VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI

MASTER OF COMPUTER APPLICATIONS

Draftcopy of MCA Syllabus 2016-2019

(as per choice based credit system)

The meeting of the Board of Studies commenced with a welcome note by the Chairman, Dr. L. Manjunatha Rao.

The Board of Studies recommends the draft syllabus for MCA Programme for the academic year 2016-17 with the following suggestions through deliberations:

1. Proposed the detailed syllabus for all 6 semesters of MCA program as per CBCS from the academic year 2016 - 2017 by finalizing the core and elective subjects with respect to current industry and all stake holders' requirements (as per the guide lines of ACM, IEEE - Computer Society and NBA).
2. Electives are offered in the light of new advancements in the field of information technology.
3. Proposed credit weightage for various components like lectures, tutorials, practical for each course.
4. Credits for curriculum components in Master of Computer Applications programme are allotted (mathematics, management, humanities, professional core, electives, project work, and seminar) as per VTU guidelines.
5. Proposed rules and regulations for MCA Programme as per Choice Based Credit System (CBCS) from the academic year 2016 – 2017 onwards.
6. Proposed the Scheme of Teaching and Examination for all six semesters of MCA programme.
7. Proposed the Course Outcomes (COs) as per the Outcome Based Education of NBA.
8. The BoS has recommended the detailed syllabus, rules and regulations, scheme of teaching and examination for all the six semesters.

The BoS meeting was concluded with vote of thanks by the chairman to all the members.

SIGNATURES OF PARTICIPANTS

SL. NO	NAME	INSTITUTION	BoS	SIGNATURE
1	Dr. L. Manjunatha Rao	Dr. Ambedkar Institute of Technology, Bengaluru	Chairman	
2	Dr. Thimmaraju. S.N	VTU PG Studies, Mysuru.	Member	9/6
3	Dr. Vasudeva	Shri Madhwa Vadiraja Institute of Technology, UDUPI.	Member	9-6-16
4	Dr. Kavya N P	RNS Institute of Technology, Bengaluru	Special invited member	9/6/16
5	Dr. Usha J	R.V College of Engineering Bengaluru.	Special invited member	Usha J 9/6/16
6	Dr. Poornima Nataraja	Dayananda Sagar College of Engineering, Bengaluru	Special invited member	Poornima 9/6/16
7	Prof . Veera Nagaiah M	Cambridge Institute of Technolgy, Bengaluru	Special invited member	09/06/16
8	Dr. Prasad Naik	Nitte Meenakshi Institute of Technology, Bengaluru	Special invited member	09/06/2016

2/3

DataStructuresUsingC [As per Choice Based Credit System (CBCS) scheme] SEMESTER – I			
Subject Code	16MCA11	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03
CREDITS – 04			

Course outcomes:

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

CO1: *Recall the concepts of data, Information and their representations*

CO2: *Understand ADT &amp; their concrete representations as data structures*

CO3: *Analyze iterative and recursive operations*

CO4: *Implement concrete data structures*

CO5: *Solve problems by creating ADT and concrete implementations using C programming*

Modules	**Teaching Hours**
Module -1	
IntroductiontoDataStructures Informationanditsmeaning:AbstractDataTypes, SequencesasValueDefinitions, ADTforVaryinglengthcharacterStrings, Data Types, Pointersandreview ofPointers, DataStructures.Arrays:ArrayasanADT, UsingOne-dimensionalArrays, ImplementingOne-DimensionalArrays, ArraysasParameters, HandlingofCharacterStringsandCharacterStrings	**10 Hours**
Module -2	
Stack and Recursion Definition and examples , Primitive operations, Example, The stack as an ADT, Representing stacks ,Implementing the pop operation, Testing for exceptional conditions , Implementing the push operations , Examples for infix , postfix, and prefix expressions, Basic definition and Examples ,Program to evaluate a postfix expression ,Converting an expression from infix to postfix, Program to convert an expression from infix to postfix, Applications of Stacks: Expression Evaluations, Recursion etc. Recursion: definition and processes, Factorial function, Multiplication of natural numbers, Fibonacci sequence, Binary search, Properties of recursive definition or algorithm. Binary search, Towers of Hanoi problem.	**10 Hours**

Module -3	
QueuesandLists The queue and its sequential representation, the queue as ADT, Insert operation, Priority queue, Array implementation of a priority queue. Linked lists, Inserting and removing nodes from a list, Linked implementations of stacks, getnode and Freenode operations, Linked implementation of queues, Linked list as a data Structure, Example of list operations, Header nodes, Array implementation of lists, Limitations of array implementation, allocating and freeing dynamic variables, Linked lists using dynamic variables, Non integer and non-homogenous lists, Other list structures: Circular lists, Stack as a circular lists, doubly linked lists, Application of Linked Lists: Stacks, Queues, Double-ended Queues, Priority Queues.	**10 Hours**
Module -4	
Sorting and Searching Bubble sort, Quick sort, Selection sort, Tree Sorting: Binary Tree Sort, Heap Sort, Insertion Sorts: Simple Insertion, Shell Sort, Address Calculation Sort, Merge and Radix Sort. Basic Search Techniques: Algorithmic Notations, Sequential searching, Searching an ordered table, Indexed sequential search, Binary search, Interpolation search, Tree searching: Inserting into a Binary Search Tree ,Deleting form a binary search tree, Hashing : Resolving hash clashes by open addressing, Choosing a hash Function.	**10 Hours**
Module -5	
BinaryTrees Tree traversals, Binary Search Tree and Operations, AVL Tree and Operations, Red-Black Tree, Threaded binary trees and operations.	**10 Hours**
Text Books: 1. **Data Structures Using C and C++** by YedidyahLangsam and Moshe J. Augenstein and Aaron M Tenanbanum, 2nd Edition, Pearson Education Asia, 2002.	
Reference Books: 1. Balaguruswamy: Data Structures Using C, McGraw Hill Education 2. Data Structures and Algorithm Analysis in C, Mark Allen Weiss, 2nd Edition, Pearson Education Aisa, 1997. 3. Richard F Giberg and Behrouz A Forouzan: Data Structures – A Pseudo code Approach with C, 2nd Edition, Cengage Learning 4. Robert Kruse, C L Tondo, Bruce Leung and Shashi Mogalla: Data Structures and Program Design in C, 2nd Edition, Pearson Education	

UNIX Programming

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

SEMESTER – I

Subject Code	16MCA12	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

CO1*: Understand and experience the UNIX environment, File system and hierarchy.*

CO 2*: Demonstrate commands to extract, interpret data for further processing.*

CO 3*: Apply commands to perform different tasks on various applications*

CO 4: *Analyze the usage of different shell commands and variables.*

CO 5*: Evaluate different commands with sample shell scripts*

Modules	**Teaching Hours**
Module -1	
Introductionof 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.	**10 Hours**
Module -2	
UNIX File System The file, what's in a filename? The 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: Is – 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	**10 Hours**

Module -3	
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, Prosperities 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 Parameters, get line, Built-in variables, Arrays, Functions, Interface with the Shell, Control Flow.	**10 Hours**
Module -4	
AdvancedShellProgramming The sh command, export, cd, the Command, expr, Conditional Parameter Substitution, Merging Streams, Shell Functions, eval, Exec Statement and Examples	**10 Hours**
Module -5	
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	**10 Hours**
Text Books: 1. YourUNIX-TheUltimateGuide, SumitabhaDas, TataMcGrawHill,	
Reference Books: 1. "UnixShellProgramming", YashwantKanetkar, 2. "BeginningShellScripting", EricFoster-Johnson, JohnCWelch, MicahAnderson, Wroxpublication. 3. UNIX: Concepts and Applications, SumitabhaDas, TataMcGrawHill,	

WebTechnologies

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

SEMESTER – I

Subject Code	16MCA13	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

CO1*: Develop Web apps using various development languages and tools.*

CO 2*:Build the ability to select the essential technology needed to develop and implement web applications*

CO 3*: Compare Scripting language utilities for static and dynamic environment*

CO 4:*Design XML document and presentation of XML document using css and xslt.*

CO 5*: Justify the need for CGI programming between PERL and various markup languages.*

Modules	**Teaching Hours**
Module -1	
WebFundamentals Internet, WWW, Web Browsers and Web Servers, URLs, MIME, HTTP, Security, the Web Programmers Toolbox. Evolution of the Web, Peak into the History of the Web, Internet Applications, Networks, TCP/IP, Higher Level Protocols, Important Components of the Web, Web Search Engines, Application Servers	**10 Hours**
Module -2	
Introduction to XHTML and JavaScript 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 <span> and <div> tags, Conflict resolution. 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,	**10 Hours**

Module -3	
JavaScriptand DynamicDocuments 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. Dynamic Documents with JavaScript: Introduction, Positioning Elements, Moving Elements, Element Visibility, Changing Colors and Fonts, Dynamic Content, Stacking Elements, Locating the Mouse Cursor, Reacting to a Mouse Click, Slow Movement of Elements, Dragging and Dropping Elements	**10 Hours**
Module -4	
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.	**10 Hours**
Module -5	
Perl and CGI Programming Origins and uses of Perl, Scalars and their operations, Assignment statements and simple input and output, Control statements, Fundamentals of arrays, Hashes, References, Functions, Pattern matching, File input and output; Examples. Using Perl for CGI Programming: The Common Gateway Interface; CGI linkage; Query string format; CGI.pm module; A survey example; Cookies.	**10 Hours**
Text Books: 1. RobertW.Sebesta:ProgrammingtheWorldWideWeb, 4thEdition, Pearsoneducation, 2012.Chapters 1, 2, 3, 4, 5, 6, 7, 8, 9, 11&13 **2.** M.Srinivasan:WebTechnologyTheoryandPractice, PearsonEducation, 2012.Chapter1, 2, 3 &4	
Reference Books: 1. Jeffrey C.Jackson:WebTechnologies-A ComputerScience Perspective,PearsonEducation, EleventhImpression,2012. 2. ChrisBates:WebProgrammingBuildingInternetApplications,3rdEdition,WileyIndia,2009. 3. InternetTechnologyandWebDesign,InstructionalSoftwareResearchandDevelopment(ISRD)Group,TataMcGrawHill,2011. Raj Kamal : Internet and Web Technologies, McGrawHill Education	

ComputerOrganization

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

SEMESTER – I

Subject Code	16MCA14	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

CO1: *Understand the Basics of Digital System*

CO2: *Understand the Basics of Computer System Organization*

CO3: *Apply the concepts of the number system in Designing Digital System.*

CO4: *Analise the need of Logic circuits in digital system*

CO5: *Create logic circuits for real time requirement*

Modules	**Teaching Hours**
Module -1	
BinarySystems and Combinational Logic DigitalComputersandDigitalSystems,BinaryNumbers,NumberBaseConversion,OctalandHexadecimalNumbers,subtractionusingr'sandr-1complements,BinaryCode,BinaryStorage andRegisters,BinaryLogic, IntegratedCircuits. AxiomaticDefinitionofBooleanAlgebra,BasicTheoremsandPropertiesofBooleanAlgebra,BooleanFunctions,CanonicalandStandardForms,OtherLogicOperations,DigitalLogicGates,ThemapMethod,Two–andThree–VariableMaps,Four–VariablesMap	**10 Hours**
Module -2	
ArithmeticCircuits and Sequential Logic NANDandNORImplementation,OtherTwo-LevelImplementations,Don'tCareConditions.Introduction,Adders, Subtractors,BinaryParallelAdder,DecimalAdder,MagnitudeComparator, Decoders, Multiplexers,BOOTHalgorithmforsignednumberswithexample. SequentialLogic: Introduction,differenttypesofFlip–Flops,TriggeringofFlip-Flops,Registers,ShiftRegisters,RipplecounterandSynchronousCounter.	**10 Hours**

Module -3	
Computer Types, Functional Units, Basic Operational Concepts, Bus structure, Software, Performance, Multiprocessing and Multi computers, Introduction to Assemblers and Compilers. Machine Instruction: Memory Locations and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes	**10 Hours**
Module -4	
AssemblyLanguage and Input/OutputOrganization Basics of Assembly Language Programme, ExamplesfromAssemblyLanguageProgramming. AccessingI/ODevices,Interrupts,DMA,ProcessorExample,Buses.CasestudyofIA32Intelprocessor	**10 Hours**
Module -5	
TheMemorySystem BasicConcepts,SemiconductorRAMMemories,Read–OnlyMemories,Speed,Size,andCost,CacheMemories,VirtualMemories,MemoryManagementRequirements,SecondaryStorage.	**10 Hours**
Text Books: 1. M.MorrisMano, “DigitalLogicandComputerDesign”, Pearson, 2012. 2. CarlHamacher, ZvonkoVranesicSafwatZaky, ”ComputerOrganization”, 5thedition, TataMcGraw-Hill, 2011	
Reference Books: 1. 1. JohnP.Hayes, “ComputerArchitectureandOrganization”, TataMcGraw-Hill, Edition, 2012. 2. SoumitrsKumarMandal,“DigitalElectronicsPrinciplesandApplications”, TataMcGraw-Hill, 2010 3. Hamacher , “ Computer Organization” , McGraw-Hill Education	

OperatingSystems
[As per Choice Based Credit System (CBCS) scheme]
SEMESTER – I

Subject Code	16MCA15	IA Marks	20
Number of Lecture Hours/Week	04	Exam Marks	80
Total Number of Lecture Hours	50	Exam Hours	03

CREDITS – 04

Course outcomes:

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

CO1: *Understand the Basics of Digital System*
CO2: *Understand the Basics of Computer System Organization*
CO3: *Apply the concepts of the number system in Designing Digital System.*
CO4: *Analise the need of Logic circuits in digital system*
CO5: *Create logic circuits for real time requirement*

Modules	**Teaching Hours**
Module -1	
ComputerandOperatingSystems Structure BasicElements,ProcessorRegisters,Instruction Execution, The Memory Hierarchy,CacheMemory,I/OCommunicationTechniques,IntroductiontoOperatingSystem,MainframeSystems, DesktopSystems,MultiprocessorSystems,DistributedSystems,ClusteredSystems,Real-TimeSystems,HandheldSystems,FeatureMigration,ComputingEnvironments. **SystemStructures:** SystemComponents,Operating–SystemServices,SystemCalls,SystemPrograms,SystemStructure,VirtualMachines,SystemDesignandImplementation,SystemGeneration	**10 Hours**
Module -2	
ProcessManagement and Mutual Execution Process,ProcessStates,ProcessDescription,ProcessControl, Executionof theOperatingSystem,SecurityIssues,Processes and Threads,Symmetric Multiprocessing(SMP),Microkernels,CPUScheduler andScheduling. PrinciplesofConcurrency, MutualExclusion: Hardware Support, Semaphores, Monitors, Message Passing, Readers/Writes Problem.	**10 Hours**

Module -3	
Deadlockand Memory Management 10 Hours PrinciplesofDeadlock,DeadlockPrevention,Deadlock Avoidance, DeadlockDetection,AnIntegratedDeadlockStrategy,DiningPhilosophersProblem MemoryManagement: Swapping,ContiguousMemoryAllocation,Paging,Segmentation,SegmentationwithPaging,DemandPaging,ProcessCreation,PageReplacement,AllocationofFrames,Thrashing	**1 0 H o ur s**
Module -4	
FileSystem and SecondaryStorage FileConcept,AccessMethods,Directory Structure,FileSystem Mounting,FileSharing,Protection,File–SystemStructure,File–SystemImplementation,DirectoryImplementation,AllocationMethods,Free–SpaceManagement**,** DiskStructure,DiskScheduling,DiskManagement.	**1 0 H o u rs**
Module -5	
ComputerSecurity and CasestudyofLinuxOperatingsystem TheSecurityProblem,UserAuthentication,ProgramThreats,SystemThreats. LinuxSystem Linuxhistory,DesignPrinciples,Kernelmodules,Process,management,scheduling, Memory management, Filesystems, Input and output,Inter-processcommunications	**10 Ho ur s**
Text Books: 1. Silberschatz, Galvin, Gagne, “OperatingSystemConcepts”John Wiley, SixthEdition, 2004 2. WilliamStallings, “OperatingSystemInternalsandDesignPrinciples”Pearson, 6th edition, 2012	
Reference Books: 1. Chakraborty,“OperatingSystems”JaicoPublishingHouse,2011. 2. DhananjayM.Dhamdhere,“OperatingSystems–AConcept–BasedApproach”,TataMcGraw–Hill,3rdEdition,2012 3. Elmasri,Carrick,Levine,“OperatingSystems–AspiralApproach”,TataMcGraw–Hill,2012 4. Dhamdhere , “ Operating System : A Concept”, McGraw–Hill Education	

DATA STRUCTURES USING C LABORATORY

Subject Code	16MCA16	CIE Marks	20
Number of Lecture Hours/Week	01Hr Tutorial (Instructions) 02 Hours Laboratory	SEE Marks	80
		SEE Hours	03

CREDITS – 02

Course outcomes:

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

CO1: *Apply data structure concepts to develop interactive applications in C.*

CO2: *Be fluent in the use of linked list based applications*

CO3: *Be fluent in the use of tree based applications.*

CO4: *Be fluent in the use of different types of sorting and searching techniques.*

Laboratory Experiments:

1. WriteaC programto Convertaprefixnotationtopostfixnotation.
2. WriteaC program to Evaluate a given postfix expression and its values for the variables
3. WriteaC program to Simulate the working of circular queue providing the following operations–Insert, Delete and Display.
4. WriteaC program to Demonstrate recursion
    a. Calculate GCD and LCM of 3 integer numbers
    b. Solve Towers of Hanoi Problem
    c. Calculate the sum for a given number ‘n’ from 1 to n.
5. WriteaC program to Simulate the working of a linked list providing the following operations
    a. Insert at the beginning
    b. Insert at the end
    c. Insert before a given element
    d. Insert at the position
    e. Display

6. WriteaC program to Simulate the working of a circular linked list providing the following operations
    a. Delete from the beginning
    b. Delete from the end
    c. Delete a given element
    d. Delete every alternate element
    e. DisplayInsertismandatory

7. WriteaC program to Simulate the working of a dequeue

8. WriteaC program to Simulate the working of a double linked list to implement stack and queue

9. WriteaC program to Create a binary tree and implement the tree traversal techniques of inorder, preorder and postorder.

10. WriteaC program to Implement quick sort

11. WriteaC program to Implement Heap sort.

12. WriteaC program to Implement the search techniques of
    a. Linear Search
    b. Binary Search

13. WriteaC program to
    a. Create AVL Tree
    b. Insert element to AVL tree
    c. Find the height of the AVL tree

UNIXProgrammingLaboratory

Laboratory Code	16MCA17	CIE Marks	20
Number of Lecture Hours/Week	01Hr Tutorial Instructions) 02 Hours Laboratory	SEE Marks	80
		SEE Hours	03

CREDITS – 02

Course outcomes:

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

CO1: Understand the Unix programming environment.

CO2: Be fluent in the use of Vi editor.

CO3: Be able to design and implement shell scripts to manage users with different types of permission and file based applications.

CO4: Be fluent to write Awk scripts.

Laboratory Experiments:

A. ExploretheUnixenvironment.

B. Explorevieditorwithvimtutor.Performthefollowingoperationsusingvieditor,butnotlimitedto:

1. Insert character,delete character,replace character
2. savethefileandcontinueworking
3. save thefileand exittheeditor
4. quittheeditor
5. quitwithoutsavingthefile
6. rename afile
7. insertlines,deletelines,
8. setlinenumbers
9. searchforapattern
10. moveforwardandbackward

1a. Writeashellscriptthattakesavaliddirectorynameasanargumentandrecursivelydescendallthesub-directories,findsthemaximumlengthofanyfileinthathierarchyandwritesthismaximumvaluetothestandard output.

1b. Writeashellscriptthatacceptsapathnameandcreatesallthecomponentsinthatpathnameasdirectories.Forexample, if the scriptis namedmpc,thenthe commandmpca/b/c/dshouldcreate directoriesa,a/b,a/b/c,a/b/c/d.

2a.Writeashellscriptthatacceptstwofilenamesasarguments,checksifthepermissionsforthesefilesareidenticalandifthepermissionsareidentical,outputcommonpermissionsandotherwiseoutputeach filenamefollowedbyitspermissions.

2b.Writeashellscriptwhichacceptsvalidlog-

innamesasargumentsandprintstheircorresponding homedirectories,ifno argumentsarespecified,printasuitableerrormessage.
3a. Createascriptfilecalledfile-propertiesthatreadsafilenameenteredandoutputsitProperties
3b. Writeshellscripttoimplementterminallocking(similartothelockcommand).Itshouldprompttheuserforapassword.Afteracceptingthepasswordenteredbytheuser,itmustpromptagainforthematchingpasswordasconfirmationandifmatchoccurs,itmustlockthekeyworduntilamatching passwordisenteredagainbytheuser,Notethatthescriptmustbewrittentodisregard BREAK,control-D.Notime limitneedbe implementedforthe lockduration.
4a. Writeashellscriptthatacceptoneormore filenamesasargument and convertallofthemto uppercase,providedtheyexistincurrentdirectory.
4b. Writeashellscriptthatdisplaysallthelinkstoafilespecifiedasthefirstargumenttothescript.Thesecondargument,whichisoptional,canbeusedtospecifyinwhichthesearchis tobegin.Ifthissecondargumentisnotpresent,thesearchistobeginincurrentworkingdirectory.Ineithercase,thestartingdirectoryaswellasallitssubdirectoriesat alllevelsmustbesearched. Thescript neednotincludeanyerrorchecking.
5a. Writeashellscriptthatacceptsasfilenameasargumentanddisplayitscreationtimeiffileexistandifitdoesnotsendoutputerror message.
5b. Writeashellscriptto displaythecalendar for currentmonth withcurrentdatereplacedby *or**dependingonwhetherthedatehasone digitortwodigits
6a. Writeashellscripttofindafile/sthatmatchesapatterngivenascommandlineargumentinthehomedirectory,displaythecontentsofthefileandcopythefileintothedirectory ~/mydir
6b. Writeashellscripttolistallthefilesinadirectorywhosefilenameisatleast10characters.(useexprcommandtocheckthelength)

7a. Writeashellscriptthatgetsexecuteddisplaysthemessageeither“GoodMorning” or“GoodAfternoon”or“GoodEvening”dependingupontimeatwhichtheuserlogsin.

7b. Writeashellscriptthatacceptalistoffilenamesasitsargument,countandreport occurrenceof eachword thatispresentinthefirstargumentfileon other argumentfiles.

8a. Writeashellscriptthatdeterminetheperiodforwhichaspecifieduserisworkingonsystemanddisplayappropriatemessage.

8b. Writeashellscriptthatreportsthelogginginofaspecifieduserwithinoneminuteafterhe/shelogin.Thescriptautomaticallyterminateifspecifieduserdoesnotlogin duringaspecifiedperiodof time.

9a. Writeashellscriptthatacceptthefilename,startingandendinglinenumberasan argumentand displayallthelines betweenthegivenlinenumber.

9b. Writeashellscriptthatfoldslonglinesinto40columns.Thusanylinethatexceeds 40charactersmustbebrokenafter40th,a“\”istobeappendedastheindicationoffoldingandtheprocessingistobecontinuedwiththeresidue.Theinputistobesuppliedthroughatextfilecreatedbytheuser.

10a. Write an awk scriptthatacceptsdateargumentin theform ofdd-mm-yyand displaysitintheform ifmonth,dayandyear.Thescriptshouldcheckthevalidityof theargumentandinthecaseoferror,displayasuitable message.

10b. Writeanawkscripttodelete duplicatedline from a text file.Theorderof theoriginallinesmustremainunchanged.

11a. Writeanawkscripttofindouttotalnumberofbookssoldineachdisciplineaswell astotalbook soldusingassociate arraydowntableasgivenbelow.

Electrical	34
Mechanical	67
Electrical	80
ComputerScience	43
Mechanical	65
Civil	98
ComputerScience	64

11b.

Writeanawkscripttocomputegrosssalaryofanemployeeaccordinglytorulegivenbelow.

If basicsalaryis<10000thenHRA=15%of basic&DA=45%ofbasicIfbasicsalaryis>=10000thenHRA=20 %ofbasic&DA=50%ofbasic.

Note: Intheexamination*each*studentpicksonequestionfromalotof*all the*11 questions.*QuestionA &B Nottobeincluded forexamination.*

WEBPROGRAMMINGLABORATORY

Laboratory Code	16MCA18	CIE Marks	20
Number of Lecture Hours/Week	01Hr Tutorial Instructions) 02 Hours Laboratory	SEE Marks	80
		SEE Hours	03

CREDITS – 02

Course outcomes:

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

CO1: Understand the concept and usages web based programming techniques.

CO2: Be fluent in the use of XHTML programs using Javascript

CO3: Be fluent in the use of CGI and Perl programs for different types of server side applications.

CO4: Design and implement user interactive dynamic web based applications.

Laboratory Experiments:

1. CreateanXHTMLpagetodemonstratetheusageof
   a. TextFormattingtags
   b. Links
   c. Images
   d. Tables

2. Develop and demonstrate the usage of inline and external style sheet using CSS.

3. Develop and demonstrate a XHTML file that includes JavaScript script for the following problems:
   a. Input: A number n obtained using prompt
      Output: The first n Fibonacci numbers
   b. Input: A number n obtained using prompt
      Output: A table of numbers from 1 to n and their squares using alert

4. Develop and demonstrate using JavaScript, a XHTML document that displays random numbers (integers).

5. a) Developanddemonstrate,usingJavaScriptscript,aXHTMLdocumentthatcollectstheUSN(thevalidformatis:Adigitfrom1to4followedbytwoupper-casecharactersfollowedbytwodigitsfollowedbytwoupper-casecharactersfollowedbythreedigits;noembeddedspacesallowed)oftheuser.Eventhandlermustbeincluded forthe formelementthatcollectsthisinformationtovalidatetheinput.Messagesinthealertwindowsmustbeproducedwhenerrorsaredetected.

b)Modifythe aboveprogramtoget thecurrentsemester also(restrictedtobea numberfrom1to

6. a.Developanddemonstrate,usingJavaScriptscript,aXHTMLdocumentthatcontainsthreeimages,stackedontopofeachother,withonlyenoughofeachshowingsothatthemousecursorcanbeplacedoversomepartofthem.Whenthecursorisplacedovertheexposedpartofanyparagraph,itshouldrisetothetoptobecomecompletely visible.

   b. Modifytheabovedocument sothat when animage is movedfromthetop stackingposition,itreturnstoitsoriginalpositionratherthantothebottom.

7. DevelopusingJavaScriptscript,anXHTMLdocumentthatuseofonloadandonfocusevents.

8. a. DesignanXMLdocumenttostoreinformationaboutastudentinanengineeringcollegeaffiliatedtoVTU.TheinformationmustincludeUSN,Name,NameoftheCollege,Brach,YearofJoining,ande-mailid.Makeupsampledatafor3students.CreateaCSSstylesheet anduseittodisplaythedocument.

   b.CreateanXSLTstylesheetforonestudentelementoftheabovedocumentanduseittocreateadisplayofthatelement.

9. Write a Perl program which demonstrates the usage of scalar variables and arrays

10. Write a Perl program to display various Server information like Server Name, Server

    Software, Server protocol, CGI Revision etc.

11. WriteaPerlprogram todisplayadigitalclock whichdisplaysthecurrenttimeoftheserver.

12. Writea Perl programto acceptthe UserName anddisplay a greeting messagerandomlychosenfromalistof4 greetingmessages.

13. WriteaPerlprogramtokeeptrackofthenumberofvisitorsvisitingthewebpageandtodisplaythiscount ofvisitors,withproperheadings.

14. WriteaCGI-Perlprogramtouseacookietorememberthedayofthelastloginfromauseranddisplayitwhenrun

Note:Intheexamination*each*studentpicksonequestionfromthelotof*all*14question.

PYTHON PROGRAMMING

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

Subject Code	16MCA21	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

CO1: Understand and comprehend the basics of python programming.
CO2: Apply knowledge in real time applications.
CO3: Understands about files and its applications.

Modules	**Teaching Hours**
Module -1	
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.	**10 Hours**
Module -2	
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.	**10 Hours**
Module -3	
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 Reading and Writing	**10 Hours**

Module -4	
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.	**10 Hours**
Module -5	
Collection of New Information Object-Oriented Programming : Understanding a Problem Domain , Function “Isinstance,” 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.	**10 Hours**

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 16 marks.
- There will be 2full 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:

1. Practical Programming: An introduction to Computer Science Using Python, second edition, Paul Gries, Jennifer Campbell, Jason Montojo, The Pragmatic Bookshelf.
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.
2. Exploring Python, Timothy A. Budd, Mc Graw Hill Education
3. Python for Informatics: Exploring Information, Charles Severance.
4. Learning Python, Fourth Edition, Mark Lutz, O’Reilly publication

Object Oriented Programming Using C++

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

Subject Code	16MCA22	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

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.

Modules	**Teaching Hours**
Module -1	
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	**10 Hours**
Module -2	
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.	**12 Hours**

Module -3	
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:** 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. Operator Overloading: Defining Operator Function, Rules for overloading Operators. Overloading unary operators, overloading binary Operators, Overloading Comma, [], (), ->, new, delete Operators. Type Conversions: Basic to class, Class to Basic and one Class to another Class type. Advanced Typecasting.	**12 Hours**
Module -4	
Templates, Exception Handling and I/O Streams Generic classes, a class template with more than one generic type, the power of templates. Exception Handling: Exception handling model, Exception handling constructs, list of Exceptions catch all exceptions, handling uncaught exceptions. I/O Streams: IO Stream basics, Output operator <<, input >>, additional I/O operators, overloading the output operator <<, Overloading the input operator >>, file input & output, manipulators	**10 Hours**
Module -5	
STL, New Features of C++ STL: An overview, containers, vectors, lists, maps **New Features of C++:** New data types, new style casts, style Header, new keyword for operators, explicit, mutable keywords, Namespaces.	**6 Hours**
Question paper pattern: • The question paper will have ten questions. • Each full question consists of 16 marks. • There will be 2full 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:

1. K.R. Venugopal, RajkumarBuyya, T. Ravishankar: Mastering C++ ,TataMcGraw Hill Publication, 2006.
2. Herbert Schildt: C++ The Complete Reference, 4th Edition, Tata McGraw Hill, 2007.
3. Object Oriented Programming in C++:By M.T Somashekara ,D.S Guru,H.S. Nagendraswamy,K.SManjunatha2nd Editions 2012

Reference Books:

1. Stephen Prata : C++ Primer Plus, 6th Edition, Pearson Education.
2. Al Stevens: C++ Programming, 7th Edition, Wiley India Publications
3. Stanley B.Lippmann, JoseeLajore: C++Primer, 4th Edition, Addison Wesley, 2005.
4. Object oriented programming with C++, E. Balaguruswamy, TMH.

DATABASE MANAGEMENT SYSTEM

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

Subject Code	16MCA23	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

CO1: Demonstrate the fundamentals of data models and conceptualize and depict a database system and Make use of ER diagram in developing ER Model

CO2: To Summarize the SQL and relational database design.

CO3: Illustrate transaction processing, concurrency control techniques and recovery

CO4: Inference the database design in the real world entities.

Modules	Teaching Hours
Module -1	
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.	**10 Hours**
Module -2	
Relational Model Relational Model and Relational Algebra:Relational Model Concepts, Relational Model Concepts, Relational Model Constraints and 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	**10 Hours**

Module -3	
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.	**10 Hours**
Module -4	
Database Design: 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, Properties of Relational Decomposition, Algorithms for relational database schema design.	**10 Hours**
Module -5	
Transaction Management: Transaction Concept, A Simple Transaction Model, Transaction Atomicity and Durability, Serializability, Transaction Isolation and Atomicity, Transaction Isolation Levels, Implementation of Isolation Levels. Concurrency Control: Lock Based Protocols, Deadlock Handling. Recovery System: Failure Classification, Storage, Recovery and Atomicity, Recovery Algorithm.	**10 Hours**
Question paper pattern: • The question paper will have ten questions. • Each full question consists of 16 marks. • There will be 2full 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: 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.	
Reference Books: 1 C.J. Date, A. Kannan, S. Swamynatham: An Introduction to Database Systems, 8th Edition, Pearson education,2009.	

SYSTEM SOFTWARE [As per Choice Based Credit System (CBCS) scheme] SEMESTER – II			
Subject Code	16MCA24	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03
CREDITS – 04			

Course outcomes:

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

CO1: Understand the introductory concepts of system software, SIC and SIC/XE machine architecture.

CO2: Understand the design and implementation of Assemblers with implementation examples.

CO3: Design and implement the linkers and loaders, macro processors and respective implementation examples.

CO4: Understands about basics of compilers.

Modules	**Teaching Hours**
Module -1	
Machine Architecture Introduction, System Software and Machine Architecture, Simplified Instructional Computer (SIC) – SIC Machine Architecture, SIC/XE Machine Architecture, SIC Programming Examples. VAX Architecture, UltraSPARC Architecture.	**07 Hours**
Module -2	
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, SPARC Assembler	**13 Hours**
Module -3	
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, SunOS Linker, Cray MPP linker	**10 Hours**

Module -4	
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 Macro Expansion, General- Purpose Macro Processors, Macro Processing Within Language Translators, Implementation Examples - MASM Macro Processor, ANSI C Macro Processor., ELENA macro processor	**10 Hours**
Module -5	
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 into passes, Interpreters, P-code Compilers, Compiler-Compilers, SunOS C compiler, YACC Compiler-compiler.	**10 Hours**
Question paper pattern: • The question paper will have ten questions. • Each full question consists of 16 marks. • There will be 2full 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: 1. Leland.L.Beck: System Software, 3rd Edition, Addison-Wesley, 1997.	
Reference Books: 1. J.Nithyashri, “System Software”, 2nd Edition, Tata McGraw Hill, 2010	

DISCRETE MATHEMATICAL STRUCTURES

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

SEMESTER – II

Subject Code	16MCA25	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

CO1: Use the logical notation to define and reason about fundamental mathematical concepts such as sets, relations, functions, and integers.

CO2: Calculate numbers of possible outcomes of elementary combinatorial processes such as permutations and combinations.

CO3: Calculate probabilities and discrete distributions for simple combinatorial processes; calculate expectations.

CO4: Apply graph theory models of data structures and state machines to solve problems of connectivity and constraint satisfaction, for example, scheduling.

Modules	**Teaching Hours**
Module -1	
Logic: Propositional logic, equivalences, predicates and quantifiers, rules of inference, introduction to proofs, proof methods.	**10 Hours**
Module -2	
Sets, Functions and Relations: Sets, set operations, functions, recursive functions, sequences and summations, relations, equivalence relations and partial ordering.	**13 Hours**
Module -3	
Counting: Basics of counting, the pigeonhole principle, permutations and combinations, recurrence relations, generating functions, inclusion-exclusion	**10 Hours**
Module -4	
Probability: Introduction to probability, axioms of probability, independence and conditional probability	**10 Hours**

Module -5	**10 Hours**
Graph Theory: Graphs, terminology and special types of graphs, isomorphism, Euler and Hamiltonian paths, planar graphs, graph coloring.	

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 16 marks.
- There will be 2full 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:

1. Discrete Mathematics and its Applications by Kenneth H Rosen, 7th edition, (Indian adaptation by Kamala Krithivasan), Tata McGraw Hill, 2011.
2. Discrete and Combinatorial Mathematics: An Applied Introduction by Ralph P. Grimaldi and B V Ramana (Indian corrupted adaptation), 5th edition, Pearson, 2011.

Reference Books:

1. Graph Theory with Applications to Engineering and Computer Science by NarsinghDeo, Prentice Hall India, 2004.
2. Discrete Mathematical Structures with Applications to Computer Science by J. P. Tremblay and R. Manohar, McGraw Hill.

Note: Unit IV to be covered from Text Book 2.
Unit I: T1: 1.1 – 1.8
Unit II: T1: 2.1- 2.5, 7.1, 7.5, 7.6
Unit III: T1:5.1 – 5.4, 6.1 – 6.4
Unit IV: T2: 3.4 – 3.6, T1: 6.5
Unit V: T1: 8.1 – 8.8

PYTHON PROGRAMMING LABORATORY

Laboratory Code	16MCA26	CIE Marks	20
Number of Lecture Hours/Week	01Hr Tutorial Instructions) 02 Hours Laboratory	SEE Marks	80
		SEE Hours	03

CREDITS – 02

Course outcomes:

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

CO1: Apply object-oriented programming concepts to develop dynamic interactive Python applications.

CO2: Be fluent in the use of procedural statements — assignments, conditional statements, loops, method calls — and arrays

CO3: Be able to design, code, and test small Python programs with a basic understanding of top-down design.

CO4: Be able to solve real-world problem using language idioms, data structures and standard library

Laboratory Experiments:

NOTE: The experiments are to be carried using discrete components only.

1.Write 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. Write a program to search an element using linear search

4. Write a program to search an element using binary search.

5. Write a program to simulate stack.

6. Using a stack evaluate an arithmetic expression.

7. Write a program to multiply two matrices.

8. Write a program to find the roots of a quadratic equation

9. Insert a number in a sorted array.

10. Write a Python Program to check whether the given string is palindrome or not using built in string manipulation methods.

11. Write a Python Program to read a word and prints the number of letters, vowels and percentage of vowels in the word using dictionary

12. Write a Python Program to check a given sentence is a pangram or not using function/Module.

13.Write 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.Write 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 addTimeto 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.

OBJECT ORIENTED PROGRAMMING USING C++ LABORATORY

Laboratory Code	16MCA27	CIE Marks	20
Number of Lecture Hours/Week	01Hr Tutorial Instructions) 02 Hours Laboratory	SEE Marks	80
		SEE Hours	03

CREDITS – 02

Course outcomes:

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 anarray of 10 STUDENT objects. Using appropriate functions, find the average of the two bettermarks for each student. Print the USN, Name and the average marks of all the students.

2. Write 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. Write a program to demonstrate friend functions and friend classes

4. Write a C++ program for scalar multiplication of two vectors using operator overloading

5. Write a C++ program to create a template function for Bubble Sort and demonstrate sorting of integers and doubles.

6. 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.

7. Write 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.

8. 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.

9. Define a class SET with Data members: array of int, int variable to indicate number of elements in a SET object; and Member functions: to read element of a SET object, to print elements of a SET object, to find union of 2 objects of SET using operator overloading (S3=S1+S2), to find intersection of 2 objects of SET using operator overloading (S4= S1*S2). S1, S2, S3 and S4 are objects of SET. Use this class in a main function to show the above operations.

10. 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.

11. Write a program to create a file to store some records and search for a particular record and display it.

12. Write a program to create a generic stack class and member functions to perform stack operations..

13. Write a program to implement Exception Handling with minimum 5 exceptions Classes including two built-in exceptions

14. Write a program to demonstrate namespaces and Volatile member functions

DATABASE MANAGEMENT SYSTEMS LABORATORY

Laboratory Code	16MCA28	CIE Marks	20
Number of Lecture Hours/Week	01Hr Tutorial Instructions) 02 Hours Laboratory	SEE Marks	80
		SEE Hours	03

CREDITS – 02

Course outcomes:

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: Able to 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:

***NOTE:* (Part A and Part B are assessed equally)**

PART A:

Unit I SQL

Data Definition Language: Create, Alter, Drop, Rename, Truncate
Data Manipulation Language: Insert, Update, Delete,
Select

Unit II Data Control Language: Grant, Revoke, Roles

Transaction Control: Commit, Rollback, SavepointSQL SELECT Statements: Selecting All Columns, Selecting Specific Columns,Column Alias, Concatenation Operator, Arithmetic Operators, ComparisonConditions, Logical Conditions, ORDER BY Clause

Unit III Functions:

Single Row Functions, Character Functions, Number Functions, DateFunctions, Conversion Functions, General Functions, Multiple Row Functions,Group FunctionSubquery: Subquery, Types of Subquery, Group Function, Having Clause

Unit IV Joins:

Equijoins, Non-Equijoins, Joining Three Tables, Self Joins, Left OuterJoins, Right Outer Joins, Full Outer Joins, Cross Joins, Natural JoinsOther Concepts: Sequence, View, Index, Synonyms

Unit V Constraints:

Not Null, Unique Key, Primary Key, Foreign Key, Check, Dropping a Constraint, Enabling & Disabling*3.* Write a program to demonstrate friend functions and friend classes

PART B: 22 Hrs.

Mini Project:

Database project:
Use of Real World Application with Technological Application by using Open Source software application and Tool
Software / Tools: Mysql, Postgre SQL 9.0

Part A:

In the Final examination any 5 Question will be given for Part A (any one question can be picked)

1. Drawing ER diagram
2. Schema Diagram
3. Table creation as per the question given (No. of Tables)
4. Inserting values (update / alter / modify etc.,)
5. Writing and Executing queries from given the question.
6. Report if any applicable to be done as per the question.

Part B:

Demonstration of project in groups (Minimum 2 or Max 3 members in a group)

Computer Networks

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

SEMESTER –III

Subject Code	16MCA31	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

CO1: trace the flow of information from one node to another node in the network

CO2: identify the components required to build different types of networks

CO3: understand the functionalities needed for data communication into layers

CO4: choose the required functionality at each layer for given application

CO5: understand the working principles of various application protocols

Modules	**Teaching Hours**
Module -1	
Introduction to Computer Networks and Physical Layer Networking Devices, Classification of Computer Networks, Network Protocol Stack (TCP/IP and ISO-OSI),Network Standardization and Examples of Networks.Data Transmission Concepts, Analog and Digital Data Transmission, Transmission Impairments and Channel Capacity, Guided and Wireless transmission, communication media, Digital modulation techniques (FDMA,TDMA,CDMA) and mobile telephone systems (1G,2G,3G and 4G).	**10 Hours**
Module -2	
Data Link layer Data link layer design issues, Error Detection and Correction Codes,Data Link Protocols and Sliding window protocols. **Medium Access Sub Layer** The Channel Allocation Problem, Multiple access protocols and Examples: Ethernet, Wireless LAN, Broadband Wireless and blue tooth, RFID, Data Link Layer Switching.	**10 Hours**
Module -3	
Network Layer Network Layer Design issues, Routing algorithms, Congestion Control Algorithms, Quality of Service, Internetworking and The Network Layer in the Internet	**10 Hours**

Module -4	
The Transport Layer The Transport Service, Elements of Transport Protocols, Congestion Control, The Internet Transport Protocol: UDP,The Internet Transport Protocols – TCP, Performance Issues.	**10 Hours**
Module -5	
The application Layer DNS, Email, WWW, Streaming audio and Video and Content Delivery	**10 Hours**
Text Books: 1. "Computer Networks" by Andrew S Tanenbaum, David J Wetheral, 5th Edition, Pearson 2012 Chapter 1, 2.2, 2.3, 2.5, 2.7, 3.1, 3.2, 3.3, 3.4, 4.2, 4.3, 4.4, 4.5, 4.6, 4.8 Chapter 5, Chapter 6 (excluding 6.7) 2 "Data and Computer Communications" by William Stallings , Above 7th Edition , 2004 Chapter 3	
Reference Books: 1. "Computer Networks" Principles,Technologies and Protocols for Network Design, by NATALA OLIFER and VICTOR OLIFER , 2010 2. http://www.ietf.org/rfc.html relevant RFC document could be used to get more detailed information about any of the concepts prescribed in the syllabus like RFC 2460 can be referred to get a detailed information about IPV6	

Java Programming

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

Subject Code	16MCA32	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

CO1: Demonstrate the basic programming constructs of Java Apply suitable OOP concepts to develop Java programs for a given scenario

CO2: Demonstrate Generalization and run time polymorphism

CO3: Exemplify the usage of Packages, Interfaces, Exceptions and Multithreading

CO4: Demonstrate Enumerations, Wrappers, Auto boxing, Generics, collection frameworkand I/O operations

CO5: Implement the concepts of Networking using Java network classes

Modules	Teaching Hours
Module -1	
Java Programming Fundamentals The Java Language, The Key Attributes of Object-Oriented Programming, TheJava Development Kit, A First Simple Program, Handling Syntax Errors, The Java Keywords, Identifies in Java, The Java Class Libraries. **Introducing Data Types and Operators** Java's Primitive Types, Literals, A Closer Look at Variables, The Scope and Lifetime of Variables, operators, Shorthand Assignments, Type conversion in Assignments, Using Cast, Operator Precedence, Expressions. **Program Control Statements** Input characters from the Keyword, if statement, Nested ifs, if-else-if Ladder, Switch Statement, Nested switch statements, for Loop, Enhanced for Loop, While Loop, do-while Loop, Use break, Use continue, Nested Loops. **Introducing Classes, Objects and Methods** Class Fundamentals, How Objects are Created, Reference Variables and Assignment, Methods, Returning from a Method, Returning Value, Using Parameters, Constructors, Parameterized Constructors, The new operator Revisited, Garbage Collection and Finalizers, The this Keyword. **More Data Types and Operators** Arrays, Multidimensional Arrays, Alternative Array Declaration Syntax, Assigning Array References, Using the Length Member, The For-Each Style for Loop, Strings, The Bitwise operators.	**10 Hours**
String Handling	

String Fundamentals, The String Constructors, Three String-Related Language Features, The Length() Method, Obtaining the characters within a string, String comparison, using indexOf() and lastIndexOf(), Changing the case of characters within a string, StringBuffer and String Builder.	
Module -2	
A Closer Look at Methods and Classes Controlling Access to Class Members, Pass Objects to Methods, How Arguments are passed, Returning Objects, Method Overloading, Overloading Constructors, Recursion, Understanding Static, Introducing Nested and Inner Classes, Varargs: Variable-Length Arguments. **Inheritance** Inheritance Basics, Member Access and Inheritance, Constructors and Inheritance, Using super to Call Superclass constructors, Using super to Access Superclass Members, Creating a Multilevel Hierarchy, When are Constructors Executed, Superclass References and Subclass Objects, Method Overriding, Overridden Methods support polymorphism, Why Overridden Methods, Using Abstract Classes, Using final, The Object Class.	**10 Hours**
Module -3	
Interfaces Interface Fundamentals, Creating an Interface, Implementing an Interface, Using Interface References, Implementing Multiple Interfaces, Constants in Interfaces, Interfaces can be extended, Nested Interfaces, Final Thoughts on Interfaces. **Packages** Package Fundamentals, Packages and Member Access, Importing Packages, Static Import **Exception Handling** The Exception Hierarchy, Exception Handling Fundamentals, The Consequences of an Uncaught Exception, Exceptions Enable you to handle errors gracefully, using Multiple catch clauses, Catching subclass Exceptions, try blocks can be nested, Throwing an Exception, A Closer look at Throwable, using finally, using throws, Java's Built-in Exceptions, New Exception features added by JDK 7, Creating Exception Subclasses.	**10 Hours**
Module -4	
Multithreaded Programming Multithreading fundamentals, The Thread Class and Runnable Interface, Creating Thread, Creating Multiple Threads, Determining When a Thread Ends, Thread Priorities, Synchronization, using Synchronization Methods,The Synchronized Statement, Thread Communication using notify(), wait() and notify All(), suspending, Resuming and stopping Threads.	**10 Hours**

Enumerations, Auto boxing and Annotations Enumerations, Java Enumeration are class types, The Values () and Valueof () Methods, Constructors, methods, instance variables and enumerations, Auto boxing, Annotations (metadata) **Generics** Generics Fundamentals Bounded Types, Generic Methods, Generic Constructors, Some Generic Restrictions.	
Module -5	
Applets Applet basics, A complete Applet Skeleton, Applet Initialization and Termination, A key Aspect of an Applet Architecture, Requesting Repainting, using the status window, Passing parameters to Applets. **Swing Fundamentals** The origin and Design philosophy of swing, Components and containers, Layout managers, A first simple swing Example, Event Handling, Exploring Swing Controls-JLabel and ImageIcon, The Swing Buttons, Trees. **Networking with Java.net** Networking fundamentals, The Networking classes and Interfaces, The InetAddress class, The Socket Class, The URL class, The URLConnection Class, The HttpURL Connection Class. **Exploring Collection Framework** Collections Overview, The Collection Interfaces, The collection Classes. The Arrays Class.	**10 Hours**
Text Books: 1. Java Fundamentals, A comprehensive Introduction by Herbert Schildt, Dale Skrien. Tata McGraw Hill Edition 2013. (Chapters:1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,17,18,22,23,24,25,26)	
Reference Books: 1. Programming with Java by T V Suresh Kumar, B Eshwara Reddy and P Raghavan, Sanguine Technical Publishers, 2011. 2. Programming in JAVA2 by Dr K Somasundaram ,Jaico publications 3. Java Programming by Hari Mohan Pandey, Pearson Education, 2012. 4 Java 6 Programming, Black Book, KoGenT ,dreamtech Press, 2012. 5. Java 2 Essentials, Cay Hortsmann, second edition, Wiley	

Analysis and Design of Algorithms

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

SEMESTER –III

Subject Code	16MCA33	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

CO1: Categorize problems based on their characteristics and practical importance.

CO2: Develop Algorithms using iterative/recursive approach

CO3: Compute the efficiency of algorithms in terms of asymptotic notations

CO4: Design algorithm using an appropriate design paradigm for solving a given problem

CO5: Classify problems as P, NP or NP Complete

CO6: Implement algorithms using various design strategies and determine their order ofgrowth.

Modules	**Teaching Hours**
Module -1	
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, Examples. Selection Sort and Bubble Sort, Sequential Search and String Matching.	**10 Hours**
Module -2	
Divide-and-Conquer Mergesort, Quicksort, Binary Search, Binary tree Traversals and related properties, Multiplication of large integers, Stressen's Matrix Multiplication.	**10 Hours**
Module -3	
Decrease-and-Conquer Insertion Sort, Depth First and Breadth First Search, Topological sorting, Algorithms for Generating Combinatorial Objects **Space and Time Tradeoffs** Sorting by Counting, Input Enhancement in String Matching, Hashing.	**10 Hours**

Module -4	
Dynamic Programming Computing a binomial coefficient, Warshall's and Floyd's Algorithms, The Knapsack Problem and Memory Functions **Greedy Technique** Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Huffmann Trees.	**10 Hours**
Module -5	
Coping with Limitations of Algorithm Power Backtracking: n-Queens problem, Hamiltonian Circuit Problem, Subset – Sum Problem. Branch-and-Bound: Assignment Problem, Knapsack Problem, Traveling Salesperson Problem, Lower-Bound Arguments, Decision Trees, P, NP and NP-Complete Problems.	**10 Hours**

Text Books:

1. Anany Levitin: Introduction to the Design and Analysis of Algorithms,Pearson Education, 2nd Edition.(Chapters 1.1-1.4, 2.1-2.5, 3.1, 3.2, 3.4, 4.1-4.5, 5.1-5.4, 7.1-7.3, 8.1, 8.2, 8.4, 9.1-9.4, 11.1-11.3, 12.1-12.3)

Reference Books:

2. Coremen T.H., Leiserson C.E., and Rivest R.L.: Introduction to Algorithms, PHI 1998.
3. Horowitz E., Sahani S., Rajasekharan S.: Computer Algorithms, Galgotia Publication 2001.
4. Michael T Goodrich and Roberto Tamassia : Algorithm Design, Wiley India
5. R C T Lee, S S Tseng, R C Chang, Y T Tsai : Introduction to Design and Analysis of Algorithms: A Strategic Approach, Tata McGraw Hill

Software Engineering

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

Subject Code	16MCA34	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

CO1: Categorize problems based on their characteristics and practical importance.

CO2: Develop Algorithms using iterative/recursive approach

CO3: Compute the efficiency of algorithms in terms of asymptotic notations

CO4: Design algorithm using an appropriate design paradigm for solving a given problem

CO5: Classify problems as P, NP or NP Complete

CO6: Implement algorithms using various design strategies and determine their order ofgrowth.

Modules	**Teaching Hours**
Module -1	
Overview Introduction: Professional Software Development Attributes of good software, software engineering diversity, IEEE/ ACM code of software engineering ethics, case studies **Software Process & Agile Software Development** Software Process models: waterfall, incremental development, reuses oriented, Process activities; Coping with change, The rational Unified process. Agile methods, Plan-driven and agile Development, Extreme Programming, Agile project management, Scaling agile methods.	**10 Hours**
Module -2	
Requirements Engineering Functional and non-functional requirements, The software requirements document, Requirements specification, Requirements engineering processes, Requirement elicitation and analysis, Requirements validation, Requirements management **Component-based software engineering** Components and component model, CBSE process, Component composition.	**10 Hours**

Module -3	
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, Documenting architectural design	**10 Hours**
Module -4	
Design and implementation Design: Design concepts, Function oriented design, detailed design, verification, matrix (Complexity matrix for function oriented design) **Distributed Software engineering** Distributed system issues, Client-server computing, Architectural patterns for distributed systems, Software as a service.	**10 Hours**
Module -5	
Planning a software Project Process planning, Effort estimation, Project scheduling and staffing, Software configuration management plan, Quality plan, Risk Management, Project monitoring plan. **Software Testing** Testing fundamentals, Black-box testing, White-box testing, Testing process	**10 Hours**
Text Books: 1. Ian Sommerville : Software Engineering, 9th edition, Person Education Ltd, 2011. (Chapters-: 1, 2, 3, 4, 5, 17, 18) 2. Pankaj Jalote: Software Engineering, Wiley India Pvt Ltd (2010) (Chapters-:4, 6.1, 6.2, 6.5, 6.6)	
Reference Books: 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	

Computer Networks Laboratory

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

SEMESTER –III

Laboratory Code	16MCA36	CIE Marks	20
Number of Lecture Hours/Week	01Hr Tutorial Instructions) 02 Hours Laboratory	SEE Marks	80
		SEE Hours	03

CREDITS – 02

Course outcomes:

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

CO1: Describe basic terminologies used for computer networking.

CO2: Describe the functions of layers in the Internet Model.

CO3: Demonstrate application layer protocols used for process to Process Communication.

CO4: Demonstrate sub netting and routing mechanisms for a given network topology. Exemplify link layer functionalities.

CO5: Describe the components and working of wireless networks.

CO6: Implement network topologies, routing, error detection and correction techniques, protocol packet analysis using open source tools such as Packet Tracer, Wire Shark etc...

1. *Write a TCL script to simulate the network described below*
Consider a small network with five nodes n0, n1, n2, n3, n4, forming a star topology. The node n4 is at the center. Node n0 is a TCP source, which transmits packets to node n3 (a TCP sink) through the node n4. Node n1 is another traffic source, and sends UDP packets to node n2 through n4. The duration of the simulation time is 10 seconds.
Write a Tcl script (in the text area below) to simulate this scenario. Once you have written your simulation script, click on the "Run" button to execute it. You can verify your code by clicking on the "View Solution" button at the bottom of the page.

2. Write a TCL script to simulate a file transfer with ns2 Consider a client and a server. The server is running a FTP application (over TCP). The client sends a request to download a file of size 10 MB from the server. Write a script to simulate this scenario. Let node #0 be the server and node #1 be the client. TCP packet size is 1500 B. Assume typical values for other parameters.
Note: This simulation require transfer of a fixed size file. Therefore, time required for the transfer would be constant for a given bandwidth of a link. To verify this, determine the time that would roughly be required for the transfer. Then look at the bottom of the trace file and verify whether there is any transmission beyond the time calculated.
To verify that the client has downloaded the entire file, plot the "Bytes Received " curve for node #1. The y-axis is in Kbits. Convert it to MB and verify whether it approximates the specified file size. TCP headers would effectively increase the count of received bytes at node # 1.

***3**.Setting up a local area network with ns2*

In this exercise you will be simulating a CSMA/CD based LAN with ns2. Consider the LAN
with seven nodes to be an isolated one i.e. not connected to the Internet. Node # 0 in the LAN act as a UDP traffic source, and node # 6 is the destination node. Assume CBR traffic to be flowing between the nodes. The simulation lasts for 25 seconds.

In Ethernet a packet is broadcasted in the shared medium, and only the destination node accepts the packet. Other nodes simply drop it. What should be the number of hops a packet from node # 0 to node # 6 travel? Verify this from the "Hop Count" plot.

Additional Task:

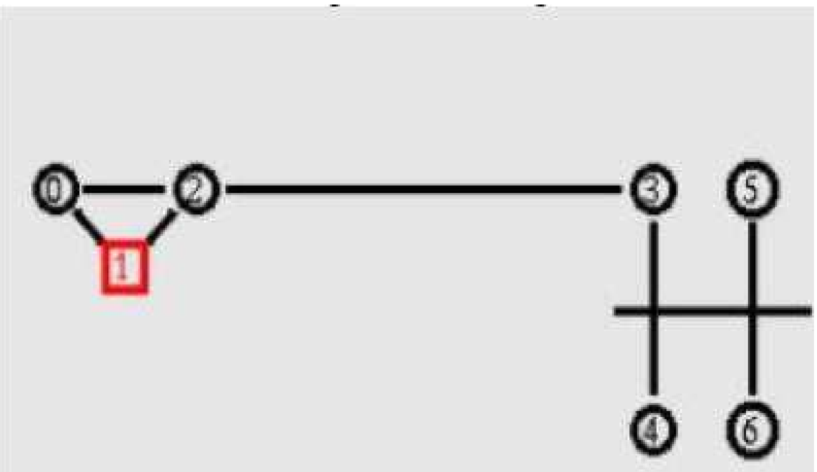
Suppose the above LAN is to be connected to the Internet. Add node # 7 into the network so that it act as the gateway. Connect node # 0 and # 7 with a 1 Mb wired link. Move the UDP source to node # 7. How the hop count should get affected in this case? Verify from the plot.

4. Simulating link errors

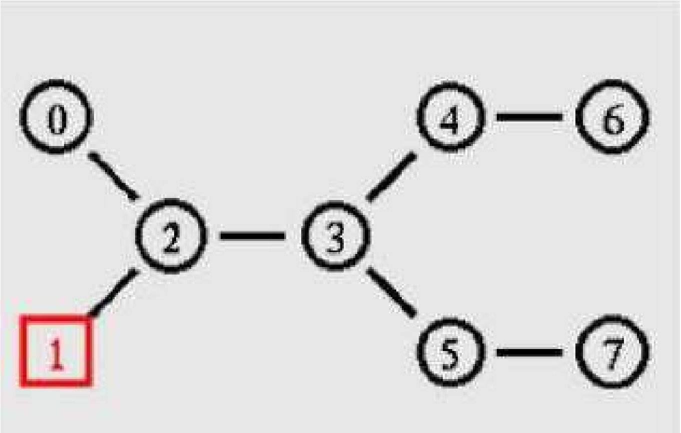
Consider the following network diagram

Here node # 2 act as a router. Any traffic to or from the LAN passes through it. Consider node # 1 running a FTP server, and node # 5 is downloading a file of size 4 MB. However, the link
between node # 2 and # 3 is fault. It drops packets with a fixed probability of 0.2. Implement a link error model to reflect this.

It may be noted here that the file download time will be more than the we had in exercise # 2 of experiment # 1. Try different values of the simulation time to ensure that the file has been entirely transferred. Is the plot of bytes received a linear curve or non-linear? Why?



Presence of link errors cause one or more packets to be retransmitted. Verify this from the "Packet Retransmissions" plot.



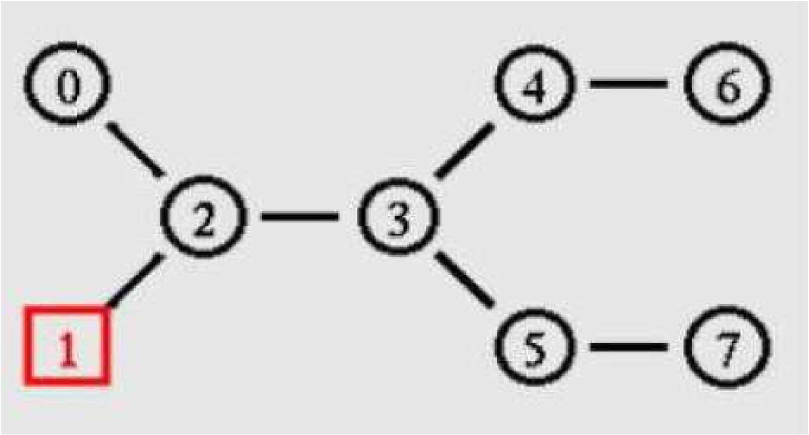
Assume node # 0 running a FTP application (over TCP) and sending data to node # 6. Node # 1 is sending CBR data node # 7. Assume all the links except 2-3 has a bandwidth of 1 Mb, propagation delay of 10ms and queue type as DropTail. (All are duplex links).
Tasks:

- *The link 2-3 has a propagation delay of 10 ms. Vary it's bandwidth from 0.5 Mb to 2.5 Mb in steps of 0.25Mb.*
- *Compute the throughput for node # 3 in each case*
- *Plot the throughput vs. bandwidth data in the "Custom Plot" section below*

Based on the above plots, suggest what should be the recommended bandwidth of the link 2-3.
Now, plot the end-to-end delay between nodes 0 and 6 for the above chosen values of link 2-3 bandwidth. Revisit your previous answer (i.e. optimum bandwidth of link 2-3) based on these graphs. Measuring Network Performances

6. Bandwidth sharing between TCP and UDP

Consider the dumbbell topology from our previous exercise:



Node # 0 is a TCP source, and the corresponding sink is at node # 6. Node # 1 is a UDP source (CBR traffic) with a null agent attached to node # 7. These two traffic flows *through the common link 2-3. The aim of this exercise is to examine how TCP and UDP share the bandwidth between themselves when the rate of CBR traffic is changed.*

Set the TCP packet size to 1460 B. The UDP and CBR packet sizes are 1500 B. All the links in the network have same bandwidths (say, 4 Mb), delay and queue types.

Part 1:

- *Set the initial rate of CBR traffic to 0.5 Mb. Run the simulation, and plot the "Bytes Received" by node #s 4 and 5 (sinks for TCP and UDP traffic)*
- *Now, increment the rate up to 4 Mb, the link bandwidth, in steps of 0.5 Mb. Run the simulation and plot the graphs again.*

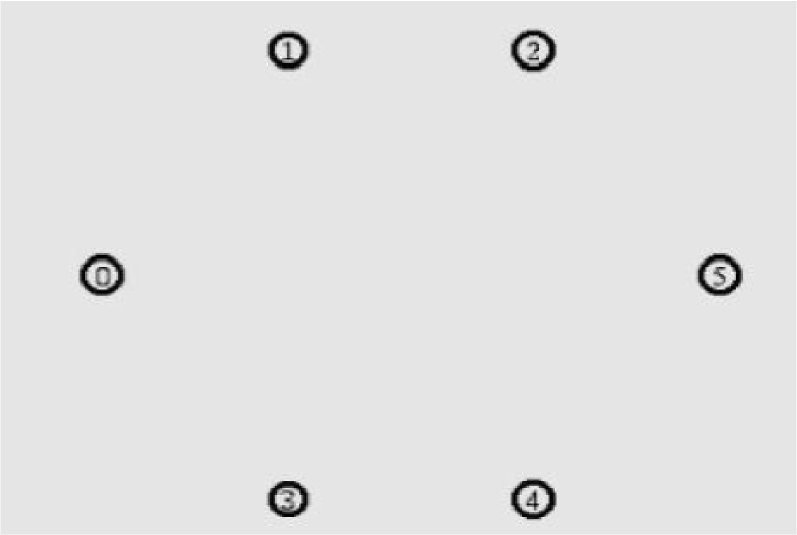
How does the graphs change after each run? In particular, what's the nature of the graphs when the rate of CBR traffic is 50% of the bandwidth?

Part 2: Behaviour of UDP

- *Reduce the bandwidth of the link 2-3 to say, 2 Mb. Repeat the above steps and observe the graphs in this case. From the graphs plotted observe how UDP occupies a larger portion of the bandwidth. How does thebehaviour change for other variations of TCP (Newreno, Vegas)?*

7. Write a TCL script to simulate the following scenario with ns2 simulator.

Consider six nodes, (as shown in the figure below) moving within a flat topology of 700m x 700m.



The initial positions of nodes are 0 (150,300) ,1 (300,500),2 (500,500),3 (300,100),4(500,100) and 5(650,300) respectively.A TCP connection is initiated between node 0 (source) and node 5 (destination) through node 3 and node 4 i.e the route is 0-3-4-5.

At time t = 3 seconds the FTP application runs over it.
After time t=4.0 sec, node 3 (300,100) moves towards node 1 (300,500) with a speed of 5.0m/sec
and after some time the path break, then the data transmit with a new path via node 1 and node 2 i.e the new route 0-1-2-5.
The simulation lasts for 60 secs. In the above said case both the route has equal cost. Use DSR as the routing protocol and the IEEE 802.11 MAC protocol.
Now Analyze the trace file and determine when the use of second route commence, and Plot the number of packets received by each node over the entire time duration of the simulation

8. Simulate a wired network and demonstrate Distance Vector Routing algorithm .

***9. Simulate a network which will create congestion in the network**. With the trace file created identfy the points at which congestion occurs by writing sed / awk scripts. Also write a mechanism to correct/control the congestion.*
***Note :**Network Free and open source software simulators like NS2 / NS3 could be used. If NS2 is used tcl scripting should be introduced . If NS3 is used c++ with python has to be introduced during first two or three weeks of the labs. Only above 9 experiments should be included for the laboratory exam.*

Note: In the examination each student should do one question out of the above 9 questions

Java Programming Laboratory

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

SEMESTER –III

Laboratory Code	16MCA37	CIE Marks	20
Number of Lecture Hours/Week	01Hr Tutorial Instructions) 02 Hours Laboratory	SEE Marks	80
		SEE Hours	03

CREDITS – 02

Course outcomes:

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

CO1: Students will get the idea of java language fundamentals and run time environment

CO2: Give broad knowledge and skill necessary to write a program in java programs

CO3: Student will be able to learn the object oriented concepts and its Implementation in Java

CO4: Student should able to solve the run time problems encountered during the excution

CO5: Students will learn the implementation of multithreading and client side providing

Provides the basics GUI using swings and network programming

1. *a) Write a JAVA Program to demonstrate Constructor Overloading and Method Overloading.*
   b) Write a JAVA Program to implement Inner class and demonstrate its Access protection.

2. *Write 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). Write 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. *Write 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 throwsLess Balance Exception if an account holder tries to withdraw money which makes thebalance 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 towithdraw more money which generates a LessBalanceException takeappropriate actionfor the same.

5. Write a JAVA program using Synchronized Threads, which demonstrates Producer Consumer concept.

6. Write 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.

8. Write a JAVA Program to create an enumeration Day of Week with seven values SUNDAY through SATURDAY. Add a method isWorkday( ) 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. Write 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 Stackoperations that demonstrates the runtime binding.

10. Write a JAVA program to print a chessboard pattern

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

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

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

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

Note: In the examination each student should do one question out of the above 14 questions

Algorithms Laboratory

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

SEMESTER –III

Laboratory Code	16MCA38	CIE Marks	20
Number of Lecture Hours/Week	01Hr Tutorial Instructions) 02 Hours Laboratory	SEE Marks	80
		SEE Hours	03

CREDITS – 02

Course outcomes:

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

CO1: Explain the basic concepts of time and space complexity, divide-and-conquer strategy, dynamic Programme greedy and approximate algorithms.

CO2: Describe the methodologies of how to analyze an algorithm

CO3: Describe the data structures of red-black tree, B-tree, heap and disjoint sets.

CO4: Design a better algorithm to solve the problems

1 *Implement Recursive Binary search and Linear search and determine the time required to search an element. Repeat the experiment for different values of n, the number of elements in the list to be searched and plot a graph of the time taken versus n.*

2 *Sort a given set of elements using the Heapsort method and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n.*

3 *Sort a given set of elements using Merge sort method and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n.*

4 *Obtain the Topological ordering of vertices in a given graph.*

5 *Implement 0/1 Knapsack problem using dynamic programming.*

6 *From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.*

7 *Sort a given set of elements using Quick sort method and determine the time required sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n.*

8 *Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal's algorithm.*

9 *Print all the nodes reachable from a given starting node in a digraph using BFS method.*

10 *Check whether a given graph is connected or not using DFS method.*

11 *Find a subset of a given set S = {s1,s2,.....,sn} of n positive integers whose sum is equal to a given positive integer d. For example, if S= {1, 2, 5, 6, 8} and d = 9 there are two solutions{1,2,6}and{1,8}.A suitable message is to be displayed if the given problem instance doesn't have a solution.*

12	*a). Implement Horspool algorithm for String Matching.* *b). Find the Binomial Co-efficient using Dynamic Programming.*
13	*Find Minimum Cost Spanning Tree of a given undirected graph using Prim's algorithm.*
14	*a).Implement Floyd's algorithm for the All-Pairs- Shortest-Paths Problem.* *b).Compute the transitive closure of a given directed graph using Warshall's algorithm.*
15	*Implement N Queen's problem using Back Tracking.*

Note: In the examination questions must be given based on above lot.

Intellectual Property Rights

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

SEMESTER –III

Subject Code	16MCA351	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

CO1:dentify which of the four main different types of intellectual property rights may be presented by an output

CO2:analyse an innovative or creative output in terms of intellectual property rights generated

CO3:discuss the appropriateness, or not, of registering an intellectual property right

CO4:apply the appropriate ownership rules to intellectual property he / she has been involved in creating

CO5:suggest ways of exploiting intellectual property rights created in his / her own work.

Modules	**Teaching Hours**
Module -1	
Introduction to Intellectual Property Law – The Evolutionary Past - The IPR Tool Kit- Para -Legal Tasks in Intellectual Property Law – Ethical obligations in Para Legal Tasks in Intellectual Property Law - Introduction to Cyber Law – Innovations and Inventions Trade related Intellectual Property Right..	**10 Hours**
Module -2	
Introduction to Trade mark – Trade mark Registration Process – Post registration Procedures – Trade mark maintenance - Transfer of Rights - Inter parties Proceeding – Infringement - Dilution Ownership of Trade mark – Likelihood of confusion - Trademarks claims – Trademarks Litigations – International Trade mark Law	**10 Hours**
Module -3	
Introduction to Copyrights – – Principles of Copyright Principles -The subjects Matter of Copy right – The Rights Afforded by Copyright Law – Copy right Ownership, Transfer and duration – Right to prepare Derivative works – Rights of Distribution – Rights of Perform the work Publicity Copyright Formalities and Registrations - Limitations - Copyright disputes and International Copyright Law – Semiconductor Chip Protection Act	**10 Hours**

Module -4	
The law of patents-patent searches –Patent owner shp and transfer-Patent infringement-International Patent Law	**10 Hours**
Module -5	
Introduction to Trade Secret – Maintaining Trade Secret – Physical Secrurity – Employee Limitation - Employee confidentiality agreement - Trade Secret Law - Unfair Competation – Trade Secret Letigation – Breach of Contract – Applying State Law	

Text Books:

1 DebiragE.Bouchoux: “Intellectual Property”. Cengage learning, New Delhi

2. M.Ashok Kumar and Mohd.Iqbal Ali: “Intellectual Property Right” Serials Pub.

3. Cyber Law. Texts & Cases, South-Western’s Special Topics Collections

Reference Books:

1. PrabhuddhaGanguli: ‘ Intellectual Property Rights” Tata Mc-Graw –Hill, New Delhi
2. J Martin and C Turner “Intellectual Property” CRC Press

Richard Stimm “ Intellectual Property” Cengage Learning

Advanced DBMS

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

Subject Code	16MCA352	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

CO1: Describe database management system storage structure and the concept of transaction mechanism.

CO2: Identify and be able to use recent and advanced database techniques (e.g. in concurrency control, buffer management, and recovery).

CO3: Decide on configuration issues related to database access mechanism and query evaluation.

CO4: Analyze and optimize relational operators and query optimization.

CO5: Decide on database design and work with latest applications

Modules	**Teaching Hours**
Module -1	
Over view 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 formats. Introduction to Transaction Processing; Transaction and System Concepts; Desirable Properties of Transactions	**10 Hours**
Module -2	
Characterizing Schedules based on Recoverability; Characterizing Schedules based on Serializability; Two-Phase Locking Techniques for Concurrency Control; Concurrency Control based on Timestamp Ordering; Multiversion Concurrency Control Techniques; Validation Concurrency Control Techniques;. Granularity of Data Items and Multiple Granularity Locking; Recovery Concepts, Recovery Techniques based on Deferred Update; Recovery Techniques based on Immediate Update; Shadow Paging; The ARIES Recovery Algorithms; Recovery in Multidatabase Systems; Database Backup and Recovery from Catastrophic Failures. Hash-Based indexing Static hashing, Extendible hashing, Linear hashing, comparisons	**10 Hours**

Module -3	
Tree Structured Indexing Intuition for tree indexes; Indexed sequential access method; B+trees, Search, Insert, Delete, Duplicates, B+tress in practice **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	**10 Hours**
Module -4	
Evaluating Relational Operators The Selection operation; General selection conditions; The Projection operation; The Join operation; The Set operations; Aggregate operations; The impact of buffering. **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.	**10 Hours**
Module -5	
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. **More Recent Applications** Mobile databases; Multimedia databases; Geographical Information Systems; Genome data management.	
Text Books: 1. Raghu Ramakrishnan and Johannes Gehrke: Database Management Systems,3rd Edition, McGraw-Hill,2003, (Chapters 8,9,10,11,12,13.1 to 13.3, 14,15,20) 2. Elmasri and Navathe:Fundamentals of Database Systems,5th Edition, Pearson Education, 2007. (Chapter 30)	
Reference Books: 1. Conolly and Begg: Database Systems,4th Edition, Pearson Education,2002.	

MIS & E-Commerce [As per Choice Based Credit System (CBCS) scheme] SEMESTER –III			
Subject Code	16MCA353	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03
CREDITS – 04			

Course outcomes:

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

CO1: Recognize the roles and uses of technology in business systems, operations and describe organizational structure &business processes.

CO2:apply the knowledge in the development of a suitable electronic file storage system for a business

CO3:demonstrate an understanding of the process in systems design, development and describe several generic business applications of data communications technology

CO4:define, analyze the principles of E-commerce, basics of World Wide Weband analyzing the concept of electronic data interchange, its legal, social and technical aspects.

CO5:identify the fundamental principles of e-commerce and its applications in business like E-banking, electronic payment system, the security issues over the web, the available solutions and future aspects of e-commerce security.

Modules	**Teaching Hours**
Module -1	
Information and Knowledge Information concepts, classification of information, methods of data and information collection, value of information, information: A quality product, General model of a human as information processor, Knowledge, **Introduction of MIS** MIS: Concept, Definition, Role of the MIS, Impact of MIS, MIS and the user, Management as a control system, MIS support to the management, Management effectiveness and MIS, Organization as system. MIS: organization effectiveness **Decision Making and DSS** 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	**10 Hours**

Module -2	
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) **Client Server Architecture and E-business Technology** Client server architecture, implementation strategies, Introduction to E-business, model of E-business, internet and World Wide Web, Intranet/Extranet, Electronic, Impact of Web on Strategic management, Web enabled business management, MIS in Web environment.	**10 Hours**
Module -3	
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).	**10 Hours**
Module -4	
E-Commerce techniques and Issues Introduction to Active Server Pages (ASP), 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	**10 Hours**
Module -5	
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, Security and Encryption, Web Security.	
Text Books: 1. Waman S Jhawadekar: Management Information System, 3rd Edition, Tata McGraw Hill. 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 4. H. M. Deitel, P. J. Deitel and T. R. Nieto, E-Business and E-Commerce: How to Programe, Prentice hall, 2001	

Reference Books:

1. Henry Chan, Raymond Lee, Tharam Dillon, Elizabeth Chang: E-Commerce Fundamentals and Applications, Wile India Edition
2. Ralph M Stair and George W Reynolds: Principles of Information Systems, 7th Edition, Thomson, 2010 .
3. Steven Alter: Information Systems - The Foundation of E-Business, 4th Edition, Pearson Education, 2001
4. Rahul De, Managing Information Systems in Business, Government and Society, , Wiley India.

Operations Research

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

SEMESTER –III

Subject Code	16MCA354	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

CO1: Formulate linear programming problems (LPP), given unambiguous problem
Statement.

CO2: Solve two-variable LPP graphically and carry out sensitivity analysis on the requiredinput parameter graphically.

CO3: Solve general LP problems using simplex or dual simplex techniques

CO4: Solve general LP problems using Apply transportation and assignment algorithms to obtain the optimum solution

CO5:apply PERT and CPM techniques for project management

CO6: Formulate and solve simple games to resolve competitive situations

Modules	**Teaching Hours**
Module -1	
Information and Knowledge **Introduction and Overview of the OR Modeling Approach** The origin of OR, the nature of OR, the impact of OR, defining the problem and gathering data, Formulating a mathematical model, deriving solutions from the model, testing the model, preparing to apply the model, implementation. **Introduction to Linear Programming** Formulation of linear programming problem (LPP), examples, Graphical solution, the LP Model, Special cases of Graphical method, assumptions of Linear Programming (LP), additional example.	**10 Hours**
Module -2	
Solving LPP - the Simplex Method The essence of the simplex method, setting up the simplex method, algebra of the simplex method, the simplex method in tabular form, special cases in the simplex method, tie breaking in the simplex method, adopting to other model forms (Two Phase method, Big-M method), post optimality analysis	**10 Hours**

Module -3	
Duality Theory and Sensitivity Analysis The essence of duality theory, economic interpretation of duality, primal dualrelationship, adapting to other primal forms, the role of duality in sensitive analysis, the dual simplex method	**10 Hours**
Module -4	
Transportation and Assignment Problems The transportation problem, a stream line simplex method for the transportation problem, the assignment problem, a special algorithm for the assignment problem **PERT and CPM** Network representation, Critical path (CPM) computations and PERT networks.	**10 Hours**
Module -5	
Game Theory The formulation of two persons, zero sum games, solving simple games- a prototype example, games with mixed strategies, graphical solution procedure, solving by linear programming, extensions	**10 Hours**
Text Books: 1 Frederick S.Hillier& Gerald J.Lieberman: Introduction to Operations Research, 8thEdition, Tata McGraw Hill, 2006. (Chapters 1.1-1.3, 2.1-2.6,3.2-3.4, 4.1-4.7, 6.1-6.7, 7.1, 8.1-8.4, 13.1-3.4, 14.1-14.6) 2. Hamdy A Taha: Operations Research - An Introduction, 7th Edition, Pearson Education 2007 Chapter 6.6(except 6.6.3 and 6.6.4)	
Reference Books: 1. Wayne L. Winston: Operations Research Applications and Algorithms, 4th Edition, Thomson Course Technology, 2003. 2. Theory and Problems of Operations Research, Richard Bronson and 51 GovindasamiNaadimuthu, Schaum's Outline, Tata McGraw Hill, 2nd Edition, 1997.	

Principles of User Interface Design

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

Subject Code	16MCA355	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

CO1:familiarizethe new technologies that provide interactive devices and interfaces.

CO2:develop the processes and evaluate UID.

CO3:understand Direct Manupulation and Virtual Environment

CO4:discuss the command, natural languages and issues in design for maintaining QoS

CO5:persuade user documentations and information search.

Modules	**Teaching Hours**
Module -1	
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.	**10 Hours**
Module -2	
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. **Evaluating Interface Design** Introduction, Expert Reviews, Usability Testing and Laboratories, Survey Instruments, Acceptance tests, Evaluation during Active Use, Controlled Psychologically Oriented Experiments	**10 Hours**
Module -3	
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	**10 Hours**

Module -4	
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 **Design Issues** Quality of Service: Introduction, Models of Response-Time Impacts, 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	**10 Hours**
Module -5	
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. **Information Search and Visualization** Introduction, Search in Textual Documents and Database Querying, Multimedia document searches, Advanced filtering and Search Interfaces, Information Visualization : Introduction, Data tyoe by task taxonomy, Challenges for information visualization.	**10 Hours**

Text Books:

1. Ben Shneiderman, Plaisant, Cohen, Jacobs: Designing the User Interface, 5th Edition, Pearson ,Education, 2010.

Reference Books:

1 Alan Dix, Janet Finalay, Gregory D AbiwdmRusselBealel: Human-Computer Interaction, III Edition, Pearson , Education, 2008.

2 Eberts: User Interface Design, Prentice Hall, 1994

3 Wilber O Galitz: The Essential Guide to User Interface Design- An Introduction to GUI Design, Principles and Techniques, Wiley-Dreamtech India Pvt Ltd, 2011

Human Resource Management

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

SEMESTER –III

Subject Code	16MCA356	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

CO1: After studying this course, students will be able to:

CO2:identify the primary external influences affecting HRM.

CO3:outline the components and the goals of staffing, training and development.

CO4:understand the selection procedure in various national and international organizations.

CO5:understand the practices used to retain the employees and able to evaluate their performance.

CO6:identify the stress and the cause of burn out.

Modules	**Teaching Hours**
Module -1	
Human Resource Management: Introduction, Meaning, Nature, Scope of HRM, Definitions, Importance of HRM, Objectives of HRM, Functions of HRM, Process of HRM, HR Organization Structure, Strategic HRM, Role of HRM in Strategic Management, Recent Trends in HRM. **Human Resource Planning:** Introduction, Meaning-HRP, Definitions, Need for HRP, Importance of HRP, Factors influencing HRP, Benefits of HRP, HRP components, Process of HRP, Steps in HRP.	**10 Hours**
Module -2	
Job Analysis: Meaning, Need for Job Analysis, Concepts in Job Analysis, Scope of Job Analysis, Objectives of Job Analysis, Significance of Job Analysis, Process of Job Analysis, Techniques of Job Analysis, Job Description, Job Specification. **Job Design:** Meaning, Need for Job Design, Principles of Job Design, Objectives of Job Design, Features of good Job Design, Approaches to Job Design, Job Rotation, Job Enlargement, Job Enrichment, Advantages of job Design, Limitation of Job Design.	**10 Hours**

Module -3	
Recruitment: Meaning, Need for Recruitment, Factors determining Recruitment, Recruitment Constraints, Recruitment Process, Recruitment Policy, Sources of Recruitment, Recruitment Methods, Techniques of Recruitment, Attributes to Recruitment. **Selection:** Introduction, Meaning, Importance, Selection Stages, Types of Test, Types of Interviews **Placement:** Meaning, Effective placements, Problems in Placements, Induction and Orientation.	**10 Hours**
Module -4	
Training and Development: Introduction, Importance of Training, Reasons for Training, Objectives of training, Identification of Training needs, Training Policy, Types of Training, Types of Training Programs, Training and Development Methods, Characteristics of good Training Programs, Training Stages, Benefits of Training, Evaluation of Training.	**10 Hours**
Module -5	
Performance Appraisal: Introduction, Meaning, Need for performance Appraisal, Objectives of Performance Appraisal, Performance Appraisal Rating Criteria, Evaluation Stages, Appraisal Methods, Requisites of effective Performance Appraisal, latest Appraisal Techniques, Limitations of Performance Appraisal. **Compensation:** Objectives of Compensation Planning, Job Evaluation, Compensation Pay Structure in India, Wage and Salary Administration, Factors Influencing Compensation Levels, Executive Compensation, International Compensation. **International Human Resource Management:** Introduction, Meaning, Definition, Role of Culture in IHRM, Impact of Country Culture in IHRM, Multinational Staffing Decision, Types of International Assignments, Successful Expatriation, Human Resource Information System (HRIS	**10 Hours**
Text Books: 5. Human Resource Management – Rao V. S. P, Excel BOOKS, 2010 6. Human Resources Management: A South Asian Perspective, Snell, Bohlander, & Vohra, Cengage Learning, 16th Rep., 2012. 7. Human Resource Management - Lawrence S. Kleeman, Biztantra , 2012. Human Resource Management – Aswathappa K HPH	
Reference Books: 5. Human Resource Management - John M. Ivancevich, 10/e, McGraw Hill. 6. Human Resource Management in practice - Srinivas R. Kandula, PHI, 2009 7. P.Subba Rao, Human Resource Management & Industrial relations, Himalaya Publishing House, Mumbai.	

ADVANCED JAVA PROGRAMMING
[As per Choice Based Credit System (CBCS) scheme]
SEMESTER – IV

Subject Code	16MCA41	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

CO1: Learn the concept of Servlet and its life cycle
CO2: Understand 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

Modules	**Teaching Hours**
Module -1	
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.	**10 Hours**
Module -2	
JSP and Controlling the Structure of generated servlets Overview of 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, Including files and applets in jsp Pages, using java beans components in JSP documents	**12 Hours**

Module -3	
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.	**07 Hours**
Module – 4	
Talking to Database, Immediate Solutions, Essential JDBC program, using prepared Statement Object, Interactive SQL tool. JDBC in Action Result sets, Batch updates, Mapping, Basic JDBC data types, Advanced JDBC data types, immediate solutions.	**09 Hours**
Module -5	
Introduction to EJB and Server Side Component Models The Problem domain, Breakup responsibilities, Code Smart not hard, the Enterprise java bean specification. Components Types. Server Side Component Types, Session Beans, MessageDriven 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. Models: The Stateless Session Bean, the Stateful Session Bean, the Singleton SessionBean, Message-Driven Beans. EJB and PERSISTENCE. Persistence Entity manager Mapping Persistence objects, Entity Relationships	**12 Hours**

Text Books:

1. Marty Hall, Larry Brown. Core Servlets and Java Server Pages. Volume 1: Core Technologies. 2ndEdition. (Chapter 3,4,5,6,7,8,9,10,11,12,13,14).
2. Java 6 Programming Black Book, Dreamtech Press. 2012 (Chapter 17,18,19,20,21,22,27,28,29,30).
3. Andrew LeeRubinger, Bill Burke. Developing Enterprise Java Components. Enterprise JavaBeans 3.1.O'reilly. (Chapter 1,2,3,4,5,6,7,8,9,10,11).

Reference Books:

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.

Herbert Schildt, Java The Complete Reference, 8thEdition. Comprehensive coverage of the Java Language. Tata McGraw-Hill Edition – 2011.

ADVANCED WEB PROGRAMMING

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

SEMESTER – IV

Subject Code	16MCA42	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

CO1: Acquire knowledge of

CO2: Build the Web Applications using PERL, PHP and CGI.

CO3: Model-View-Controller (MVC) Architecture.

CO4: Design the Web Pages using Ruby, Rails and Layouts.

CO5: Apply the knowledge gained in the Building a web portals.

CO6: Evaluate web site performance against user acceptance testing.

Modules	**Teaching Hours**
Module -1	
Programming in Perl and CGI Scripting and Building Web Applications with Perl Origins and uses of Perl, Scalars and their operations, Assignment statements and simpleinput and output, Control statements, Fundamentals of arrays, Hashes, References, Fun What is CGI? Developing CGI Applications actions, Pattern matching, File input and output; Examples. CGI.pm methods, Example, Creating HTML Pages Dynamically, Using CGI. pm – An Example, Adding Robustness, libwww, Carp, Cookies, Uploading files, Tracking users with Hidden Data, Using Relational Databases	**10 Hours**
Module -2	
Introduction to PHP and Building Web applications with PHP Origins and uses of PHP, Overview of PHP, General syntactic characteristics, Primitives, operations and expressions, Output, Control statements, Arrays, Functions, Pattern matching, Form handling, Files, Tracking users, cookies, sessions, Using databases, Handling XML	**10 Hours**
Module -3	
Introduction to Ruby and Introduction to Rails Origins and uses of Ruby, Scalar types and their operations, Simple input and output, Control statements, Arrays, Hashes, Methods, Classes, Code blocks and iterates, Pattern matching. Overview of Rails, Document requests, Processing forms, Layouts. Rails applications with Databases.	**10 Hours**

Module – 4	
Introduction to web 2.0 and Web Services What is Web 2.0?, Folksonomies and Web 2.0, Software As a Service (SaaS), Data and Web 2.0, Convergence, Iterative development, Rich User experience, Multiple Delivery Channels, Social Networking. Web Services: SOAP, RPC Style SOAP, Document style SOAP. WSDL, REST services, JSON format, What is JSON? Array literals, Object literals, Mixing literals, JSON Syntax, JSON Encoding and Decoding, JSON versus XML	**10 Hours**
Module -5	
D3.js (Data Driven Documents): Data visualization tool for web apps Introduction to D3: Building a Simple Subway Train Status Board, Graphing Mean Daily Plaza Traffic. Scales, Axes, and Lines, Graphing Turnstile Traffic, Interaction and Transitions, Subway Connectivity, Scheduled Wait Time Distribution	**10 Hours**

Text Books:

1. **ChrisBates:** Web Programming Building Internet Applications. 3rd Edn,Wiley India,2013
2. **RobertW.Sebesta**:Programming the Worldwide Web, 4th Edition, Pearson Education, 2012
3. **Francis Shanahan:**Mashups,WileyIndia, 2012
4. **Mike Dewar**: "Getting Started with D3": O'Reilly Media, 2012

Reference Books:

M.Deitel,P.J.Deitel,A.B.Goldberg:Internet &Internet & World Wide Web How to program, 3rdEdition, Pearson Education/PHI, 2004

SOFTWARE TESTING AND PRACTICES

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

Subject Code	16MCA43	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

- Acquire knowledge of basic principles and knowledge of softwaretesting and debugging and test cases.
- Will be able to understand the perceptions on testing like levels of testing, generalized pseudo code and with related examples
- To study the various types of testing.
- Will be able to understand analyses the difference between functional testing and structural testing.

Analyzethe performance of fault based testing, planning and Monitoring the process, Documentation testing.

Modules	**Teaching Hours**
Module -1	
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; Test Generation from Predicates.Sensitivity, Redundancy, Restriction, Partition, Visibility and Feedback, Test Specification andcases, Adequacy Criteria, Comparing Criteria	**12 Hours**
Module -2	
A perspective on Testing, Examples Basic definitions, Test cases, Insights from a Venn diagram, Identifying test cases, Error and fault taxonomies, 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	**7 Hours**
Module -3	
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 testcases for triangle problem,	**7 Hours**

Next Date function and commission problem, Guidelines and observations, Decision tables, Test cases for triangle problem	
Module -4	
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, Guidelines and observations.	**12 Hours**
Module -5	
Fault Based Testing, Planning and Monitoring the Process, Documenting Analysis and Test 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.	**12 Hours**
Text Books: 1. AdithyaP.Mathur " Foundations of Software Testing – Fundamental Algorithms and Techniques", Pearson Education India, 2011 2. Mauro Pezze, Michael Young, Software testing and Analysis- Process, Principles andTechniques, Wiley India, 2012 3. Paul C Jorgensen, "Software Testing A Craftsman's Approach", Aueredachpublications, 3rd edition, 2011.	
Reference Books: 1. KshirasagaraNaik, PriyadarshiTripathy: Software Testing and Quality Assurance, Wiley India 2012 2. M.G.Limaye: Software Testing-Principels, Techniques and Tools – McGraw Hill, 2009	

NCED JAVA PROGRAMMING LABORATORY

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

Laboratory Code	16MCA46	CIE Marks	*20*
Number of Lecture Hours/Week	01Hr Instructions 02 Hrs Laboratory	SEE Marks	*80*
Total Number of Lecture Hours	42	SEE Hours	*3 Hrs*

CREDITS – 02

NOTE: In the examination each student should do one question out of the above 13 questions.

Course outcomes:

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

- Designing HTML pages to demonstrate Java Servlets, JSP, Bean and EJB programs.
- Implementing Dynamic HTML using Servlet and demonstration of sevices methods, auto web page refresh, Session tracking using cookie and Http Session in Servlet.
- Learn the fundamental of connecting to the database
- Demonstrate JSP (page attributes, action tags and all basic tags) and types of EJB application.

Program Statements

1. Write a 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. Write a 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. Write a JAVA Servlet Program to implement and demonstrate get() and Post methods(Using HTTP Servlet Class).
4. Write a JAVA Servlet Program using cookies to remember user preferences.
5. Write a JAVA Servlet program to track HttpSession by accepting user name and password using HTML and display the profile page on successful login.
6. Write a JAVA JSP Program which uses jsp:include and jsp:forward action to display a Webpage.
7. Write a JAVA JSP Program which uses <jsp:plugin> tag to run a applet
8. Write a JAVA JSP Program to get student information through a HTML and create a JAVA Bean class, populate Bean and display the same informationthrough another JSP

9.	Write a JSP program to implement all the attributes of page directive tag.
10.	Write 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 logic).
12.	An EJB application that demonstrates MDB (with appropriate business logic).
13.	An EJB application that demonstrates persistence (with appropriate business logic).

ADVANCED WEB PROGRAMMING LABORATORY

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

SEMESTER – IV

Laboratory Code	16MCA47	CIE Marks	*20*
Number of Lecture Hours/Week	01Hr Instructions 02 Hrs Laboratory	SEE Marks	*80*
Total Number of Lecture Hours	42	SEE Hours	*3 Hrs*

CREDITS – 02

NOTE:

1. In the examination, one exercise from part A is to asked for 5 marks
2. Mini project student group size is limited to two students only.
3. The mini project under part B has to be evaluated for 15 marks.
4. Project report duly signed by the Guide and HoD need to be submitted during examination.

Course outcomes:

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

- Understand, analyse and apply the role of languages like HTML, XHTML, CSS, XML, perl, PHP script.
- Build web application using PERL and store values in MYSQL.
- Analyse, build and consume web services.
  Analyse a web project and identify its elements and attributes In comparison to traditional projects.

PART – A
Server Side Web Programming

1.	Write a PERL program to insert name and age information entered by the user into a table created using MySQL and to display the current contents of this table.
2.	a) Write a PHP program to store current date-time in a COOKIE and display the‘Last visited on’ date-time on the web page upon reopening of the same page. b) Write a PHP program to store page views count in SESSION, to increment the count on each refresh, and to show the count on web page.
3.	Write a PHP script to **upload** and **download** a file into/from server folder.
4.	Create a XHTML form with Name, Address Line 1, Address Line 2, and E-mail text fields. On submitting, store the values in MySQL table. Retrieve and display the data based on Name.
5.	Write a PHP program to read student data from an XML file and store into the MySQL database. Retrieve and display.
6.	Create a XHTML form with Name, Address Line 1, Address Line 2, and E-mail text fields. On submitting, store the values in MySQL table. Provide buttons to update and delete data for the same.

7.	Build a Rails application to accept book information viz. accession number, title, authors, edition and publisher from a web page and store the information in a database and to search for a book with the title specified by the user and to display the search results with proper headings.
8.	Create a XHTML form to collect Viewers Opinion based on rating (1 to 5). Finally Display the survey report in terms of a bar-graph using D3-JS

PART B

Develop a web application project using thelanguages 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. In the examination each student picks one question from part A.
2. A team of two or three students must develop the mini project. However during the examination, each student must demonstrate the project individually.
3. The team must submit a brief project report (25-30 pages) that must include the following

    a. Introduction
    b. Requirement Analysis
    c. Software Requirement Specification
    d. Analysis and Design
    e. Implementation
    f. Testing

4. The report must be evaluated for 5 Marks. Demonstration and Viva for 10 Marks.

SOFTWARE TESTING LABORATORY

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

SEMESTER – IV

Laboratory Code	16MCA48	CIE Marks	*20*
Number of Lecture Hours/Week	01Hr Instructions 02 Hrs Laboratory	SEE Marks	*80*
Total Number of Lecture Hours	42	SEE Hours	*3 Hrs*

CREDITS – 02

Laboratory Experiments:

NOTE: The experiments are to be carried using discrete components only.

Course outcomes:

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

- Analyze the performance of fault based testing, planning and Monitoring the process, Documentation testing
- This course provides to experience on software testing projects using software testing tools.
- Understand the process to be followed in software development life cycle.
- Practical solutions to the problems
- Define, formulate and analyze a problem.

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.
7.	Design, develop, code and run the program in any suitable language to solve the commission problem. Analyze it from the perspective of decision table-based testing, derive different test cases, execute these test cases and discuss the test results
8.	Design, develop, code and run the program in any suitable language to implement the binary search algorithm. Determine the basis paths and using them derive different test cases, execute these test cases and discuss the test results.
PART B **MINI PROJECT**	

ADVANCED COMPUTER NETWORKS

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

SEMESTER – IV

Subject Code	16MCA441	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

- Understand the terminology and concepts of OSI reference model and TCP-IP reference model.
- Acquire theconceptsofprotocols,networkinterfaces,anddesign/performanceissuesinlocalareanetworksandwideareanetworks.
- Analyze the difference between wireless networks and satellite network.
- Evaluate the performance of over symmetric networks

Modules	Teaching Hours
Module -1	
Introduction to Computer Networks History of TCP/IP, TCP Applications and Services, Performance Study of TCP/IP, earning 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.	**09 Hours**
Module -2	
TCP/IP Network Simulation and TCP Modeling The Role of Simulation, Steps of a Systematic Simulation Study,Types of Simulations, Continuous versus Discrete Event, Terminating versus Steady State, Synthetic versus Trace Driven Simulation, Simulation Validation andVerification, Confidence Level of Simulation Results, Confidence Level Formula, Terminating Simulation, Steady-State Simulation, Common Simulation Mistakes, Simulation with Self-Similar TrafficNetwork Simulators: Model Construction and Parameter SettingData Collection, Simulation Execution, Presentation of Results and Examples of TCP/IP Simulation.	**09 Hours**

Motivation for Mathematical. Modelling of TCP, Essentials of TCP Modelling, Window Dynamics Packet-Loss Process, Gallery of TCP Models, Periodic Model, Detailed Packet Loss Model,Stochastic Model with General Loss Process, Control System Model and Network System Model.	
Module - 3	
TCP/IP Performance over Wireless Networks & TCP/IP Performance over Mobile Networks & Optical Networks Layer 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. WirelessSystem 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 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, Optical Burst Switching, Optical Packet Switching:Optical Packet Format, Congestion Resolution in Optical Packet Switches, Performance of TCP/IP over Optical Networks, Optical Packet Network End-to-End Performance, Mapping of TCP in Optical Packets, Optical Packet Design in the TCP/IP Environment	**12 Hours**
Module - 4	
TCP/IP Performance over Satellite Networks & TCP/IP Performance over Asymmetric Networks A Brief History of Data Satellites, Motivations for Using Satellites, Types of Satellites Satellite Internet Architectures,Satellite Characteristics Affecting TCP:Long Feedback Loop, Link Impairment,Bandwidth-Delay Product, Bandwidth Asymmetry, Variable Delays,LEO Handoff Spectral Congestion,Security. TCP Enhancements for Satellite Networks: Path MTU Discovery,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	**12 Hours**

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 Satellite Links: Satellite Transport Protocol, Space Communications Protocol Specifications-Transport Protocol. 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	
Module – 5	
TCP/IP Performance over Asymmetric Networks & New TCP Standards and Flavors 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	**8 Hours**
Text Books: High Performance TCP/IP: Networking Concepts, Issues, and Solutions, Mahbub Hassan and Raj Jain, IST Edition, 2009 PHI Learning Chapters 1,2,3,4,5,6,7,8,9,10,11,12,13 (excluding those topics which are not in the syllabus)	
Reference Books: TCP/IP Illustrated (Volume I, Volume II and Volume III), W. Richard Stevens, *Addison-Wesley*	

DATA WAREHOUSING AND DATA MINING

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

SEMESTER –IV

Subject Code	16MCA442	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

- Learn the concept of Data warehousing and OLAP.
- Understand storage and retrieval technique of data from DATA CUBE.
- Analyze different types of data and different preprocessing techniques.
- Evaluate various Association algorithms and its applications.
- Apply different Classification technique.
- Evaluate different type’s classifiers.
- Analyze different clustering techniques and their applications

Modules	**Teaching Hours**
Module -1	
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 CubeImplementations, Data Cube operations, Implementation of OLAP and overview on OLAP Softwares.	**08 Hours**
Module -2	
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	**10 Hours**
Module-3	
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	**8 Hours**
Module-4	
Classification : Methods, Improving accuracy of classification Basics, General approach to solve classification problem, Decision Trees, Rule Based Classifiers,	**12 Hours**

Nearest Neighbor Classifiers. Bayesian Classifiers, Estimating Predictive accuracy of classification methods, Improving accuracy of classification methods, Evaluation criteria for classification methods, Multiclass Problem.	
Module-5	
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	**12 Hours**

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 ComputingMcGrawHill Publisher, 1997.

WEB ENGINEERING

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

SEMESTER – IV

Subject Code	16MCA443	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

- This course aims to introduce the methods and techniques used in Web-based system development.
- In contrast to traditional Software Engineering efforts, Web Engineering methods and techniques must incorporate unique aspects of the problem domain such as: document oriented delivery, fine-grained lifecycles, user-centric development, client-server legacy system integration and diverse end user skill levels.
- This course draws upon previous programming and computing experience to develop practical web development and maintenance skills.

Modules	**Teaching Hours**
Module -1	
Introduction to Web Engineering An Introduction to Web Engineering, History of web Development, Time line, Motivation, Categories of Web Applications, Characteristics of Web Applications. Evolution and Need for Web Engineering, Web Engineering Models, Software Engineering v/s Web Engineering. World Wide Web: Introduction to TCP/IP and WAP, DNS, Email, TelNet, HTTP and FTP, Introduction to Browser and search engines, Search fundamentals, Search strategies, Directories search engines and Meta search engines, Working of the search engines, Miscellaneous Web Browser details, Introduction to Web Servers: Features of web servers, caching, case study-IIS, Apache, Configuring web servers.	**10 Hours**
Module -2	
Information Architecture for Web Engineering Information Architecture: The role of the Information Architect, Collaboration and Communication, Organizing Information, Organizational Challenges, Organizing Web sites parameters and Intranets Creating Cohesive Websites: Conceptual Overview Website Development, Website Design issues, Conceptual Design, High-Level Design, Indexing the Right Stuff, Grouping Content.Architectural Page Mockups, Design Sketches,Navigation Systems. Searching Systems Good & bad web design,	**12 hours**

Process of Web Publishing. Phases of Web Site development, enhancing your web-site, submission of website to search engines Web security issues, security audit of websites, Web effort estimation, Productivity, Measurement, Quality usability and reliability. Requirements Engineering for Web Applications: Introduction, Fundamentals, Requirement Source, Type, Notations Tools. Principles Requirements Engineering Activities, Adapting RE Methods to Web Application	
Module – 3	
XML and Web XML Language Basics, XML - Name Spaces - Structuring With Schemas and DTD - Presentation Techniques - Transformation - XML Infrastructure, Overview Of SOAP - HTTP - XML-RPC - SOAP: Protocol - Message Structure - Intermediaries - Actors - Design Patterns And Faults - SOAP With Attachments.	**07 Hours**
Module - 4	
Web Services Overview - Architecture - Key Technologies - UDDI - WSDL - ebXML - SOAP And Web Services In E-Com - Overview Of .NET And J2EE, XML SECURITY-Security Overview - Canonicalization - XML Security Framework - XML Encryption - XML Digital Signature - XKMS Structure - Guidelines For Signing XML Documents - XML In Practice	**09 Hours**
Module - 5	
Cloud Services Collaborating on Calendars, Schedules, and Task Management, Collaborating on Event management, Collaborating on Contact management, collaborating on Project Management, Collaborating on Word Processing, Collaborating on Spread sheets, Collaborating on Databases, Collaborating on presentations, Storing and sharing Files and other online content, sharing Digital Photographs, controlling the collaborations with Web-Based Desktops.	**12 Hours**
Text Books: 1. Roger S.Pressman, David Lowe, “Web Engineering”, Tata Mcgraw Hill Publication, 2007 2. Frank. P. Coyle, XML, Web Services And The Data Revolution, Pearson Education, 2002. 3. Ramesh Nagappan , Robert Skoczylas and Rima Patel Sriganesh, " Developing Java Web Services", Wiley Publishing Inc., 2004.	
Reference Books: 1. Cloud Computing a Practical approach, Anthony TVelte, Toby J Velte, Robert Elsenpeter, 2. Tata McGraw-HILL, 2010 Edition Sandeep Chatterjee, James Webber, "Developing Enterprise Web Services", Pearson Education, 2004. 3. McGovern, et al., "Java Web Services Architecture", Morgan Kaufmann Publishers,2005 4. Cloud Computing-Web Based applications that change the way you work and collaborate online, Michael Miller, Pearson Education, 2009 Edition	

SOFTWARE ARCHITECTURE
[As per Choice Based Credit System (CBCS) scheme]
SEMESTER – IV

Subject Code	16MCA444	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

- Acquire knowledge of
    - Working principles, characteristics and basic applications of Architectural patters.
    - information on project life cycle context.
    - How the architecture is influenced.
    - understand the quality attributes of architecture.
- Modeling quality attributes through
    - check lists.
    - experiments.
    - Back-of-the envelope analysis
- Requirements gathering.
    - Interviewing stack holders, etc,
- Understand different design strategies
- Different types of design patters can be understood.

Modules	**Teaching Hours**
Module -1	
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?	**10 Hours**
Module -2	
Understanding Quality Attributes Architecture & Requirements; Functionality; quality attribute considerations; Specifying and achieving Quality attribute requirements; Guiding quality design decisions; Availability; Interoperability; Modifiability; Performance; Security; Testability; Usability.	**12 hours**
Module - 3	
Quality Attribute modeling and Analysis, Architecture and requirements Gathering Modeling Architecture to enable quality attribute analysis;	**12 Hours**

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	
Module - 4	
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.	**10 Hours**
Module -5	
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	**6 Hours**
Text Books: 1. Len Bass, Paul Clements, Rick Kazman: Software Architecture in Practice, 3rdEdition, Pearson Education, 2013(Listed Topics only from Chapters1,2,3,4,5,6,7,8,9,10,11,14,16,17,18,19) 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 (chapter 2).	
Reference Books: 1. Richard N.Taylor, NenadMedvidovic and Eric M. Dashofy: Software Architecture: Foundations, Theory, and Practice, Wiley- India 2012 2. Mary Shaw and DavidGarlan : Software Architecture -Perspectives on an Emerging Discipline, Prentice Hall of India, 2007.	

PARALLEL AND DISTRIBUTED COMPUTING

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

SEMESTER – IV

Subject Code	16MCA445	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

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

- Understand fundamentals of parallel and distributing computing
- Learn the algorithms and key technologies of parallel/distributed.
- Find and analyze performance issues parallel/distributed computing and an ability to make appropriate design trade-off during problem solving
- Develop parallel program using parallel algorithms

Modules	**Teaching Hours**
Module -1	
Introduction to Parallel Computing Need of Performance, Building Parallel Systems, Why to Write Parallel Programs? How to Write Parallel Programs? Approach: Concurrent, Parallel, Distributed.	**07 Hours**
Module -2	
Parallel Hardware and Parallel Software Background, Modifications to the von Neumann Model, Parallel Hardware, Parallel Software, Input and Output, Performance, Parallel Program Design and Writing and Running Parallel Programs	**07Hours**
Module - 3	
Distributed Memory Programming with MPI Getting Started, The Trapezoidal Rule in MPI, Dealing with I/O, Collective Communication, MPI Derived Data types, A Parallel Sorting Algorithm	**10 Hours**
Module - 4	
Shared Memory Programming with Pthreads Processes, Threads and Pthreads, Hello, World program ,Matrix-Vector Multiplication, Critical Sections Busy-Waiting, Mutexes, Producer-Consumer Synchronization and Semaphores, Barriers and Condition Variables, Read-Write Locks, Caches, Cache-Coherence, and False Sharing and Thread-Safety	**12 Hours**

Module -5	
Parallel Program Development and Parallel Algorithms Introduction to OpenMP, The Trapezoidal Rulem Scope of Variables, The Reduction Clause, The Parallel For Directive, More About Loops in OpenMP: Sorting, Scheduling Loops, Producers and Consumers, Caches, Cache-Coherence, and False Sharing and Thread-Safety, Two N-Body Solvers, Tree Search and Case Studies	**14 Hours**
Text Books: 1. An introduction to parallel programming by peter s. Pacheco. 2011.I Edition Morgan Kaufmann publishers	
Reference Books: 1.Using OpenMP: Portable Shared Memory Parallel Programming ,Gabriele Jost and Ruud van der Pas The MIT Press (October 12, 2007) 2. Using MPI - 2nd Edition: Portable Parallel Programming with the Message PassingInterface, William Gropp and Ewing Lusk, 1999, 2nd edition, MIT Press 3. Pthreads Programming: A Posix Standard for Better Multiprocessing, Dick Buttlar, Jacqueline Farrell & Bradford Nichols .1996, I Edition, Oreilly	

CRYPTOGRAPHY AND NETWORK SECURITY
[As per Choice Based Credit System (CBCS) scheme]
SEMESTER – IV

Subject Code	16MCA446	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

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

- To be able to identify common network security vulnerabilities/attacks; explain the foundations of Cryptography and network security;
- Understand Encryption and decryption of messages using block ciphers
- Demonstrate detailed knowledge of the role of encryption to protect data.
- Analyze Network Security Practice And System Security

Modules	**Teaching Hours**
Module -1	
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.	**08 Hours**
Module -2	
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, Diffie Hellman Key exchange.	**12Hours**
Module - 3	
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	**11 Hours**

Module - 4	
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 SecurityPayload; Combining Security Associations; Key Management.	**10 Hours**
Module –5	
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.	**09 Hours**
Text Books: William Stallings, “Cryptography and Network Security – Principles and Practices”, 4th Edition, Pearson Education, 2009. (Chapters: 1, 2.1-2.3, 3.1,3.2,3.5, 5.1,5.2, 6.2, 9.1,9.2, 10.1,10.2, 11.1-11.4, 13.1, 13.3, 14.1, 4.2, 15.1, 15.2, 16.1-16.6, 17.1-17.3, 18.1, 18.2, 20.1; Exclude the topic not mentioned in the syllabus)	
Reference Books: **1.** Behrouz A. Forouzan and DebdeepMukhopadhyay: “Cryptography and Network Security”, 2nd Edition, Tata McGraw-Hill, 2010. 2. AtulKahate, “Cryptography and Network Security” 2nd Edition TMH.	

WIRELESS COMMUNICATION&MOBILE COMPUTING

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

SEMESTER – IV

Subject Code	16MCA451	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

After studying this course, students will be able to:

- Acquire knowledge of classification of devices, Limitations of devices, interfaces, protocol and system Architecture
- Understand the importance of Wireless Medium Access control and CDMA based communication and its Applications.
- Importance of Network layer,Transport layer and Databases
- Data Dissemination and Broadcasting Systems
- Data Synchronization Server and Management Application languages – XML, Java, J2ME and JavaCard, Mobile Operating Systems

Modules	**Teaching Hours**
Module -1	
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 GSM – Services and System Architectures, Radio Interfaces, Protocols, Localization, Calling, Handover.	**10 Hours**
Module -2	
Wireless Medium Access Control and CDMA – based Communication Medium Access Control, Introduction to CDMA – based Systems. OFDM	**08Hours**
Module - 3	
Mobile IP Network Layer Mobile Transport Layer and Databases 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. **Databases** Database Hoarding Techniques, Data Caching, Client – Server Computing and Adaptation, Transactional Models, Query Processing, Data Recovery Process	**10 Hours**

Module -4	
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.	**10 Hours**
Module -5	
Data Synchronization in Mobile Computing Systems Synchronization, Synchronization Protocols, SyncML – Synchronization Language for Mobile Computing. Mobile Devices, Server and Management, Wireless LAN, Mobile Internet Connectivity and Personal Area NetworkMobile 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, Bluetooth – enabled Devices Network, Zigbee. Mobile Application languages – XML, Java, J2ME and JavaCard, Mobile Operating SystemsIntroduction, XML, JAVA, Java 2 Micro Edition (J2ME), JavaCard Operating System, PalmOS, Windows CE, Symbian OS.	**12 Hours**
Text Books: Raj Kamal: Mobile Computing, Oxford University Press, 2007	
Reference Books: 1. AsokeTalkukder, Roopa R Yavagal: Mobile Computing – Technology, Applications and Service Creation, Tata McGraw Hill, 2005. 2. 2 Reza B'Far: Mobile Computing Principles – Designing and Developing Mobile Applications with UML and XML, 5th Edition, Cambridge University press, 2006. 3. Uwe Hansmann, LothatMerk, Martin S Nicklous and Thomas Stober: Principles of Mobile Computing, 2nd Edition, Springer International Edition, 2003. 4. Schiller: Mobile Communication, Pearson Publication, 2004.	

NoSQL

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

Subject Code	16MCA452	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

After studying this course, students will be able to:

- Learn NoSQL, its characteristics and history, and the primary benefits for using NSQL databases.
- Define the major types of NoSQL databases including a primary use case and advantages/disadvantages of each type.
- Describe the factors affecting return on investment for using locally hosted database vs. database-as-a-service.
- List the key benefits of IBM Cloudant, a NoSQL Database-as-a-Service for Web and mobile applications.
- Create a document database, add documents, and run queries using IBM Cloudant.

Modules	**Teaching Hours**
Module -1	
Introduction to NoSQL Definition of NOSQL, History of NOSQL and Different NOSQL products, Exploring MondoDB Java/Ruby/Python, Interfacing and Interacting with NOSQL.	**06 Hours**
Module -2	
NoSQL Basics NOSQL Storage Architecture, CRUD operations with MongoDB, Querying, Modifying and Managing NOSQL Data stores, Indexing and ordering datasets(MongoDB/CouchDB/Cassandra).	**12Hours**
Module - 3	
Advanced NoSQL NOSQL in CLOUD, Parallel Processing with Map Reduce, BigData with Hive	**08 Hours**
Module -4	
Working with NOSQL Surveying Database Internals, Migrating from RDBMS to NOSQL, Web Frameworks and NOSQL, using MySQL as a NOSQL.	**10 Hours**

Module-5	
Developing Web Application with NOSQL and NOSQL Administration PHP and MongoDB, Python and MongoDB, Creating Blog Application with PHP, NOSQL Database Administration.	**14 Hours**

Text Books:

1. "Professional NOSQL" by Shashank Tiwari, 2011, WROX Press(Chapter 1,2,3,4,5,6,7,8,9,10,11,12,13,15)
2. The Definitive guide to MongoDB, The NoSQL Database for Cloud and Desktop Computing, Apress 2010 (Chapter 6,7,8,9)

Reference Books:

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, 8thEdition. Comprehensive coverage of the Java Language. Tata McGraw-Hill Edition – 2011.

INTERNET OF THINGS (IoT)

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

SEMESTER – IV

Subject Code	16MCA453	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

After studying this course, students will be able to:

- Understand constraints and opportunities of wireless and mobile networks for Internet of Things.
- Analyze the societal impact of IoT security events.
- Develop critical thinking skills.
- Analyze, design or develop parts of an Internet of Things solution and map it toward selected business model(s)
- Evaluate ethical and potential security issues related to the Internet of Things.

Modules	**Teaching Hours**
Module -1	
M2M to IoT Introduction:The Vision-Introduction, From M2M to IoT, M2M towards IoT-the global context, A use case example, Differing Characteristics	**10 Hours**
Module -2	
M2M to IoT A Market Perspective– Introduction, Some Definitions, M2M Value Chains, IoT Value Chains, An emerging industrial structure for IoT, The international driven global value chain and global information monopolies. M2M to IoT-An Architectural Overview– Building an architecture, Main design principles and needed capabilities, An IoT architecture outline, standards considerations.	**07Hours**
Module - 3	
M2M and IoT Technology Fundamentals Devices and gateways, Local and wide area networking, Data management, Business processes in IoT, Everything as a Service(XaaS), M2M and IoT Analytics, Knowledge Management	**06 Hours**
Module -4	
IoT Architecture-State of the Art Introduction, State of the art, Architecture Reference Model-Introduction, Reference Model and architecture, IoT reference Model	**09 Hours**

Module-5	
IoT Reference Architecture Introduction, Functional View, Information View, Deployment and Operational View, Other Relevant architectural views. Real-World Design Constraints- Introduction, Technical Design constraints-hardware is popular again, Data representation and visualization, Interaction and remote control. Industrial Automation- Service-oriented architecture-based device integration, SOCRADES: realizing the enterprise integrated Web of Things, IMC-AESOP: from the Web of Things to the Cloud of Things, Commercial Building Automation- Introduction, Case study: phase one-commercial building automation today, Case study: phase two- commercial building automation in the future.	**10 Hours**
Text Books: Jan Holler, VlasiosTsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle, **“From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence”,**1st Edition, Academic Press, 2014.	
Reference Books: 1. Vijay Madisetti and ArshdeepBahga, “Internet of Things (A Hands-on-Approach)”, 1stEdition, VPT, 2014. 2. Francis daCosta, “Rethinking the Internet of Things: A Scalable Approach to Connecting Everything”, 1st Edition, Apress Publications, 2013	

SOFTWARE QUALITY MANAGEMENT

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

SEMESTER – IV

Subject Code	16MCA454	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

After studying this course, students will be able to:

- Acquire knowledge of
  - Software Quality through Hierarchical models
- Improve the quality of software by
  - SQA plan
  - Reviews and Audits
- Quality control through CASE tools.
- Understand different quality standards

Modules	**Teaching Hours**
Module -1	
INTRODUCTION TO SOFTWARE QUALITY Software Quality - Hierarchical models of Boehm and McCall – Quality measurement – Metricsmeasurement – Metrics measurement and analysis–Gilb's approach – GQM Model	**10 Hours**
Module -2	
SOFTWARE QUALITY ASSURANCE Quality tasks – SQA plan – Teams – Characteristics Implementation – Documentation– Reviews and Audits.	**10Hours**
Module - 3	
QUALITY CONTROL AND RELIABILITY Tools for Quality – Ishikawa's basic tools – CASE tools Defect prevention and removal – Reliablitymobels Rayleigh model – Reliability growth models for quality assessment.	**10 Hours**
Module -4	
QUALITY MANAGEMENT SYSTEM Elements of QMS – Rayleigh model framework Reliability Growth models for QMS – Complexity metrics and models Customer satisfaction analysis.	**10 Hours**
Module -5	
QUALITY STANDARDS Need for standards – ISO 9000 Series – ISO 9000 3 for software development – CMMand CMMI – Six Sigma concepts..	**10 Hours**

Text Books:

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

Soft Computing

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

Subject Code	16MCA455	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

- Learn the concepts of feed forward neural networks.
- Understand the basic mathematical elements of the theory of fuzzy sets, fuzzy logic inference with emphasis on their use in the design of intelligent or humanistic systems.
- Analyse the design of a genetic algorithm, and comment its weaknesses and strengths
- Develop genetic algorithms for single and multiple objective optimization problems

Modules	**Teaching Hours**
Module -1	
Neural Networks History, overview of biological Neuro-system, Mathematical Modelsof Neurons, ANN architecture, Learning rules, Learning Paradigms-Supervised, Unsupervisedand reinforcement Learning, ANN training Algorithms perceptions, Training rules, Delta, BackPropagation Algorithm, Multilayer Perceptron Model, Hopfield Networks, AssociativeMemories, Applications of Artificial Neural Networks.	**10 Hours**
Module -2	
Fuzzy Logic Introduction to Fuzzy Logic, Classical and Fuzzy Sets: Overview ofClassical Sets, Membership Function, Fuzzy rule generation. Compliment, Intersections, Unions,Combinations of Operations, Aggregation Operations .Fuzzy Numbers, Linguistic Variables,Arithmetic Operations on Intervals & Numbers, Lattice of Fuzzy Numbers, Fuzzy Equations.Classical Logic, Multivalued Logics, Fuzzy Propositions, Fuzzy Qualifiers, Linguistic Hedges.Information & Uncertainty, Nonspecificity of Fuzzy & Crisp Sets, Fuzziness of Fuzzy Sets	**12Hours**
Module - 3	
Neuro-Fuzzy Systems Architecture of Neuro Fuzzy Networks, Applications of FuzzyLogic: Medicine, Economics etc.	**08 Hours**

Module - 4	
Artificial Intelligence AI problems, AI intelligent agents: Agents and Environments, the concept of rationality, thenature of environments, structure of agents, problem solving agents, problem formulation.Searching: Searching for solutions, uniformed search strategies, Heuristic functions. Constrainsatisfaction problems: Game Playing Alpha-Beta pruning, Evaluation functions, cutting ofsearch, Knowledge Representation & Reasons logical Agents, logic, propositional logic,Resolution patterns in propositional logic, Resolution,Forward & Backward Chaining, Planning – Classical planning problem, Language of planningproblems, Expressiveness and extension, planning with state.	**10 Hours**
Module-5	
Genetic Algorithms An Overview, GA in problem solving, and Implementation of GA,Genetic Algorithms: survival of the fittest principle in Biology, Genetic Algorithms, Significanceof Genetic operators, termination parameters, Evolving Neural nets, Ant Algorithms.	**10 Hours**

TEXT BOOKS:

1. Anderson J.A.: An Introduction to Neural Networks, PHI, 1999.
2. Hertz J. Krogh, R.G. Palmer: Introduction to the Theory of Neural Computation, Addison- Wesley, 1991.
3. G.J. Klir& B. Yuan: Fuzzy Sets & Fuzzy Logic, PHI, 1995.
4. Stuart Russel, Peter Norvig, “Artificial Intelligence – A Modern Approach”, Second Edition, PHI/Pearson Education.
5. Patrick Henry Winston, “Artificial Intelligence”, 3rd Edition, Pearson Education.
6. Melanie Mitchell: An Introduction to Genetic Algorithm, PHI, 1998.

ENTERPRISE RESOURCE PLANNING

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

SEMESTER – IV

Subject Code	16MCA456	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

Course outcomes:

After studying this course, students will be able to:

- Acquire knowledge of
    - Benefits of ERP, Process Re-engineering
    - Project management and Monitoring
- Analysethe performance of
    - Project
    - Quality management
- ERP evolves in market place
- Develop the ERP system, ERP with E-Commerce & Internet

Modules	Teaching Hours
Module -1	
Introduction To ERP Overview, Benefits of ERP, ERP and Related Technologies, Business Process Reengineering, Data Warehousing, Data Mining, On–line Analytical Processing, Supply Chain Management	**10 Hours**
Module -2	
ERP Implementation Implementation of Life Cycle, Implementation Methodology, Hidden Costs, Organizing Implementation, Vendors, Consultants and Users, Contracts, Project Management and Monitoring	**10Hours**
Module - 3	
BusinessModules Business Modules in an ERP Package, Finance, Manufacturing, Human Resource, Plant Maintenance, Materials Management, Quality Management, Sales and Distribution	**10 Hours**
Module -4	
ERPMarket ERP Market Place, SAP AG, PeopleSoft ,Baan Company , JD Edwards World Solutions Company, Oracle Corporation, QAD , System Software Associates.	**10 Hours**
Module -5	
ERP–PresentAndFuture Turbo Charge the ERP System, EIA, ERP and E–Commerce, ERP and Internet, Future Directions in ERP.	**10 Hours**

Text Books:

1. Alexis Leon, “ERP Demystified”, Tata McGraw Hill, 1999.
2. JosephA. Brady, Ellen F. Monk, Bret J. Wangner, “Concepts in Enterprise ResourcePlanning”, Thomson Learning, 2001.

Reference Books:

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

OBJECT-ORIENTED MODELING AND DESIGN PATTERNS

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

Subject Code	16MCA51	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:
After studying this course, students will be able to:

- Acquire knowledge of
    - Basic UML Concepts and terminologies
    - Life Cycle of Object oriented Development
    - Modeling Concepts
- Identify the basic principles of Software modeling and apply them in real world applications
- Produce conceptual models for solving operational problems in software and IT environment using UML
- Analyze the development of Object Oriented Software models in terms of
    - Static behaviour
    - Dynamic behaviour
- Evaluate and implement various Design patterns

Modules	**Teaching Hours**
Module -1	
Modeling Concepts & Class Modeling What is Object Orientation? What is OO development? OO themes; Evidence for usefulness of OO development; OO modeling history. Modeling as Design Technique: Modeling; 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; Practical Tips	**10 Hours**
Module -2	
State Modeling and Interaction Modeling State Modeling: Events, States, Transitions and Conditions; State diagrams; State diagram behavior; Practical tips. Advanced State Modeling: Nested state diagrams; Nested states; Signal generalization; Concurrency; A sample state model; Relation of class and state models; Practical tips.	**10Hours**

Interaction Modeling: Use case models; Sequence models; Activity models. Use case relationships; Procedural sequence models; Special constructs for activity models	
Module -3	
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.	**10Hours**
Module -4	
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 behavior	**10Hours**
Module -5	
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, Organization of work: Master-Slave, Access Control: Proxy; Management Patterns: Command processor; Communication Patterns: Forwarder-Receiver; Client-Dispatcher-Server; Publisher-Subscriber	**10 Hours**
Question paper pattern: 1. The question paper will have ten questions. 2. Each full question consists of 16 marks. 3. There will be 2 full questions (with a maximum of four sub questions) from each module. 4. Each full question will have sub questions covering all the topics under a module. 5. The students will have to answer 5 full questions, selecting one full question from each module.	

Text Books:

1. Michael Blaha, James Rumbaugh, “Object-Oriented Modeling and Design with UML”, 2nd Edition, Pearson Education / PHI, 2005. (Chapters 1 to 9, 11 to 14.10,15.1 to 15.8)
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. (Chapters 1, 2.4, 3)

Reference Books:

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.
3. K. Barclay, J. Savage, “Object-Oriented Design with UML and JAVA”, Elsevier, 2008.
4. Booch, G., Rumbaugh, J., and Jacobson, I., “The Unified Modeling Language User Guide”, 2nd Edition, Pearson, 2005.
5. E. Gamma, R. Helm, R. Johnson, J. Vlissides, “Design Patterns-Elements of Reusable Object- Oriented Software”, Addison-Wesley, 1995.
6. Michael R Blaha, James R Rumbaugh, “Object Oriented Modeling and Design with UML”, 2nd Edition, Prentice Hall, 2004

PROGRAMMING USING C#.NET

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

Subject Code	16MCA52	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:
After studying this course, students will be able to:

1. Understand C# and client-server concepts using .Net Frame Work Components.
2. Apply delegates, event and exception handling to incorporate with ASP, Win Form, ADO.NET.
3. Analyze the use of .Net Components depending on the problem statement.
4. Implement & develop a web based and Console based application with Database connectivity

Modules	**Teaching Hours**
Module -1	
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, Widows Card Space and LINQ.	**10 Hours**
Module -2	
Introducing C# , Namespaces, Classes, Structures and Objects Need of C#, C# Pre-processor Directives, Creating a Simple C# Console Application, Identifiers and Keywords. Data Types, Variables and Constants: Value Types, Reference Types, Type Conversions, Boxing andUnBoxing , Variables and Constants . Expression and Operators : Operator Precedence, Using the ?? (Null Coalescing) Operator, Using the :: (Scope Resolution) Operator and Using the is and as Operators. Control Flow statements: Selection Statements, Iteration Statements and Jump Statements. Namespaces, The System namespace, Classes and Objects: Creating a Class, Creating an Object, Using this Keyword, Creating an Array of Objects, Using the Nested Classes, Defining Partial Classes and Method, Returning a Value from a Method and Describing Access Modifiers. Static Classes and Static Class	**10Hours**

Members. Properties: Read-only Property, Static Property, Accessibility of accessors and Anonymous types. Indexers, Structs: Syntax of a struct and Access Modifiers for structs.	
Module -3	
Object- Oriented Programming, Delegates, Events and Exception Handling **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 Delegates, Muticasting with Delegates. Events: Event Sources, Event Handlers, Events and Delegates, Multiple Event Handlers. Exception Handling: The try/catch/finally statement, Checked and Unchecked Statements.	**10 Hours**
Module -4	
Graphical User Interface with Windows Forms Introduction: Definitions and Amplifier Types, Series Fed Class A Amplifier, Operation of Amplifier Stage, Transformer Coupled Class A Amplifier, Class B Amplifier Operation, Class B Amplifier Circuits: Transformer Coupled Push-Pull Circuits, Complementary –Symmetry Circuits, Amplifier Distortion, Class C and Class D Amplifier. Relevant Problems.	**10 Hours**
Module -5	
Web App Development and Data Access using ADO.NET Introduction, Web Basics, Multitier Application Architecture, Your First Web Application: Building WebTime Application, Examining WebTime.aspx's Code-Behind File, Standard Web Controls: Designing a Form, Validation Controls, Session Tracking: Cookies, Session Tracking with http Session State, Options.aspx :Selecting a Programming Language, ecommenations.aspx: Displaying Recommendations based on Session Values.	**10 Hours**

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 16 marks.
- There will be 2full 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:

1. NET 4.0 Programming (6-in-1), Black Book, Kogent Learning Solutions Inc., Wiley- Dream Tech Press. (Chapters: 1,10,11,12,13,14 and 19).
2. Paul Deitel and Harvey Deitel: C# 2010 for Programmers, 4th Edition, Pearson Education. (Chapters: 14,15,19 and 27.3)

Reference Books:

1. Andrew Trolsen: Pro C# 5.0 and the .NET 4.5 Framework, 6th Edition, Wiley-Appress.
2. Bart De Smet: C# 4.0 Unleashed, Pearson Education- SAMS Series.
3. Herbert Schildt: Complete Reference C# 4.0, Tata McGraw Hill, 2010.

MOBILE APPLICATIONS

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

SEMESTER – V

Subject Code	16MCA53	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

After studying this course, students will be able to:

- Illustrate effective user interfaces that leverage evolving mobile device capabilities
- Develop applications using software development kits (SDKs), frameworks and toolkits
- Establish various methods to integrate database and server-side technologies
- Design and develop open source software based mobile applications

Build and deploy competent mobile development solutions

Modules	**Teaching Hours**
Module -1	
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.	**10 Hours**
Module -2	
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	**10Hours**
Module -3	
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	**10 Hours**
Module -4	
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	**10 Hours**

Module -5	
Feedback and Oscillator Circuits 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	**10 Hours**

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 16 marks.
- There will be 2full 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:

Jeff McWherter and Scott Gowell, “Professional Mobile Application Development”, 1st Edition, 2012, ISBN: 978-1-118-20390-3

Reference Books:

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

MOBILE AND AD HOC SENSOR NETWORKS [As per Choice Based Credit System (CBCS) scheme] SEMESTER – V			
Subject Code	16MCA541	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03
CREDITS – 04			

Course outcomes:

After studying this course, students will be able to:

- Describe the basic concepts related to wireless networks
- Understand the devices and protocols used in the wireless network architectures
- Analyze various types of wireless technology and concepts
- Apply the concepts and solve problems relevant to performance of wireless networks

Modules	**Teaching Hours**
Module -1	
Mobile Ad-Hoc Networking with a View of 4G Wireless: Imperatives and Challenges, Off-the-Shelf Enables of Ad Hoc Networks, IEEE 802.11 in Ad Hoc Networks: Protocols, Performance and Open Issues, Scatternet Formation in Bluetooth Networks	**10 Hours**
Module -2	
Topology Control in Wireless Ad Hoc Networks, Location Discovery, Mobile Ad Hoc Networks (MANETs): Routing Technology for Dynamic, Wireless Networking, Routing Approaches in Mobile Ad Hoc Networks.	**10Hours**
Module -3	
Energy-Efficient Communication in Ad Hoc Wireless Networks, Ad Hoc Networks Security, Simulation and Modeling of Wireless, Mobile, and Ad Hoc Networks, Algorithmic Challenges in Ad Hoc Networks	**10 Hours**
Module -4	
Introduction and Overview of Wireless Sensor Networks: Applications of Wireless Sensor Networks, Examples of Category 1 WSN Applications, Another Taxonomy of WSN Technology. Basic Wireless Sensor Technology: Sensor Node Technology, Sensor Taxonomy, WN Operating Environment, WN Trends**.**	**10 Hours**
Module -5	
Wireless Transmission Technology and Systems: Radio Technology Primer, Available Wireless Technologies. Medium Access Control Protocols for Wireless Sensor Networks: Fundamentals of MAC Protocols, MAC Protocols for WSNs, Sensor-MAC Case Study, IEEE 802.15.4 LR-WPANs Standard Case Study**.**	**10 Hours**

Question paper pattern:

1. The question paper will have ten questions.
2. Each full question consists of 16 marks.
3. There will be 2full questions (with a maximum of four sub questions) from each module.
4. 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:

1. "Adhoc and Sensor Networks" by Stefano Basagni, Silvia Giordano, Ivan Stojmenvic. IEEE Press, A John Wiley & Sons, Inc., Publication 2004.

2. KazemSohraby, Daniel Minoli, TaiebZnati. Wireless Sensor Networks, A John Wiley & Sons, Inc., Publication 2007

Reference Books:

1.C. Siva Ram Murthy and B.S.Manoj, Ad hoc Wireless Networks Architectures and protocols, 2nd edition, Pearson Education. 2007.

2. Holger Karl & Andreas Willig, " Protocols and Architectures for Wireless Sensor Networks", John Wiley, 2005.

WEB 2.0 AND RICH INTERNET APPLICATIONS

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

SEMESTER – V

Subject Code	16MCA542	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

After studying this course, students will be able to:

- Define and illustrate rich internet concepts and applications
- Analyze the working of development models in web designing
- Illustrate appropriate component lifecycle techniques using frameworks
- Evaluate and implement state based systems using data models and data binding

Modules	**Teaching Hours**
Module -1	
Introduction What is Web 2.0?, Folksonomies and Web 2.0, Software as a Service (SaaS), Data and Web 2.0, Convergence, Iterative development, Rich User experience, Multiple Delivery Channels, Social Networking. What is JSON?, Array literals, Object literals, Mixing literals, JSON Syntax, JSON Encoding and Decoding, JSON versus XML. BUILDING 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 XMLHTTP Object – properties and methods, handling different browser implementations of XMLHTTP; AJAX Patterns (Only algorithms – examples not required): Predictive fetch pattern, Submission throttling pattern, Periodic refresh, Multi stage download, Fall back patterns. Introduction to JQuery	**10 Hours**
Module -2	
SOAP The Case for SOAP; What Does SOAP Define? SOAP Message Structure; SOAP Message Elements; SOAP Processing Model; SOAP Encoding; WSDL: Describing a Web Service; Describing Functional Characteristics of Services; WSDL 1.2; UDDI: Discovering Web Services; Categorizing Services; Identifiers; Business Entity Relationships; UDDI's SOAP Interfaces; UDDI and SOAP/WSDL Relationships; Publishing WSDL Service Interfaces in UDDI; Internationalization and Multiple Languages; Extending a UDDI Registry; Private UDDI Registries; ebXML: Architectural Overview of ebXML.	**10Hours**

Module -3	
Java Web Service Developer Pack: Java Web Service Developer Pack: Setting up Java WSDP, Java WSDP components. JAXP: JAXP Architecture; SAX; DOM; When to Use SAX; When to Use DOM; When Not to Use Either; JAXP and XML Schemas; XSLT; XSLTc; JDOM; JAXP RI; JAX-RPC: JAX-RPC Service Model; Data Types and Serialization; JAX-RPC Development; Advanced JAX-RPC; JAX-RPC Interoperability; JAX-RPC and J2EE;	**10 Hours**
Module -4	
JAKM JAXM: Messaging and MOM; Messaging and Web Services; Messaging in Java; JAXM Architecture; Designing with JAXM; Developing with JAXM; JAXR: Registries and Repositories; JAXR Architecture; The JAXR Information Model; The JAXR API; JAXR to UDDI Mapping; JAXR and ebXML Registry; JAXB: The Need for Binding and JAXB; When to Use JAXB; JAXB Architecture; Developing with JAXB; XML-to-Java Mapping; The JAXB API; Validation with JAXB; Customizing JAXB; When to Use Custom Declarations;	**10 Hours**
Module -5	
Transaction Management Transaction Management: Concepts; A Transaction Model for Web Services; New Transaction Specifications; JSRs for Web Service Transaction Support; Security: Security Considerations for Web Services; Web Services Security Initiatives; Canonical XML; XML Digital Signatures; Apache XML Security; XML Encryption; Security Assertions Markup Language; Web Services Security Assertions; XML Access Control Markup Language; XML Key Management Specification; WS-I Specifications; SOAP and Firewalls;	**10 Hours**

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 16 marks.
- There will be 2full 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:

1. Professional AJAX – Nicholas C Zakas et al, Wrox publications, 2006.
2. James McGovern, Sameer Tyagi, “Michael E. Stevens, Sunil Mathew: Java Web Services Architecture”, Morgan Kaufmann – 2003.

Reference Books:

1. Mashups – Francis Shanahan, Wrox, 2007. 2. Richard Monsol-Haefel, “J2EE Web Services”, Pearson 2003. 3. Steven Graham, Dong Davis, “Building Web Services with Java”, II Edition, Pearson-2005

BIG DATA ANALYTICS

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

Subject Code	16MCA543	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:
After studying this course, students will be able to:

- Design algorithms by employing Map Reduce technique for solving Big Data problems
- Design algorithms for Big Data by deciding on the apt Features set
- Design algorithms for handling peta bytes of datasets
- Design algorithms and propose solutions for Big Data by optimizing main memory consumption
- Design solutions for problems in Big Data by suggesting appropriate clustering techniques

Modules	**Teaching Hours**
Module -1	
Big Data and Analytics Example Applications, Basic Nomenclature, Analysis Process Model, Analytical Model Requirements , types of Data Sources, Sampling, Types of data elements, data explorations, exploratory statistical analysis, missing values, outlier detection and Treatment, standardizing data labels, categorization	**10 Hours**
Module -2	
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 – Inter- and Trans-Firewall Analytics	**10Hours**
Module -3	
Meet Hadoop Data, Data Storage and Analysis ,Comparison with Other Systems,RDBMS,Grid Computing Volunteer Computing, A Brief History of Hadoop,Apache Hadoop and the Hadoop Ecosystem Hadoop Releases Response	**10 Hours**
Module -4	
The Hadoop Distributed File system The Design of HDFS, HDFS Concepts, Blocks, Namenodes and Datanodes, HDFS Federation, HDFS High-Availability, The Command-Line Interface, Basic Filesystem Operations, Hadoop Filesystems Interfaces ,The Java Interface, Reading Data from a Hadoop URL, Reading Data Using the FileSystem API, Writing Data, Directories, Querying the Filesystem, Deleting Data, Data Flow Anatomy of a File Read ,Anatomy of a File Write, Coherency Model, Parallel Copying with distcp Keeping an HDFS Cluster Balanced, Hadoop Archives	**10 Hours**

Module -5	
Map Reduce A Weather Dataset ,Data Format, Analyzing the Data with Unix Tools, Analyzing the Data with Hadoop, Map and Reduce, Java MapReduce, Scaling Out, Data Flow, Combiner functions, Running a Distributed MapReduce Job, Hadoop Streaming, Hadoop Pipes, Compiling and Running, Developing a MapReduce Application, The Configuration API, Combining Resources, Variable Expansion, Configuring the Development Environment, Managing Configuration, GenericOptionsParser, Tool and ToolRunner, Writing a Unit Test, Mapper, Reducer, Running Locally on Test Data, Running a Job in a Local Job Runner, Testing the Driver, Running on a Cluster, Packaging, Launching a Job, The MapReduce Web UI, Retrieving the Results, Debugging a Job, Hadoop Logs, Remote Debugging.	**10 Hours**

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 16 marks.
- There will be 2full 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:

1. Bart Baesens, " Analytics in a Big Data World : The Essential Guide to Data Science and its Applications" Wiley
2. Michael Minelli, Michehe Chambers, "Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses", 1st Edition, Michael Minelli, Michele Chambers, AmbigaDhiraj, 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, ISBN: 9788126551071, 2015.
2. Chris Eaton, Dirk deroos et al. , "Understanding Big data ", McGraw Hill, 2012.
3. VigneshPrajapati, "Big Data Analytics with R and Haoop", Packet Publishing 2013.
4. Tom Plunkett, Brian Macdonald et al, "Oracle Big Data Handbook", Oracle Press, 2014.

PATTERN RECOGNITION

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

Subject Code	16MCA544	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

After studying this course, students will be able to:

- State and illustrate different techniques used in phases for pattern recognition problems
- Recognize and examine the real world problems and design appropriate algorithm for detection
- Apply the pattern recognition algorithm for detection, classification of different applications
- Test the PRsystems by considering different parameters of the problem domain and strategies of the software development
- Verify the quality of the PRsystems by considering different parameters of the problem domain

Modules	**Teaching Hours**
Module -1	
Introduction to PR and probability Machine perception, pattern recognition systems, design cycle, learning and adaptation, Applications of pattern recognition Introduction, probability of events, Random variables,Joint distributions, Densities, Moments of random variables, Estimation of parameters from samples Minimum risk estimators	**10 Hours**
Module -2	
Statistical Decision Making Introduction, Baye's Theorem, multiple features, conditionally independent features, decision boundaries, unequal costs of error, Estimation of error rates, the leaving one- out technique, Characteristic curves, Estimating the composition of populations	**10Hours**
Module -3	
Non parametric Decision Making& Clustering Introduction, histograms, Kernel estimators, window estimators, nearest neighbor classification techniques, adaptive decision boundaries, Adaptive discriminate Functions, minimum squared error discriminate functions, choosing a decision making technique, clustering, criterion functions and clustering, Hierarchical clustering	**10 Hours**
Module -4	
Artificial Neural Networks nets without hidden layers, nets with hidden layers, The back Propagation algorithms Hopfield nets, An Application	**10 Hours**

Module -5	
Case studies using tools: - Study feature extraction techniques, methodology Bio metric systems, Prediction of Natural calamities, Bio medical Application, Application of PR in image analysis	**10 Hours**

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 16 marks.
- There will be 2full 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:

1.Pattern Classification Duda R. O., and Hart P E., and Stork D G., Wiley Publishers

2.Pattern Recognition and Image Analysis, Earl Gose, Richard J and Steve J, PHI

3.Pattern recognition (Statistical, structural and Neural Approaches), Robert Schalkoff

Reference Books:

1. Pattern Recognition,SergiosThedoridis& Konstantinos Koutrumbas, Elsevier Academic Press, 4th Edition.

SERVICE ORIENTED ARCHITECTURES (SOA)

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

Subject Code	16MCA545	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

After studying this course, students will be able to:

- Describe about evolution, characteristics and services in SOA with SOA architecture, WSDL, SOAP and UDDI
- Analyze the SOA Architectural style, SOA strategies, modeling web services
- Design, implementing process of SOA in web service.
- Apply the SOA operational style for the web services.

Modules	**Teaching Hours**
Module -1	
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 SOAMinimum risk estimators	**10 Hours**
Module -2	
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	**10Hours**
Module -3	
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	**10 Hours**
Module -4	
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.	**10 Hours**

Module -5	
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.	**10 Hours**

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 16 marks.
- There will be 2full 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:

1.ThomasErl: 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:

1. I. J. Nagrath, “Electronics: Analog and Digital”, PHI.
2. David A. Bell, “Electronic Devices and Circuits”, Oxford University Press.

STORAGE AREA NETWORKS

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

SEMESTER – V

Course Code	16MCA546	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

After studying this course, students will be able to:

- Understand the fundamentals of storage and storage networking concepts
- Analyze Network Attached and Storage Area Networks Requirements
- Apply and Integrate SAN and NAS solutions for an enterprise requirements
- Design a secured, scalable SAN / NAS enterprise solutions

Modules	**Teaching Hours**
Module -1	
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	**10 Hours**
Module -2	
Storage Fundamentals Storage Architectures, Device Overviews, Connectivity Options, Data Organizational Methods	**10Hours**
Module -3	
Network Attached Storage Putting Storage on the Network, NAS Hardware Devices , NAS Software Components, NAS Connectivity Options	**10 Hours**
Module -4	
Storage Area Networks Architecture Overview, Hardware Devices, Software Components, Configuration Options for SANs.	**10 Hours**
Module -5	
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	**10 Hours**

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 16 marks.
- There will be 2full 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:

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), ISBN-13: 978-8126537501
2. Storage Are networks Essentials : A complete guide to understanding and Implementing SANs, Richard Barker, Paul Massiglia, Wiley

SOFTWARE DEFINED NETWORKS

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

SEMESTER – V

Course Code	16MCA551	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

After studying this course, students will be able to:

- Recognize the fundamentals and characteristics of Software Defined Networks
- Understand the basics of Software Defined Networks Operations and Data flow
- Discriminate different Software Defined Network Operations and Data Flow
- Analyse alternative definitions of Software Defined Networks
- Apply different Software Defined Network Operations in real world problem

Modules	**Teaching Hours**
Module -1	
Introduction to SDN Understanding the SDN,Understanding the SDN technology,Evolution versus Revolution, 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.	**10 Hours**
Module -2	
Working of SDN Evaluation of Switches and Control planes, SDN Implications, Data centre Needs, The Evaluation of Networking technology. 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.	**10Hours**
Module -3	
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.	**10 Hours**

Module -4	
SDN via APIS,SDN via Hypervisor-Based Overlays, SDN via Opening up the device, Network function virtualization, Alternative Overlap and Ranking. Data centres definition, Data centres demand, tunnelling technologies for Data centres	**10 Hours**
Module -5	
Path technologies in data centres, Ethernet fabrics in Data centres, SDN use case in Data centres, Real World Data centre implementation, Campus, Hospitality and Mobile Networks	**10 Hours**

Question paper pattern:

1. The question paper will have ten questions.
2. Each full question consists of 16 marks.
3. There will be 2 full questions (with a maximum of four sub questions) from each module.
4. Each full question will have sub questions covering all the topics under a module.
5. 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.
2. Software Define Networks, A Comprehensive Approach, Paul Goransson, Chuck Black. MK Publications.

Reference Books:

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

CLOUD COMPUTING

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

SEMESTER – V

Course Code	16MCA552	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

After studying this course, students will be able to:

- Understand the cloud computing delivery model and the enabling technologies.
- Explain and cloud computing platforms, key technology drivers and cloud programming/software environments
- Identify the need for cloud computing model and compare various key enabling technologies.
- Analyze and choose an appropriate programming environment for building cloud applications.

Modules	**Teaching Hours**
Module -1	
Distributed System Models and Enabling Technologies Scalable Computing Service over the Internet, System Models for Distributed and Cloud Computing, Software Environments for Distributed Systems and Clouds, Performance, Security and Energy Efficiency.	**10 Hours**
Module -2	
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.	**10Hours**
Module -3	
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	**10 Hours**
Module -4	
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.	**10 Hours**

Module -5	
Ubiquitous Clouds and the Internet of Things Cloud Trends in Supporting Ubiquitous Computing, Performance of Distributed Systems and the Cloud, Enabling Technologies for the Internet of Things, Innovative Applications of the Internet of Things, Online Social and Professional Networking.	**10 Hours**

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 16 marks.
- There will be 2full 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:

1. "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, GeethaManjunath, Elsevier Publication, 2012.
2. "Cloud Computing, A Practical Approach", Anthony T. Volte, Toby J. Volte and Robert Elsenpeter, McGraw Hill, 2010.
3. "Cloud Computing for Dummies", J. Hurwitz, ISBN 978-0-470-484-8
4. "Web-Based Applications that Change the Way You Work and Collaborate Online", Michael Miller, Pearson Publication, 2012.

INFORMATION RETRIEVAL & SEARCH ENGINES

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

SEMESTER – V

Course Code	16MCA553	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

After studying this course, students will be able to:

- Provide a high-level review of the field of informationretrieval and its relationship to search engines its architecture.
- Knowledge on crawling, document feeds, and other techniques for acquiring the information that will be searched based on statistical nature of text and the techniques.
- Create indexes for efficient search and how those indexes are used to process queries
- Understand techniques that are used to process queries and transform them into better representations of the user's information need
- Knowledge on Ranking algorithms and the retrieval models and important classes of techniques used for classification, filtering, clustering

Modules	**Teaching Hours**
Module -1	
INTRODUCTION Information retrieval: Boolean retrieval, dictionaries and tolerant retrieval, evaluation in information retrieval, XML retrieval, probabilistic information retrieval, language models for information retrieval, matrix and hierarchical clustering,Search Engines, Search Engineers, Architecture, Basic Building Blocks, Text Acquisition, Text Transformation Index Creation, User Interaction, Ranking and Evaluation.	**10 Hours**
Module -2	
CRAWLS , FEEDS Deciding what to search, Crawling the Web, Directory Crawling, Document Feeds, Conversion Problem, Storing the Documents, Detecting Duplicates, removes noise. Text Statistics, Document Parsing, Document Structure and Markup, Link Analysis, Information Extraction, Internationalization	**10Hours**
Module -3	
RANKING WITH INDEXES, QUERIES Abstract Model of Ranking, Inverted indexes, Compression, Entropy and Ambiguity, Delta, Encoding, Bit-aligned codes, Auxiliary Structures, Index Construction, Query Processing, Information Needs and Queries ,Query Transformation and Refinement , Showing theResults Cross-Language Search.	**10 Hours**

Module -4	
RETRIEVAL MODELS Overview of Retrieval Models , Boolean Retrieval , The Vector Space Model, Probabilistic Models, Information Retrieval as Classification, BM25 Ranking Algorithm, Complex Queries and Combining Evidence, Web Search, Machine Learning and Information Retrieval	**10 Hours**
Module -5	
EVALUATING SEARCH ENGINES The Evaluation Corpus , Logging , Effectiveness Metrics, Recall and Precision Averaging and Interpolation , Efficiency Metrics, Training, Testing, and Statistics	**10 Hours**
Question paper pattern: 1. The question paper will have ten questions. 2. Each full question consists of 16 marks. 3. There will be 2 full questions (with a maximum of four sub questions) from each module. 4. Each full question will have sub questions covering all the topics under a module. 5. The students will have to answer 5 full questions, selecting one full question from each module.	
Text Books: 1. Search Engines: Information Retrieval in Practice: Trevor Strohman, Bruce Croft Donald Metzler, Kindle Edition.	
Reference Books: Introduction to information retrieval:Christopher D. Manning, PrabhakarRaghavan and HinrichSchütze Cambridge University Press. 2008	

ARTIFICIAL INTELLIGENCE

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

SEMESTER – V

Course Code	16MCA554	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

After studying this course, students will be able to:

- Acquire knowledge of
  - Uncertainty and Problem solving techniques
  - Symbolic knowledge representation to specify domains
  - Reasoning tasks of a situated software agent
- Comprehend on
  - different logical systems for inference over formal domain representations
  - trace on particular inference algorithm working on a given problem specification
- Apply and Analyze AI technique to any given concrete problem
- Interpret and Implement non-trivial AI techniques in a relatively large system

Modules	**Teaching Hours**
Module -1	
What is Artificial Intelligence: The AI Problems, The Underlying assumption, What is an AI Technique?, The Level of the model, Criteria for success, some general references, One final word and beyond. Problems, problem spaces, and search: Defining, the problem as a state space search, Production systems, Problem characteristics, Production system characteristics, Issues in the design of search programs, Additional Problems.	**10 Hours**
Module -2	
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. **Using predicate logic**: Representing simple facts in logic, representing instance and ISA relationships, Computable functions and predicates, Resolution, Natural Deduction	**10Hours**

Module 3	
Symbolic Reasoning Under Uncertainty: Introduction to nonmonotonic reasoning, Logic for nonmonotonic reasoning, Implementation Issues, Augmenting a problem-solver **Implementation**: Depth-first search, Implementation: Breadth-first search. Statistical Reasoning: Probability and Bayes Theorem, Certainty factors and rule-based systems, Bayesian Networks, Dempster-Shafer Theory, Fuzzy logic.	**10 Hours**
Module -4	
Weak Slot-and-filter structures: Semantic Nets Frames, Strong slot-and –filler structures: Conceptual dependency, scripts, CYC	**10 Hours**
Module -5	
Advanced Topics in AI: Minimax search, Goal Stack Planning, Expert System- Representation, Expert System Shell, explanation, Knowledge acquisition.	**10 Hours**

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 16 marks.
- There will be 2full 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:

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. 2.. Nils J. Nilsson: “Principles of Artificial Intelligence”, Elsevier, ISBN-13: 9780934613101

SOFTWARE PROJECT MANAGEMENT
[As per Choice Based Credit System (CBCS) scheme]
SEMESTER – V

Course Code	16MCA555	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:
After studying this course, students will be able to:

- Understand the practices and methods for successful software project management
- Identify techniques for requirements, policies and decision making for effective resource management
- Apply the evaluation techniques for estimating cost, benefits, schedule and risk
- Devise a framework for software project management plan for activities, risk, monitoring and control
- Devise a framework to manage people

Modules	**Teaching Hours**
Module -1	
INTRODUCTION TO SOFTWARE PROJECT MANAGEMENT Introduction, Why is Software Project Management important? What is a Project?, Contract Management, Activities Covered by Software Project Management, Plans, Methods and Methodologies, Some ways of categorizing software projects, Stakeholders, Setting Objectives, Business Case, Project Success and Failure, What is Management? Management Control, Traditional versus Modern Project Management Practices	**10 Hours**
Module -2	
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 (Simple problems)	**10Hours**
Module -3	
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	**10 Hours**

Module -4	
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	**10 Hours**
Module -5	
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	**10 Hours**

Question Paper Pattern:

- The question paper will have ten questions.
- Each full question consists of 16 marks.
- 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:

1. Bob Hughes, Mike Cotterell, Rajib Mall, "Software Project Management", Fifth Edition, Tata McGraw Hill, 2011.
2."Accounting for Management" Jawahar Lal, 5th Edition, Wheeler Publications, Delhi

Reference Books:

1.JackMarchewka," 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 Practise", Pearson Education, 2002

CYBER SECURITY

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

Course Code	16MCA556	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	50	SEE Hours	03

CREDITS – 04

Course outcomes:

After studying this course, students will be able to:

- Define and illustrate cyber security concepts and applications
- Analyze the working of cyber security principles to system design
- Illustrate appropriate techniques to solve cyber security threats
- Evaluate and implement cyber security through network security protocols

Modules	**Teaching Hours**
Module -1	
Systems Vulnerability Scanning Overview of vulnerability scanning, Open Port / Service Identification, Banner / Version Check, Traffic Probe, Vulnerability Probe, Vulnerability Examples, OpenVAS, Metasploit. Networks Vulnerability Scanning - Netcat, Socat, understanding Port and Services tools - Datapipe, Fpipe, WinRelay, Network Reconnaissance – Nmap, THC-Amap and System tools. Network Sniffers and Injection tools – Tcpdump and Windump, Wireshark, Ettercap, Hping Kismet	**10 Hours**
Module -2	
Network Defense tools Firewalls and Packet Filters: Firewall Basics, Packet Filter Vs Firewall, How a Firewall Protects a Network, Packet Characteristic to Filter, Stateless Vs Stateful Firewalls, Network Address Translation (NAT) and Port Forwarding, the basic of Virtual Private Networks, Linux Firewall, Windows Firewall, Snort: Introduction Detection System	**10Hours**
Module -3	
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, L0htcrack, Pwdump, HTC-Hydra	**10 Hours**
Module -4	
Cyber Crime Introduction to Cyber Crime and law Cyber Crimes, Types of Cybercrime, Hacking, Attack vectors, Cyberspace and Criminal Behavior, Clarification of Terms, Traditional Problems Associated with Computer Crime,	**10 Hours**

Introduction to Incident Response, Digital Forensics, Computer Language, Network Language, Realms of the Cyber world, A Brief History of the Internet, Recognizing and Defining Computer Crime, Contemporary Crimes, Computers as Targets, Contaminants and Destruction of Data, Indian IT ACT 2000.	
Module -5	
Cyber Investigation Introduction to Cyber Crime Investigation Firewalls and Packet Filters, password Cracking, Keyloggers and Spyware, Virus and Warms, Trojan and backdoors, Steganography, DOS and DDOS attack, SQL injection, Buffer Overflow, Attack on wireless Networks	**10 Hours**

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 16 marks.
- There will be 2full 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:

1. Anti-Hacker Tool Kit (Indian Edition) by Mike Shema, Publication Mc Graw Hill
2. Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives by Nina Godbole and SunitBelpure, Publication Wiley

Reference Books:

1. Marjie T. Britz - Computer Forensics and Cyber Crime: An Introduction - Pearson
2. Chwan-Hwa (John) Wu,J. David Irwin - Introduction to Computer Networks and Cybersecurity - CRC Press
3. Bill Nelson, Amelia Phillips, Christopher Steuart - Guide to Computer Forensics and Investigations -cengage Learning

SOFTWARE DESIGN LABORATORY

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

Course Code	16MCA56	CIE Marks	20
Number of Practical Hours/Week Number of Instructional Hours/Week	02 01	SEE Marks	80
Total Number of Practical Hours	42	SEE Hours	03

CREDITS – 02

Laboratory Experiments:

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 specified by the examiner. For analysis and design models - Use-Case, Class Diagram, Sequence, Communication/Collaboration, Activity and State-machine diagrams should be drawn

NOTE: Any supporting tool may be used.

Course Outcomes

After studying this course, students will be able to:

- Understand the fundamental principles of Object-Oriented analysis, design, development and programming
- Demonstrate and represent the UML model elements, to enable visual representation of the system being developed
- Implement object oriented design model with the help of modern tool, Rational software Architect
- Analyze and differentiate the static and dynamic behavior of the system for achieving the intended functionalities of the system
- Evaluate Various design patterns for applicability, reasonableness, and relation to other design criteria

The design pattern is allotted based on lots from the following list:

1. Controller
2. Publisher-Subscriber (Observer)
3. Command Processor
4. Forwarder- Receiver
5. Client-Dispatcher
6. Proxy
7. Polymorphism
8. Whole-Part
9. Master-Slave

.NET LABORATORY

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

Course Code	16MCA57	CIE Marks	20
Number of Practical Hours/Week Number of Instructional Hours/Week	02 01	SEE Marks	80
Total Number of Lecture Hours	42	SEE Hours	03

CREDITS – 02

NOTE:

1. Students are required to execute one question from Part A and one from Part B.
2. Part A has to be evaluated for 50 marks and Part B has to be evaluated for 30 marks.

Course Outcomes

After studying this course, students will be able to:

- Understand C# and client-server concepts using .Net Frame Work Components
- Apply delegates, event and exception handling to incorporate with ASP, Win Form, ADO.NET
- Analyze the use of .Net Components depending on the problem statement
- Implement & develop a web based and Console based application with Database connectivity

1.Write a Program in C# to demonstrate Command line arguments processing.
2. Write a Program in C# to demonstrate boxing and Unboxing.
3. Write a program to demonstrate Operator overloading.
4. Find the sum of all the elements present in a jagged array of 3 inner arrays.
5. Using Try, Catch and Finally blocks write a program in C# to demonstrate error handling.
6. Demonstrate Use of Virtual and override key words in C# with a simple program.
7. Write a program to demonstrate delegates.
8. Write a program to demonstrate abstract class and abstract methods in C#.
9. Write a program to illustrate the use of different properties in C#.
10. Demonstrate arrays of interface types (for runtime polymorphism) with a C# program.

PART-B

1. Consider the Database STUDENT consisting of following tables: tbl_Course (CourseID: int, CourseName: string) tbl_Student (USN: string, StudName: string, Address: string, CourseID: int, YrOfAdmsn: int) Develop suitable windows application using C#.NET having following options:

1. Entering new course details.
2. Entering new student details.
3. Display the details of students (in a Grid) who belong to a particular course.
4. Display the details the students who have taken admission in a particular year.

2. Consider the Database BLOODBANK consisting of following tables: tbl_BloodGroup (BloodID: int, BloodGroup: string) tbl_Donor (DonorID: int, DonorName: stirng, Address: string, ContactNo: int, DOB: date, Gender: string, Weight: int, BloodID: int)
Develop suitable windows application using C#.NET having following options:

1. Entering Blood group details.
2. Entering new donor details.
3. Display the details of donors (in a Grid) having particular blood group.
4. Display the details of donors (in a Grid) based on gender.
5. Display the details of donors (in a Grid) based on age (above 18), weight (above 45KG) and
Gender(user's choice).

3. Consider the Database STUDENT consisting of following tables: tbl_Course (CourseID: int, CourseName: string) tbl_Book (BookID :int, BookTitle: string, Author: string, CourseID: int) tbl_Student (USN: string, StudName: string, CourseID: int) tbl_BookIssue(USN: string, BookID: int, IssueDate: Date) Develop suitable windows application using C#.NET having following options:

1. New Course Entry.
2. New Book Entry
3. New Student Entry
4. Issue of books to a student.
5. Generate report (display in a grid) showing all the books belonging to particular course.
6. Generate report (display in a grid) showing all the books issued on a particular date.
7. Generate report (display in a grid) showing all the books issued to a particular student.

4. Develop a Web Application using C#.NET and ASP.NET for an educational institution. The master page should consist of Institution Name, Logo and Address. Also, it should provide hyperlinks to Departments, Facilities Available and Feedback. Each department page and facilities page should be designed as static pages. The hyperlinks should navigate to these static pages in the form of Content Pages associated with Master Page designed. The Feedback page should have fields to enter Name, Email and Message with Submit and Cancel Buttons. Database should be created to store these three data.

5. Develop a Web Application using C#.NET and ASP.NET for a Bank. The BANK Database should consist of following tables: tbl_Bank (BankID: int, BankName: string) tbl_Branch (BranchID: int, BankID: int, BranchName: string) tbl_Account (AccountNo: int, BankID: int, BranchID: int, CustomerName: string, Address: string, ContactNo: int, Balance: real) (Note: AccountNo and BankID together is a composite primary key).

The master page of this web application should contain hyperlinks to New Bank Entry, New Branch Entry (of selected Bank), New Customer Entry (based on branch and bank) and Report Generation. The hyperlinks should navigate to respective content pages. These content pages provide the fields for respective data entry. The reports should be generated (display in grid) as below:

1. Display all records of particular bank.
2. Display all records of a branch of particular bank.
3. The balance should be displayed for the entered account number (Bank and Branch are input through ComboBox controls and Account number is input through TextBox).

MOBILE APPLICATIONS LABORATORY

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

Course Code	16MCA58	CIE Marks	20
Number of Practical Hours/Week Number of Instructional Hours/Week	02 01	SEE Marks	80
Total Number of Lecture Hours	42	SEE Hours	03

CREDITS – 02

Laboratory Programs:

The laboratory can be carried out only using any mobile application software.

Note:

1. Students are required to execute one question from Part A and give demo from Part B.

Part A has to be evaluated for 50 marks and Part B has to be evaluated for 30 marks.

Course outcomes:

After studying this course, students will be able to:

- Illustrate effective user interfaces that leverage evolving mobile device capabilities
- Develop applications using software development kits (SDKs), frameworks and toolkits
- Establish various methods to integrate database and server-side technologies
- Design and develop open source software based mobile applications
- Build and deploy competent mobile development solutions

PART – A

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

PART – B

Mini-Project

Students should be able to build a complete app using multiple features learnt in Part – A with user interfaces and database connectivity.

VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELGAUM
CHOICE BASED CREDIT SYSTEM (CBCS)
SCHEME OF TEACHING AND EXAMINATION 2016 - 2017

9th June 2016

Programme: Master of Computer Applications

VI Semester (PROJECT WORK / INTERNSHIP)

CREDIT BASED

Subject Code	Subject	Evaluation of Seminar/Synopsis/Dissertation					CREDITS
		Internal		External		Total	
		Seminar / Presentation	Synopsis	Dissertation Evaluation	Via-Voce		
16MCA61	Phase – I Synopsis - Project work / Internship	50	50	--		100	04
16MCA62	Phase – II Project Dissertation	75	-	125	100	300	16
Total		125	50	125	100	400	20
Grand Total (I to VI Sem.) : 4400 Marks; 150 Credits							

1/2

VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELGAUM
CHOICE BASED CREDIT SYSTEM (CBCS)
SCHEME OF TEACHING AND EXAMINATION 2016 - 2017

Guidelines for 1st to 5th Semester:

1. Internal Assessment can be awarded based on Test/Assignment/Seminar.
2. Weightage for Test is minimum 10 marks.

Guidelines for Phase – I: Synopsis - Project work / Internship (6th Semester)

1. The student shall undergo Internship for 4-6 weeks during vacation.
2. The Internship should be carried out in Industry/R&D Labs/ Institution
3. Internship report and Synopsis of the project has to be submitted before the end of the first month of 6th semester and evaluated by a Internal panel.
4. The synopsis of the project should include:
   a. Project Problem formulation and Literature Survey.
   b. Training undergone on required tools and technologies for the development of his/her project.
   c. not less than 15 pages of report.
5. Internal Assessment (Presentation/Seminar) is evaluated by internal panel for 50 Marks.

Guidelines for Phase – II: Project Dissertation

1. The Project should be carried out in his/her Institution or any Industry/R&D Labs based on relevant tools and techniques learned in MCA courses/internship for a minimum period of 16 weeks.
2. Internal Assessment is done by the internal Panel for 75 Marks.
3. The Internal Examiner (the project guide with at least 3 years experience) and External Examiner shall be appointed by the University for the final evaluation of Project.
4. Internal and External Examiner shall carry out the evaluation of Dissertation Report for 125 Marks individually. The average of the marks allotted by the Internal Examiner and the External examiner shall be the final marks of the Project dissertation report evaluation.
5. The Project Presentation and Viva-Voce shall be evaluated jointly by Internal Examiner and External Examiner for 100 Marks.
6. The outcome of the project should be encouraged to present/publish in reviewed Conferences/Journals as papers.The Board of Studies would like to recommend draft syllabus

09/06/16 9/6 09/06/2016 Poornima 9/6/16 9/6/16 9/6/16 9.6.16

2/2