, Expression, Program Blocks, Control Sections and Programming Linking, Assembler Design Operations - One-Pass Assembler, Multi-Pass Assembler, Implementation Examples - MASM Assembler.</p>			
Module 3			10 Hours
<p>Loaders and Linkers: Basic Loader Functions - Design of an Absolute Loader, A Simple Bootstrap Loader, Machine- Dependent Loader Features – Relocation, Program Linking, Algorithm and Data Structures for a Linking Loader; Machine-Independent Loader Features – Automatic Library Search, Loader Options, Loader Design Options - Linkage Editor, Dynamic Linkage, Bootstrap Loaders, Implementation Examples - MS-DOS Linker.</p>			
Module 4			10 Hours
<p>Macro Processor: Basic Macro Processor Functions - Macro Definitions and Expansion, Macro Processor Algorithm and Data Structures, Machine-Independent Macro Processor Features –Concatenation of Macro Parameters, Generation of Unique Labels, Conditional Macro Expansion, Keyword Macro Parameters, Macro Processor Design Options – Recursive</p>			

Macro Expansion, General- Purpose Macro Processors, Macro Processing Within Language Translators, Implementation Examples - MASM Macro Processor, ANSI C Macro Processor.	
Module 5	10 Hours
Compilers: Basic Compilers Functions- Grammars, Lexical Analysis, Syntactic Analysis, Code Generation. Machine Dependent Compiler Features- Intermediate Form of the Program, Machine dependent code Optimization. Machine Independent Compiler Features- Structured variables, Machine Independent code Optimization. Compiler Design Options- Division System Software: introduction into passes, Interpreters, P-code Compilers.	
<p>Question paper pattern:</p> <ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. <p>The students will have to answer 5 full questions, selecting one full question from each module</p>	
<p>Text Book:</p> <ol style="list-style-type: none"> 1. System Software: An introduction to System Programming, Leland L. Beck, 3rd edition, Pearson Education, 1997. 	
<p>Reference Book:</p> <ol style="list-style-type: none"> 1. D.M. Dhamdhare: System Programming and Operating System, 2nd edition, Tata McGraw – Hill, 1999. 	

Object Oriented Programming with C ++ Laboratory

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – II

Subject Code	17MCA206L	CIE Marks	50
		SEE Marks	50
Number of Lecture Hours/Week	03 Hrs Laboratory	SEE Hours	03

CREDITS – 1.5**Course Outcome (CO): At the end of this course, the students will be able to:****CO1:** Apply and implement major object oriented concepts like function overloading, operator overloading, Encapsulations, and inheritance, message passing to solve real-world problems.**CO2:** Use major C++ features such as Virtual functions, Templates for data type independent designs and File I/O to deal with large data sets.**CO3:** Analyze, design and develop solutions to real-world problems applying OOP Concepts of C++.**Laboratory Experiments:****NOTE: The experiments are to be carried using discrete components only.**

1	Define a STUDENT class with USN, Name, and Marks in 3 tests of a subject. Declare an array of 10 STUDENT objects. Using appropriate functions, find the average of the two better marks for each student. Print the USN, Name and the average marks of all the students.
2	Implement a C++ program to create a class called COMPLEX and implement the following overloading functions ADD that return a complex number: (i) ADD (a, s2) – where ‘a’ is an integer (real part) and s2 is a complex number (ii) ADD (s1, s2) – where s1 and s2 are complex numbers.
3	Create a class called MATRIX using two-dimensional array of integers. Implement the following operations by overloading the operator == which checks the compatibility of two matrices to be added and subtracted. Perform the addition and subtraction by overloading the + and – operators respectively. Display the results by overloading the operator <<. If (m1==m2) then m3 = m1+m2 and m4 = m1-m2 else display error.
4	Implement a C++ program to create a class called STACK using an array of integers. Implement the following operations by overloading the operators ‘+’ and ‘--’: (i) s1 = s1 + element; where s1 is an object of the class STACK and element is an integer to be pushed on the top of the stack (ii) s1 = --s1 ; where s1 is an object of the class STACK. ‘--’ operator pops the element. Handle the STACK empty and full conditions. Also display the contents of the stack after each operation, by overloading the << operator.
5	Implement a program to demonstrate friend functions and friend classes
6	Write a C++ program to create a class template called QUEUE with member functions to add an element and to delete an element from the queue. Using the member functions, implement a queue of integers and double. Demonstrate the operations by displaying the contents of the queue after every operation.
7	Implement a C++ program to create a class called STRING and implement the following operations. Display the results after every operation by <i>overloading the operator <<</i> . (i) STRING s1 = “NMIT” (ii) STRING s2 = “AUTONOMOUS”

	(iii) STRING s3 = s1 + s2 (Use copy constructor)
8	Create an abstract base class EMPLOYEE with data members: Name, EmpID and BasicSal and a pure virtual function Cal_Sal(). Create two derived classes MANAGER (with data members: DA and HRA) and SALESMAN (with data members: DA, HRA and TA). Write appropriate constructors and member functions to initialize the data, read and write the data and to calculate the net salary. The main() function should create array of base class pointers/references to invoke overridden functions and hence to implement run-time polymorphism
9	Implement a C++ program to create a class called STUDENT with data members USN, Name and Age. Using inheritance, create the classes UGSTUDENT and PGSTUDENT having fields as Semester, Fees and Stipend. Enter the data for at least 5 students. Find the semester wise average age for all UG and PG students separately.
10	Implement a program to create a file to store some records and search for a particular record and display it.
11	Create a file employee with fields employee number, name, age and department, create a second file salary with fields gross salary and net salary. Merge the two files and display the contents of the third file
12	Implement a program to implement Exception Handling using Classes

Note: In the examination each student should pick one program from the above list and another program as per examiner choice.

Database Management Systems Laboratory
[As per Choice Based Credit System (CBCS) scheme]
SEMESTER – II

Subject Code	17MCA207L	CIE Marks	50
Number of Lecture Hours/Week	03 Hrs Laboratory	SEE Marks	50
		SEE Hours	03

CREDITS – 02

Course Outcome (CO): At the end of this course, the students will be able to:

- CO1:** Understand, appreciate the underlying concepts of database technologies
CO2: Create database with different types of integrity constraints and use the SQL Commands such as DDL, DML, DCL, TCL to access data from database objects.
CO3: Design and implement a database schema for a given problem domain
CO4: Perform embedded and nested queries
CO5: Take up real world problems independently

DBMS Lab Experiments:

Instructions for the Exercises:

1. Draw ER diagram based on given scenario with various Constraints.
2. Create Relational Database Schema based on the above scenario using Mapping Rules.
3. Perform the given queries using any RDBMS Environment.
4. Suitable tuples have to be entered so that queries are executed correctly
5. The results of the queries may be displayed directly

Laboratory Experiments:

1. Create the following tables with properly specifying Primary keys, Foreign keys and solve the following queries.

BRANCH(Branchid,Branchname,HOD)
 STUDENT(USN,Name,Address,Branchid,sem)
 BOOK(Bookid,Bookname,Authorid,Publisher,Branchid)
 AUTHOR(Authorid,Authername,Country,age)
 BORROW(USN,Bookid,Borrowed_Date)

Queries:

1. List the details of Students who are all studying in 2nd sem MCA.
2. List the students who are not borrowed any books.
3. Display the USN, Student name, Branch_name, Book_name, Author_name, Books_Borrowed_Date of 2nd sem MCA Students who borrowed books.
4. Display the number of books written by each Author.
5. Display the student details who borrowed more than two books.
6. Display the student details who borrowed books of more than one Author.
7. Display the Book names in descending order of their names.
8. List the details of students who borrowed the books which are all published by the same Publisher.

2. Design an ER-diagram for the following scenario, Convert the same into a relational model and then solve the following queries.

Consider a Cricket Tournament “ABC CUP” organized by an organization. In the tournament

there are many teams are contesting each having a Teamid, Team_Name, City, a coach. Each team is uniquely identified by using Teamid. A team can have many Players and a captain. Each player is uniquely identified by Playerid, having a Name, and multiple phone numbers, age. A player represents only one team. There are many Stadiums to conduct matches. Each stadium is identified using Stadiumid, having a stadium_name, Address (involves city, area_name, pincode).A team can play many matches. Each match played between the two teams in the scheduled date and time in the predefined Stadium. Each match is identified uniquely by using Matchid. Each match won by any of the one team that also wants to record in the database. For each match man_of_the match award given to a player.

Queries:

1. Display the youngest player (in terms of age) Name, Team name, age in which he belongs of the tournament.
2. List the details of the stadium where the maximum number of matches were played.
3. List the details of the player who is not a captain but got the man_of _match award at least in two matches.
4. Display the Team details who won the maximum matches.
5. Display the team name where all its won matches played in the same stadium.

3. Consider the following Scenario and design an ER-Diagram, map the designed ER-diagram into a Relational model.

Consider an organization “ABC” having many employees. An employee works for one department. Each employee identified by using Empid, having Name, address (described as House_no, city, district, state, pin code) and more than one phone numbers. Department identified by using Dno, having Dname, Dlocation. Each Department having a manager . Each department having many employees. There are many Projects, each project is controlled by the department. Each Project uniquely identified by Pno, having Project_name,Project_location. An employee works on many Projects. Number of hours per week worked on each project by an Employee also needs to be recorded in the database. A project is worked by many employees. Each employee supervised by the supervisor. Employee having many dependents. Dependents having the dependent_name, gender, age, address. Dependents are identified by Empid.

T1(Empid, Emp_Name,city, district, state, pin_code, phoneno, Dno,Dname,Dlocation, Dept_mgr_id, Pno, Project_name, Project_location, Number_of_Hours,Supervisor_Empid, Dependent_name, gender, address)

Deduce the above Relation T1 into the 3NF and then solve the following queries.

Queries:

1. Display the details of the employees who are working on both the projects having project_no 5 and 10.
2. Display the details of employees having atleast two dependents.
3. Display the project name on which more number of employees are working.
4. Retrieve the employees who do not have any dependents.
5. Display the Employee details whose total number of hours per week working on various projects is maximum than all other employees.
6. create a view to display the number of employees working in each department

4. Design an ER-diagram for the following scenario, Convert the same into a relational model, normalize Relations into a suitable Normal form and then solve the following queries.

A country can have many Tourist places. Each Tourist place is identified by using tourist_place_id, having a name, belongs to a state, Number of kilometers away from the capital city of that state, history. There are many Tourists visits tourist places every year. Each tourist is identified uniquely by using Tourist_id, having a Name, age, Country and multiple emailids. A tourist visits many Tourist places, it is also required to record the visted_date in the database. A tourist can visit a Tourist place many times at different dates. A Tourist place can be visited by

many tourists either in the same date or at different dates.

Queries:

1. List the state name which is having maximum number of tourist places.
2. List details of Tourist place where maximum number of tourists visited.
3. List the details of tourists visited all tourist places of the state “KARNATAKA”.
4. Display the details of the tourists visited at least one tourist place of the state, but visited all states tourist places.
5. Display the details of the tourist place visited by the tourists of all country.

5. Design an ER-diagram for the following scenario, Convert the same into a relational model, normalize Relations into a suitable Normal form and then solve the following queries.

A country wants to conduct an election for the parliament. A country having many constituencies. Each constituency is identified uniquely by Constituency_id, having the Name, belongs to a state, Number_of_voters. A constituency can have many voters. Each voter is uniquely identified by using Voter_id, having the Name, age, address (involves Houseno, city, state, pincode). Each voter belongs to only one constituency. There are many candidates contesting in the election. Each candidates are uniquely identified by using candidate_id, having Name, phone_no, age, state. A candidate belongs to only one party. There are many parties. Each party is uniquely identified by using Party_id, having Party_Name, Party_symbol. A candidate can contest from many constituencies under a same party. A party can have many candidates contesting from different constituencies. No constituency having the candidates from the same party. A constituency can have many contesting candidates belongs to different parties. Each voter votes only one candidate of his/her constituency.

Queries:

1. List the details of the candidates who are contesting from more than one constituencies which are belongs to different states.
2. Display the state name having maximum number of constituencies.
3. Create a stored procedure to insert the tuple into the voter table by checking the voter age. If voter’s age is at least 18 years old, then insert the tuple into the voter else display the “Not an eligible voter msg”.
4. Create a stored procedure to display the number_of_voters in the specified constituency. Where the constituency name is passed as an argument to the stored procedure.
5. Create a TRIGGER to UPDATE the count of “Number_of_voters” of the respective constituency in “CONSTITUENCY” table, AFTER inserting a tuple into the “VOTERS” table.

Note: In the examination each student should pick one program from the above list and Examiner may ask the students for modifications in the query.

Technical Seminar

17MCA208S Seminar

- Students should present the seminar on cutting edge/emerging/state of the art technologies in the field of Computer Science and Applications.
- Duration of the seminar should be approximately 45 minutes.
- Student should submit the write up on seminar topic containing at least 10 pages

Report format:

- 1. Cover Page**
- 2. Acknowledgement**
- 3. Abstract**

Table of Contents:

Sl. No	Description	Page No.
1.	Introduction	
2.	Literature Survey	
3	Topic Specific Description	
4	Conclusions	
5	Bibliography	

Rules:

- 1) Put page borders
- 2) Main Heading font size 17(Times New Roman)
- 3) Subheading : font size 14(Times New Roman)
- 4) Text: font size 12(Times New Roman)
- 5) Figures should have Fig numbers at the bottom of the figure e.g. Fig 3.2 is second figure in 3rd chapter
- 6) Tables should have table numbers at the top example Table 5.1 i.e, 1st table in 5th chapter
- 7) Paragraph spacing 1.5

SEMESTER - III

Computer Networks

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER –III

Subject Code	17MCA301	CIE Marks	50
Number of Lecture Hours/Week	04	SEE Marks	50
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course Outcome (CO): At the end of this course, the students will be able to:

CO1: Understand the various applications & Communication medias.

CO2: Compare various Error detection and Correction techniques, Understand Internet Control protocols and Internet transport protocols in computer network.

CO3: Comprehend different network layer functionalities. Compare various Routing and Congestion control algorithms.

CO4: Analyze the working of transport layer and application layer.

CO5: Understand the concept of packet tracer

Modules	Teaching Hours
Module 1	10 Hours
Introduction Uses of Computer Applications: Business Applications, Home Application, Mobile Users, Network Hardware, Reference Models: OSI reference model, TCP/IP reference model. The Physical Layer Guided Transmission Media, Wireless Transmission Media, Digital modulation and Multiplexing, Mobile telephone systems (1G,2G,3G and 4G).	
Module 2	10 Hours
Data Link layer Data link layer design issues, Error Detection: Parity, Checksum, CRC, Correction Codes: Hamming codes, Data Link Protocols: Simplex, Simplex Stop and Wait for an error-free channel and Nosi channel, Sliding window protocols: one bit sliding window, Go Back N and	

<p>Selective repeat.</p> <p>Medium Access Control Sub Layer</p> <p>The Channel Allocation Problem, Multiple access protocols: ALOHA, Collision free protocols: Bit Map and Token Passing.</p>	
Module 3	10 Hours
<p>Network Layer</p> <p>Network Layer Design issues, Routing algorithms: Optimality Principle, Shortest Path, Flooding, Distance Vector Link State, Congestion Control Algorithms: Quality of Service: Application Requirements and Traffic Shaping, Internetworking and The Network Layer in the Internet: IPV4, IP addresses, IPV6.</p>	
Module 4	10 Hours
<p>The Transport Layer</p> <p>The Transport Service: Services Provided to the Upper Layers, Transport Service Primitives, Berkley Sockets; Elements of Transport Protocols: Addressing, Connection Establishment, Connection Release, Flow Control and Buffering, Multiplexing, Crash Recovery, The Transport Service, Elements of Transport Protocols, Congestion Control, The Internet Transport Protocol: UDP, The Internet Transport Protocols – TCP.</p>	
Module 5	10 Hours
<p>The application Layer</p> <p>DNS: Domain Name Space, Domain Resource Records, Domain Name Servers. Electronic mail: SMTP, The World Wide Web: Static and dynamic web pages, web applications, HTTP, mobile web. Streaming audio and Video: Digital audio and video, streaming stored and line media, real-time conferencing, Content Delivery: content and internet traffic, server forms, web proxies, content delivery networks, peer-to-peer networks, Network packet tracer.</p>	
<p>Question paper pattern:</p> <ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. <p>The students will have to answer 5 full questions, selecting one full question from each</p>	

module

Text Books:

1. “Computer Networks” by Andrew S Tanenbaum, David J Wetheral, 5th Edition, Pearson 2011
2. “Data and Computer Communications” by William Stallings , Above 7th Edition , 2004

Reference Books:

1. Behrouz A. Forouzan, Data Communications and Networking, Tata McGraw-Hill 5th Edition, 2017.
2. James F. Kurose and Keith W. Ross, Computer Networking- A Top-Down Approach Featuring the Internet, 7th Edition, Pearson, 2016.

Programming with Java [As per Choice Based Credit System (CBCS) scheme] SEMESTER – III			
Subject Code	17MCA302	CIE Marks	50
Number of Lecture Hours/Week	04	SEE Marks	50
Total Number of Lecture Hours	50	SEE Hours	03
Credits – 04			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Understand the concepts of object oriented programming like Encapsulation, Abstraction and polymorphism to solve the real-life problems.</p> <p>CO2: Demonstrate the usage of Interfaces, packages and Exception handling.</p> <p>CO3: Apply the concepts event handling and applets for the internet programming.</p> <p>CO4: Apply the concepts of Collections and maps</p> <p>CO5: Analyze the software requirements needed to develop an Enterprise Application.</p>			
Modules			Teaching Hours
Module 1			10 Hours
<p>Introduction To Java: How java changed the internet; Java Buzz words, Byte Code; Object oriented programming; First Simple Java program, Introducing Classes: Classes Fundamentals; Declaring Objects, Assigning Object Reference Variable; Introducing Methods; Inheritance Basics-using Super; Creating Multilevel Hierarchy, When constructors are called, method Overriding, Dynamic Method Dispatch, Abstract classes, final with inheritance.</p>			
Module 2			10 Hours
<p>Packages and Interfaces, Exception handling in java: Packages, Access Protection, importing packages, interfaces. Exception Handling Fundamentals, Exception types, uncaught Exception, using try and catch, multi catch clause, nested try catch, throw, throws, finally, java's built in exceptions, creating your own exception subclasses, chained exceptions, using exceptions.</p>			
Module 3			10 Hours
<p>Multi-Threaded Programming ; The java tread model, The main thread, Creating thread, creating multiple threads, Using isAlive() and join(), Thread priorities, Synchronization, Inter thread communication; Suspending , resuming and stopping threads. Applets, The Applet Class: Two types of Applets; Applet basics; Applet Architecture; An Applet skeleton; Simple Applet display methods; Using the Status Window; The HTMLAPPLET tag; Passing parameters to Applets; Introduction to I/O classes and class hierarchy.</p>			
Module 4			10 Hours
<p>Event Handling: Two event handling mechanisms; The delegation event model; Event classes; Sources of events; Event listener interfaces; Using the delegation event model; Adapter classes; Inner classes. Swings: The origins of Swing; Two key Swing features; Components and Containers; The Swing Packages; A simple Swing Application;</p>			
Module 5			10 Hours
Collections: the collection interfaces and classes, accessing collection via			

iterator, storing user defined classes in collections, the random access interface, working with maps, comparators, the collection algorithms, arrays, the generic collections, Introduction to client/server architecture, Java Servlets and JDBC with sample programs.	
--	--

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 20 marks.
- There will be 2 full questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module

Text Books:

1. Java - The Complete Reference – Herbert Schildt, 7th Edition, Tata McGraw Hill, 2007.
2. J2EE - The Complete Reference – Jim Keogh, Tata McGraw Hill, 2007.

Reference Books:

1. Introduction to JAVA Programming – Y. Daniel Liang , 6thEdition, Pearson Education, 2007.
2. The J2EE Tutorial – Stephanie Bodoff et al, 2nd Edition, Pearson Education, 2004.

Python Programming [As per Choice Based Credit System (CBCS) scheme] SEMESTER – III			
Subject Code	17MCA303	CIE Marks	50
Number of Lecture Hours/Week	04	SEE Marks	50
Total Number of Lecture Hours	50	SEE Hours	03
CREDITS – 04			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Understand and comprehend the basics of python programming.</p> <p>CO2: Implement advanced concepts like List, Dictionaries, Sets and Tuples for optimized data storage.</p> <p>CO3: Understands about files and its applications.</p> <p>CO4: Apply object-oriented programming concepts to develop dynamic interactive Python applications</p> <p>CO5: Apply knowledge in real time applications.</p>			
Modules			Teaching Hours
Module 1			10 Hours
Installing Python, Simple program using Python, Expressions and Values, Variables and Computer Memory, error detection, Multiple line statements, Designing and using functions, functions provided by Python, Tracing function calls in memory model, omitting return statement. Working with Text: Creating Strings of Characters, Using Special Characters in Strings, Creating a Multiline String, Printing Information, Getting Information from the Keyboard.			
Module 2			10 Hours
A Boolean Type , Choosing Statements to Execute, Nested If Statements , Remembering the Results of a Boolean Expression Evaluation , A Modular Approach to Program Organization, Importing Modules , Defining Your Own Modules, Testing Code Semi automatically Grouping Functions Using Methods: Modules, Classes, and Methods , Calling Methods the Object-Oriented Way, Exploring String Methods, Underscores.			
Module 3			10 Hours
Storing Collections of Data Using Lists: Storing and Accessing Data in Lists, modifying Lists, Operations on Lists, Slicing Lists, Aliasing, List Methods, Working with a List of Lists. Repeating Code Using Loops: Processing Items in a List, Processing Characters in Strings, Looping Over a Range of Numbers, Processing Lists Using Indices, Nesting Loops in Loops, Looping Until a Condition Is Reached, Repetition Based on User Input, Controlling Loops Using Break and Continue .			
Module 4			10 Hours
Files: Kinds of files, Opening a File, Techniques for Reading Files, Files over the Internet, Writing Files, and Writing Algorithms That Use the File-Reading Techniques, Multiline Records. Storing Data Using Other Collection Types: Storing Data Using Sets, Storing Data Using Tuples, Storing Data Using Dictionaries, Inverting a Dictionary, Using the In Operator on Tuples , Sets, and Dictionaries, Comparing Collections.			
Module 5			10 Hours

Collection of New Information Object-Oriented Programming : Understanding a Problem Domain , Function “Instance,” Class Object, and Class Book , Writing a Method in Class Book, Plugging into Python Syntax: More Special Methods ,Creating Graphical User interface: Building a Basic GUI, Models, Views, and Controllers, Customizing the Visual Style Introducing few more Widgets, Object-Oriented GUIs, Keeping the Concepts from Being a GUI Mess.	
---	--

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 20 marks.
- There will be 2 full questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module

Text Books:

1. Practical Programming: An introduction to Computer Science Using Python, second edition, Paul Gries, Jennifer Campbell, Jason Montojo, The Pragmatic Bookshelf, 2013.
2. Learning with Python: How to Think Like a Computer Scientist Paperback – Allen Downey , Jeffrey Elkner, 2015

Reference Books:

1. Introduction to Python for Computational Science and Engineering (A beginner's guide), Hans Fangohr, 2015
2. Exploring Python, Timothy A. Budd, Mc Graw Hill Education, 2009

Discrete Mathematical Structures [As per Choice Based Credit System (CBCS) scheme] SEMESTER – III			
Subject Code	17MCA304	CIE Marks	50
Number of Lecture Hours/Week	04	SEE Marks	50
Tutorial Hours	02		
Total Number of Lecture Hours	50	SEE Hours	03
CREDITS – 04			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Apply concepts of logic and build proof techniques in the problem arising in Computer Science</p> <p>CO2: Apply knowledge of relations and functions to situations arising in Computer Science</p> <p>CO3: Apply counting techniques to different situations arising in Computer Science</p> <p>CO4: Apply concepts of groups and rings to coding theory</p> <p>CO5: Apply concepts of graph theory to Computer Science</p>			
Modules			Teaching Hours
Module 1			10 Hours
Logic and Proof: Propositional logic, Propositional equivalences, Predicates and quantifiers, rules of inference, Introduction to proofs, proof methods and strategy			
Module 2			10 Hours
Relations and Functions : Recap of basic Set Theory, relations and their properties, representing relations, Partial orderings Functions, 1-1, onto, invertible functions, cardinality applications			
Module 3			10 Hours
Counting Techniques: Basics of counting, the pigeonhole principle, permutations and combinations, recurrence relations, generating functions, inclusion-exclusion principles , Applications			
Module 4			10 Hours
Algebraic structure and coding theory: Semigroups, monoids, groups, homomorphism, normal subgroups, congruence relations, Rings, integral domains, fields, coding theory applications			
Module 5			10 Hours
Graph Theory: Graphs and graph models, graph terminology and special types of graphs, isomorphism, Euler and Hamiltonian paths, planar graphs, Shortest path problems, Introduction to trees, spanning trees, Applications			
Question paper pattern:			
· The question paper will have ten questions.			

- Each full question consists of 20 marks.
- There will be 2 full questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module.

Text Book:

1. Discrete Mathematics and its Applications by Kenneth H Rosen, 7th edition, (Indian adaptation by Kamala Krithivasan), Tata McGraw Hill, 2011.

Reference Books:

- 1 Discrete Mathematics with proof: Eric Gornet 2nd Edition Wiley
2. Discrete Mathematics Schaum's Outline series, Seymour Lipschutz, Marc Lorr Lipson, 3rd edition McGrawHill 2009

Software Engineering [As per Choice Based Credit System (CBCS) scheme] SEMESTER – III			
Subject Code	17MCA305	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
CREDITS – 03			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Understand the various software development process models and their suitability along with professional ethics and responsibilities.</p> <p>CO2: Understand and evaluate the requirements for a software system</p> <p>CO3: Apply the principles and techniques of software engineering in the architectural design, detail design, and implementation of software applications</p> <p>CO4: Understand the various aspects of Planning a software project, Risk & Management activities</p>			
Modules			Teaching Hours
Module 1			08 Hours
<p>Software Engineering Ethics Introduction: Professional Software Development Attributes of good software, software engineering diversity, IEEE/ ACM code of software engineering ethics.</p> <p>Software Process & Agile Software Development Software Process models: waterfall, incremental development, Reuse-oriented, Process activities; Coping with change, The rational Unified process. Agile methods, Plan-driven and Agile Development, Extreme Programming, Rapid Application Development.</p>			
Module 2			08 Hours
<p>Requirements Engineering Functional and non-functional requirements, The software requirements document, Requirements specification, Requirements engineering processes, Requirement elicitation and analysis, Requirements validation, Requirements management.</p>			
Module 3			08 Hours
<p>System Modeling, Architectural Design Context models, Interaction models, Structural models, Behavioral models, Model-driven engineering, Software architecture: the role of software architecture, architectural views, component and connector view, Architectural styles for C&C view.</p>			
Module 4			08 Hours
<p>Design and implementation Design: Design concepts, Function oriented design, detailed design, verification</p> <p>Component-based software engineering Components and component model, CBSE process, Component composition.</p>			

Module 5	08 Hours
<p>Planning a software Project Process planning, Effort estimation, Project scheduling and staffing, Software configuration management plan, Quality plan, Risk Management, Project monitoring plan.</p> <p>Software Testing Testing fundamentals, Black-box testing, White-box testing, Testing process</p>	
<p>Question paper pattern:</p> <ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. <p>The students will have to answer 5 full questions, selecting one full question from each module</p>	
<p>Text Books:</p> <ol style="list-style-type: none"> 1. Ian Sommerville : Software Engineering, 9th edition, Person Education Ltd, 2011. 2. Pankaj Jalote: Software Engineering, Wiley India Pvt Ltd, 2010 	
<p>Reference Books:</p> <ol style="list-style-type: none"> 1. Roger S Pressman: Software Engineering-A Practitioners approach, 6th Edition, McGraw-Hill, 2010. 2. Hans Van Vliet: Software Engineering Principles and Practices, 3rd Edition, Wiley-India, 2010 	

Java Programming Laboratory [As per Choice Based Credit System (CBCS) scheme] SEMESTER – III			
Subject Code	17MCA307L	CIE Marks	50
Number of Lecture Hours/Week	03 Hrs Laboratory	SEE Marks	50
		SEE Hours	03
CREDITS – 1.5			
Course Outcomes(CO): At the end of this course, the students will be able to:			
<p>CO1: Understand design, develop, implement, analyze and evaluation/testing. CO2: Understand Java programming language fundamentals and run time environment. CO3: Learn the object oriented concepts and its implementation in Java. CO4: Implement the multithreading and client side programming.</p>			
Laboratory Experiments:			
SECTION A			
1. a. Implement a JAVA Program to demonstrate Constructor Overloading and Method Overloading. b. Develop a JAVA Program to implement Inner class and demonstrate its Access protection.			
2. Implement a program in Java for String handling which performs the following: i) Checks the capacity of StringBuffer objects. ii) Reverses the contents of a string given on console and converts the resultant string in upper case. iii) Reads a string from console and appends it to the resultant string of ii.			
3. a. Implement a JAVA Program to demonstrate Inheritance. b. Simple Program on Java for the implementation of Multiple inheritance using interfaces to calculate the area of a rectangle and triangle.			
4. Develop a JAVA program which has i. A Class called Account that creates account with 500Rs minimum balance, a deposit() method to deposit amount, a withdraw() method to withdraw amount and also throws LessBalanceException if an account holder tries to withdraw money which makes the balance become less than 500Rs. ii. A Class called LessBalanceException which returns the statement that says withdraw amount (Rs) is not valid. iii. A Class which creates 2 accounts, both account deposit money and one account tries to withdraw more money which generates a LessBalanceException take appropriate action for the same.			
5. Implement a JAVA program using Synchronized Threads, which demonstrates Producer Consumer concept.			
6. Implement a JAVA program to implement a Queue using user defined Exception Handling (also make use of throw, throws.).			
7. Complete the following: 1. Create a package named shape. 2. Create some classes in the package representing some common shapes like Square, Triangle, and Circle. 3. Import and compile these classes in other program.d. Exit			
8. Implement a JAVA Program			

a. Create an enumeration Day of Week with seven values SUNDAY through SATURDAY. Add a method is Workday() to the DayofWeek class that returns true if the value on which it is called is MONDAY through FRIDAY. For example, the call DayOfWeek.SUNDAY.isWorkDay () returns false.

9. Implement a JAVA program which has

- i. A Interface class for Stack Operations
- ii. A Class that implements the Stack Interface and creates a fixed length Stack.
- iii. A Class that implements the Stack Interface and creates a Dynamic length Stack.
- iv. A Class that uses both the above Stacks through Interface reference and does the Stack operations that demonstrates the runtime binding.

10. Implement a JAVA program to print a chessboard pattern.

11. Implement a JAVA Program which uses FileInputStream / FileOutPutStream Classes.

12. Implement JAVA programs which demonstrates utilities of LinkedList Class.

13. Implement a JAVA program which uses Datagram Socket for Client Server Communication.

14. Implement a JAVA applet program, which handles keyboard event.

Note: In the examination each student should pick one program from the above list and another program as per examiner choice.

Python Programming Laboratory [As per Choice Based Credit System (CBCS) scheme] SEMESTER –III			
Subject Code	17MCA308L	CIE Marks	50
Number of Lecture Hours/Week	03 Hours Laboratory	SEE Marks	50
		SEE Hours	03
CREDITS – 1.5			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Apply object-oriented programming concepts to develop dynamic interactive Python applications.</p> <p>CO2: Use the procedural statements: assignments, conditional statements, loops, method calls and arrays</p> <p>CO3: Design, code, and test small Python programs with a basic understanding of top-down design.</p> <p>CO4: Learn how to create GUI and solve real-world problem using language idioms, data structures and standard library</p>			
Laboratory Experiments			
1. Develop a program to sum all the elements from n1 to n2 where n1 and n2 are positive integers			
2. Input an array of n numbers and find separately the sum of positive numbers and negative numbers.			
3. Implement a program to search an element using linear search			
4. Implement a program to search an element using binary search.			
5. Implement a program to simulate stack.			
6. Using a stack evaluate an arithmetic expression.			
7. Program to multiply two matrices.			
8. Program to find the roots of a quadratic equation			
9. Program to Insert a number in a sorted array.			
10. Implement a Python Program to check whether the given string is palindrome or not using built in string manipulation methods.			
11. Implement a Python Program to read a word and prints the number of letters, vowels and percentage of vowels in the word using dictionary.			
12. Implement a Python Program to check a given sentence is a pangram or not using function/Module.			
13. Implement a Python Event driven Program for file operations Press 1: to open file in read mode			

2: open the file in write mode

3: current position of the file pointer #4: Reposition the pointer at the beginning 5: exit.

14. Implement an Object oriented Python program to create two Time objects: currentTime, which contains the current time; and breadTime, which contains the amount of time it takes for a bread maker to make bread. Then we'll use addTime to figure out when the bread will be done. Write the printTime function to display the time when the bread will be done by the bread maker.

Note: In the examination each student should pick one program from the above list and another program as per examiner choice.

Intellectual Property Rights [As per Choice Based Credit System (CBCS) scheme] SEMESTER – III			
Subject Code	17MCA3061E	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
Credits – 03			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Identify intellectual property rights, CO2: Understand trademark and copyrights. CO3: Understand patents and Laws relating to patents. CO4: Understand Protection of trade secrets.</p>			
Modules			Teaching Hours
Module 1			08 Hours
Introduction to Intellectual Property Law, Para -Legal Tasks in Intellectual Property Law, Ethical obligations in Para Legal Tasks, Introduction to Cyber Law, Innovations and Inventions, Trade related Intellectual Property Rights.			
Module 2			08 Hours
Introduction to TradeMark(TM), Registration Process , Post registration Procedures , Maintenance, Transfer of Rights , Infringement , Dilution Ownership of TradeMark , TM claims , Trademarks Litigations – International Trade mark Law.			
Module 3			08 Hours
Introduction to Copyrights , Principles of Copyright -The subjects Matter of Copy right , Copy right Ownership, Transfer and duration , Right to prepare Derivative works , Rights of Distribution , Copyright Formalities and Registrations ,Limitations, Copyright disputes and International Copyright Law. Copyright Acts.			
Module 4			08 Hours
Introduction to Patents, filling procedures, The law of patents, patent searches ,Patent ownership and transfer, Patent infringement, International Patent Law			
Module 5			08 Hours
Introduction to Trade Secret, Maintaining Trade Secret , Physical Security , Employee Limitation , Employee confidentiality agreement , Trade Secret Law , Unfair Competition , Trade Secret Litigation , Breach of Contract , Applying State Law			
Question paper pattern:			
<ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. <p>The students will have to answer 5 full questions, selecting one full question from each module</p>			

Text Books:

1. DebiragE.Bouchoux: "Intellectual Property". Cengage learning, New Delhi, 2010
2. M.Ashok Kumar and Mohd.Iqbal Ali: "Intellectual Property Right" Serials Pub,2009
3. Cyber Law. Texts & Cases, South-Western's Special Topics Collections,2010

Reference Books:

1. Prabhuddha Ganguli: ' Intellectual Property Rights' Tata Mc-Graw –Hill, New Delhi,2009
2. J Martin and C Turner "Intellectual Property" CRC Press Richard Stimm " Intellectual Property" Cengage Learning, 2010

Enterprise Resource Planning [As per Choice Based Credit System (CBCS) scheme] SEMESTER – III			
Subject Code	17MCA3062E	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
Credits – 03			
Course Outcome (CO): At the end of this course, the students will be able to:			
CO1: Acquire knowledge and benefits of ERP, Process Re-engineering. CO2: Develop the ERP system. CO3: Understand the different components of ERP CO4: Understand the ERP market and growth of ERP. CO5: Analyze the performance of transaction processing.			
Modules			Teaching Hours
Module 1			08 Hours
Introduction To ERP Overview, Benefits of ERP, Examples of ERP & SAP, Business Process Re-engineering, Data Warehousing, Data Mining, On-line Analytical Processing, Supply Chain Management (SCM).			
Module 2			08 Hours
ERP Implementation Implementation of Sales Life Cycle, Implementation Methodology, Hidden Costs, Vendors, Consultants and Users, Contracts, Project, Management and Monitoring.			
Module 3			08 Hours
Business Modules Business Modules in an ERP Package, Finance, Manufacturing, Human Resource, Plant Maintenance, Materials Management, Quality Management, Sales and Distribution, CRM Overview Package.			
Module 4			08 Hours
ERP – Present And Future ERP & SAP, ERP & EIA, ERP and E-Commerce, ERP and Internet, Future Directions in ERP. ERP Market ERP Market, SAP AG, PeopleSoft ,Baan Company , JD Edwards World Solutions Company, Oracle Corporation, QAD, Case Study.			
Module 5			08 Hours
Transaction Processing Transaction Fundamentals, OLTP Environment, System Design, Query Processing, Purpose, Backup and Recovery, Transaction Processing Modes, Activities, Data Capture, Properties of Transaction and Transaction States. OLAP Overview , Types, Characteristics, Technologies, Applications of OLAP Models, OLAP v/s OLTP.			

Question Paper Pattern:	
<ul style="list-style-type: none">• The question paper will have ten questions.• Each full question consists of 20marks.• There will be 2 full questions (with a maximum of four sub questions) from each module.• Each full question will have sub questions covering all the topics under a module.• The students will have to answer 5 full questions, selecting one full question from each module.	
Text Books:	
<ol style="list-style-type: none">1. Alexis Leon, "ERP Demystified", Tata McGraw Hill, 1999.2. Joseph A. Brady, Ellen F. Monk, Bret J. Wangner, "Concepts in Enterprise Resource Planning", Thomson Learning, 2001.	
Reference Books:	
<ol style="list-style-type: none">1. Vinod Kumar Garg and N.K .Venkata Krishnan, "Enterprise Resource Planning concepts and Planning", Prentice Hall, 1998.2. Jose Antonio Fernandz, " The SAP R /3 Hand book", Tata McGraw Hill, 2009	

Advanced Topics in DBMS			
[As per Choice Based Credit System (CBCS) scheme]			
SEMESTER – III			
Subject Code	17MCA3063E	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
Credits – 03			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO 1: Identify the storage & Indexing of data and also the disks and files data storage</p> <p>CO 2: Analyze the tree structured indexing and hash based indexing</p> <p>CO 3: Illustration of different RAID levels and page formats</p> <p>CO 4: Analyze and understand the query evaluation, external sorting techniques.</p> <p>CO 5: Understand evaluating relational operations.</p>			
Modules			Teaching Hours
Module 1			08 Hours
Overview of Storage and Indexing, Disks and files: Data on external storage; File organizations and Indexing, Index data structures; Comparison of file organizations; Indexes and performance tuning. Memory hierarchy: RAID; Disk space management; Buffer manager: Files of records; Page formats and record format.			
Module 2			08 Hours
<p>Tree Structured Indexing Intuition for tree indexes; Indexed sequential access method; B+trees, Search, Insert, Delete, Duplicates, B+tress in practice</p> <p>Hash-Based Indexing Static hashing, Extendible hashing, Linear hashing, comparisons</p>			
Module 3			08 Hours
<p>Overview of Query Evaluation, External Sorting The system catalog, Introduction to operator evaluation; Algorithm for relational operations; Introduction to query optimization; Alternative plans; A motivating example; what a typical optimizer does. When does a DBMS sort data? A simple two-way merge sort; External merge sort</p>			
Module 4			08 Hours
<p>Evaluating Relational Operators The Selection operation; General selection conditions; The Projection operation; The Join operation; The Set operations; Aggregate operations; The impact of buffering.</p> <p>A Typical Relational Query Optimizer Translating SQL queries in to Relational Algebra; Estimating the cost of a plan; Relational algebra equivalences; Enumeration of alternative plans; Nested sub-queries; other approaches to query optimization.</p>			
Module 5			08Hours
<p>Physical Database Design and Tuning Introduction; Guidelines for index selection ,examples; Clustering and indexing; Indexes that enable index-only plans, Tools to assist in index selection; Overview of database tuning; Choices in tuning the conceptual schema; Choices in tuning queries and views; Impact of Concurrency; DBMS benchmarking.</p>			

Case study and implementation of Commercial database: Mobile databases; Multimedia databases; Geographical Information Systems; Genome data management.	
---	--

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 20 marks.
- There will be 2 full questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module

Text Books:

1. Raghu Ramakrishnan and Johannes Gehrke: Database Management Systems, 3rd Edition, McGraw-Hill, 2003.
2. Elmasri and Navathe: Fundamentals of Database Systems, 5th Edition, Pearson Education, 2007.

Reference Books:

1. Conolly and Begg: Database Systems, 4th Edition, Pearson Education, 2002.

Cyber Security [As per Choice Based Credit System (CBCS) scheme] SEMESTER – III			
Subject Code	17MCA3064E	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
Credits – 03			
Course Outcome (CO): At the end of this course, the students will be able to:			
CO1: Illustrate cyber security concepts and applications			
CO2: Analyze the working of cyber security principles to system design			
CO3: Illustrate appropriate techniques to solve cyber security threats			
CO4: Evaluate and implement cyber security through network security protocols			
Modules		Teaching Hours	
Module 1		08 Hours	
Systems Vulnerability Scanning Overview, Open Port / Service Identification, Banner / Version Check, Traffic Probe, Vulnerability Probe, Vulnerability Examples, OpenVAS, Metasploit. Netcat, Socat, Port and Services tools - Datapipe, Fpipe, WinRelay, Network Reconnaissance – Nmap, THC-Amap and System tools. Network Sniffers and Injection tools – Tcpcap and Windump, Wireshark, Ettercap, Hping Kismet.			
Module 2		08 Hours	
Network Defense tools Firewalls and Packet Filters,: Overview, Protects a Network, Packet Characteristic to Filter, Stateless Vs Stateful Firewalls, Network Address Translation (NAT) and Port Forwarding, Virtual Private Networks, Linux Firewall, Windows Firewall, Snort: Introduction Detection System			
Module 3		08 Hours	
Web Application Tools Scanning for web vulnerabilities tools: Nikto, W3af, HTTP utilities - Curl, OpenSSL and Stunnel, Application Inspection tools – Zed Attack Proxy, Sqlmap. DVWA, Webgoat, Password Cracking and Brute-Force Tools – John the Ripper, LOhtcrack, Pwdump,HTC-Hydra			
Module 4		08 Hours	
Cyber Crime Introduction to Cyber Crime and law's , Types of Cybercrime, Ethical Hacking, Cyberspace and Criminal Behavior, Traditional Problems Associated with Computer Crime, Introduction to Incident Response, Digital Forensics, Computer Language, Network Language, Realms of the Cyber world, Recognizing and Defining Computer Crime, Contemporary Crimes, destruction of evidence/data, Indian IT ACT 2000.			
Module 5		08 Hours	
Cyber Investigation			

Introduction to Cyber Crime Investigation, Firewalls and Packet Filters, password Cracking, Key logger's and Spyware, attaching with Virus and Worms, Trojan and backdoors.	
<p>Question paper pattern:</p> <ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. <p>The students will have to answer 5 full questions, selecting one full question from each module</p>	
<p>Text Books:</p> <ol style="list-style-type: none"> 1. Anti-Hacker Tool Kit (Indian Edition) by Mike Shema, Publication McGraw Hill, 2006 2. Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives by Nina Godbole and SunitBelpure, Publication Wiley, 2009 	
<p>Reference Books:</p> <ol style="list-style-type: none"> 1. Marjie T. Britz - Computer Forensics and Cyber Crime: An Introduction – Pearson, 2013 2. Chwan-Hwa (John) Wu,J. David Irwin - Introduction to Computer Networks and Cybersecurity - CRC Press, 2013 3. Bill Nelson, Amelia Phillips, Christopher Steuart - Guide to Computer Forensics and Investigations -cengage Learning, 2013 	

SEMESTER - IV

Advanced Java Programming [As per Choice Based Credit System (CBCS) scheme] SEMESTER –IV			
Subject Code	17MCA401	CIE Marks	50
Number of Lecture Hours/Week	04	SEE Marks	50
Total Number of Lecture Hours	50	SEE Hours	03
CREDITS – 04			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Learn the concept of Servlet and its life cycle CO2: Comprehend JSP tags and its services CO3: Create packages and interfaces CO4: Build Database connection CO5: Develop Java Server Pages applications using JSP Tags. CO6: Develop Enterprise Java Bean Applications</p>			
Modules			Teaching Hours
Module 1			10 Hours
<p>Servlet Servlet Structure, Servlet packaging, HTML building utilities, Lifecycle, SingleThreadModel interface, Handling Client Request: Form Data, Handling Client Request: HTTP Request Headers. Generating server Response: HTTP Status codes, Generating server Response: HTTP Response Headers, Handling Cookies, Session Tracking</p>			
Module 2			10 Hours
<p>JSP and Controlling the Structure of generated servlets Overview of JSP: JSP Technology, Need of JSP, Benefits of JSP, Advantages of JSP, Basic syntax, Invoking java code with JSP scripting elements, creating Template Text, Invoking java code from JSP, Limiting java code in JSP, using jsp expressions, comparing servlets and jsp, writing scriptlets. For example Using Scriptlets to make parts of jsp conditional, using declarations, declaration example. Controlling the Structure of generated servlets: The JSP page directive, import attribute, session attribute, isElignore attribute, buffer and autoflush attributes, info attribute, errorPage and isErrorPage attributes, isThreadSafe Attribute, extends attribute, language attribute</p>			
Module 3			10 Hours
<p>Annotations and Java Beans Creating Packages, Interfaces, JAR files and Annotations. The core java API package, New java.Lang Sub package, Built-in Annotations with examples. Working with Java Beans. Introspection, Customizers, creating java bean, manifest file, Bean Jar file, new bean, adding controls, Bean properties, Simple properties, Design Pattern events, creating bound properties, Bean Methods, Bean an Icon, Bean info class, Persistence, Java Beans API</p>			
Module 4			10 Hours
<p>JDBC Talking to Database, Immediate Solutions, Essential JDBC program, using</p>			

prepared Statement Object, Interactive SQL tool. JDBC in Action Result sets, Batch updates, Mapping, Basic JDBC data types, Advanced JDBC data types, immediate solutions. Introduction to EJB: The Problem domain, Breakup responsibilities, Code Smart not hard, the Enterprise java bean specification. Components Types.	
Module 5	10 Hours
EJB and Server Side Component Models Server Side Component Types, Session Beans, Message Driven Beans, Entity Beans, The Java Persistence Model. Container services. Dependency Injection, Concurrency, Instance pooling n caching, Transactions, security, Timers, Naming and object stores Interoperability, Life Cycle Callbacks, Interceptors, platform integration. Developing your first EJB. Preparation, Definitions, naming conventions, convention for the Examples, coding the EJB, the contract, the bean Implementation class, out of Container Testing, Integration Testing.	
<p>Question paper pattern:</p> <ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. <p>The students will have to answer 5 full questions, selecting one full question from each module</p>	
<p>Text Books:</p> <ol style="list-style-type: none"> 1. Marty Hall, Larry Brown. Core Servlets and Java Server Pages. Volume 1: Core Technologies. 2nd Edition, 2003. 2. Java 6 Programming Black Book, Dreamtech Press. 2012. 	
<p>Reference Books:</p> <ol style="list-style-type: none"> 1. Michael Sikora, EJB 3 Developer Guide, A practical guide for developers and architects to the Enterprise Java Beans Standard, Shroff Publishers & Distributors PVT LTD. July 2008. 2. Herbert Schildt, Java the Complete Reference, 8th Edition. Comprehensive coverage of the Java Language. Tata McGraw-Hill Edition – 2011. 	

Analysis and Design of Algorithms [As per Choice Based Credit System (CBCS) scheme] SEMESTER –IV			
Subject Code	17MCA402	CIE Marks	50
Number of Lecture Hours/Week	04	SEE Marks	50
Total Number of Lecture Hours	50	SEE Hours	03
CREDITS – 04			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Categorize problems based on their characteristics and practical importance.</p> <p>CO2: Develop Algorithms using iterative/recursive approach</p> <p>CO3: Compute the efficiency of algorithms in terms of asymptotic notations</p> <p>CO4: Design algorithm using an appropriate design paradigm for solving a given problem</p> <p>CO5: Classify problems as P, NP or NP Complete</p> <p>CO6: Implement algorithms using various design strategies and determine their order of growth.</p>			
Modules			Teaching Hours
Module 1			10 Hours
<p>Introduction, Fundamentals of the Analysis of Algorithm Efficiency, Brute Force Notion of Algorithm, Fundamentals of Algorithmic Problem Solving, Important Problem Types, Fundamental data Structures. Analysis Framework, Asymptotic Notations and Basic efficiency classes, Mathematical analysis of Recursive and Non-recursive algorithms.</p>			
Module 2			10 Hours
<p>Brute Force: Selection Sort and Bubble Sort, Sequential Search and String Matching.</p> <p>Divide-and-Conquer: Mergesort, Quicksort, Binary Search, Binary tree Traversals and related properties, Multiplication of large integers, Strassen’s Matrix Multiplication.</p>			
Module 3			10 Hours
<p>Decrease-and-Conquer: Insertion Sort, Depth First and Breadth First Search, Topological sorting, Algorithms for Generating Combinatorial Objects: generating permutations.</p> <p>Space and Time Tradeoffs: Sorting by Counting, Input Enhancement in String Matching, Hashing</p>			
Module 4			10 Hours
<p>Dynamic Programming: Computing a binomial coefficient, Warshall’s and Floyd’s Algorithms, The Knapsack Problem and Memory Functions</p> <p>Greedy Technique: Prim’s Algorithm, Kruskal’s Algorithm, Dijkstra’s Algorithm, Huffmann Trees.</p>			
Module 5			10 Hours
<p>Limitations of Algorithm Power: Lower-Bound Arguments, Decision Trees, P, NP and NP-Complete Problems.</p> <p>Coping with Limitations of Algorithm Power: Backtracking: n-Queens problem, Hamiltonian Circuit Problem, Subset – Sum Problem. Branch-and-Bound: Assignment Problem, Knapsack Problem, Traveling</p>			

Salesperson Problem.

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 20 marks.
- There will be 2 full questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module

Text Books:

1. Anany Levitin: Introduction to the Design and Analysis of Algorithms, Pearson Education, 2nd Edition, 2008

Reference Books:

1. Cormen T.H., Leiserson C.E., and Rivest R.L.: Introduction to Algorithms, PHI 1998.
2. Horowitz E., Sahani S., Rajasekharan S.: Computer Algorithms, Galgotia Publication 2001.

Software Testing And Practices
[As per Choice Based Credit System (CBCS) scheme]
SEMESTER – IV

Subject Code	17MCA403	CIE Marks	50
Number of Lecture Hours/Week	04	SEE Marks	50
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course Outcome (CO): At the end of this course, the students will be able to:

CO1: Acquire knowledge of basic principles and knowledge of software testing and debugging and test cases.

CO2: Understand the perceptions on testing like levels of testing, generalized pseudo code and with related examples

CO3: Understand the various types of testing.

CO4: Understand and analyze the difference between functional testing and structural testing.

CO5: Analyze the performance of fault based testing, planning and Monitoring the process, Documentation testing

Modules	Teaching Hours
Module 1	10 Hours
<p>Basics of Software Testing, Basic Principles, Test case selection and Adequacy Humans, Errors and Testing, Software Quality; Requirements, Behavior and Correctness, Correctness Vs Reliability; Testing and Debugging; Test Metrics; Software and Hardware Testing; Testing and Verification; Defect Management; Execution History; Test Generation Strategies; Static Testing. Sensitivity, Redundancy, Restriction, Partition, Visibility and Feedback, Test Specification and cases, Adequacy Criteria, Comparing Criteria</p>	
Module 2	10 Hours
<p>A perspective on Testing, Examples Basic definitions, Test cases, Insights from a Venn diagram, Identifying test cases, Level of testing, Examples: Generalized pseudo code, The triangle problem, the Next Date function, The commission problem, The SATM (Simple Automation Teller Machine) problem, The currency converter, Saturn windshield wiper</p>	
Module 3	10 Hours
<p>Boundary value testing, Equivalence class testing, Decision table based testing Boundary value analysis, Robustness testing, Worst-case testing, special value testing, Examples, Random testing, Equivalence classes, Equivalence test cases for triangle problem, Next Date function and commission problem, Guidelines and observations, Decision tables, Test cases for triangle problem</p>	
Module 4	10 Hours

<p>Path Testing, Data flow testing, Levels of Testing, Integration Testing DD Paths, Test coverage metrics, Basis path testing, guidelines and observations, Definition Use testing, Slice based testing, Guidelines and observations. Traditional view of testing levels, Alternative life cycle models, the SATM systems, separating integration and system testing.</p>	
<p>Module 5</p>	<p>10 Hours</p>
<p>Fault Based Testing, Planning and Monitoring the Process, Documenting Analysis Assumptions in fault-based testing, Mutation Analysis, Fault-based Adequacy Criteria; Variations on mutation Analysis; From Test case specification to Test Cases, Scaffolding, Generic vs. specific Scaffolding, Test Oracles, Self checks as oracles, Capture and Replay. Quality and Process, Test and Analysis strategies and plans, Risk Planning, Monitoring the Process, Improving the process, The quality team, Organizing documents, Test strategy document, Analysis and test plan, Test design specifications documents, Test and analysis reports.</p>	
<p>Question paper pattern:</p> <ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. <p>The students will have to answer 5 full questions, selecting one full question from each module</p>	
<p>Text Books:</p> <ol style="list-style-type: none"> 1. Adithya P.Mathur “ Foundations of Software Testing – Fundamental Algorithms and Techniques”, Pearson Education India, 2011 2. Mauro Pezze, Michael Young, Software testing and Analysis- Process, Principles and Techniques, Wiley India, 2012 3. Paul C Jorgensen, “Software Testing A Craftsman's Approach”, Aueredach publications, 3rd edition, 2011. 	
<p>Reference Books:</p> <ol style="list-style-type: none"> 1. Kshirasagara Naik, Priyadarshi Tripathy: Software Testing and Quality Assurance, Wiley India 2012 2. M.G.Limaye: Software Testing-Principels, Techniques and Tools – McGraw Hill, 2009 	

Principles of User Interface Design [As per Choice Based Credit System (CBCS) scheme] SEMESTER –IV			
Subject Code	17MCA404	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
CREDITS – 03			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Familiarize the new technologies that provide interactive devices and interfaces.</p> <p>CO2: Develop the processes and evaluate UID.</p> <p>CO3: Understand Direct Manipulation and Virtual Environment</p> <p>CO4: Discuss the command, natural languages and issues in design for maintaining QoS</p> <p>CO5: Persuade user documentations and information search.</p>			
Modules			Teaching Hours
Module 1			8 Hours
<p>Introduction Usability of Interactive Systems: Introduction, Usability Goals and Measures, Usability Motivation, Universal Usability, Goals for our profession. Guideline, principles, and theories: Introduction, Guidelines, principles, Theories.</p>			
Module 2			8 Hours
<p>Development Processes Managing Design Processes: Introduction, Organizational Design to support Usability, The Four Pillars of Design, Development methodologies: Ethnographic Observation, Participatory Design, Scenario Development, Social Impact statement for Early Design Review, Legal Issues.</p> <p>Evaluating Interface Design Introduction, Expert Reviews, Usability Testing and Laboratories, Survey Instruments, Acceptance tests, Evaluation during Active Use, Controlled Psychologically Oriented Experiments</p>			
Module 3			8 Hours
<p>Direct Manipulation and Virtual Environments: Introduction, Examples of Direct Manipulation, Discussion of direct manipulation, 3D Interfaces, Tele-operation, Virtual and Augmented Reality Menu Selection, Form Filling and Dialog Boxes: Introduction, Task-Related Menu Organization, Single Menus, Combination of Multiple Menus, Content Organization, Fast Movement Through Menus, Data Entry With Menus, Form Filling, Dialog Boxes and Alternatives, Audio Menus and Menus for Small Displays</p>			
Module 4			8 Hours
<p>Command and Natural Languages Introduction, Command-organization functionality strategies and structure, Naming and Abbreviations, Natural Language in computing. Interaction Devices: Introduction, Keyboards and Keypads, Pointing Devices, Speech and Auditory interfaces, Displays-Small and Large</p> <p>Design Issues Quality of Service: Introduction, Models of Response-Time Impacts,</p>			

<p>Expectations and Attitudes, User Productivity, Variability in Response time, Frustrating Experiences Balancing Function and Fashion: Introduction, Error Messages, Non anthropomorphic Design, Display design, web page design, Window Design, Color</p>	
<p>Module 5</p>	<p>8 Hours</p>
<p>User Documentation and Online Help : Introduction, Online versus paper documentation, Reading from paper versus Displays, Shaping the content of the Manuals, Accessing the Documentation, Online Tutorials and animated demonstrations, Online Communities for User Assistance The Development Process.</p> <p>Information Search and Visualization Introduction, Search in Textual Documents and Database Querying, Multimedia document searches, Advanced filtering and Search Interfaces, Information Visualization : Introduction, Datatype by task taxonomy, Challenges for information visualization.</p>	
<p>Question paper pattern:</p> <ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. <p>The students will have to answer 5 full questions, selecting one full question from each module</p>	
<p>Text Books:</p> <ol style="list-style-type: none"> 1. Ben Shneiderman, Plaisant, Cohen, Jacobs: Designing the User Interface, 5th Edition, Pearson, Education, 2010. 2. Alan Dix, Janet Finalay, Gregory D Abiwdm Russel Bealel: Human-Computer Interaction, III Edition, Pearson , Education, 2008. 	
<p>Reference Books:</p> <ol style="list-style-type: none"> 1 Eberts: User Interface Design, Prentice Hall, 1994 2 Wilber O Galitz: The Essential Guide to User Interface Design- An Introduction to GUI Design, Principles and Techniques, Wiley-Dreamtech India Pvt Ltd, 2011 	

Advanced Java Programming Laboratory
[As per Choice Based Credit System (CBCS) scheme]
SEMESTER IV

Subject Code	17MCA407L	CIE Marks	50
Number of Lecture Hours/Week	03 Hrs Laboratory	SEE Marks	50
		SEE Hours	03
CREDITS – 1.5			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>Co1: Design HTML pages to demonstrate Java Servlets, JSP, Bean and EJB programs.</p> <p>Co2: Implement Dynamic HTML using Servlet and demonstration of services methods, auto web page refresh, Session tracking using cookie and Http Session in Servlet.</p> <p>Co3: Learn the fundamental of connecting to the database</p> <p>Co4: Demonstrate JSP (page attributes, action tags and all basic tags) and types of EJB application.</p>			
Laboratory Experiments:			
1. JAVA Servlet Program to implement a dynamic HTML using Servlet (user name and Password should be accepted using HTML and displayed using a Servlet).			
2. JAVA Servlet Program to Auto Web Page Refresh (Consider a webpage which is displaying Date and time or stock market status. For all such type of pages, you would need to refresh your web page regularly; Java Servlet makes this job easy by providing refresh automatically after a given interval).			
3. Implement a JAVA Servlet Program to implement and demonstrate get() and Post methods(Using HTTP Servlet Class).The script need not include any error checking.			
4. Implement a JAVA Servlet Program using cookies to remember user preferences.			
5. Implement a JAVA Servlet program to track HttpSession by accepting user name and password using HTML and display the profile page on successful login.			
6. Implement a JAVA JSP Program which uses jsp:include and jsp:forward action to display a Webpage.			
7. Implement a JAVA JSP Program which uses <jsp:plugin> tag to run a applet			
8. Implement a JAVA JSP Program to get student information through a HTML and create a JAVA Bean class, populate Bean and display the same information through another JSP			
9. Implement a JSP program to implement all the attributes of page directive tag			
10. Implement a JAVA Program to insert data into Student DATA BASE and retrieve info based on particular queries (For example update, delete, search etc...).			
11. An EJB application that demonstrates Session Bean (with appropriate business			
12. An EJB application that demonstrates persistence (with appropriate business logic).			

Note: In the examination each student should pick one program from the above list and another program as per examiner choice.

Software Testing Laboratory [As per Choice Based Credit System (CBCS) scheme] SEMESTER – IV			
Subject Code	17MCA408L	CIE Marks	50
Number of Lecture Hours/Week	03Hrs Laboratory	SEE Marks	50
		SEE Hours	03
CREDITS – 1.5			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Analyze the performance of fault based testing, planning and Monitoring the process, Documentation testing</p> <p>CO2: Experience on software testing projects using software testing tools.</p> <p>CO3: Understand the process to be followed in software development life cycle.</p> <p>CO4: Practical solutions to the problems</p> <p>CO5: Formulate and analyze a problem.</p>			
Laboratory Experiments:			
1.	Design and develop a program in a language of your choice to solve the triangle problem defined as follows: Accept three integers which are supposed to be the three sides of a triangle and determine if the three values represent an equilateral triangle, isosceles triangle, scalene triangle, or they do not form a triangle at all. Derive test cases for your program based on decision-table approach, the test cases and discuss the results.		
2.	Design and develop a program in a language of your choice to solve the triangle problem defined as follows: Accept three integers which are supposed to be the three sides of a triangle and determine if the three values represent an equilateral triangle, isosceles triangle, scalene triangle, or they do not form a triangle at all. Assume that the upper limit for the size of any side is 10. Derive test cases for your program based on boundary-value analysis, execute the test cases and discuss the results.		
3.	Design and develop a program in a language of your choice to solve the triangle problem defined as follows: Accept three integers which are supposed to be the three sides of a triangle and determine if the three values represent an equilateral triangle, isosceles triangle, scalene triangle, or they do not form a triangle at all. Assume that the upper limit for the size of any side is 10. Derive test cases for your program based on equivalence class partitioning, execute the test cases and discuss the results.		
4.	Design, develop, code and run the program in any suitable language to solve the commission problem. Analyze it from the perspective of dataflow testing, derive different test cases, execute these test cases and discuss the test results.		
5.	Design, develop, code and run the program in any suitable language to solve the commission problem. Analyze it from the perspective of boundary value testing, derive different test cases, execute these test cases and discuss the test results.		
6.	Design, develop, code and run the program in any suitable language to solve the commission problem. Analyze it from the perspective of equivalence class testing, derive different test cases, execute these test cases and discuss the test results.		

Note: In the examination each student should pick one program from the above list and another program as per examiner choice.

Technical Seminar

17MCA409S Seminar

- Students should present the seminar on cutting edge/emerging/state of the art technologies in the field of Web Technology.
- Duration of the seminar should be approximately 45 minutes.
- Student should submit the write up on seminar topic containing at least 10 pages

Report format:

- 1. Cover Page**
- 2. Acknowledgement**
- 3. Abstract**

Table of Contents:

Sl. No	Description	Page No.
1.	Introduction	
2.	Literature Survey	
3	Topic Specific Description	
4	Conclusions	
5	Bibliography	

Rules:

- 8) Put page borders
- 9) Main Heading font size 17(Times New Roman)
- 10) Subheading : font size 14(Times New Roman)
- 11) Text: font size 12(Times New Roman)
- 12) Figures should have Fig numbers at the bottom of the figure e.g. Fig 3.2 is second figure in 3rd chapter
- 13) Tables should have table numbers at the top example Table 5.1 i.e, 1st table in 5th chapter
- 14) Paragraph spacing 1.5

Advanced Computer Networks [As per Choice Based Credit System (CBCS) scheme] SEMESTER –IV			
Subject Code	17MCA4051E	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
CREDITS – 03			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Understand the terminology and concepts of TCP-IP reference model.</p> <p>CO2: Use NS3 simulator.</p> <p>CO3: Analyze the performance of TCP/IP on wireless and mobile networks.</p> <p>CO4: Evaluate the performance of TCP/IP over Asymmetric networks.</p> <p>CO5: Understand TCP/IP Standards and flavours</p>			
Modules			Teaching Hours
Module 1			8 Hours
<p>Introduction to Computer Networks History of TCP/IP, TCP Applications and Services, Performance Study of TCP/IP, Learning of TCP Performance? TCP, TCP Services , Header Format, Encapsulation in IP, Acknowledgment Mechanism, Retransmission Mechanism, Connection Establishment and Termination, Control and Sliding Window, Congestion Control ,UDP ,UDP Services, Header Format ,Encapsulation in IP,IP Services, Fragmentation and Reassembly , Header Format and IP Version 6, Reasons for Network Measurement, Measurement Tasks, Classification of Measurement Tools, Popular Measurement Tools and Their Applications, Tcpdump, Tcpstat, Ttcp& Netperf. Distributed Benchmark System.</p>			
Module 2			8 Hours
<p>NS3 NS3 Components, Models, Node basics & packets, Net devices and Channels, Network Elements, Routing Protocols, Traffic Generator, Main Program Structure, Installation steps for NS-3, Steps in writing scripts, Implementation and execution of Routing protocols.</p>			
Module 3			8 Hours
<p>TCP/IP Performance over Wireless Networks & TCP/IP Performance over Mobile Networks Wireless Networks: Generic Characteristics, Wireless Local Area Networks and Cellular Communications Networks. TCP Performance Issues over Wireless Links, Inappropriate Reduction of Congestion Window, Throughput Loss in WLANs and Throughput Loss in Cellular Communication Systems. Improving TCP Performance over Wireless Links: Splitting TCP Connections, Snooping TCP at Base Stations, Notifying the Causes of Packet Loss, Adding Selective Acknowledgments to TCP and Comparison of Enhancement Schemes. Wireless System Evolution and TCP/IP: Trends in Cellular Communication Systems, Trends in Wireless LAN Systems, TCP/IP over Heterogeneous Wireless systems. Cellular and Ad Hoc Networks: TCP Performance in Cellular Networks, Mobile IP, Impact of Mobility on</p>			

TCP Performance, Approaches to Improve TCP Performance, TCP Performance in Ad Hoc Networks, Dynamic Source Routing, Impact of Mobility on TCP Performance, Approaches to Improve TCP Performance. Evolution of Optical Networks, IP over DWDM, Multiprotocol Label Switching, Multiprotocol Lambda Switching,	
Module 4	8 Hours
<p>TCP/IP Performance over Asymmetric Networks</p> <p>TCP: Long Feedback Loop, Link Impairment, Bandwidth-Delay Product, Bandwidth Asymmetry, Variable Delays, LEO Handoff Spectral Congestion, Security. TCP for Transactions, Window Scaling, Large Initial Window, Byte Counting, Delayed ACKs after Slow Start, Explicit Congestion Notification, Multiple Connections, Pacing TCP Segments, TCP/IP Header Notification, Multiple Connections, Pacing TCP Segments, TCP/IP Header Compression, and Security Issues Conclusions for TCP Enhancements. Advanced Enhancements and New Versions of TCP: Quick-Start TCP, High Speed TCP, TCP Peach, Explicit Transport Error Notification TCP Westwood and XCP. New Transport Protocols for Types of Network Asymmetry: Bandwidth Asymmetry, Media-Access Asymmetry, Loss Rate. Asymmetry Impact of Asymmetry on TCP performance: Bandwidth Asymmetry, Media Access Asymmetry. Improving TCP Performance over Asymmetric Networks: Uplink Bandwidth Management Handling Infrequent ACK. Experimental Evaluation Of Performance Improvement Techniques Experiments with Bandwidth Asymmetry, Experiments with Media Access Asymmetry</p>	
Module 5	8 Hours
<p>TCP/IP Standards and Flavors</p> <p>Types of Network Asymmetry: Bandwidth Asymmetry, Media-Access Asymmetry, Loss Rate. Asymmetry Impact of Asymmetry on TCP Performance: Bandwidth Asymmetry, Media Access Asymmetry. Improving TCP Performance over Asymmetric Networks: KS: Uplink Bandwidth Management Handling Infrequent ACK. Experimental Evaluation of Performance Improvement Techniques Experiments with Bandwidth Asymmetry, Experiments with Media Access Asymmetry. Duplicate Acknowledgments and Fast Retransmit, Fast Recovery and TCP Reno, TCP New Reno, TCP with Selective Acknowledgments, Forward Acknowledgments, TCP Vegas, Overview of Other Features and Options and Performance Comparison of TCP Flavors.</p>	
<p>Question paper pattern:</p> <ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. <p>The students will have to answer 5 full questions, selecting one full question from each module</p>	
<p>Text Book:</p> <ol style="list-style-type: none"> 1. High Performance TCP/IP: Networking Concepts, Issues, and Solutions, Mahbub Hassan and Raj Jain, IST Edition, 2009. 	
<p>Reference Book:</p> <ol style="list-style-type: none"> 1. TCP/IP Illustrated (Volume I, Volume II and Volume III), W. Richard Stevens, Addison-Wesley, 2013. 	

Data Warehousing and Data Mining [As per Choice Based Credit System (CBCS) scheme] SEMESTER – IV			
Subject Code	17MCA4052E	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
CREDITS – 03			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Learn the concept of Data warehousing and OLAP.</p> <p>CO2: Understand storage and retrieval technique of data from DATA CUBE.</p> <p>CO3: Analyze different types of data and different preprocessing techniques.</p> <p>CO4: Evaluate various Association algorithms and its applications.</p> <p>CO5: Apply different Classification technique.</p> <p>CO6: Evaluate different type's classifiers.</p> <p>CO7: Analyze different clustering techniques and their applications</p>			
Modules			Teaching Hours
Module 1			8 Hours
<p>Data warehousing and OLAP Data Warehouse basic concepts, Data Warehouse Modeling, Data Cube and OLAP : Characteristics of OLAP systems, Multidimensional view and Data cube, Data Cube Implementations, Data Cube operations, Implementation of OLAP and overview on OLAP Softwares.</p>			
Module 2			8 Hours
<p>Data Mining and its Applications Introduction, What is Data Mining, Motivating Challenges, Data Mining Tasks, Which technologies are used, which kinds of applications are targeted by Data Mining Which technologies are used, which kinds of applications are targeted by Data Mining, Types of Data, Data Mining Applications, Data Preprocessing.</p>			
Module 3			8 Hours
<p>Association Analysis: Basic Concepts and Algorithms Frequent Item set Generation, Rule Generation, Compact Representation of Frequent Item sets, Alternative methods for generating Frequent Item sets, FP Growth Algorithm, Evaluation of Association Patterns.</p>			
Module 4			8 Hours
<p>Classification : Methods, Improving accuracy of classification Basics, General approach to solve classification problem, Decision Trees, Rule Based Classifiers, Nearest Neighbor Classifiers. Bayesian Classifiers, Estimating Predictive accuracy of classification methods, Improving accuracy of classification methods, Evaluation criteria for classification methods, Multiclass Problem.</p>			
Module 5			8 Hours

Clustering Techniques and Outlier Analysis

Overview, Features of cluster analysis, Types of Data and Computing Distance, Types of Cluster Analysis Methods, Partitional Methods, Hierarchical Methods, Density Based Methods, Quality and Validity of Cluster Analysis, Outlier detection methods, Statistical Approaches, Clustering based applications, Classification based approached.

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 20 marks.
- There will be 2 full questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module

Text Books:

1. Jiawei Han and Micheline Kamber: Data Mining - Concepts and Techniques, 2nd Edition, Morgan Kaufmann Publisher, 2006.
2. Pang-Ning Tan, Michael Steinbach, Vipin Kumar: Introduction to Data Mining, Addison-Wesley, 2005.

Reference Books:

1. Arun K Pujari: Data Mining Techniques University Press, 2nd Edition, 2009.
2. G. K. Gupta: Introduction to Data Mining with Case Studies, 3rd Edition, PHI, New Delhi, 2009.
3. Alex Berson and Stephen J. Smith: Data Warehousing, Data Mining, and OLAP Computing McGrawHill Publisher, 1997.

Software Architecture [As per Choice Based Credit System (CBCS) scheme] SEMESTER –IV			
Subject Code	17MCA4053E	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
CREDITS – 03			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Acquire knowledge of Architectural patterns. CO2: Understand Model quality attributes CO3: Use different methods of Requirements gathering. CO4: Understand different design strategies CO5: Understand design patterns</p>			
Modules			Teaching Hours
Module 1			08 Hours
<p>Introduction and Context of Software Architecture What software architecture is and what it is not; Architectural Structures and views; Architectural patterns; What makes a “good” architecture? Why is software important? Technical Context; Project life-cycle context; Business context; Professional context; Stake holders; How is Architecture influenced? What Do Architecture influence?</p>			
Module 2			08 Hours
<p>Understanding Quality Attributes Architecture & Requirements; Functionality; quality attribute considerations; Specifying and achieving Quality attribute requirements; Guiding quality Specifying and achieving Quality attribute requirements; Guiding quality decisions; Availability; Interoperability; Modifiability; Performance; Security; Testability; Usability.</p>			
Module 3			08 Hours
<p>Quality Attribute modeling and Analysis, Architecture and requirements Gathering Modeling Architecture to enable quality attribute analysis; Quality attribute check lists; Through experiments and Back-of-the envelope analysis; Experiments; Simulations and prototypes; Analysis at different stages of the life cycle. Architecture and requirements Gathering ASRs from requirements documents; ASRs by interviewing stake holders; ASRs by understanding the business; capturing ASRs in a utility tree; Typing the methods together</p>			
Module 4			08 Hours
<p>Designing an Architecture, Documenting Software Architecture and Architecture, Implementation & Testing Design strategy; the attribute driven design methods; the steps of ADD, Uses and Audiences for architecture documentation; Notations, View and Behavior; Documentation and quality attributes, Architecture and implementation; Architecture and testing.</p>			
Module 5			08 Hours
Architectural Patterns			

Introduction to patterns; From Mud to structure; Layers; Pipes and filters; Blackboard; Distributed systems; Brocker; Interactive systems; Model-view- control; Presentation- abstraction- control; Adaptable systems; Microkernel	
--	--

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 20 marks.
- There will be 2 full questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module

Text Books:

1. Len Bass, Paul Clements, Rick Kazman: Software Architecture in Practice, 3rd Edition, Pearson Education, 2013.
2. Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal: Pattern- Oriented Software Architecture, A System of Patterns, Volume 1, John Wiley and Sons, 2012.

Reference Books:

1. Richard N.Taylor, Nenad Medvidovic and Eric M. Dashofy: Software Architecture: Foundations, Theory, and Practice, Wiley- India 2012
2. Mary Shaw and David Garlan: Software Architecture - Perspectives on an Emerging Discipline, Prentice Hall of India, 2007.

Cryptography And Network Security [As per Choice Based Credit System (CBCS) scheme] SEMESTER – IV			
Subject Code	17MCA4054E	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
Credits – 03			
Course Outcome (CO): At the end of this course, the students will be able to:			
CO1: Identify common network security vulnerabilities/attacks; explain the foundations of Cryptography and network security;			
CO2: Understand Encryption and decryption of messages using block ciphers			
CO3: Demonstrate detailed knowledge of the role of encryption to protect data.			
CO4: Analyze Network Security Practice And System Security			
Modules		Teaching Hours	
Module 1		08 Hours	
Introduction to Cryptography Introduction: OSI Security Architecture, Security Attacks, Security Services, Security Mechanism, model for Network Security. Classical Encryption Technique: Symmetric Cipher Model, Substitution Techniques, Transposition Techniques.			
Module 2		08 Hours	
Data Encryption and advanced encryption techniques Block Ciphers, Data Encryption Standard and Advanced Encryption Standard Block Cipher Principles, The Data Encryption Standard, Block Cipher Design Principles and Modes of operation, Evaluation Criteria for AES, AES Cipher-Encryption and Decryption, Data Structure, Encryption Round. Public Key Cryptography and Key Management Principles of Public Key Cryptosystem, RSA algorithm, Key management, Daffier Hellman Key exchange.			
Module 3		08 Hours	
CRYPTOGRAPHY techniques Message Authentication and Hash Function: Authentication Requirement, Authentication Functions, Message Authentication Code, Hash Functions, Digital Signatures, Digital Signature Standard. Authentication Applications: Kerberos, X.509 Authentication Service			
Module 4		08 Hours	
E-MAIL AND IP SECURITY Electronic Mail Security: Pretty Good Privacy (PGP), S/MIME IP Security: IP Security Overview;IP Security Architecture; Authentication Header; Encapsulating Security Payload; Combining Security Associations; Key Management.			
Module 5		08 Hours	

WEB AND SYSTEM SECURITY

Web Security: Web security Considerations; Secure Socket layer (SSL) and Transport layer Security (TLS); Secure Electronic Transaction (SET).

System Security: Intruders, Intrusion Detection, Firewall Design Principles- Characteristics, Types of Firewall and Firewall Configuration.

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 20 marks.
- There will be 2 full questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module

Text Books:

1. William Stallings, "Cryptography and Network Security – Principles and Practices", 4th Edition, Pearson Education, 2009.
2. Atul Kahate, "Cryptography and Network Security" 2nd Edition TMH, 2013.

Reference Book:

1. Behrouz A. Forouzan and Debdeep Mukhopadhyay: "Cryptography and Network Security", 2nd Edition, Tata McGraw-Hill, 2010.

Mobile Technologies [As per Choice Based Credit System (CBCS) scheme] SEMESTER – IV			
Subject Code	17MCA4061E	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
CREDITS – 03			
Course outcomes: After studying this course, students will be able to:			
<p>CO1: Acquire knowledge of classification of devices, Limitations of devices, interfaces, protocol and system Architecture.</p> <p>CO2: Understanding standards like GSM, NB-IoT, LTE.</p> <p>CO3: Understand performance of mobile Network and Transport layer and the HLR System.</p> <p>CO4: Analyze Data Dissemination and Broadcasting Systems.</p> <p>CO5: Understand Data Synchronization in Mobile Computing Systems.</p>			
Modules			Teaching Hours
Module 1			08 Hours
Mobile Devices and Systems, Architectures Mobile phones, Digital Music Players, Handheld Pocket Computers, Handheld Devices, Operating Systems, Smart Systems, Limitations of Mobile Devices, Automotive Systems, Radio Interfaces, Protocols, Localization, Calling, Handover.			
Module 2			08 Hours
<p>GSM, NBIOT and LTE GSM frequency bands, GSM PLMN, GSM PLMN Services, GSM Subsystems, GSM interfaces, The radio interface (MS to BSC). NBIoT introduction, benefits, deployments, applications, architecture, LPWA Technologies and examples. LTE Overview and benefits, motivation for LTE, Elements of LTE system, LTE network elements, LTE technologies, LTE and 4G, LTE services and future.</p>			
Module 3			08 Hours
<p>Mobile IP Network Layer Mobile Transport Layer and HLR System Packet Delivery and Handover Management, Registration, Tunneling and Encapsulation, Route Optimization, Dynamic Host Configuration Protocol Indirect TCP, Snooping TCP, Mobile TCP, Other Methods of TCP – layer Transmission for Mobile Networks, Home Location Register (HLR) Systems HLR and VLR overview, its functions, HLR basic parameters, Location identity, Architecture and technologies.</p>			
Module 4			08 Hours
Data Dissemination and Broadcasting Systems Communication Asymmetry, Classification of Data – Delivery Mechanisms, Data Dissemination Broadcast Models, Selective Tuning and Indexing Techniques, Digital Audio Broadcasting. Digital video Broadcasting.			
Module 5			08 Hours
Data Synchronization in Mobile Computing Systems Synchronization, Synchronization Protocols, SyncML – Synchronization			

<p>Language for Mobile Computing. Mobile Devices, Server and Management, Wireless LAN, Mobile Internet Connectivity and Personal Area Network Mobile agent, Application Server, Gateways, Portals, Service Discovery, Device Management, Mobile File Systems. Wireless LAN (WiFi) Architecture and Protocol Layers, WAP 1.1 and WAP 2.0 Architectures.</p>	
<p>Question paper pattern:</p> <ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. <p>The students will have to answer 5 full questions, selecting one full question from each module.</p>	
<p>Text Books:</p> <ol style="list-style-type: none"> 1. Raj Kamal: Mobile Computing, Oxford University Press, 2007 2. AsokeTalkukder, Roopa R Yavagal: Mobile Computing – Technology, Applications and Service Creation, Tata McGraw Hill, 2005. 	
<p>Reference Books:</p> <ol style="list-style-type: none"> 1. Reza B’Far: Mobile Computing Principles – Designing and Developing Mobile Applications with UML and XML, 5th Edition, Cambridge University press, 2006. 2. Uwe Hansmann, LothatMerk, Martin S Nicklous and Thomas Stober: Principles of Mobile Computing, 2nd Edition, Springer International Edition, 2003. 	

<p align="center">Big Data Analytics [As per Choice Based Credit System (CBCS) scheme] SEMESTER – IV</p>			
Subject Code	17MCA4062E	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
Credits – 03			
<p>Course Outcome (CO): At the end of this course, the students will be able to:</p> <p>Co1: Design algorithms by employing Map Reduce technique for solving Big Data problems Co2: Design algorithms for Big Data by deciding on the apt Features set Co3: Design algorithms for handling peta bytes of datasets Co4: Design algorithms and propose solutions for Big Data by optimizing main memory consumption Co5: Design solutions for problems in Big Data by suggesting appropriate clustering techniques</p>			
Modules			Teaching Hours
Module 1			08 Hours
<p>Big Data and Analytics Example Applications, Basic Nomenclature, Analysis Process Model, Analytical Model Requirements , types of Data Sources, Types of data elements, data explorations, exploratory statistical analysis, missing values, outlier detection and Treatment, standardizing data labels, categorization</p>			
Module 2			08 Hours
<p>Big Data Technology Hadoop’s Parallel World – Data discovery – Open source technology for Big Data Analytics – cloud and Big Data –Predictive Analytics – Mobile Business Intelligence and Big Data – Crowd Sourcing Analytics</p>			
Module 3			08 Hours
<p>Meet Hadoop Data Storage and Analysis, Comparison with Other Systems, RDBMS,Grid Computing , Cloud Computing, A Brief History of Hadoop, Apache- Hadoop and the Hadoop Ecosystem.</p>			
Module 4			08 Hours
<p>The Hadoop Distributed File system The Design of HDFS, HDFS Concepts, Blocks, Namenodes and Datanodes, HDFS Federation, Hadoop File systems Interfaces ,The Java Interface, Reading Data from a Hadoop URL, Reading Data, Writing Data, Querying the File system, Deleting Data, Coherency Model, Parallel Copying with distcp, Hadoop Archives.</p>			
Module 5			08 Hours
<p>Map Reduce A Weather Dataset , Analyzing the Data with Unix Tools, Analyzing the Data with Hadoop, Java MapReduce, Scaling Out, Running a Distributed MapReduce Job, Hadoop Streaming, Hadoop Pipes, Compiling and</p>			

Running, Developing a MapReduce Application,

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 20 marks.
- There will be 2 full questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Bart Baesens, “ Analytics in a Big Data World : The Essential Guide to Data Science and its Applications” Wiley, 2010
2. Michael Minelli, Michele Chambers, “Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today’s Businesses”, 1st Edition, Michael Minelli, Michele Chambers, Ambiga Dhiraj, Wiley CIO Series, 2013.
3. Tom White, “Hadoop: The Definitive Guide”, 3rd Edition, O’reilly, 2012

Reference Books:

1. Boris lublinsky, Kevin t. Smith, Alexey Yakubovich, “Professional Hadoop Solutions”, Wiley, 2015.
2. Chris Eaton, Dirk deroos et al. , “Understanding Big data ”, McGraw Hill, 2012.
3. Vignesh Prajapati, “Big Data Analytics with R and Haoop”, Packet Publishing 2013.

Software Quality Management			
[As per Choice Based Credit System (CBCS) scheme]			
SEMESTER – IV			
Subject Code	17MCA4063E	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
CREDITS – 03			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>Co1: Acquire knowledge of Software Quality through Hierarchical models Co2: Improve the quality of software by SQA plan and Reviews and Audits Co3: Analyze Quality control through usage of CASE tools. Co4: Understand different quality standards</p>			
Modules			Teaching Hours
Module 1			8 Hours
Introduction To Software Quality Software Quality - Hierarchical models of Boehm and McCall – measurement – Metrics measurement – Metrics measurement and analysis – Gilb’s approach – GQM Model			
Module 2			8 Hours
Software Quality Assurance Quality tasks – SQA plan – Teams – Characteristics Implementation – Documentation– Reviews and Audits.			
Module 3			8 Hours
Quality Control And Reliability Tools for Quality – Ishikawa’s basic tools – CASE tools Defect prevention and removal– Reliability models Rayleigh model – Reliability growth models for quality assessment.			
Module 4			8 Hours
Quality Management System Elements of QMS – Rayleigh model framework Reliability Growth models for QMS –Complexity metrics and models Customer satisfaction analysis.			
Module 5			8 Hours
Quality Standards Need for standards – ISO 9000 Series – ISO 9000 3 for software development – CMM and CMMI – Six Sigma concepts.			
Question paper pattern:			
<ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. <p>The students will have to answer 5 full questions, selecting one full question from each module.</p>			
Text Books:			
<ol style="list-style-type: none"> 1. Allan C. Gillies, “Software Quality: Theory and Management”, Thomson Learning, 2003. 2. Stephen H. Kan, “Metrics and Models in Software Quality Engineering”, Pearson Education (Singapore) Pte Ltd., 2002. 			

Reference Books:

1. Norman E. Fenton and Shari Lawrence Pfleeger, "Software Metrics" Thomson, 2003
2. Mordechai Ben Menachem and Garry S. Marliss, "*Software Quality*", Thomson Asia Pvt Ltd, 2003
3. Mary Beth Chrissis, Mike Konrad and Sandy Shrum, "*CMMI* ", Pearson Education (Singapore) Pvt Ltd, 2003

MIS & E-Commerce [As per Choice Based Credit System (CBCS) scheme] SEMESTER – IV			
Subject Code	17MCA4064E	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
CREDITS – 03			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Understand Decision making and DSS.</p> <p>CO2: Analyze different electronic business systems.</p> <p>CO3: Understand the concepts of e-commerce.</p> <p>CO4: Understand the techniques of the e-commerce and internet communication</p>			
Modules			Teaching Hours
Module 1			8 Hours
<p>Decision Making and DSS Overview of MIS, Decision making concepts; decision making process, decision-making by analytical modeling, Behavioral concepts in decision making, organizational decision-making, Decision structure, DSS components, Management reporting alternatives</p>			
Module 2			8 Hours
<p>Electronic Business systems Enterprise business system – Introduction, cross-functional enterprise applications, real world case, Functional business system, - Introduction, marketing systems, sales force automation, CIM, HRM, online accounting system, Customer relationship management, ERP, Supply chain management (real world cases for the above).</p>			
Module 3			8 Hours
<p>E-Commerce Introduction Course overview; Introduction to e-commerce, E-commerce Business Models and Concepts, Ecommerce Infrastructure: The Internet and World Wide Web, Web design, JavaScript Internet Information Server (IIS); Personal Web Server (PWS).</p>			
Module 4			8 Hours
<p>E-Commerce techniques and Issues Introduction to Active Server Pages (ASP), Introduction to Product Recommendation Systems, Building an E-Commerce Web Site, E-Commerce Payment Systems, E-Commerce Marketing Techniques, Building product catalogue, Search product catalogue, Web Spider and search agent, Ethical, Social and Political Issues in E-Commerce</p>			
Module 5			8 Hours

Internet Communication

Transaction Systems, Shopping Carts, XML, E-Commerce Applications: Business-to-Consumer (B2C), Consumer-to-Consumer (C2C), Business-to-Business (B2B), Digital Government, Marketplaces, and Communities, E-Governance, Internet Banking.

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 20 marks.
- There will be 2 full questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Waman S Jhawadekar: Management Information System, 6th Edition, Tata McGraw Hill, 2010.
2. James A O'Brien and George M Marakas: Management Information System, 7th Edition, Tata McGraw Hill, 2006,
3. Turban, Rainer, and Potter, Introduction to E-Commerce, second edition, 2003

Reference Books:

1. 1H. M. Deitel, P. J. Deitel and T. R. Nieto, E-Business and E-Commerce: How to Programe, Prentice hall, 2001
2. Henry Chan, Raymond Lee, Tharam Dillon, Elizabeth Chang: E-Commerce Fundamentals and Applications, Wile India Edition
3. Ralph M Stair and George W Reynolds: Principles of Information Systems, 7th Edition, Thomson, 2010 .

SEMESTER - V

Object-Oriented Modelling And Design Patterns [As per Choice Based Credit System (CBCS) scheme] SEMESTER – V			
Subject Code	17MCA501	CIE Marks	50
Number of Lecture Hours/Week	04	SEE Marks	50
Total Number of Laboratory Hours	02		
Total Number of Lecture Hours	50	SEE Hours	03
CREDITS – 05			
Course Outcomes (CO): After studying this course, students will be able to:			
CO1: Acquire knowledge of <ul style="list-style-type: none"> ○ Basic UML Concepts and terminologies ○ Life Cycle of Object oriented Development ○ Modelling Concepts 			
CO2: Identify the basic principles of Software modelling and apply them in real world applications			
CO3: Produce conceptual models for solving operational problems in software and IT environment using UML			
CO4: Analyze the development of Object Oriented Software models in terms of <ul style="list-style-type: none"> ○ Static behaviour ○ Dynamic behaviour 			
CO5: Evaluate and implement various Design patterns			
Modules			Teaching Hours
Module 1			10 Hours
Modelling Concepts & Class Modelling What is Object Orientation? What is OO development? OO themes; Evidence for usefulness of OO development; OO modelling history. Modelling as Design Technique: Modelling; abstraction; The three models. Object and class concepts; Link and associations concepts; Generalization and inheritance; A sample class model; Navigation of class models; Practical tips. Advanced object and class concepts; Association ends; N-array associations; Aggregation; Abstract classes; Multiple inheritance; Metadata; Reification; Constraints; Derived data; Packages;			
Module 2			10 Hours
State Modelling and Interaction Modelling State Modelling: Events, States, Transitions and Conditions; State diagrams; State diagram behaviour; Practical tips. Advanced State Modelling: Nested state diagrams; Nested states; Signal generalization; Concurrency; A sample state model; Relation of class and state models; Interaction Modelling: Use case models; Sequence models; Activity models. Use case relationships; Procedural sequence models; Special constructs for activity models			
Module 3			10 Hours
System Conception and Analysis System Conception: Devising a system concept; elaborating a concept; preparing a problem statement. Overview of analysis; Domain class model; Domain state model; Domain interaction model; Iterating the analysis. Application Analysis: Application interaction model; Application class model; Application state model; adding operations.			
Module 4			10 Hours

<p>System Design and Class Design Overview of system design; Estimating performance; Making a reuse plan; Breaking a system in to sub-systems; Identifying concurrency; Allocation of sub-systems; Management of data storage; Handling global resources; Choosing a software control strategy; Handling boundary conditions Class Design: Overview of class design; Bridging the gap; Realizing use cases; Designing algorithms; Recording downwards, Refactoring; Design optimization; Reification of behaviour.</p>	
<p>Module 5</p>	<p>10 Hours</p>
<p>Patterns and Design Patterns What is a pattern and what makes a pattern? Pattern categories; Relationships between patterns; Pattern description. Introduction, Model View Controller, Structural decomposition: Whole-Part, Access Control: Proxy; Management Patterns: Command processor; Publisher-Subscriber</p>	
<p>Question paper pattern:</p> <ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. <p>The students will have to answer 5 full questions, selecting one full question from each module.</p>	
<p>Text Books:</p> <ol style="list-style-type: none"> 1. Michael Blaha, James Rumbaugh, “Object-Oriented Modelling and Design with UML”, 2nd Edition, Pearson Education / PHI, 2005. 2. Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal, “Pattern-Oriented Software Architecture, A System of Patterns”, Volume 1, John Wiley and Sons, 2006. 	
<p>Reference Books:</p> <ol style="list-style-type: none"> 1. Grady Booch et al, “Object-Oriented Analysis and Design with Applications”, 3rd Edition, Pearson, 2007. 2. Mark Priestley, “Practical Object-Oriented Design with UML”, 2nd Edition, Tata McGraw-Hill, 2003. 	
<p><i>Laboratory Experiments:</i></p>	
<ul style="list-style-type: none"> • The student has to draw the necessary UML diagrams using any suitable UML drawing tool and implement a program in Java or C++ or C# to demonstrate the design pattern. • For analysis and design models - Class Diagram, Use-Case, Activity Diagram, Sequence diagrams and Collaboration diagrams should be drawn. 	
<p>NOTE: ANY SUPPORTING TOOL MAY BE USED.</p> <p>Part A: UML Diagrams</p> <p>1. UML class representation</p> <p>In the employee referral process, the HR head of the region where a vacancy exists informs employees of that region and other regional HR heads. The other regional HR heads inform employees by putting up a notice informing them about the vacancy. The employees send on their recommendations to the regional HR head of a region where a vacancy exists. The</p>	

regional HR head then matches the skills of these candidates with the skills required for the vacant position and short lists them. An interview schedule is drawn up and the short listed candidates are informed. Based on the interview proceedings, interview details are updated and all the selected candidates are given offer letter. The candidate informs the HR either by accepting or declining the offer letter. When a candidate referred by the employee joins the organization, the employee who has referred the candidate is paid the bonus.

- a) Identify the various use cases and actors involved and represent the user view of the system.
- b) Identify the various classes and attributes and bring out a class diagram. and sequence diagram

2: UML class representation

Design a student class with the following attributes: Reg. no, Name of student, Marks in subject-1, Marks in subject-2, Marks in subject-3 and Total marks

- a) Identify the various use cases and actors involved and represent the user view of the system.
- b) Identify the various classes and attributes and bring out a class diagram, and a sequence diagram.

3. C library information system:

A library lends books and magazines to members, who are registered in the system. Also it handles the purchase of new titles for the library. Popular titles are bought in multiple copies. Old books and magazines are removed when they are out of date or in poor condition. A member can reserve a book or magazine that is not currently available in the library, so that when it is returned or purchased by the library, the person is notified. The library can easily create, replace and delete information about the titles, members loans and reservations in the system.

For the above problem specification devise the following UML diagrams:

- use case diagram
- class diagram
- state transition diagram
- sequence diagram
- collaboration diagram
- activity diagram

Part B: Design Patterns

1. Publisher-Subscriber: Define a one-to-many dependency between objects so that when one object changes state all its dependents are notified and updated automatically. The Observer pattern is also known as Dependents, Publisher-Subscriber. The need to maintain consistency between related objects without making classes tightly coupled.

2. Command Processor: The command processor design pattern separates the request for Service from its execution. A command processor component manages request as separate objects, schedules their execution and provides additional service.

3. Whole-Part: The idea of the Whole-Part pattern is to introduce a component that encapsulates smaller objects, and prevents clients from accessing these constituent parts directly. Define an interface for the aggregate that is the only means of access to the functionality of the encapsulated objects, allowing the aggregate to appear as a semantic unit.

<p align="center">Programming Using C# and .NET [As per Choice Based Credit System (CBCS) scheme] SEMESTER – V</p>			
Subject Code	17MCA502	CIE Marks	50
Number of Lecture Hours/Week	04	SEE Marks	50
Total Number of Lecture Hours	50	SEE Hours	03
Credits – 04			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Understand the concepts of .NET Framework, Architecture, Framework 4.0, Metadata and Assemblies. Knowledge on Windows Forms, ASP .NET and ASP .NET AJAX, ADO .NET.</p> <p>CO2: Understand the fundamentals of C# and .Net framework. Understand object inheritance and its use in C#.</p> <p>CO3: Develop the Windows Forms and apply user controls in Windows Forms applications.</p> <p>CO4: Design menus in a Windows Forms application and create Multiple Document Interface (MDI) applications.</p> <p>CO5: Bind Windows Forms applications to various data sources by using Microsoft ADO.NET.</p>			
Modules			Teaching Hours
Module 1			10 Hours
Getting started with .NET Framework 4.0 Benefits of .NET Framework, Architecture of .NET Framework 4.0, Components of .NET Framework 4.0: CLR, CTS, Metadata and Assemblies, .NET Framework Class Library, Windows Forms, ASP .NET and ASP .NET AJAX, ADO .NET, Windows workflow Foundation, Windows Presentation Foundation, Windows Communication Foundation.			
Module 2			10 Hours
Introducing C# Boxing and Undoing, Namespaces, The System namespace, Access Modifiers, Static Classes and Static Class Members. Static Property, Accessibility of accessors and Anonymous types. Encapsulation: Encapsulation using accessors and mutators, Encapsulation using Properties. Inheritance: Inheritance and Constructors, Sealed Classes and Sealed Methods, Extension methods. Polymorphism: Compile time Polymorphism/Overloading, Runtime Polymorphism/Overriding. Abstraction: Abstract classes, Abstract methods. Interfaces: Syntax of Interfaces, Implementation of Interfaces and Inheritance. Delegates: Creating and using.			
Module 3			10 Hours
Delegates and Events and Exception Handling Delegates, Multicasting with Delegates. Events: Event Sources, Event Handlers, Events and Delegates, Multiple Event Handlers. Exception Handling: The try/catch/finally statement, Checked and Unchecked Statements.			
Graphical User Interface with Windows Forms			

Windows Forms, Event Handling and basic controls	
Module 4	10 Hours
<p>Data Access with ADO.NET Describing the Architecture of ADO.NET, ADO.NET, ADO.NET Entity Framework. Creating Connection Strings: Syntax for Connection Strings. Creating a Connection to a Database: SQL Server Database, OLEDB Database, ODBC Data Source. Creating a Command Object. Working with DataAdapters: Creating DataSet from DataAdapter, Paging with DataAdapters, Updating with DataAdapters, Adding Multiple Tables to a DataSet, Creating Data View. Using DataReader to Work with Databases.</p>	
Module 5	10 Hours
<p>Web App Development with ASP.NET: Introduction, Web Basics, Multitier Application Architecture, First Web Application: Building WebTime Application, Standard Web Controls: Designing a Form, Validation Controls, Session Tracking: Cookies, Session Tracking with http Session State, Options.</p>	
<p>Question paper pattern:</p> <ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. <p>The students will have to answer 5 full questions, selecting one full question from each module.</p>	
<p>Text Books:</p> <ol style="list-style-type: none"> 1. .NET 4.0 Programming (6-in-1), Black Book, Kogent Learning Solutions Inc., Wiely- Dream Tech Press, 2010. 2. Paul Deitel and Harvey Deitel: C# 2010 for Programmers, 4th Edition, Pearson Education, 2010. 	
<p>Reference Books:</p> <ol style="list-style-type: none"> 1. Andrew Trolsen: Pro C# 5.0 and the .NET 4.5 Framework, 6th Edition, Wiely-Appress, 2012. 2. Bart De Smet: C# 4.0 Unleashed, Pearson Education- SAMS Series, 2013 	

Mobile Applications			
[As per Choice Based Credit System (CBCS) scheme]			
SEMESTER – V			
Subject Code	17MCA503	CIE Marks	50
Number of Lecture Hours/Week	04	SEE Marks	50
Total Number of Lecture Hours	50	SEE Hours	03
CREDITS – 04			
Course Outcome (CO): At the end of this course, the students will be able to:			
CO1: Illustrate effective user interfaces that leverage evolving mobile device capabilities			
CO2: Develop applications using software development kits (SDKs), frameworks and toolkits			
CO3: Establish various methods to integrate database and server-side technologies			
CO4: Design and develop open source software based mobile applications			
CO5: Build and deploy competent mobile development solutions			
Modules			Teaching Hours
Module 1			10 Hours
Introduction Preliminary Considerations – Cost of Development – Importance of Mobile Strategies in the Business World – Effective use of Screen Real Estate – Understanding Mobile Applications Understanding Mobile Applications Users – Understanding Mobile Information Design – Understanding Mobile Platforms – Using the Tools of Mobile Interface Design.			
Module 2			10 Hours
Getting Started with Android Programming What is Android – Obtaining the required tools– Anatomy of an Android Application – Components of Android Applications – Activities – Fragments – Utilizing the Action Bar			
Module 3			10 Hours
Android UI Design and Location Based Services Views and View Groups – Basic Views – Fragments – Displaying Maps – Getting Location Data – Publishing for Publishing – Deploying APK Files			
Module 4			10 Hours
Android Messaging and Networking SMS Messaging – Sending Email – Networking – Downloading Binary Data, Text files – Accessing Web Services – Performing Asynchronous Call – Creating your own services – Communicating between a service and an activity – Binding activities to services			
Module 5			10 Hours
IOS and Windows Phone 7 IOS – Obtaining the tools and SDK – Components of XCODE – Architecture of iOS –Building Derby App in iOS – Other useful iOS things – Windows Phone: Getting the tools you need – Windows Phone 7 Project – Building Derby App in Windows Phone 7 – Distribution – Other useful Windows Phone Thing			

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 20 marks.
- There will be 2 full questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module.

Text Book:

1. Jeff McWherter and Scott Gowell, "Professional Mobile Application Development", 1st Edition, 2012.

Reference Books:

1. Wei-Meng Lee, "Beginning Android Application Development", Wiley 2011.
2. Reto Meier, "Professional Android 4 Application Development", Wrox Publications 2012

.NET Laboratory [As per Choice Based Credit System (CBCS) scheme] SEMESTER – V			
Subject Code	17MCA506L	CIE Marks	50
Number of Lecture Hours/Week	03 Hrs Laboratory	SEE Marks	50
		SEE Hours	03
CREDITS – 1.5			
Course Outcomes(CO):After studying this course, students will be able to:			
CO1: Understand C# and client-server concepts using .Net Frame Work Components			
CO2: Apply delegates, event and exception handling to incorporate with ASP, Win Form, ADO.NET			
CO3: Analyze the use of .Net Components depending on the problem statement			
CO4: Implement & develop a web based and Console based application with Database connectivity			
Laboratory Experiments:			
1. Implement a Program in C# to demonstrate Command line arguments processing for the following. a) To find the square root of a given number. b) To find the sum & average of three numbers.			
2. Program in C# to demonstrate the following a) Boxing and Unboxing b) Invalid Unboxing.			
3. Implement a program in C# to add Two complex numbers using Operator Overloading.			
4. Develop a Program in C# to find the sum of each row of given jagged array of 3 inner arrays.			
5. Implement a Program in C# to demonstrate Array Out of Bound Exception using Try, Catch and Finally blocks.			
6. Implement a Program to Demonstrate Use of Virtual and override key words in C# with a simple program.			
7. Implement a Program in C# to create and implement a Delegate for any two arithmetic operations			
8. Implement a Program in C# to demonstrate abstract class and abstract methods in C#.			
9. Implement a program to Set & Get the Name & Age of a person using Properties of C# to illustrate the use of different properties in C#.			
10. Implement a Program in C# Demonstrate arrays of interface types (for runtime polymorphism).			
11. Consider the Database db_EMS (Employee Management System) consisting of the following tables : tbl_Designations (IdDesignation: int, Designation: string) tbl_EmployeeDetails(IdEmployee: int, EmployeeName: string, ContactNumber: string, IdDesignation: int, IdReportingTo: int) Develop a suitable window application using C#.NET having following options. 1. Enter new Employee details with designation & Reporting Manager. 2. Display all the Project Leaders (In a Grid) reporting to selected Project Managers (In a Combo box).			

3. Display all the Engineers (In a Grid) reporting to selected Project Leader (In a Combo box).
4. Display all the Employees (In a Grid) with their reporting Manager (No Value for PM).

12. Consider the Database db_LSA (Lecturer Subject Allocation) consisting of the following tables:

tbl_Subjects(IdSubject: int, SubjectCode: string, SubjectName: string)

tbl_Lecturers(IdLecturer: int, LecturerName: string, ContactNumber: string)

tbl_LecturerSubjects(IdSubject: int, SubjectCode: string, IdLecturer: int)

Develop a suitable window application using C#.NET having following options.

1. Enter new Subject Details.
2. Enter New Lecturer Details.
3. Subject Allocation with Lecturer Name in a Combo box and subjects to be allocated in Grid with checkbox Column.
4. Display all the subjects allocated (In a Grid) to the selected Lecturer (In a Combo Box).

13. Consider the database db_VSS (Vehicle Service Station) consisting of the following tables:

tbl_VehicleTypes(IdVehicleType: int, VehicleType: string, ServiceCharge: int)

tbl_ServiceDetails(IdService: int, VehicleNumber: string, ServiceDetails: string, IdVehicleType: int)

Develop a suitable window application using C#.NET having following options.

1. Enter new Service Details for the Selected Vehicle Type (In a Combo Box).
2. Update the Existing Service Charges to Database.
3. Total Service Charges Collected for the Selected Vehicle (In a Combo box) with total amount displayed in a text box.

14. Develop a web application using C#.NET and ASP.NET for the Postal System Management. The master page should contain the hyper links for adding **Area Details**, **Postman details**, **Letter distributions** and **View Letters**.

Consider the database db_PSM (Postal System Management) consisting of the following tables:

tbl_AreaDetails(IdArea: int, AreaName: string)

tbl_PostmanDetails(IdPostman: int, PostmanName: string, ContactNumber: string, IdArea: int)

tbl_AreaLetters(IdLetter: int, LetterAddress: string, IdArea: int)

Develop the suitable content pages for the above created 4 hyper links with the following details:

1. Enter New Area Details
2. Enter New Postman Details with the Area he/she is in-charge of (display Area in a Combo box)
3. Enter all the Letters distributed to the selected Area (display Area in a Combo box)
4. Display all the Letter addresses (In a Grid) to be distributed by the selected Postman (In a Combo box)

Note: In the examination each student should pick one program from the above list and another program as per examiner choice.

Mobile Applications in Android Laboratory [As per Choice Based Credit System (CBCS) scheme] SEMESTER – V			
Subject Code	17MCA507L	CIE Marks	50
Number of Lecture Hours/Week	03Hrs	SEE Marks	50
	Laboratory	SEE Hours	03
CREDITS – 1.5			
Course Outcome (CO): At the end of this course, the students will be able to:			
CO1: Illustrate effective user interfaces that leverage evolving mobile device capabilities			
CO2: Develop applications using software development kits (SDKs), frameworks and toolkits			
CO3: Establish various methods to integrate database and server-side technologies			
CO4: Design and develop open source software based mobile applications			
CO5: Build and deploy competent mobile development solutions			
Sl. No.	Laboratory Experiments		
1	Design an application that contains Phone Contacts in vertical linear manner. Selected contact appears at the top of the list with a large italicized font and a blue background		
2	Create an application that uses Layout Managers and Event Listeners		
3	Develop a standard calculator application to perform basic calculations like addition, subtraction, multiplication and division		
4	Devise an application that draws basic graphical primitives (rectangle, circle) on the screen		
5	Build an mobile application that create, save, update and delete data in a database		
6	Devise an application that implements Multi threading		
7	Develop a mobile application that uses GPS location information		
8	Create an application that writes data to the SD card.		
9	Implement an application that creates an alert upon receiving a message.		
10	Devise a mobile application that creates alarm clock		

Note: In the examination each student should pick one program from the above list and another program as per examiner choice.

17MCA508P Internet of Things (IoT)

Internet of Things (IoT): A self-study project based on IoT will be implemented and presented by a group of students (group size not exceeding four)

Report format:

- 1. Cover Page**
- 2. Acknowledgement**
- 3. Abstract**

Table of Contents:

Sl. No	Description	Page No.
1.	Introduction	
2.	Literature Survey	
3	Topic Specific Description	
4	Conclusions	
5	Bibliography	

Rules:

1. Put page borders
2. Main Heading font size 17(Times New Roman)
3. Subheading : font size 14(Times New Roman)
4. Text: font size 12(Times New Roman)
5. Figures should have Fig numbers at the bottom of the figure e.g. Fig 3.2 is second figure in 3rd chapter
6. Tables should have table numbers at the top example Table 5.1 i.e, 1st table in 5th chapter
7. Paragraph spacing 1.5

Rich Internet Applications [As per Choice Based Credit System (CBCS) scheme] SEMESTER – V			
Subject Code	17MCA5041E	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
Credits – 03			
Course Outcomes(CO): After studying this course, students will be able to:			
CO1: Define and illustrate rich internet concepts and applications			
CO2: Analyze the working of development models in web designing			
CO3: Illustrate appropriate component lifecycle techniques using frameworks			
CO4: Evaluate and implement state based systems using data models and data binding			
Modules			Teaching Hours
Module 1			08 Hours
<p>Rich Internet Applications With Ajax: Limitations of Classic Web application model, AJAX principles, Technologies behind AJAX, Examples of usage of AJAX; Asynchronous communication and AJAX application model</p> <p>Ajax with XMLHTTP object: Creating Ajax Applications: An example, Analysis of example ajax.html, Creating the JavaScript, Creating and opening the XMLHttpRequest object, Data download, Displaying the fetched data, Connecting to the server, Adding Server-side programming, Sending data to the server using GET and POST.</p>			
Module 2			08 Hours
Handling multiple XMLHttpRequest objects in the same page, Using two XMLHttpRequest objects, Using an array of XMLHttpRequest objects, Using inner functions, Downloading JavaScript, connecting to Google Suggest, Creating google.php, Downloading from other domains with Ajax, HTML header request and Ajax, Defeating caching, Examples.			
Module 3			08 Hours
<p>Working with XML DOM in Ajax Building XML and working with XML in JavaScript, Getting the document element, Accessing any XML element, Handling whitespace in Firefox, Handling cross-browser whitespace, Accessing XML data directly, Validating XML, Further examples of Rich Internet Applications with Ajax.</p> <p>Working with PHP and Ajax Working with PHP server variables, Getting the data in to array format, Wrapping applications in to a single PHP page, Validating input from the user, Validating integers and text, DOM, Appending new elements to a web page using the DOM and Ajax, Replacing elements using the DOM, Handling timeouts in Ajax, Downloading images with Ajax, Example programs.</p>			
Module 4			08 Hours
<p>Introduction to Bootstrap. What Is Bootstrap? Bootstrap File Structure, Basic HTML Template,</p>			

Global Styles, Default Grid System, Basic Grid HTML, Offsetting Columns, Nesting Columns, Fluid Grid System, Container Layouts, Responsive Design. Typography, Emphasis Classes, Lists, Code, Tables, Optional Table Classes, Table Row Classes, Forms, Buttons, Images, Icons..	
Module 5	08 Hours
<p>Dropdown Menus, Button Groups, Button Groups as Radio Buttons and Checkboxes, Buttons with Dropdowns, Split Button Dropdowns, Dropup Menus, Navigation Elements, Tabular Navigation, Basic Pills Navigation, Stackable Navigation, Dropdowns, Navigation Lists, Tabable Navigation, Navbar, Forms, Navbar Menu Variations, Breadcrumbs, Pagination, Pager, Labels, Badges, Typographic Elements.</p> <p>Programmatic API, Transitions, Modal, Dropdown, Dropdown Usage via JavaScript, Scrollspy, Toggleable Tabs, Tooltips, Popover, Alerts, Buttons, Collapse, Carousel, Typeahead, Affix.</p>	
<p>Question paper pattern:</p> <ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. <p>The students will have to answer 5 full questions, selecting one full question from each module.</p>	
<p>Text Books:</p> <ol style="list-style-type: none"> 1. Professional AJAX – Nicholas C Zakas et al, Wrox publications, 2008. 2. Steven Holzner: Ajax: A Beginner’s Guide, Tata McGraw Hill, 2014. 3. Jake Spurlock: "Bootstrap: Responsive Web Development", O'Reilly Media, 2014. 	
<p>Reference Books:</p> <ol style="list-style-type: none"> 1. Thomas A. Powel: Ajax The Complete reference, McGraw Hill, 2008. 2. Aravind Shenoy, Ulrich Sossou: Learning Bootstrap, Packt, Dec 2014. 3. Dana Moore, Raymond Budd, Edward Benson: Professional Rich Internet Applications: AJAX and Beyond, Wiley 2012. 	

Storage Area Networks [As per Choice Based Credit System (CBCS) scheme] SEMESTER – V			
Subject Code	17MCA5042E	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
Credits – 03			
Course Outcomes (CO): After studying this course, students will be able to: CO1: Understand the fundamentals of storage and storage networking concepts CO2: Analyze Network Attached and Storage Area Networks Requirements CO3: Apply and Integrate SAN and NAS solutions for an enterprise requirements CO4: Design a secured, scalable SAN / NAS enterprise solutions			
Modules		Teaching Hours	
Module 1		08 Hours	
Concepts of Storage Networking The Data Storage and Data Access Problem, The Battle for Size and Access Decoupling the Storage Component: Putting Storage on the Network, Decoupling the Storage Component: Creating a Network for Storage			
Module 2		08 Hours	
Storage Fundamentals Storage Architectures, Device Overviews, Connectivity Options, Data Organizational Methods			
Module 3		08 Hours	
Network Attached Storage Putting Storage on the Network, NAS Hardware Devices , NAS Software Components, NAS Connectivity Options			
Module 4		08 Hours	
Storage Area Networks Architecture Overview, Hardware Devices, Software Components, Configuration Options for SANs.			
Module 5		08 Hours	
Application Defining the I/O Workload, Applying the SAN Solution, Applying the NAS Solution Considerations When Integrating SAN and NAS Management Planning Business Continuity, Managing Availability, Maintaining Serviceability, Capacity Planning and Security Considerations Case Studies NAS Case Study, SAN Case Study, SAN/NAS Management Case Study			

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 20 marks.
- There will be 2 full questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. The Complete Reference “ Storage networks”, Robert Spalding, Mc Graw Hill Education (India) 2003

Reference Books:

1. Information Storage and Management (Misl-Wiley) : 2nd Edition, Emc Education Services, Wiley; Second edition (29 August 2012)
2. Storage Area Networks Essentials : A complete guide to understanding and Implementing SANs, Richard Barker, Paul Massiglia, Wiley, 2002

Artificial Intelligence [As per Choice Based Credit System (CBCS) scheme] SEMESTER –V			
Subject Code	17MCA5043E	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
CREDITS – 03			
Course Outcomes (CO): After studying this course, students will be able to:			
<p>CO1: understand concept of AI. CO2: Analyze Heuristic search techniques. CO3: Understand predicate logic and uncertainty. CO4: Understand the working of Neural Networks. CO5: Interpret and Implement non-trivial AI techniques in a relatively large system.</p>			
Modules			Teaching Hours
Module 1			08 Hours
Artificial Intelligence: The AI Problems, The Underlying assumption, AI Technique, The Level of the model, Criteria for success, Problem spaces and search: Production systems, Problem characteristics.			
Module 2			08 Hours
Heuristic search techniques: Generate-and-test, Hill climbing, Best-first search, Problem reduction, Constraint satisfaction, Mean-ends analysis. Knowledge representation issues: Representations and mappings, Approaches to knowledge representation, Issues in knowledge representation, The frame problem.			
Module 3			08 Hours
Using predicate logic: Representing simple facts in logic, representing instance and ISA relationships, Computable functions and predicates, Resolution, Natural Deduction. Symbolic Reasoning Under Uncertainty: Introduction to nonmonotonic reasoning, Logic for nonmonotonic reasoning.			
Module 4			08 Hours
Neural Networks Introduction, Humans and Computers, Organization of the Brain, Biological Neuron, Biological and Artificial Neuron Models, Characteristics of ANN, McCulloch-Pitts Model, Historical Developments, Potential Applications of ANN.			
Module 5			08 Hours
Weak Slot-and-filter structures: Semantic Nets Frames, Strong slot-and –filler structures: Conceptual dependency, scripts, CYC.			
Question paper pattern: <ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. · Each question will have questions covering all the topics under a module. The students will have to answer 5 full questions, selecting one full question from each module.			

Text Books:

1. Elaine Rich, Kevin Knight, Shivashankar B Nair: Artificial Intelligence, Tata McGraw Hill 3rd edition. 2013

Reference Books:

1. Stuart Russel, Peter Norvig: Artificial Intelligence A Modern Approach, Pearson 3rd edition 2013.

Multimedia			
[As per Choice Based Credit System (CBCS) scheme]			
SEMESTER – V			
Subject Code	17MCA5044E	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
CREDITS – 03			
Course Outcome (CO): At the end of this course, the students will be able to:			
<p>CO1: Understand the terminology and concepts of Multimedia Elements and audio technology</p> <p>CO2: Acquire the concepts of Graphics and Images, Video Technology, Computer-Based Animation</p> <p>CO3: Analyze the different modes of data compression</p> <p>CO4: Analyze the different data and file format standards</p>			
Modules			Teaching Hours
Module 1			8 Hours
<p>Introduction, Media and Data Streams, Audio Technology Multimedia Elements; Multimedia Applications; Evolving Technologies for Multimedia Systems; Defining Objects for Multimedia Systems; Characterizing Data Streams: Asynchronous Transmission Mode, Synchronous Transmission Mode, Isochronous Transmission Mode; Characterizing Continuous Media Data Streams. Sound: Frequency, Amplitude, Sound Perception and Psychoacoustics; Audio Representation on Computers</p>			
Module 2			8 Hours
<p>Graphics and Images, Video Technology, Computer-Based Animation: Capturing Graphics and Images Computer Assisted Graphics and Image Processing; Graphics and Image Output Options. Basics; Television Systems; Digitalization of Video Signals; Digital Television; Basic Concepts; Specification of Animations; Methods of Controlling Animation.</p>			
Module 3			8 Hours
<p>Data Compression Basic Compression Techniques; JPEG: Image Preparation, Lossy Sequential DCT-based Mode, Expanded Lossy DCT-based Mode, Lossless Mode, Hierarchical Mode. H.261 (Px64) and H.263,Fractal Compression.</p>			
Module 4			8 Hours
<p>Data and File Format Standards Rich-Text Format; TIFF File Format; Resource Interchange File Format (RIFF); MIDI File Format; JPEG DIB File Format for Still and Motion Images; AVI Indeo File Format; MPEG Standards; TWAIN</p>			
Module 5			8 Hours
<p>Multimedia Application Design Multimedia Application Classes; Types of Multimedia Systems; Virtual Reality Design; Components of Multimedia Systems; Application Workflow Design Issues; Distributed Application Design Issues.</p>			

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 20 marks.
- There will be 2 full questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Ralf Steinmetz, Klara Narstedt: Multimedia Fundamentals: Vol 1-Media Coding and Content Processing, 2nd Edition, Pearson Education, 2003.
2. Prabhat K. Andleigh, Kiran Thakrar: Multimedia Systems Design, PHI, 2003.

Reference Books:

1. K.R Rao, Zoran S. Bojkovic and Dragorad A. Milovanovic: Multimedia Communication Systems: Techniques, Standards, and Networks, Pearson Education, 2002.
2. Nalin K Sharad: Multimedia information Networking, PHI, 2002

Software Defined Networks [As per Choice Based Credit System (CBCS) scheme] SEMESTER – V			
Subject Code	17MCA5051E	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
CREDITS – 03			
Course Outcomes (CO): After studying this course, students will be able to:			
<p>CO1: Recognize the fundamentals and characteristics of Software Defined Networks CO2: Understand the basics of Software Defined Networks Operations and Data flow CO3: Discriminate different Software Defined Network Operations and Data Flow CO4: Analyze alternative definitions of Software Defined Networks CO5: Apply different Software Defined Network Operations in real world problem</p>			
Modules			Teaching Hours
Module 1			08 Hours
<p>Introduction to SDN Understanding the SDN, Understanding the SDN technology, Control Plane, Data Plane, Moving information between planes, separation of the control and data planes, Distributed control planes, Load Balancing , Creating the MPLS Overlay, Centralized control planes.</p>			
Module 2			08 Hours
<p>Working of SDN Evaluation of Switches and Control planes, SDN Implications, Data centre Needs, Forerunner of SDN ,Software Defines Networks is Born, Sustain SDN interoperability, Open source contribution, Fundamental Characteristics of SDN, SDN Operations, SDN Devices, SDN Controllers, SDN Applications, Alternate SDN methods.</p>			
Module 3			08 Hours
<p>The Open Flow Specifications Open Flow Overview, Open Flow Basics, Open Flow 1.0 additions, Open Flow 1.1 additions, Open Flow 1.2 additions, Open Flow 1.3 additions, Open Flow limitations.</p>			
Module 4			08 Hours
<p>SDN via APIS, SDN via Hypervisor-Based Overlays, SDN via Opening up the device, Network function virtualization, Alternative Overlap and Ranking.</p>			
Module 5			08 Hours
<p>Data centres definition, Data centres demand, tunnelling technologies for Data centres Path technologies in data centres, Ethernet fabrics in Data centres, SDN use case in Data centres</p>			
Question paper pattern:			
<ul style="list-style-type: none"> · The question paper will have ten questions. · Each full question consists of 20 marks. · There will be 2 full questions from each module. 			

- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Software Defined Networking by Thomas D Nadeau and Ken Gray, 2013.
2. Software Define Networks, A Comprehensive Approach, Paul Goransson, Chuck Black. MK Publications, 2013.

Reference Books:

1. Software Defined Networking for Dummies brought you by cisco, Brian Underdahl and Gary Kinghorn, 2015.

Cloud Computing [As per Choice Based Credit System (CBCS) scheme] SEMESTER – V			
Subject Code	17MCA5052E	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
Credits – 03			
Course outcomes: After studying this course, students will be able to:			
<p>CO1: Understand the cloud computing delivery model and the enabling technologies.</p> <p>CO2: Explain and cloud computing platforms, key technology drivers and cloud programming/software environments</p> <p>CO3: Identify the need for cloud computing model and compare various key enabling technologies.</p> <p>CO4: Analyze and choose an appropriate programming environment for building cloud applications.</p>			
Modules		Teaching Hours	
Module 1		08 Hours	
Distributed System Models and Enabling Technologies Scalable Computing Service over the Internet, System Models for Distributed and Cloud Computing, SAAS Characteristics & Architecture.			
Module 2		08 Hours	
Software Environments for Distributed Cloud Computing, Performance, Security and Energy Efficiency, Edge Computing & Architecture.			
Module 3		08 Hours	
Virtual Machines and Virtualization of Clusters and Data Centers Implementation Levels of Virtualization, Virtualization Structures /Tools and Mechanisms, Virtual Cluster and Resource Management, Virtualization for Data-Center Automation.			
Module 4		08 Hours	
Cloud Platform Architecture over Virtualized Data Centers Cloud Computing and Service Models, Data-Center Design and Interconnection Networks, Architectural Design of Compute and Storage Clouds, Public Cloud Platforms: GAE, AWS, and Azure, Cloud Security and Trust Management			
Module 5		08 Hours	
Cloud Programming and Software Environments Features of Cloud and Grid Platforms, Programming Support of Google App Engine, Programming on Amazon AWS and Microsoft Azure, Emerging Cloud Software Environments			
Question paper pattern:			
<ul style="list-style-type: none"> · The question paper will have ten questions. 			

- Each full question consists of 20 marks.
- There will be 2 full questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

2. “Distributed and Cloud Computing, From Parallel Processing to the Internet of Things”, Kai Hwang, Geoffrey C. Fox. Jack J Dongarra, MK Publishers, 2012.

Reference Books:

1. “Moving to the Cloud, Developing Apps in the New World of Cloud Computing”, DinakarSitaram, Geetha Manjunath, Elsevier Publication, 2012.
2. “Cloud Computing, A Practical Approach”, Anthony T. Volte, Toby J. Volte and Robert Elsenpeter, McGraw Hill, 2010.

Service Oriented Architectures [As per Choice Based Credit System (CBCS) scheme] SEMESTER – V			
Subject Code	17MCA5053E	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
Credits – 03			
Course Outcomes (CO): After studying this course, students will be able to:			
<p>CO1: Describe about evolution, characteristics and services in SOA with SOA architecture, WSDL, SOAP and UDDI</p> <p>CO2: Analyze the SOA Architectural style, SOA strategies, modelling web services</p> <p>CO3: Design, implementing process of SOA in web service.</p> <p>CO4: Apply the SOA operational style for the web services.</p>			
Modules			Teaching Hours
Module 1			08 Hours
<p>Service Oriented Architecture Considerations for Enterprise-Wide SOA, Strawman Architecture for Enterprise-Wide SOA, Enterprise SOA Layers, Application Development Process, SOA Methodology for Enterprise, Architectural Considerations , Solution Architecting for Enterprise Applications ,Solution Architecture for Enterprise Applications Based on SOA Minimum risk estimators</p>			
Module 2			08 Hours
<p>Service Oriented Architecture Considerations for Enterprise-Wide SOA, Strawman Architecture for Enterprise-Wide SOA, Enterprise SOA Layers, Application Development Process, SOA Methodology for Enterprise, Architectural Considerations, Solution Architecting for Enterprise Applications, Solution Architecture for Enterprise Applications Based on SOA</p>			
Module 3			08 Hours
<p>SOA Analysis and Design Service-oriented Analysis and Design – Design of Activity, Data, Client and business process services – Technologies of SOA – SOAP – WSDL – JAX – WS – XML WS for .NET –Service integration with ESB – Scenario – Business case for SOA – stakeholder objectives – benefits of SPA – Cost Savings</p>			
Module 4			08 Hours
<p>SOA Implementation SOA based integration – integrating existing application – development of web services – Integration - SOA using REST – RESTful services – RESTful services with and without JWS – Role of WSDL, SOAP and Java/XML mapping in SOA – JAXB Data binding.</p>			
Module 5			08 Hours
<p>Application Integration JAX –WS 2.0 client side/server side development – Packaging and Deployment of SOA component – SOA shopper case study –WSDL centric java WS with SOA-J – related software – integration through service composition (BPEL) – case study - current trends.</p>			

Question paper pattern: <ul style="list-style-type: none">· The question paper will have ten questions.· Each full question consists of 20 marks.· There will be 2 full questions from each module.· Each question will have questions covering all the topics under a module. <p>The students will have to answer 5 full questions, selecting one full question from each module.</p>	
Text Books: <ol style="list-style-type: none">1. Thomas Erl: Service Oriented Architecture- Concepts, Technology and Design, Pearson Education, 2013.2. Shankar Khambhapaty, Service Oriented Architecture for Enterprise and Cloud Applications, 2nd Edition, Wiley-India, 2012.	
Reference Books: <ol style="list-style-type: none">1. I. J. Nagrath, “Electronics: Analog and Digital”, PHI.2nd edition, 2013.2. David A. Bell, “Electronic Devices and Circuits”, Oxford University Press, 2005.	

\

Software Project Management [As per Choice Based Credit System (CBCS) scheme] SEMESTER – V			
Subject Code	17MCA5054E	CIE Marks	50
Number of Lecture Hours/Week	03	SEE Marks	50
Total Number of Lecture Hours	40	SEE Hours	03
CREDITS – 03			
Course Outcomes (CO): After studying this course, students will be able to:			
CO1: Understand the practices and methods for successful software project management			
CO2: Identify techniques for requirements, policies and decision making for effective resource management			
CO3: Apply the evaluation techniques for estimating cost, benefits, schedule and risk			
CO4: Devise a framework for software project management plan for activities, risk, monitoring and control			
CO5: Devise a framework to manage people			
Modules			Teaching Hours
Module 1			08 Hours
Introduction To Software Project Management Introduction, Software Project Management Project Overview, Contract Management, Activities Covered by Software Project Management, Plans, Methods and Methodologies, Ways of categorizing software projects, Stakeholders, Setting Objectives, Business Case, Project Success and Failure, Management Control, Traditional versus Modern Project Management Practices.			
Module 2			08 Hours
Project Evaluation & Finance Evaluation of Individual Projects, Cost Benefit Evaluation Techniques, Risk Evaluation, Programme Management, Managing allocation of Resources within Programmes, Financial Accounting: An overview, Accounting concepts, Principles & Standards, Ledger posting, Trial balance, Profit and Loss account, Balance sheet			
Module 3			08 Hours
Activity Planning Objectives of Activity Planning, When to Plan, Project Schedules, Sequencing and Scheduling Activities, Network Planning Models, Forward Pass – Backward Pass , Identifying critical path, Activity Float, Shortening Project Duration, Activity on Arrow Networks Risk Management, Nature of Risk, Categories of Risk, A framework for dealing with Risk, Risk Identification, Risk analysis and prioritization, risk planning and risk monitoring			
Module 4			08 Hours
Monitoring And Control Creating the Framework, Collecting the Data, Review, Project Termination Review, Visualizing Progress, Cost Monitoring, Earned Value Analysis, Prioritizing Monitoring, Getting Project Back To Target, Change Control, Software Configuration Management			
Module 5			08 Hours

Managing People And Working In Teams

Introduction, Understanding Behavior, Organizational Behavior: A Background, Selecting the Right Person for the Job, Instruction in the Best Methods, Motivation, The Oldham –Hackman Job Characteristics Model, Stress –Health and Safety , Working In Teams, Becoming a Team, Decision Making, Leadership

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 20 marks.
- There will be 2 full questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Bob Hughes, Mike Cotterell, Rajib Mall, “Software Project Management”, Fifth Edition, Tata McGraw Hill, 2011.
2. “Accounting for Management” JawaharLal, 5th Edition, Wheeler Publications, Delhi, 2004.

Reference Books:

1. Jack Marchewka,” Information Technology-Project Management”, Wiley Student Version, 4th Edition, 2013.
2. James P Lewis, “Project Planning, Scheduling & Control”, McGraw Hill, 5th Edition, 2011.
3. Pankaj Jalote,” Software Project Management in Practice”, Pearson Education, 2002

SEMESTER VI

SEM VI PROJECT

VI SEMESTER		CREDIT BASED			
Subject Code	Subject Name	Examination			Credits
		CIE	SEE	Total	
17MCA601P	Dissertation Assessment	75	75	150	10
17MCA602P	Internal Evaluation and Viva voce	75	75	150	15
	Total	150	150	300	25

Dissertation Assessment (17MCA601P)

- The project shall be carried out in the same institution or in industry/R&D labs based on software tools and technologies learnt in MCA course/internship for minimum period of 16 weeks.
- The internal examiners (Project Guide with at least 2 years experience) and the external examiners shall be appointed by the College for the Final Evaluation of the project.
- Internal and external examiners shall carry out the evaluation of Dissertation report for 75 marks individually.
- Dissertation Assessment by the internal examiner will be considered as **CIE** and Dissertation Assessment by the external examiner will be considered as **SEE**

Internal Evaluation and Viva Voce (17MCA602P)

- Internal assessment (CIE) shall be evaluated by both the **internal and external guide** for 75 marks individually. The average of the marks allotted by the internal and external guides shall be the final marks for the Internal Evaluation.
- The project presentation and Viva-voce (SEE) shall be evaluated jointly by both the **internal and external examiners** for 75 marks.