University of Kerala

Career Related First Degree Programme in BCA

<u>SCHEME AND SYLLABUS</u>

SCHEME

Semester 1

Course code	Credits	Course Name	Lecture	Tutorial	Lab	Total
			...			

Course code	Credits	Course Name	Lecture	Tutorial	Lab	Total
EN1111.4	2	Speaking and listening skills	3	-	-	3
MM1131.9	3	Mathematics I	3	1	-	4
CP1121	3	Introduction to IT	3	-	-	3
CP1131	3	Digital Electronics	3	-	-	3
CP1141	4	Introduction to Programming	3	1	-	4
CP1142	3	Programming Lab – I	-	-	4	4
CP1122	2	PC Software Lab	-	-	4	4
TOTAL	20		15	2	8	25

Semester 2

Course code	Credits	Course Name	Lecture	Tutorial	Lab	Total
EN1211.4	2	Writing and Presentation Skills	3	-	-	3
MM1231.9	3	Mathematics II	3	1	-	4
CP1241	3	Computer Organization & Architecture	3	1	-	4
CP1242	3	Object Oriented Programming	3	-	-	3
CP1243	3	Data Structures	3	-	-	4
CP1244	3	Programming Lab – II	-	-	4	4
CP1245	3	Data Structure Lab	-	-	4	4
TOTAL	20		15	2	8	25

Semester 3

Course code	Credits	Course Name	Lecture	Tutorial	Lab	Total
CP1331	3	Computer Oriented Numerical Methods	3	1	-	4
CP1341	3	Computer Networks	3	-	-	3
CP1342	3	Operating Systems	3	1	-	4
CP1343	2	System Analysis and Design	3	-	-	3
CP1344	3	Programming in JAVA	3	-	-	3
CP1345	3	Programming Lab – IV	-	-	4	4
CP1346	3	Numerical Methods Lab	-	-	4	4
TOTAL	20		15	2	8	25

University of Kerala

Semester 4			Hrs per week			
Course code	Credits	Course Name	Lecture	Tutorial	Lab	Total
CP1441	2	Introduction to Information Security	3	-	-	3
CP1442	3	Visual Tools	3	-	-	3
CP1443	3	Database Management Systems	3	1	-	4
CP1444	3	Design and Analysis of Algorithms	3	1	-	4
CP1445	3	System Software	3	-	-	3
CP1446	3	Visual Tools Lab	-	-	4	4
CP1447	3	Databases Lab	-	-	4	4
TOTAL	20		15	2	8	25

Semester 5			Hrs per week			
Course code	Credits	Course Name	Lecture	Tutorial	Lab	Total
CP1541	3	Free and Open Source Softwares (FOSS)	3	-	-	3
CP1542	3	Computer Graphics	3	1	-	4
CP1543	3	Internet Programming	3	1	-	4
CP1551.1 CP1551.2 CP1551.3	2	**Open Course** Internet Technology LINUX Environment Business Informatics	3	-	-	3
CP1544	3	Management Information Systems	3	-	-	3
CP1545	3	Internet Programming Lab	-	-	4	4
CP1546	3	Computer Graphics Lab	-	-	4	4
TOTAL	20		15	2	8	25

Semester 6			Hrs per week			
Course code	Credits	Course Name	Lecture	Tutorial	Lab	Total
CP1641	2	Business Informatics	3	-	-	3
CP1642	4	Object Oriented Analysis and Design	3	1	-	4
CP1643	4	Data Mining & Data Warehousing	3	-	-	3
CP1661.1 CP1661.2 CP1661.3	4	**Elective** Bioinformatics Trends in Computing Software Testing	3	-	-	3
CP1644	3	Free and Open Source Software (FOSS) Lab	-	-	4	4
CP1645	3	Major Project & VIVA	-	-	8	8
TOTAL	20		12	1	12	25

Division of Marks (Lab Examination)

(Computer Science)
1. First program should be sufficiently simple   **– 25 marks**
   (Logic – 10 marks, Successful compilation – 10 marks, Result – 5 marks)
2. Second program should be based on advanced concepts   **- 30 marks**
   (Logic – 15 marks, Successful compilation – 10 marks, result – 5 marks)
3. Viva Voce   **- 15 marks**
4. Lab Record   **- 10 marks**

                                           Total Marks   **- 80 marks**

(Digital Electronics)
1. Procedure- Theory/Connection Diagram/ Equation   **- 20 Marks**
2. Manipulation of Experiment- Connection/Soldering   **- 15 Marks**
3. Observation/Tabulation/Calculation   **- 10 Marks**
4. Viva   **- 10 Marks**
5. Result   **- 10 Marks**
6. Identification of Circuit Components   **- 15 Marks**
   (Resistors Using Color Codes, Capacitors, Diodes, Transistors etc)

                                           Total Marks   **- 80 marks**

SEMESTER ONE

Semester 1			Hrs per week			
Course code	**Credits**	**Course Name**	**Lecture**	**Tutorial**	**Lab**	**Total**
EN1111.4	2	Speaking and listening skills	3	-	-	3
MM1131.9	3	Mathematics I	3	1	-	4
CP1121	3	Introduction to IT	3	-	-	3
CP1131	3	Digital Electronics	3	-	-	3
CP1141	4	Introduction to Programming	3	1	-	4
CP1142	3	Programming Lab – I	-	-	4	4
CP1122	2	PC Software Lab	-	-	4	4
TOTAL	**20**		**15**	**2**	**8**	**25**

EN1111.4 SPEAKING AND LISTENING SKILLS

1. AIM:
- To familiarize students with English sounds and phonemic symbols.
- To enhance their ability in listening and speaking.

2. OBJECTIVES:
On completion of the course, the students should be able to
- ✓ listen to lectures, public announcements and news on TV and radio.
- ✓ engage in telephonic conversation.
- ✓ communicate effectively and accurately in English.
- ✓ use spoken language for various purposes.

3. SYLLABUS

Module I: Pronunciation: Phonemic symbols – consonants – vowels –syllables - word stress - strong and weak form intonation.

Module II: **Listening Skills:** Difference between listening and hearing –active listening – barriers to listening - academic listening - listening for details - listening and note-taking - listening for sound contents of videos - listening to talks and descriptions -listening for meaning - listening to announcements - listening to news programmes.

Module III: **Speaking Skills:** Interactive nature of communication -importance of context - formal and informal - set expressions in different situations –greeting – introducing – making requests - asking for / giving permission - giving instructions and directions – agreeing / disagreeing - seeking and giving advice - inviting and apologizing telephonic skills - conversational manners.

Module IV: **Dialogue Practice:** (Students should be given ample practice in dialogue, using core and supplementary materials)

4. REFERENCES
4.1 Core
- ❖ **Listening and Speaking***:* A Course for Undergraduate Students (Foundation Books)
4.2 Additional References
- ▪ Marks, Jonathan. *English Pronunciation in Use*. New Delhi: CUP, 2007.
- ▪ Lynch, Tony. *Study Listening*. New Delhi: CUP, 2008.
- ▪ Kenneth, Anderson, Tony Lynch, Joan MacLean. *Study Speaking*. New Delhi: CUP, 2008

NB:- Activities and assignments are not meant for End_Semester_Examination

MM1131.9 MATHEMATICS I

1. SYLLABUS

Module–I: Review of basic differentiation, Differentiation of hyperbolic functions, derivatives of hyperbolic functions, inverse hyperbolic functions logarithmic differentiation, implicit differentiation, Lebnitz's theorem, Mean value theorem, Rolle's theorem, Lagrange's mean-value theorem, Maxima and minima.

Module–II: Differential equations, General Concepts, Formulation and solution of differential equations, solution of higher order linear DEs. Partial Des**,** Laplace and Inverse Laplace transforms.

Module–III: Theory of Numbers, prime numbers, Unique factorization theorem, Euclidean algorithm, congruences, Fermat's theorem, Wilson's theorem.

Module–IV: Complex Numbers, Separation into real and imaginary parts, Complex mapping

Module V: Miscellaneous Topics: Markov processes. Harmonic analysis and Fourier series, Linear Programming

2. REFERENCES
2.1 Core
- ❖ Erwin Kreyzig *Advanced Engineering Mathematics*, New Age International Pvt Ltd.
- ❖ Shanthi Narayan, *Differential Calculus*, S Chand & Company
- ❖ Zafar Ahsan, *Differential Equations and their applications*.
- ❖ Rudra Pratap, *Getting Started with MATLAB*, Oxford University Press
2.2 Internet resources:
- o www.ams.org/mathweb
- o www.falstad.com/mathphysics.html
- o //cs.jsu.edu/mcis/faculty/leathrum/Mathlets
- o //archives.math.utk.edu/visual.calculus/

CP1121 INTRODUCTION TO IT

1. AIM:
- ➢ To create overall generic awareness about scope of the field of IT and to impart basic personal computing skills.
- ➢ To create background knowledge for the various courses in the programme.

2. OBJECTIVES:
- ✓ To introduce the basic terminology in the field of IT
- ✓ To impart functional knowledge about PC hardware, operations and concepts
- ✓ To impart functional knowledge in the use of GUI Operating System
- ✓ To impart functional knowledge in a standard office package (word processor, spread sheet and presentation softwares) and popular utilities
- ✓ To impart functional knowledge about networks and internet.
- ✓ To give an overview of computer application in various fields and an overall generic awareness about the scope of the field of IT

3. SYLLABUS

Module–I: Computer characteristics: Speed, storage, accuracy, diligence; Digital signals, Binary System, ASCII; Historic Evolution of Computers; Classification of computers: Microcomputer, Minicomputer, mainframes, Supercomputers; Personal computers: Desktop, Laptops, Palmtop, Tablet PC; Hardware & Software; Von Neumann model.

Module–II: Hardware: CPU, Memory, Input devices, output devices. Memory units: RAM (SDRAM, DDR RAM, RDRAM etc. feature wise comparison only); ROM-different types: Flash memory; Auxiliary storage: Magnetic devices, Optical Devices; Floppy, Hard disk, Memory stick, CD, DVD, CD-Writer; Input devices - keyboard, mouse, scanner, speech input devices, digital camera, Touch screen, Joystick, Optical readers, bar code reader; Output devices: Display device, size and resolution; CRT, LCD; Printers: Dot-matrix, Inkjet, Laser; Plotters, Sound cards & speaker.

Module-III: Software- System software, Application software; concepts of files and folders, Introduction to Operating systems, Different types of operating systems: single user, multitasking, time-sharing multi-user; Booting, POST; Basic features of two GUI operating systems: Windows & Linux (Basic desk top management); Programming Languages, Compiler, Interpreter, Databases; Application softwares: Generic Features of Word processors, Spread sheets and Presentation softwares; Generic Introduction to Latex for scientific typesetting; Utilities and their use; Computer Viruses & Protection, Free software, open source.

Module–IV: Computer Networks- Connecting computers, Requirements for a network: Server, Workstation, switch, router, network operating systems; Internet: brief history, World Wide Web, Websites, URL, browsers, search engines, search tips; Internet connections: ISP, Dial-up, cable modem, WLL, DSL, leased line; email, email software features (send receive, filter, attach, forward, copy, blind copy); characteristics of web-based systems, Web pages, introduction to HTML.

NB:- Activities and assignments are not meant for End_Semester_Examination

Activities & Assignments: Applications of Computers in various fields: office automation, education, entertainment, medicine, commerce, governance, resource management, law and order, communications, science and technology, defense; Historic evolution of IT; Pioneers in IT; Debates in IT : Computer Creativity, Digital Divide, IT Policy, IT and Development etc; IT in India (major initiatives, key institutions, statistics), IT in Kerala (major initiatives, key institutions, statistics); Careers in IT.

4. REFERENCES
4.1 Core
❖ E. Balaguruswamy, *Fundamentals of Computers*, McGraw hill, 2014

4.2 Additional
▪ Dennis P Curtain, *Information Technology: The Breaking wave*, McGrawhill, 2014
▪ Peter Norton, *Introduction to Computers*, McGrawhill, Seventh edition

4.3 Internet resources:
o www.fgcu.edu/support/office2000
o www.openoffice.org *Open Office Official web site*
o www.microsoft.com/office *MS Office web site*
o www.lgta.org *Office on-line lessons*
o www.learnthenet.com *Web Primer*
o www.computer.org/history/timeline
o www.computerhistory.org
o http://computer.howstuffworks.com
o http://vmoc.museophile.org *Computer History*
o www.dell.com *Dell Computers*
o www.intel.com *Intel*
o www.ibm.com *IBM*
o www.keralaitmission.org *Kerala Govt. IT Dept*.
o www.technopark.org

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1131 DIGITAL ELECTRONICS

1. AIM:
  ➢ To impart basic knowledge in digital logic and circuits and to introduce basic concepts of data communications.

2. OBJECTIVES:
  ✓ To review basic electronics concepts
  ✓ To review data representation techniques
  ✓ To introduce student to basic concepts of digital logic
  ✓ To introduce students to the design of basic logic circuits
  ✓ To introduce students to some commonly used combinational and sequential circuits

3. SYLLABUS

Module–I: Review of Basic Electronics: Review of basic operations of passive and active electronic components: Resistors, Capacitors, Inductors, Diodes, LEDs and Transistors, Operation of rectifiers (half and full wave), RC Coupled Feedback Amplifiers, Oscillators, Multi-vibrators, 555 timer (All the preceding topics shall be covered with stress on behaviour of component/circuit).

Module–II: Data Representation: Data Representation: Concept of number system bases – binary, decimal and hexadecimal number systems and conversion between each, Binary arithmetic: Addition, subtraction, 1s and 2s complement system, multiplication, Codes: BCD, ASCII, Floating Point Representation.

Module III: Boolean Algebra: Basic Functions: AND, OR and NOT, Truth tables; Combinational logic: Laws of Boolean Algebra; Combinational Logic in Venn diagrams; Other Boolean functions: NAND, NOR, XOR, Implication; Flip Flops and Latches; Realising Boolean Functions: Min-terms, SOP Expressions, Max-terms, POS Expressions; Karnaugh maps, McClarley method.

Module IV: Digital Circuits: Multiplexer, Adders (full and half), comparators, counters, Decoders and display, shift registers, de-multiplexer and key-board encoder; Digital ICs: TTL, CMOS and ECL families, SSI, MSI, LSI and VLSI classification, noise, fan-out, power dissipation, propagation delay.

Activities and assignments*: Miscellaneous Topics: Advances in Electronics: Evolution of Transistor Technology, Nano Technology, Molecular Electronics.*

4. REFERENCES
4.1 Core
  ❖ M.Morris Mano, *Digital Logic and Computer Design*, Pearson, 2013
4.2 Additional
  ▪ Thomas L Floyd, *Digital fundamentals*, Pearson, 2013
4.3 Internet resources:
  o www.prenhall.com/mano

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1141 INTRODUCTION TO PROGRAMMING

1. AIM:
> ➢ To Expose students to algorithmic thinking and problem solving and impart moderate skills in programming in a industry-standard programming language

2. OBJECTIVES:
- ✓ To expose students to algorithmic thinking and algorithmic representations
- ✓ To introduce students to basic data types and control structures in C.
- ✓ To introduce students to structured programming concepts
- ✓ To introduce students to standard library functions in C language

3. SYLLABUS

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

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

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

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

Activities and assignments: Pre-processor directives: #include, #define, macros with arguments, the operators # and ##, conditional compilations, multiple file programming; creating header files, program verification, algorithm efficiency analysis; int86 functions and graphic functions.

4. REFERENCES
4.1 Core
- ❖ Ashok N. Kamthene, *Programming in C*, Pearson Education, Second edition
4.2 Additional
- ▪ E.Balaguruswamy, *Programming in ANSI C*, McGrawhill, Sixth Edition

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1142 PROGRAMMING LAB – I

1. AIM:
➢ To provide an opportunity for hands-on practice of basic features of DOS, Windows, software tools(word processor, spread sheet, presentation s/w) and algorithmic thinking and problem solving in a industry standard programming language

2. OBJECTIVES:
After the completion of this course, the student should be able to:
✓ Create, Save, Copy, Delete, Organise various types of files and manage the desk top in general
✓ Use a standard word processing package Exploiting popular features
✓ Use a standard spread-sheet processing package Exploiting popular features
✓ Use a standard presentation package Exploiting popular features

Also, this course will provide hands-on practice in the following topics, under a variety of programming situations with a focus on writing, debugging and analyzing structured programs:
✓ basic data types in C.
✓ basic control structures in C.
✓ arrays, structures and files
✓ standard library functions in C language
✓ solving moderately complex problems involving the above and requiring selection of appropriate data structures and efficient algorithms

3. SYLLABUS
1. Familiarization of important DOS/Windows/Linux features
2. Practice on basic features of word processor, spread sheet and presentation software.

Part A
The C laboratory work will consist of 15-20 Experiments
1. Testing out and interpreting a variety of simple programs to demonstrate the syntax and use of the following features of the language: basic data types, operators and control structures.

Part II
2. 1-D Arrays: A variety of programs to declare, intitialise, read, print and process 1-D arrays of various basic data types. Processing to include, selection, sum, counting, selective sum, selective counting, reversing etc.
3. Pointers: A large number of trivial programs involving all possible data types to familiarize the syntax of pointers in a variety of situations and to draw memory diagrams based on the observations.
4. Structures: A variety of programs to declare, intitialise, read, print and process structures made up of a variety of data types and structures.
5. 2-D Arrays: A variety of programs to declare, intitialise, read, print and process 2-D arrays of various basic data types. Processing to include, selection, sum, counting, selective sum, selective counting, reversing etc.
6. Array of Structures and Structure of Arrays: Programs to demonstrate declaration and processing of structure of arrays and array of structures.
7. Pointers to Arrays: A number of programs to demonstrate handling of 1-D and 2-D arrays using pointers and to draw memory diagrams based on the observations.

8. Pointers to Structures: A number of programs to demonstrate use of pointers to structures and to draw memory diagrams based on the observations.

9. Functions –I: Simple Examples of declaring and using functions of the following categories (i) no argument, no return, (ii) argument, no return, (iii) no argument, return, (iv) argument, return, all pass by value

10. Functions –II: Declaring and using functions with pass by reference, Passing and Returning structures, Recursive functions.

11. Files: Simple Example involving use of multiple files: declaring, opening, closing, reading from and writing to text files.

12. Files: Example involving use of multiple files: declaring, opening, closing, reading from and writing to binary files.

13. Library functions: A variety of Examples demonstrating (i) string processing functions (ii) a variety of selected library functions

14. Debugging programs involving syntactic and/or logical errors

16-20: Developing programming solutions to problems including program design, algorithm development and data structure selection.

4. REFERENCES
4.1 Core
- ❖ Deitel & Deital, *C: How to Program*, Pearson Education
- ❖ Alan R Feuer, *The C Puzzle Book*, Pearson Education
- ❖ Yashvant Kanetkar, *Test Your C Skills*, BPB Publications, 3rd Edition

4.2 Internet resources:
- o www.cprogramming.com
- o www.programmersheaven.com
- o www.cplusplus.com
- o //cwx.prenhall.com/bookbind/pubbooks/deitel
- o www.fgcu.edu/support/office2000
- o www.openoffice.org Open Office Official web site
- o www.microsoft.com/office MS Office web site
- o www.lgta.org Office on-line lessons
- o www.learnthenet.com Web Primer

CP1122 PC SOFTWARE LAB
PART A
Lab Sessions to Practice the following features (depending on availability) on a selected GUI Operating System: Mouse Practice, Starting, Login, Shutdown, Exploring Directories, Resizing, Moving, Minimizing, closing of software windows, familiarization with file icons, Launching Applications, Deleting, Renaming files, Managing Directories, Searching for files, Using Accessories.

Lab Sessions to Practice the following features (depending on availability) on a selected Word Processor: General: Menus, Shortcut menus, Toolbars, Customizing toolbars, Creating and opening documents, Saving documents, Renaming documents, Working on multiple documents, **Working With Text** : edit menu, Formatting toolbar, Format Painter, Formatting Paragraphs: Paragraph attributes, The clipboard, Columns, Drop caps; **Styles** : Apply a style, Apply a style from the style dialog box, Create a new simple style from the style dialog box, Delete a style; **Lists :** Bulleted and numbered lists, Nested lists, Formatting lists **Tables :**Insert Table button, Draw a table, Inserting rows and columns, Moving and resizing a table, Tables and Borders toolbar, Table properties **Graphics:** Adding clip art, Add an image from a file, Editing a graphic, AutoShapes; **Spelling and Grammar**: AutoCorrect, Spelling and grammar check, Synonyms, Thesaurus; **Page Formatting:** Page margins, Page size and orientation, Headers and footers, Page numbers, Print preview and printing; **Macros :**Recording a macro, Running a macro; **Table of Contents**: Mark TOC entries; Generate a table of contents;

Lab Sessions to Practice the following features (depending on availability) on a selected Spread Sheet: Spreadsheet Basics: Screen elements,, Adding and renaming worksheets, The standard toolbar - opening, closing, saving, and more; Customising:, Menus, Customize toolbars, Recording a macro, Running a macro **Modifying A Worksheet**, Moving through cells, Adding worksheets, rows, and columns, Resizing rows and columns, Selecting cells, Moving and copying cells,, Freeze panes; **Formatting Cells:** Formatting toolbar, **Formulas and Functions:** Formulas, Linking worksheets, Relative, absolute, and mixed referencing, Basic functions, Function Wizard, Autosum, **Sorting and Filling:** Basic ascending and descending sorts, Complex sorts, Graphics; Adding clip art; Add an image from a file; Editing a graphics; AutoShapes; **Charts:** Chart Wizard; Resizing a chart; Moving a chart, Chart formatting toolbar; **Page Properties and Printing;** Keyboard Shortcuts

PART B
Presentation Softwares: Beginning a presentation, Templates and Slide Master, Drawing Tools, ClipArt and WordArt, Organization Charts, Graph, Output and Presentation Options, Integrating with Animation and Multimedia packages.

Lab Sessions to Practice the following features (depending on availability) on a selected Browser: Standard Tool Bars: Basic Browsing Buttons; Saving an Image from the Web, printing, Specifying a Home Page, **Browsing**: Using Web URLs, Anatomy of a URL, Membership Websites: Signing up for email service, Downloading and installing Plug-Ins; **Searching:** Academic Search on the web**;** Advanced search features.

Lab Session to Practice the following with Utilities:
• Play/use a variety of content on a standard PC
• Compress and decompress a file
• Burn a CD
• Convert between standard image file formats

SEMESTER TWO

Semester 2			Hrs per week			
Course code	**Credits**	**Course Name**	**Lecture**	**Tutorial**	**Lab**	**Total**
EN1211.4	2	Writing and Presentation Skills	3	-	-	3
MM1231.9	3	Mathematics II	3	1	-	4
CP1241	3	Computer Organization & Architecture	3	1	-	4
CP1242	3	Object Oriented Programming	3	-	-	3
CP1243	3	Data Structures	3	-	-	4
CP1244	3	Programming Lab – II	-	-	4	4
CP1245	3	Data Structure Lab	-	-	4	4
TOTAL	**20**		**15**	**2**	**8**	**25**

EN1211.4 WRITING AND PRESENTATION SKILLS

1. AIM:
- ➢ To familiarize students with different modes of general and academic writing.
- ➢ To help them master writing techniques to meet academic and professional needs
- ➢ To introduce them to the basics of academic presentation
- ➢ To sharpen their accuracy in writing.

2. OBJECTIVES:
On completion of the course, the students should be able to
- ✓ Understand the mechanism of general and academic writing.
- ✓ Recognize the different modes of writing
- ✓ Improve their reference skills, take notes, refer and document data and materials
- ✓ Prepare and present seminar papers and project reports effectively.

3. SYLLABUS

Module I: Writing as a skill – its importance - mechanism of writing – words and sentences - paragraph as a unit of structuring a whole text - combining different sources – functional use of writing –personal, academic and business writing – creative use of writing.

Module II: Writing process - planning a text - finding materials - drafting – revising – editing - finalizing the draft - computer as an aid - key board skills - word processing - desk top publishing.

Module III: Writing models – essay - précis - expansion of ideas – dialogue - letter writing - personal letters - formal letters - CV – surveys – questionnaire - e-mail – fax - job application - report writing.

Module IV: Presentation as a skill - elements of presentation strategies – audience – objectives – medium- key ideas - structuring the material - organizing content - audio-visual aids - hand-outs - use of power point - clarity of presentation - non-verbal communication - seminar paper presentation and discussion.

4. REFERENCES
4.1 Core
- ❖ **Part I to III :** *Write Rightly* A Course for Sharpening Your Writing Skills. (CUP)
- ❖ **Part IV:** *Guide to Presentations*. (by Mary Munter and Lynn Rusell. Pearson Education)

4.2 Additional
- ▪ **Part I to III:** Robert, Barraas. Students Must Write. London: Routledge, 2006.
  - o Bailey, Stephen. Academic Writing. Routledge, 2006.
  - o Hamp-Lyons, Liz, Ben Heasley. Study Writing. 2nd Edition. Cambridge University Press, 2008.
  - o Ilona, Leki. Academic Writing. CUP, 1998.
  - o McCarter, Sam, Norman Whitby. Writing Skills. Macmillan India, 2009.
- ▪ **Part IV:** Jay. *Effective Presentation*. New Delhi: Pearson, 2009.
- ▪ Mayor, Michael, et al, Ed. *Longman Dictionary Of Contemporary English*. 5th Edition. London: Pearson Longman Ltd, 2009

NB:- Activities and assignments are not meant for End_Semester_Examination

MM1231.9 MATHEMATICS II

1. SYLLABUS

Module–I: Proof Methods, Logic: Formal proofs, Propositional reasoning, Proofs by contradiction, False Proofs, Proofs by Induction, Symbolic Logic: Boolean expressions, Logical Equivalance, DeMorgan's Law, tautologies, Implications, Arguments, Fallacies, Normal forms in prepositional logic, Resolution

Module–II: Set Theory, Relations, Functions: Review of Set theory concepts, set operations, characteristic functions, fuzzy set theory basics, Relations: operations on relations, equivalence relations & partitions, partial orders, ordered sets, Warshal's algorithm, Functions, Recursion.

Module–III: Algebraic Structures: Algebra, DeMorgan's Law, Group, Ring, Polish Expressions, Communication Model and error corrections, Hamming Codes

Module–IV: Graph Theory: Introduction, Graph Notation, Topological sort, Graph Propagation algorithm, Depth First, Breadth-first searches, Shortest Path algorithms, Directed acyclic graphs.

Module V: Miscellaneous Topics: Graphical representations of functions, Graphical interpretation of convergence, Complex Mapping, Fractals, Grammars, Languages and Automaton. Introduction to Matlab (Matrix, Linear Algebra, Graphics operations)

2. REFERENCES
2.1 Core
  ❖ Rajendra Akerkar, Rupali Akerkar, *Discrete Mathematics*, Pearson Education

2.2 Additional
  ▪ RM Somasundaram, *Discrete Mathematical structures*
  ▪ Calvin C. Clawson, *Mathematical Mysteries, The beauty and magic of Numbers*, Viva Books Pvt Ltd,
  ▪ Rudra Pratap, *Getting Started with MATLAB*, Oxford University Press

2.3 Internet resources:
  o www.prenhall.com//goodaire www.ams.org/mathweb
  o www.falstad.com/mathphysics.html //cs.jsu.edu/mcis/faculty/leathrum/Mathlets
  o www.ealnet.com/ealsoft/fracted.htm Fractals
  o www.math.umass.edu/~mconnors/fractal/fractal.html
  o //info.lboro.ac.uk/departments/ma/gallery/index.html
  o www.ee.umd.edu/~yavuz/logiccalc.html logic calculator
  o //archives.math.utk.edu/visual.calculus/ www.utm.edu/departments/math/graph/
  o www.math.odu.edu/~bogacki/cgi-bin/lat.cgi *Linear Algebra Tool Kit*

CP1241 COMPUTER ORGANIZATION & ARCHITECTURE

1. AIM:
  ➢ To impart knowledge in the functional organization of physical components and architecture of a computer.

2. OBJECTIVES:
  ✓ To understand the functional units of a standard PC and its working
  ✓ To understand the memory organization in a computer.
  ✓ To introduce the concept of parallel processing and multiprocessing.

3. SYLLABUS

Module–I: Functional units of a PC; basic operational concepts; memory address, word, instruction set, programs, assembly language instructions; CPU registers; addressing modes, instruction format, system buses, instruction cycle, memory, example-organization of 8085 computer; encoding of information, unsigned numbers, signed numbers, operations, Booth's algorithm (circuit design and RTL not required), floating point number representation, operations.

Module–II: Processing unit: Specifying a CPU, design of a simple CPU, fetching instructions, decoding and executing instructions, branching, design of a simple ALU, design of control unit, multiple buses in CPU, Micro-program, micro sequencer, micro subroutine, microinstruction format, design and implementation of a simple micro-sequencer; micro-programmed control and hardwired control, RISC & CISC (feature-wise comparison only); Pipelining and Parallel processing**,** Pentium microprocessor.

Module–III: Memory: memory hierarchy, speed, size, cost; RAM, ROM, internal chip organization; cache memory, operations in cache memory, hit ratio, multilevel organization of cache memory; virtual memory, page fault, TLB, segmentation, memory protection, multiple module memories, memory interleaving.

Module–IV: Input Output operations: Accessing I/O devices; Asynchronous data transfers, handshaking, programmed I/O (concept only), polling, interrupts: types of interrupts, processing interrupts, priority, interrupt hardware, ISR, daisy chaining; Direct memory access, DMA controller, transfer modes, I/O processors, serial communication, UART, standards: RS-232, USB.

Activities and Assignments:; parallelism in uniprocessor systems, organization of general-purpose multiprocessors; RTL, VHDL; hardware essentials: CPU sockets; FDC, HDC, I/O cards, display adapter, modem; motherboard architecture; bus system: PCI, AGP, USB; clustering, grid computing; Computer faults: hardware & software; types of faults; diagnostic programs and tools; printer problems; monitor problems, problem diagnosis, organization of a modern PC.

4. REFERENCES
4.1 Core
  ❖ Carpinelli, John D., *Computer systems Organization & Architecture*, Pearson Education
4.2 Additional
  ▪ Carl Hamacher, Vranesic, Zaky, *Computer Organization* 4/e, McGraw-Hill
  ▪ ISRD Group, *Computer Organization*, McGrawhill, Tenth edition

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1242 OBJECT ORIENTED PROGRAMMING

1. AIM:
  ➢ To introduce the student to the basic concepts of object orientation and impart skills in an industry standard object oriented language

2. OBJECTIVES:
On the completion of this course, the student will be able to
  ✓ Understand the concepts of classes and object
  ✓ Define classes for a given situation and instantiate objects for specific problem solving
  ✓ Reuse available classes after modifications if possible
  ✓ Possess skill in object oriented thought process

3. SYLLABUS

Module–I: Concept of Object orientation – why related data and methods should be kept as a single unit – comparison with procedural and structured programming – Classes and objects – data abstraction, encapsulation, inheritance, polymorphism, dynamic binding, message passing. Advantages of object orientation – reusability, maintenance, security, comfort in programming. Input and output streams in C++; Basic data types and declarations.

Module–II: Classes and objects in C++, access modifiers, static members, friend functions, Constructors and Destructors, polymorphism, Operator Overloading and type conversion, anonymous objects

Module-III: Inheritance- parent and child classes, private, public and protected inheritance, Multiple inheritance and multi-level inheritance, Virtual base classes. C++ and memory models – new and delete operators, Heap, dynamic objects.

Module–IV: Binding & Polymorphism: Early binding, Late Binding, Pointers to derived class objects, virtual functions, Pure virtual functions, abstract classes, object slicing, exception handling in C++: try, throw and catch.

Assignments and activities: Evolution of OOP – history of C and C++, Review of features of C++ common with C and also minor variations; study of File stream classes in C++. Templates – class and function templates, Templates versus macros, String objects in C++, Standard Template Library in C++ - containers, associative containers Name spaces.

4. REFERENCES
4.1 Core
  ❖ Ashok N. Kamthane, *Object oriented Programming with ANSI & Turbo C++,* Pearson

4.2 Additional
  ▪ H M Deitel and P J Deitel, *C++: how to program*, Pearson Education
  ▪ Robert Lafore, *Object Oriented Programming in Turbo C++*, Galgotia Publications

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1243 DATA STRUCTURES

1. AIM:
 ➢ To introduce students to various data structures and their features and applicability.

2. OBJECTIVES:
By the end of the course, students should be:
 ✓ Able to write well-structured programs in C
 ✓ Be familiar with data structures like array, structures, lists, stacks, queues, trees and graphs
 ✓ Able to implement the above data structures in C/C++
 ✓ Able to appreciate various searching and sorting strategies
 ✓ Able to select appropriate data structures for solving a given problem

3. SYLLABUS

Module–I: Linked Lists: Concept of static versus dynamic data structures, implementation of linked lists using pointers, operations on linked lists: insertion, deletion and traversing. Doubly linked lists and circular linked lists, applications of linked lists.

Module- II: Stacks and Queues: FIFO and LIFO data structures – stacks using (i) pointers and (ii) arrays. Queues using (i) pointers and (ii) arrays, applications, polish notation.

Module-III: Trees: Concept of linear versus non-linear data structures, various types of trees – binary, binary search trees. Creating a binary search tree, traversing a binary tree (in-order, pre-order and post-order), operations on a tree – insertion, deletion and processing, expression trees, implementation using pointers, applications.

Module–IV: Searching: sequential searching, binary searching, Hashing – linear hashing, hash functions, hash table searching, Sorting: exchange sort, selection sort and insertion sort, Graphs, graph traversal- depth-first and breadth-first traversal of graphs, applications.

Assignments and Activities: Multi-way search trees, B-trees, Huffman trees, case studies.

4. REFERENCES
4.1 Core
 ❖ A.K.Sharma, *Data Structures Using C*, Pearson, Second edition, 2011
4.2 Additional
 ▪ Nair A.S., Makhalekshmi, *Data Structures in C*, PHI, Third edition 2011.
4.3 Internet resources:
 o www.keralauniversity.edu/csbos
 o http://warrior-101.tripod.com/dstut/dstut.html : Tutorial on data structures
 o http://crasseux.com/books/ctutorial/Data-structures.html

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1244 PROGRAMMING LAB – II

1. AIM:
> To provide an opportunity for hands-on practice of object oriented programming and problem solving in a industry-standard programming language and also hands-on practice in various user-defined static and dynamic data structures.

2. OBJECTIVES:
This course will provide hands-on practice in a the following topics, under a variety of programming situations with a focus on writing, debugging and analyzing object oriented programs:
- ✓ basic data types and control structures in C++.
- ✓ managing classes and objects in a variety of situations
- ✓ solving moderately complex problems involving the above and requiring selection of appropriate structures and algorithms

3. SYLLABUS
The laboratory work will consist of 15-20 experiments, only by using class concept
Part A
1. Testing out and interpreting a variety of simple programs to demonstrate the syntax and use of the following features of the language: basic data types, operators and control structures.
2. Solving a problem using (i) structures and (ii) classes and comparison between the two (the problem logic and details should be kept minimal and simple to enable focus on the contrast between the two methods, for example declaring result of a set of students defining the name and total marks in the program itself).
3. Class definitions and usage involving variety of constructors and destructors
Part B
4. Programs involving various kinds of inheritances,
5. Programs involving operator overloading and type conversions
6. Programs involving virtual base classes, friend functions
7. Program to demonstrate early and late binding
8. Program to allocate memory dynamically
9. Program involving class and function templates
10. Programs to demonstrate(i) string processing (ii) file streams (iii) a variety of selected library functions
11. Exception handling
12. Handling of 2-D arrays using pointers
13. Debugging programs involving syntactic and/or logical errors

4. REFERENCES
- ❖ Deitel & Deital, *C++: How to Program*, Pearson Education

CP1245 DATA STRUCTURES LAB

1. AIM:
- ➢ To provide an opportunity for hands-on practice on different algorithms using various data structures.

2. OBJECTIVES:
This course will provide hands-on practice in all the following topics, using either C or C++:
- ✓ Stack and queues
- ✓ managing both singly and doubly linked list
- ✓ different trees, construction and traversal
- ✓ Searching and sorting

3. SYLLABUS
The laboratory work will consist of 15-20 experiments like
<u>Part A</u>
• Linked list: traversal, node deletion, node insertion in singly, doubly and circular lists
• Implementation of different searching techniques
• Implementation of different sorting techniques
<u>Part B</u>
• Stacks: matrix representation and linked list representation: Push, Pop
• Queues: matrix representation and linked list representation: Add, delete
• Circular queue implementation
• Evaluation of expression using stacks
• Tree traversal
• Evaluation of expression using binary trees.
• Infix to postfix and prefix conversion
• Creating and processing binary search tree

SEMESTER THREE

Semester 3			Hrs per week			
Course code	Credits	Course Name	Lecture	Tutorial	Lab	Total
CP1331	3	Computer Oriented Numerical Methods	3	1	-	4
CP1341	3	Computer Networks	3	-	-	3
CP1342	3	Operating Systems	3	1	-	4
CP1343	2	System Analysis and Design	3	-	-	3
CP1344	3	Programming in JAVA	3	-	-	3
CP1345	3	Programming Lab – IV	-	-	4	4
CP1346	3	Numerical Methods Lab	-	-	4	4
TOTAL	20		15	2	8	25

CP1331 COMPUTER ORIENTED NUMERICAL METHODS

1. SYLLABUS

Module I: Introduction and Non-linear Equations, Interpolations: Introduction to Numerical Methods: Nature of numerical problems; computer based solutions; number representations; Notions of accuracy, convergence, efficiency, complexity; Solutions of Nonlinear equations: (Bisection method; Regula-Falsi; Newton-Raphson; Secant; Successive approximation method); Interpolation techniques: Linear Interpolation; Newton's forward and backward formulae; Lagrange's interpolation; Bessel Functions; Linear Regression; Cubic Splines.

Module–II: Numerical Integration & Differentiation: Concept of differentiation and Integration, graphical interpretation; Cubic Spline based Numerical Differentiation; Numerical Integration: Trapezoidal, Simpson's Romberg, Gaussian, and Filon's methods.

Module–III : Matrix Computations: Addition, subtraction, transpose, multiplication of matrices; Matrix-based solutions of simultaneous linear equations: Gauss Jordan Method; Gauss elimination with back-substitution; LU Decomposition method.

Module–IV: Differential Equations: Ordinary Differential Equations: Picard's method; Euler's and modified Euler's method; Runge-Kutta; Predictor-Corrector methods; Partial Differential Equations: Jacobi and Gauss-Siedel methods.

2. REFERENCES
2.1 Core
   ❖ V Rajaraman, Computer Oriented Numerical Methods, PHI
2.2 Additional
   • E V Kameshwar, *Numerical Techniques in C*, BPB Publications

CP1341 COMPUTER NETWORKS

1. AIM:
> To introduce computer networks and through knowledge of data communication networks, their structures, techniques as well as some common standards.

2. OBJECTIVES:
On completion of this course student shall:
- ✓ Be aware of evolution of development of networks
- ✓ understand the basic transmission technologies and characteristics
- ✓ understand the use of layer architecture for networking systems
- ✓ understand the main design issues of transport protocols and the mechanism to control traffic flow and congestion.

3. SYLLABUS

Module I Introduction to networks – Data Communication – Data flow simplex, Half duplex, Full duplex- Type of Connection – Point–to-Point, multi-drop. Bandwidth- bit rate, baud rate. Transmission media – Copper wires, fiber optics, Radio transmission, microwave, Satellite. switching-circuit, packet, message.

Module II Protocols – standards- Layering, packets, Layered PDUs, ISO-OSI model, TCP/IP model – Comparison. Framing- bit oriented, byte oriented, Error correction – detection – parity, hamming code, CRC. Flow control, error control- Piggybacking, pipelining, Protocols- Noiseless and noisy channels – stop &wait, Stop &wait ARQ, Sliding window.

Module III Access control - pure- slotted ALOHA, CSMA, CSMA/CD. LAN Standards – Ethernet, Token bus, Token ring. Interfacing devices – bridge, hub, switch, router, gateway.

Module IV Internetworking- datagrams, fragmentation – routing-Distance vector routing, Link state routing. Concepts of congestion control-leaky bucket algorithm. Process to Process delivery -TCP, UDP, Application Layer -DNS, Remote login, file transfer protocol(FTP).

Assignments and activities: Practical networking- networking in LINUX, Peer–to-peer networking, Measurement and packet analysis, blue tooth, emerging topics

4. REFERENCES
4.1 Core
- ❖ Brijendra Singh, *Data Communication and Computer Networks* 2/e, PHI
4.2 Additional
- ▪ Behrouz A Forouzan, *Data Communication and Computer networks*, 4th ed, McGraw Hill
- ▪ Achyut S Godbole, *Data communications and networks*, McGrawHill, Second
4.3 Internet resources:
- o www.netbook.cs.purdue.edu, www.labbook.cs.purdue.edu,
- o www.edugrid.ac.in/webfolder/courses/cn/cn_indEX.htm

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1342 OPERATING SYSTEMS

1. AIM:
➢ To introduce students to basic functions and the theoretical underpinnings of modern operating systems

2. OBJECTIVES:
To introduce students to:
✓ Fundamental concepts of systems software
✓ Functions of operating systems as a resource manager
✓ Strategies for constrained resource allocation
✓ Strategies for process scheduling
✓ Memory and I/O Management techniques
✓ Salient features of popular operating systems.

3. SYLLABUS

Module I: Introduction to operating system: Operating system as the main component of system software; OS as a resource manager, Structure of OS- shell, utilities, resource management routines, kernel, evolution of OS, multiprogramming, time sharing, real-time systems, parallel systems, distributed systems, OS functions, Characteristics of modern OS; **Process Management:** Process description and control: process control block, Process states: operations on processes; concurrent process; threads; processes and threads; symmetric multiprocessing; micro Kernels. CPU Scheduling: Schedulers, Scheduling methodology, CPU Scheduling algorithms, performance comparison.

Module II: Process synchronization- independent and concurrent processes, critical section, mutual Exclusion, Petersons solution, semaphore, classical synchronization problem-bounded buffer and reader/writer problem. Concept of inter-process communication. **Deadlock**- deadlock and starvation, conditions for deadlock, resource allocation problem, methods for handling deadlock-deadlock prevention, deadlock avoidance- Bankers algorithm, deadlock detection, deadlock recovery.

Module III: Memory Management & Protection: Concept of memory, address binding, Logical address, physical address, swapping, contiguous allocation- fixed partition, variable partition, fragmentation. Non-contiguous allocation– paging, segmentation. Virtual memory-demand paging, page fault, replacement algorithms, thrashing. Protection and security – mechanisms and policies, threats, accidental data loss, protection mechanisms, user authentication, attacks from inside, virus, antivirus.

Module IV: I/O & File Management I/O management – I/O hardware, application I/O interface, kernel I/O subsystem. Disk I/O, disk scheduling, RAID, disk cache. File management-file concept, access methods, directory structure, file system structure & implementation, directory implementation, allocation methods, free space management.

Assignments and activities: case study of popular Operating Systems- MS DOS, UNIX, Windows 2000, Windows NT, Linux, Sun OS, Solaris. Process management –Windows, Linux, Solaris. Concurrency management - Windows, Linux, Solaris. Memory management - Windows, Linux, Solaris.

NB:- Activities and assignments are not meant for End_Semester_Examination

4. REFERENCES
4.1 Core
- ❖ Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, *Operating System Principles*

4.2 Additional
- ▪ Achyut S Godbole, *Operating systems*, McGRawhill, Third Edition

4.3 Internet resources:
- o www.aw.com/cs_supplements/nutt3/index.html
- o www.aw.com/cs_supplements/nutt/index.html
- o //cwx.prenhall.com/bookbind/pubbooks/tanenbaum2/
- o www.gnu.org, www.linux.org, www.linuxcentral.com

CP1343 SYSTEM ANALYSIS AND DESIGN

1. AIM:
- ➢ To provide an overview of how a software is developed and what are the different stages by which the development take place

2. OBJECTIVES
At the end of this course, the students will be able to
- ✓ Explain the background study required for developing a system
- ✓ Design a new system
- ✓ Discuss types of testing
- ✓ Select the hardware and software required for a system

3. SYLLABUS

Module I: Overview of System analysis and Design: Business system concepts, project selection, sources of project requests, preliminary investigation, System development life cycle - Feasibility analysis, design, implementation, testing and evaluation, project review. **Feasibility study** - technical and economical feasibility, cost and benefit analysis, fact finding techniques, DFD, Data dictionaries, Decision analysis, decision trees and tables.

Module II: System design-Design objectives, Process and stages of system design, Design methodologies, structured design, structured walkthrough, audit considerations, audit trials, detailed design, modularization, module specification, software design and documentation tools, top down and bottom up approaches

Module III: Testing & System Conversion: Unit and integration testing, testing practices and plans, system control and quality assurance, training, conversion, operation plans, system administration.

Module IV: Hardware and Software selection: Benchmarking, Financial considerations in selection software selection, vendor selection, performance and acceptance criteria.

4. REFERENCES
4.1 Core
- ❖ Award, EM, *Systems Analysis and Design*, Galgotia Pub, 1991
4.2 Additional
- ▪ Lesson, *System analysis and Design,* SRA pub, 1985
- ▪ Rajaraman V, *Analysis and Design of Information systems*, PHI, 1991

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1344 PROGRAMMING IN JAVA

1. AIM:
  ➢ To introduce students to basic features of Java language and selected APIs

2. OBJECTIVES:
  ✓ Let students install and work with JDK, also make them aware the use of java doc.
  ✓ Practice basic data types, operators and control structures in Java
  ✓ Practice basic handling of classes and objects in Java
  ✓ Introduce the following selected APIs: I/O, Strings, Threads, AWT, Applet, Networking
  ✓ Idea to approach and use a new package

3. SYLLABUS

Module–I: Brief History of Java, Special Features of Java, Data Type & Operators in Java, Arrays, Objects, the Assignment Statement, Arithmetic Operators, Relational and Logical Operators in Java, control Structures, The Java Class, Constructor, Simple Java Application, simple Java Applet, Finalizers, Classes inside classes : composition.

Module-II: Inheritance & Interface, Deriving Classes, Method Over-riding, Method Overloading, Access Modifiers, Abstract Class and Method, Interfaces, Packages, Imports and Class Path.

Module-III: Exception Handling, The Try-Catch Statement, Catching more than one Exception, The Finally Clause, Generating Exceptions, Threads: Introduction, Creating Threads in Applications, Method in Thread Class, Threads in Applets.

Module-IV: Java APIs – overview of APIs, IO Packages, Java Input Stream Classes, Java Output Stream Classes, File Class, Graphic & Sound: AWT and Swing, Graphic methods, Fonts, Loading and Viewing Images, Loading and Playing Sound, AWT & Event Handling, Layouts, JDBC.

4. REFERENCES
4.1 Core
  ❖ *Java Programming*, *Schaum Outline Series*
4.2 Additional
  ▪ Deitel, *Java: How To Program*, Pearson Education
4.3 Internet resources:
  o http://javaboutique.internet.com/articles/ITJqanda/
  o http://java.sun.com/
  o http://freewarejava.com/
  o http://java.sun.com/developer/onlineTraining/

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1345 PROGRAMMING LAB – IV

1. AIM:
➢ To provide an opportunity for hands-on practice in Java.

2. OBJECTIVES:
This course will provide hands-on practice, under a variety of programming situations with a focus on writing, debugging and analysing object oriented programs:
- ✓ basic data types and control structures in Java
- ✓ installing and using JDK
- ✓ writing applications and applets
- ✓ managing classes and objects in a variety of situations
- ✓ using i/o, string, threads and net APIs
- ✓ solving moderately complex problems involving the above.

3. SYLLABUS
The laboratory work will consist of 15-20 Experiments

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

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

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

1. Programs to demonstrate methods of string class

2. Program to demonstrate AWT/Swing graphic methods

3. Program for Loading and Viewing Images, Loading and Playing Sound

4. Programs to demonstrate various Layouts

17-18 Programs to demonstrate event handling

19. Program to demonstrate simple server-client (using a single m/c both as client and server)

20. Debugging programs involving syntactic and/or logical errors

4. INTERNET RESOURCES
- o http://java.about.com/od/idesandeditors/
- o http://www.programmingtutorials.com/java.aspx

CP1346 NUMERICAL METHODS LAB

1. SYLLABUS
PART A
- Solutions of Non-linear equations:
  - Bisection method;
  - Regula-Falsi;
  - Newton-Raphson;
  - Secant;
  - Sucessive approximation method;
  - Schroder's Method;
  - Bairstrow's method
- Interpolation techniques:
  - Linear Interpolation;
  - Newton's forward and backward formulae;
  - Lagrange's interpolation;
  - Linear Regression; Cubic Splines; Chebyshev Polynomial.
- Numerical Integration & Differentiation:
  - Cubic Spline based Numerical Differentiation;
  - Numerical Integration:
  - Trapezoidal,
  - Simpson's
  - Gaussian, and Filon's methods.
- Matrix Computations:
  - Addition, subtraction, transpose, multiplication of matrices;
  - Matrix-based solutions of simultaneous linear equations:
  - Gauss Jordan Method;
  - Gauss elimination with back-substitution;
  - LU Decomposition method

PART B
- Differential Equations:
  - Ordinary Differential Equations:
  - Picard's method;
  - Euler's and modified Euler's method;
  - Runge-Kutta; Predictor-Corrector methods;
- Partial Differential Equations:
  - Jacobi and Gauss-Siedel methods.
- Sorting (Any two of student's choice and comparison)
  - Bubble;
  - Shell;
  - Quick;
  - Selection;
  - Heap;

SEMESTER FOUR

Semester 4			Hrs per week			
Course code	**Credits**	**Course Name**	**Lecture**	**Tutorial**	**Lab**	**Total**
CP1441	2	Introduction to Information Security	3	-	-	3
CP1442	3	Visual Tools	3	-	-	3
CP1443	3	Database Management Systems	3	1	-	4
CP1444	3	Design and Analysis of Algorithms	3	1	-	4
CP1445	3	System Software	3	-	-	3
CP1446	3	Visual Tools Lab	-	-	4	4
CP1447	3	Databases Lab	-	-	4	4
TOTAL	**20**		**15**	**2**	**8**	**25**

CP1441 INTRODUCTION TO INFORMATION SECURITY

1. AIM:
- ✓ To introduce internetworking and the issues and methods of information security over internetworks.

2. OBJECTIVES:
On completion of this course student shall:
- ✓ Be aware of principles and protocols of internetworks
- ✓ Understand the basic issues in information security
- ✓ Understand the concept of ciphers and cryptography.
- ✓ To impart an idea on various ciphers
- ✓ Understand the concept of digital signatures and e-mail security policies
- ✓ Impart an idea on malicious softwares and remedies.

3. SYLLABUS

Module I: Information Security: Network security, Confidentiality, integrity, authentication, security policy, basic network security terminology, cryptography, symmetric encryption, substitution ciphers, transposition ciphers, steganography, Block ciphers, modes of operation, Data Encryption Standard, Public key cryptography, applications, strength and weakness, RSA algorithm, key distribution (concepts only).

Module II: Authentication, authentication methods, message digest, digital signatures, digital signature algorithm, DSS, E-mail security: Pretty Good Privacy, working of PGP, S/MIME, MIME, IP Security, Architecture, IPSec: strengths and benefits, IPv4, IPv6, ESP protocol, Web Security: Secure Socket layer, SSL session and connection.

Module III: Malicious Software, viruses, working of anti-virus software, worms, Trojans, spyware, firewall, characteristics of firewall, packet filters, application level gateways, firewall architecture, trusted systems.

Module IV: Security and Law:- Regulations in India. Information Technology Act 2000/2008. Cyber Crime and the IT Act 2000/2008. Indian Contract Act 1872, Indian Penal Code, Indian Copyright Act, Consumer Protection Act. Future Trends – The Law of Convergence.

Assignments and activities: AES, Blowfish algorithms, Kerberos, Comparison of PGP and S/MIME, study of common malicious software, antiviruses.

4. REFERENCES
4.1 Core
- ❖ Brijendra Singh, *Cryptography & Network Security*, PHI.
- ❖ Pachghare, V.K., *Cryptography and Information Security,* PHI.

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1442 VISUAL TOOLS

1. AIM:
- ✓ To get an exposure to visual language

2. OBJECTIVES:
At the end of this course, the students will be able to:
- ✓ Give an introduction about visual basic
- ✓ Explain the fundamentals of visual basic
- ✓ Discuss the various controls in visual basic
- ✓ Narrate database connectivity in visual basic

3. SYLLABUS

Module I: Introduction Visual Basic IDE - Menu bar, Tool bar, Tool Box, Form designer, Code Editor, Properties Window, Form Layout, Project explorer, Immediate Window, Locals Window and Watch Window; Features in Visual Basic 6.0: The Language Object Based Programming, Different Data Access Models, Internet enabled Features. Wizards. **Visual Basic Language:** Variables, Constants, Data Types, User Defined Data Types, Scope of Variables, - Life Time of a Variables, Naming conventions in VB, Arrays, Dynamic Arrays, Collections, procedures,- Calling Procedures, Subroutines, Functions, calling Functions, Arguments, Named and Positional arguments, Control flow Statements, Loop Statements, Exit Statement.

Module II: Visual Basic Standard ActiveX Controls & Objects: Standard Controls: TextBox, CheckBox, ComboBox, Option Button, Massage Box, PictureBox, Image Control, Shape control, System Controls, Timer Control, OLE container Control, Common Dialogs control. Event driven Programming and an overview of Properties, Methods and Events, Important Methods, Properties and Events of Standard Active X controls. i.e., TextBox, ListBox, ComboBox etc. Creating Groups, Creating and Adding Project files. Standard EXE Project file, adding Form file. Designing Menu Objects. Single Document Interface and Multiple Documents Interface. Drag Drop Operations.

Module III: Input Validation: Over View of Validation Field Level Validation, Form Level Validation Masking. E**rror handling:** General Error Handling in Visual Basic, Built in VB Error Objects Error Types, Creating Error Handlers. **Compilation:** P-Code compilation, Native Code Compilation, Conditional Compilation, Optimization.

Module IV: Database Programming: Introduction to different databases. Data-bound controls. Data Control. Data Access Object Model. Accessing Native Database with DAO, ODBC connectivity to any RDBMS, Remote Data Access, Remote Data Control, Accessing ODBC database with RDO, **Data Report Designer:** Creating Data Reports, Various Types of Reports, **ActiveX Data Objects:** Active Data Object Model, Uses of ADO Model, Accessing ODBC, database with ADO. **Object Linking and Embedding:** Overview of OLE, In place Activation, Creating OLE Servers. **ActiveX Controls:** Ms Flex Grid, Flex Grid, Image List Control, List View Control, Tool Bar, Status Bar etc.

Activities and Assignment: Programming with Windows32 Application Programming Interface (API) Declaring and Calling DLL procedure, Usage of API Viewer application, Passing string and Arrays to DLLs procedure, Passing User Defined Types and Function Pointers to DLL procedure, Converting C Declarations to Visual Basic.

4. REFERENCES
4.1 Core
- ❖ Halvorson, Microsoft Visual Basic 6.0 Professional-Step by Step, PHI, Second Edition

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1443 DATABASE MANAGEMENT SYSTEMS

1. AIM:
- ❖ To introduce basic concepts of data bases, and related techniques and tools

2. OBJECTIVES:
- ✓ Be aware of basic concepts of data bases and data base management systems
- ✓ Be aware of concepts of relational data bases.
- ✓ Know to normalize relational data bases
- ✓ Skilled in using relational algebra and relational calculus
- ✓ Develop skills to write database queries

3. SYLLABUS

Module–I: Introduction: evolution of data base systems, overview of database management systems, Relational data model, mathematical definition, candidate, primary and foreign keys, set operations on relations, insertion, deletion and update operations, attribute domains.

Module-II: Relational algebra and relational calculus, Introduction to SQL, Table creation, selection, projection and join using SQL

Module-III: Functional Dependencies – Inference axioms, normalization, 1NF, 2NF, 3NF and Boyce-Codd Normal forms, Lossless and lossy decompositions.

Module-IV: The E-R Model, Entities and attributes, 1-1 and many-1, many-many relationships. Security – Physical and Logical, Design and maintenance issues, integrity.

Assignments and activities: Study of features of MS Access, Open Office Base, Oracle, mySQL, emerging areas.

4. REFERENCES
4.1 Core
- ❖ Ramon A. Mata-toledo and Pauline K. Cushman, *Fundamentals of Relational Data Bases*, Schaum Outlines, Tata McGraw Hill
4.2 Additional
- ▪ Atul Kahate, *Introduction to Data Base Management Systems*, Pearson Education
4.3 Internet resources:
- o www.pearson.co.in/AtulKahate,
- o www.edugrid.ac.in/webfolder/courses/dbms/dbms_indEX.htm

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1444 DESIGN AND ANALYSIS OF ALGORITHMS

1. AIM:
> ➢ To make students able to devise and analyze new algorithms by themselves.

2. OBJECTIVES:
On completion this course, student should:
> ➢ Be able to analyse the complexity of algorithms
> ➢ Be able to select good algorithms from among multiple solutions for a problem
> ➢ Have better knowledge on fundamental strategies of algorithm design
> ➢ Have better awareness on complex algorithm design strategies
> ➢ Implement some typical algorithms

3. SYLLABUS

Module–I: Algorithm Analysis: properties of a good algorithm, efficiency considerations, time complexity, space complexity, Asymptotic notations: Big O notation, best case, worst case, average case, simple examples, recursion and its elimination- recursive and no-recursive algorithms for binary search.

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

Module–III: Dynamic programming: principle of optimality, all pair shortest paths, single source shortest paths, travelling sales person's problem Back tracking: implicit constraints and explicit constraints, 8 queen's problem, Branch and bound: LC search

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

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

4. REFERENCES
4.1 Core
> ❖ Anany Levitin, *Introduction to design and analysis of algorithms*, Pearson, Second Edition

4.2 Additional
> ▪ Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekharan – *Computer Algorithms / C++*, Second Edition- Universities Press.

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1445 SYSTEM SOFTWARE

1. AIM:
  ➢ Provide an overall picture of the system related software

2. OBJECTIVES:
At the end of the course, the students should be able to
  ✓ Explain the internal working of the system
  ✓ Discuss the principles of assemblers
  ✓ Narrate the working of loaders and linkers
  ✓ Discuss system development tools

3. SYLLABUS:

MODULE I: INTRODUCTION System software and machine architecture – The simplified Instructional Computer (SIC) - Machine architecture - Data and instruction formats - addressing modes - instruction sets - I/O and programming.

MODULE II: ASSEMBLERS Basic assembler functions - A simple SIC assembler – Assembler algorithm and data structures - Machine dependent assembler features - Instruction formats and addressing modes – Program relocation - Machine independent assembler features - Literals – Symbol-defining statements – Expressions - One pass assemblers and Multi pass assemblers.

MODULE III: LOADERS AND LINKERS Basic loader functions - Design of an Absolute Loader Machine dependent loader features - Relocation – Program Linking – Algorithm and Data Structures for Linking Loader - Machine-independent loader features – Automatic Library Search – Loader Options - Loader design options - Linkage Editors – Dynamic Linking – Bootstrap Loaders

MODULE IV: MACROPROESSOR AND SYSTEM SOFTWARE TOOLS Basic macro processor functions - Macro Definition and Expansion – Macro Processor system software tools, Text editors - Overview of the Editing Process - User Interface – Editor Structure. -Interactive debugging systems - Debugging functions and capabilities – Relationship with other parts of the system – User-Interface Criteria.

4. REFERENCES
4.1 Core
  ❖ Leland L. Beck, *System Software – An Introduction to Systems Programming*, 3rd Edition, Pearson Education Asia, 2006.
4.2 Additional
  ▪ D. M. Dhamdhere, *Systems Programming and Operating Systems*, Second Revised Edition, Tata McGraw-Hill, 2000.

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1446 VISUAL TOOLS LAB

The laboratory work may consist of 15-20 experiments covering the following topics

1. Introduction to visual basic IDE- demonstration on various types of windows in IDE

2. Mouse, keyboard, focus and Scroll events.

3. Demonstrate the syntax and use of various VB data types, operators and control structures

4. Demonstration and use arrays and its types- (static, dynamic and control arrays)

5. Implementation of Functions, Procedures and Sub routines

6. Implementation of various standard activeX controls- Text box, list box, Check box, Combo box, option button, picture box, image box, timer, OLE control and message box

7. Programs to demonstrate various methods, events and properties of activeX controls

8. Program to implement the Menu Editor

9. Implementation of Single and Multiple Document Interface

10. Implementation of Input validation(form and field level)

11. Implementation of Error handling in VB

12. Program to implement Data bound controls

13. Implementing ADO and RDO to access database.

14. Creating and generating data reports using report designer

CP1447 DATABASES LAB

1. AIM:
This course will provide hands-on practice in the following topics, under a variety of computing situations with a focus on writing and analysing SQL statements:

➢ Installing and configuring a proper SQL tool
➢ Database design and implementation
➢ Writing and analysing SQL statements
➢ Create user interface (using java AWT) and study the working of a data base in a front end application

2. SYLLABUS
The laboratory work will consist of 15-20 Experiments. Tools to be used include: Personal Oracle 8/ MS Access/OpenOffice Base/Java. Experiments will cover creating tables including defining relations between them, practicing SQL, Experiments designed around a case study, miscellaneous topics including security, connecting databases to front-end applications. Some sample topics are given below:

Part A
1. SQL statement for creating, listing, dropping, checking, updating tables
2. Record manipulation using-insert, delete, update
3. Experiments that clarify the importance of keys (Except foreign key)
4. Queries with an Expression and a column alias
5. A simple query that aggregates (groups) over a whole table
6. A query with a literal string in the SELECT list
7. Queries with sub string comparison and ordering
8. Query using the "IS NULL" syntax to list (compare '=NULL' instead of IS NULL")
9. Finding values within a certain range
10. Using the --"BETWEEN" keyword

Part B
11. A Join between two tables (foreign key)
12. Nested queries
13. The EXISTS and UNIQUE function in SQL
14. Renaming attributes and joined tables
15. Statements related with VIEWs

SEMESTER FIVE

Semester 5			Hrs per week			
Course code	Credits	Course Name	Lecture	Tutorial	Lab	Total
CP1541	3	Free and Open Source Softwares (FOSS)	3	-	-	3
CP1542	3	Computer Graphics	3	1	-	4
CP1543	3	Internet Programming	3	1	-	4
CP1551.1 CP1551.2 CP1551.3	2	**Open Course** Internet Technology LINUX Environment Business Informatics	3	-	-	3
CP1544	3	Management Information Systems	3	-	-	3
CP1545	3	Internet Programming Lab	-	-	4	4
CP1546	3	Computer Graphics Lab	-	-	4	4
TOTAL	20		15	2	8	25

CP1541 FREE AND OPEN SOURCE SOFTWARES (FOSS)

1. AIM:
➢ To introduce different free and open source softwares

2. OBJECTIVES:
At the end of this course, the students will be able to
- ✓ Explain the features of free & open source software
- ✓ Familiarization with LINUX
- ✓ Work with PHP
- ✓ Demonstrate the working of MySQL

3. SYLLABUS
Module I: Open source software: Features, advantages over proprietary software, examples**, Free software:** concepts, features, Free software Vs Open Source software, Free software movements. Policies, GPL, Free OS, History and Features of Linux, Various flavours of Linux, Linux Kernel and Shell, Graphical Desktops- GNOME, KDE, Linux File System and Directories

Module-II: The building blocks of PHP: variables, globals & super globals **Data types**: Settype, type casting, test type, **Operators & Expressions**, **Flow control functions in PHP**, **Functions**: Defining a function variable scope, calling a function, returning values ,setting default values for arguments, passing variable reference **Arrays**: creating arrays(associative & multidimensional), Array related functions **Working with strings:** Formatting strings, indexing, strlen() functions

Module-III: Forms in PHP: Creating a simple input form, combining HTML & PHP code on a single page, redirecting the user ,creating a send mail form, File upload form **Cookies**: Introduction, setting a cookie with PHP, deleting a cookie, session function overview: starting a session, working with session variables, passing session IDs in the query string, destroying sessions & unsetting variables

Module-IV: Database concepts: Open source database software: MySQL features MySQL data types: Numeric, date & time, string Table creation in MySQL: insert, select, where clause, ordering the result, like operator Selecting Multiple tables: using join, using queries Modifying records: update command, replace command, delete command date & time functions in MySQL Interacting with MySQL using PHP: connecting to MYSQL ,Executing queries, Retrieving error messages, inserting data with PHP, retrieving data with PHP

4. REFERENCES
4.1 Core
- ❖ Julie C.Meloni, PHP, MySQL and Apache, Pearson Education
- ❖ Ivan Byross, HTML, DHTML, Javascript, Perl, BPB Publication

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1542 COMPUTER GRAPHICS

1. AIM:
> ➢ To introduce basic theoretical underpinnings and concepts behind computer graphics and Expose student to algorithms, tools and techniques for implementing the same.

2. OBJECTIVES:
On completion of this course, students should be able to:
- ✓ handle basic graphic primitives in C/C++ for developing 2D and 3D graphics
- ✓ program basic scan-conversion algorithms
- ✓ apply various transformations to 2D and 3D graphic objects
- ✓ derive various projections of 3D objects
- ✓ give realistic rendering to 3D wireframe objects
- ✓ be familiar with current trends in computer graphics

3. SYLLABUS

Module I: Introduction: graphic data representation, concept of pixels, resolution, aspect ratio, Raster scan display, Random Scan display, video adapter, frame buffer, display technology-CRT, LCD, LED, smart devices (feature wise comparison only), Output Primitives: Straight Line, DDA algorithm, Bresenham's Line Algorithm, Circle- Mid Point Circle Algorithm, polygon filling algorithms- boundary fill, scan-line algorithm, Aliasing and Anti-aliasing.

Module II: Two dimensional Transformations: Translation, scaling, fixed point scaling, rotation, reflection, transformation with respect to arbitrary points. Application of homogeneous coordinates for uniform matrix operations, composite transformations, Windowing and clipping: Window to viewport transformation, Clipping- Point clipping, Line Clipping, Cohen-Sutherland Line Clipping algorithms, Polygon Clipping-Sutherland-hodgeman algorithm.

Module III: 3D Concepts and Techniques: 3D display techniques, 3D Transformations, 3D modelling schemes, Projection-parallel projections, perspective projection, Visible Surface Detection- Hidden Surface removal Algorithms-scan line method, Z-buffer method.

Module IV: Colour Illumination methods: color models-RGB, HSI, CMYK, Illumination model and light sources, Specular reflection, Intensity attenuation, shadow, Polygon Shading methods, animation**,** morphing - tweening, warping (Concepts only) zooming, panning, rubber band lines (concepts only)

4. REFERENCES
4.1 Core
> ❖ Amarendra N Sinha and Arun D Udai, *Computer Graphics*, Mc Graw Hill publications

4.2 Additional
> ▪ Donald Hearn, M. Pauline Baker, *Computer Graphics (C Version) 2/e,* Pearson

4.3 Internet resources:
- o www.prenhall.com/hearn,
- o www.prenhall.com/bookbind/pubbooks/hill4
- o www.povray.org ray tracing and 3D morphing,
- o www.cs.unc.edu/~pxpl/home.html

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1543 INTERNET PROGRAMMING

1. AIM:
➢ To Expose students to technology of web sites and to introduce various tools and languages required for technical and creative design of state-of-the-art web sites

2. OBJECTIVES:
To impart basic skills in moderately complex use of the following tools/scripts/languages:
- ✓ HTML, DHTML, CGI Script, Perl, CSS, Java script, ASP and JSP.
- ✓ To impart necessary ability to choose the appropriate web tools/languages for creating state-of-the art web sites
- ✓ To Expose students to current trends and styles in web design and applications

3. SYLLABUS

Module–I: HTML: General Introduction to Internet and WWW; Text tags; Graphics, Video and Sound Tags; Link and Anchor Tags; Table Tags; Frame Tags; Miscellaneous tags (layers, image maps etc); CSS; DHTML; Example Applications; simple introduction to XML and VRML

Module–II: CGI Programming: HTML Forms and Fields; Perl: Basic control structures, data types and basic features; CGI Programs: GET & POST methods, simple applications; Cookies; Server Side Includes; Example Applications;

Module–III: Javascript: Basic data types; control structures; standard functions; arrays and objects, event driven programming in Javascript; Example Applications;

Module–IV: Architecture of java Servelets; Servelet Structure; Servelet Life Cycle; Request and Response Objects; Sessions; Invoking Servelets;

Assignments and Activities: JDBC; PHP; .NET Technology; C#; Creative Design of Web sites; Macromedia flash, Web Servers, Web databases, Web Administration and Maintenance.

4. REFERENCES
4.1 Core
- ❖ V.K. Jain, *Advanced Programming in Web Design*, Cyber Tech Publications
4.2 Additional
- ▪ Joel Sklar, *Principles of Web Design*, Vikas
- ▪ H M Deitel, P J Deitel & A B Goldberg, *Internet and Worldwide web programming: How to Program*, 3/e, Pearson Education
4.3 Internet resources:
- o ww.learnasp.com/learnasp/
- o http://notes.corewebprogramming.com/
- o www.rh.edu/~heidic/webtech/notes/
- o www.redbrick.dcu.ie/help/slides/week7_perl/perl.ppt.

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1551 OPEN COURSE
CP1551.1 INTERNET TECHNOLOGY

1. AIM:
- ✓ Give an introduction about the components of internet, its working and the way in which web pages are designed.

2. OBJECTIVE:
At the end of this course, the students will be able to
- ✓ Discuss various components of internet
- ✓ Explain different devices used for networking
- ✓ Explain the working principle of Internet
- ✓ Design web pages using HTML

3. SYLLABUS:

MODULE I - Introduction to Computer Networks- Advantages of Networks, Goals of Networks, Types of Networks- LAN,MAN,WAN, Internet , Public Networks, LAN topologies-Bus, Star, Ring, Mesh.

MODULE II - Networking Devices- Interconnecting Issues, Connectivity Devices, Hubs, Switch, Bridges, Routers.

MODULE III - Introduction to Internet -Meaning of Internet, WWW- History, Working of Internet, Browsing, Searching the Web, Internet protocols- TCP/IP Protocol suite, UDP, IP addresses, IP Versions – IPV4, IPV6, Services of the Internet- FTP, HTTP, Email.

MODULE IV - HTML- Understanding HTML, Text tags; Graphics, Video and Sound Tags; Link and Anchor Tags; Table Tags; Frame Tags; Miscellaneous tags (layers, image maps etc);

Assignments and Activities

4. REFERENCES
4.1 Core
- ❖ Douglas E Comer, *Computer Networks and Internets*, 4/e, Pearson Education

4.2 Additional
- ▪ Andrew S. Tanenbaum, *Computer Networks*, 4/e, Pearson Education

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1551.2 LINUX ENVIRONMENT

1. AIM:
- ✓ To familiarize with Linux working environment

2. OBJECTIVES:
- ✓ Introduction to Operating Systems
- ✓ Introduction to linux
- ✓ Introduction to OpenOffice.org

3. SYLLABUS

Module I- Operating Systems- Necessity of OS, Types of OS-Batch Systems, Time Sharing Systems, Real time Systems, Basic Structure of An OS- Kernel, Shell, File System, OS as a Resource Manager, General principles of Resource Management

Module II- Introduction to Linux- History and Features of Linux, Various flavours of Linux, Linux Kernel and Shell, Graphical Desktops- GNOME, KDE, Linux File System and Directories, Linux commands bc, cal, cat, cd, chgrp, chmod, clear, cmp, cp, kill, rm, rmdir, tty, wc, who, grep, write, telnet, whois, mv, find, ps, mkdir, more, date, mount, show, mount etc. Pipeline and redirection concepts, using floppy and cd- rom in linux

Module III-Open Office.org-Open Office Writer-Parts of the OpenOffice.org Window, Editing and Writing a Writer document, spell checker, autocorrect, Thesaurus, create table, table formatting, finding items in a document, header and footer, create and modify page numbers, adding graphics, borders and colours

Module IV- Open Office.org-Open Office Calc- Entering data in a spreadsheet, spreadsheet math, columns, lookup functions, charting data, Open Office Impress- Create a new presentation , insert, copy and delete slides, formatting text, bulleted and numbered lists, adding clipart, pictures, charts and spreadsheets, slide settings and transitions, animating slides, previewing and running a slideshow

Assignments and Activities: Packages in Linux, Case study of open source softwares, comparison of Linux with Windows

4. REFERENCES
4.1 Core
- ❖ Pramod Chandra P Bhat, *An Introduction to Operating Systems: Concepts and Practice,* 2/e PHI,2007

4.2 Additional
- ▪ Richard Peterson, *Linux Programming: A Beginners Guide*, DreamTech

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1551.3 BUSINESS INFORMATICS

1. AIM:
  ➢ To create an awareness about role of IT in business and to introduce concepts and techniques of e-commerce

2. OBJECTIVES:
By the end of this course, the student should be able to:
  ✓ Have an awareness about role of IT in business
  ✓ Have knowledge of basic concepts of e-commerce
  ✓ Be aware of different types of e-commerce web sites and different modes of payments
  ✓ Be aware of security and legal issues in e-commerce

3. SYLLABUS

Module–I: History of e-commerce, definition, classification- B2B, B2C, C2C, G2C, B2G sites, ecommerce in education, financial, auction, news, entertainment sectors, Doing e-Commerce.

Module–II: Electronic payment systems – relevance of currencies, credit cards, debit cards, smart cards, e-credit accounts, e-money, security concerns in e commerce, authenticity, privacy, integrity, non-repudiation, encryption, secret key cryptography, public key cryptography, digital signatures, firewalls

Module–III: Mass marketing, segmentation, one-to-one marketing, personalization and behavioural marketing, web advertising, online advertising methods, advertising strategies and promotions, special advertising and implementation topics.

Module IV- Mobile Commerce: attributes and benefits, Mobile Devices, Computing software, Wireless Telecommunication devices, Mobile finance applications, Web 2.0 Revolution, social media and industry disruptors, Virtual communities, Online social networking: Basics and examples, Web 3.0 and Web 4.0, Civil law, intellectual property law, common law and EC legal issues

Assignments and Activities: Case study of two internationally successful e-commerce web sites and two Kerala-based e-commerce web sites; IT act (India) and e-commerce.

4. REFERENCES
4.1 Core
  ❖ Erfan Turban et.al., *Electronic Commerce–A Managerial Perspective*, Pearson Education
4.2 Additional
  ▪ R Kalokota, Andrew V. Winston, *Electronic Commerce – a Manger's guide*, Pearson
4.3 Internet resources:
  o www.ecommercetimes.com,
  o www.online-commerce.com,
  o www.rsa.com,
  o www.ntsecurity.com
  o www.easystorecreator.com/ecommercetutorial.asp

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1544 MANAGEMENT INFORMATION SYSTEMS

1. AIM:
- To introduce role, types, techniques and tool used in various information systems

2. OBJECTIVES:
At the end of this course, the students will be able to
- ✓ Define the components of information systems
- ✓ Discuss merits and demerits of information systems
- ✓ Explain optimization techniques used in information systems

3. SYLLABUS

Module I: An introduction to information systems, Information systems in organizations, Information Technology Concepts, The IS Revolution; Information requirement for the different levels of management, transaction processing system, Management information system, Decision support system. Strategic Role of Information Systems. Business Processes; Information management, and Decision Making. Computers and Information Processing;

Module II: Transaction processing system; hardware and software requirements, tools used, case studies, merits and demerits of transaction processing system, Managerial control, Information and tools required, difference between transactional system and managerial system. Frequency of taking outputs, Need for interconnected system, common database, Redundancy control, case studies.

Module III: Decision support system, concept and tools, case studies, virtual organizations, strategic decisions-unstructured approach, cost and values of unstructured information.

Module IV: Optimization techniques, difference between optimization tools and DSS tools expert system, difference between expert system and management information system. Role of chief information officer.

4. REFERENCES:
4.1 Core
- ❖ Rajaraman, *Management Information Systems*
4.2 Additional
- ▪ S. Sadagopan, *Management Information Systems*, Prentice-Hall of India
- ▪ Jawadekarw. S, *Management Information Systems*, Tata McGraw-Hill

CP1545 INTERNET PROGRAMMING– LAB

1. AIM:
  ➢ To give hands-on Exposure to various tools and languages required for technical and creative design of web sites

2. OBJECTIVES:
To practice moderately complex use of the following scripts/languages/technologies:
  ✓ HTML, DHTML, CSS,
  ✓ Java script,
  ✓ CGI Script, Perl,

3. SYLLABUS
The laboratory work will consist of 15-20 Experiments
Part A (HTML)
Part B (Javascript, XML, Perl, CGI)
  1. Practicing basic HTML tags, text tags test styles, paragraph styles, headings, lists
  2. Tables in HTML, Frames in HTML, nested frames, Link and Anchor Tags
  3. Including graphics, video and sound in web pages, including Java applets
  4. Layers & Image Maps
  5. Creating animated Gifs, simple flash animations
  6. Cascading Style sheets
  7. DHTML
  8. Creating and browsing XML database
  9. Installing VRML plugins and viewing VRML source files
  10. HTML forms and Fields
  11. Exercises covering basic introduction to perl
  12. Installing web server, setting CGI, connecting HTML forms to Perl Scripts (CGI programming)
  13. Exercises covering basic introduction to Javascript

  14-20: Development of a web site involving a variety of tools practiced above

4. REFERENCES
4.1 Core
  ❖ V.K. Jain, *Advanced Programming in Web Design*, Cyber Tech Publications

4.2 Additional
  ▪ H M Deitel, P J Deitel & A B Goldberg, *Internet and Worldwide web programming: How to Program*, 3/e, Pearson Education

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1546 COMPUTER GRAPHICS LAB

1. AIM:
- ✓ To provide hands-on Exposure to tools, techniques and algorithms in computer graphics

2. OBJECTIVES:
In this course, students shall:
- ✓ implement basic scan-conversion algorithms
- ✓ implement clipping algorithms
- ✓ implement various transformations to 2D and 3D graphic objects
- ✓ implement orthographic and perspective projections of 3D objects
- ✓ create 3D wireframe objects

3. SYLLABUS
The laboratory work will consist of 10-15 Experiments

Part A
1. Implementing DDA & Bresenham algorithm for line drawing, effecting different line styles
2. Implementing circle drawing algorithms, drawing ellipses and sectors
3. Representing 2D object data files (containing vertex and edge lists) and implementing programs which read and plot these objects.
4. Implementing 2D transformations (programs which prompt for type of transformation, parameters and name of object data file and plot object and transformed object in 2 colors)
5. Implementing composite transformations (modification on the above, program prompts for number of transformations, accepts parameters for each and then plots all stages of transformations in different colors

Part B
6. Implementing Windowing and Clipping algorithms
7. Implementing a filling algorithm, reading the object from data file
8. Representing 3D object data files (containing vertex and edge lists) and implementing programs which read and plot these objects.
9. Implementing 3D transformations (programs which prompt for type of transformation, parameters and name of object data file and plot object and transformed object in 2 colors)
10. Implementing composite transformations
11. Implementing hidden surface removal by surface normal computation: to be tried out on a cube and/or a sphere

4. REFERENCES
4.1 Core
- ❖ Malay K. Pakhira, *Computer Graphics Multimedia and Animation,* PHI, 2008
- ❖ D P Mukherjee, *Fundamentals of Computer Graphics and Multimedia*, PHI
- ❖ Peter Cooley, *The Essence of Computer Graphics*, Pearson Education

SEMESTER SIX

Semester 6			Hrs per week			
Course code	Credits	Course Name	Lecture	Tutorial	Lab	Total
CP1641	2	Business Informatics	3	-	-	3
CP1642	4	Object Oriented Analysis and Design	3	1	-	4
CP1643	4	Data Mining & Data Warehousing	3	-	-	3
CP1661.1 CP1661.2 CP1661.3	4	**Elective** Bioinformatics Trends in Computing Software Testing	3	-	-	3
CP1644	3	Free and Open Source Software (FOSS) Lab	-	-	4	4
CP1645	3	Major Project & VIVA	-	-	8	8
TOTAL	20		12	1	12	25

CP1641 BUSINESS INFORMATICS

1. AIM:
> To create an awareness about role of IT in business and to introduce concepts and techniques of e-commerce

2. OBJECTIVES:
By the end of this course, the student should be able to:
- ✓ Have an awareness about role of IT in business
- ✓ Have knowledge of basic concepts of e-commerce
- ✓ Be aware of different types of e-commerce web sites and different modes of payments
- ✓ Be aware of security and legal issues in e-commerce

3. SYLLABUS

Module–I: History of e-commerce, definition, classification- B2B, B2C, C2C, G2C, B2G sites, ecommerce in education, financial, auction, news, entertainment sectors, Doing e-Commerce.

Module–II: Electronic payment systems – relevance of currencies, credit cards, debit cards, smart cards, e-credit accounts, e-money, security concerns in e commerce, authenticity, privacy, integrity, non-repudiation, encryption, secret key cryptography, public key cryptography, digital signatures, firewalls

Module–III: Mass marketing, segmentation, one-to-one marketing, personalization and behavioural marketing, web advertising, online advertising methods, advertising strategies and promotions, special advertising and implementation topics.

Module IV- Mobile Commerce: attributes and benefits, Mobile Devices, Computing software, Wireless Telecommunication devices, Mobile finance applications, Web 2.0 Revolution, social media and industry disruptors, Virtual communities, Online social networking: Basics and examples, Web 3.0 and Web 4.0, Civil law, intellectual property law, common law and EC legal issues

Assignments and Activities: Case study of two internationally successful e-commerce web sites and two Kerala-based e-commerce web sites; IT act (India) and e-commerce.

4. REFERENCES
4.1 Core
- ❖ Erfan Turban et.al., *Electronic Commerce–A Managerial Perspective*, Pearson Education
4.2 Additional
- ▪ R Kalokota, Andrew V. Winston, *Electronic Commerce – a Manger's guide*, Pearson
4.3 Internet resources:
- o www.ecommercetimes.com,
- o www.online-commerce.com,
- o www.rsa.com,
- o www.ntsecurity.com
- o www.easystorecreator.com/ecommercetutorial.asp

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1642 OBJECT ORIENTED ANALYSIS AND DESIGN

1. AIM:
➢ Provide an environment for the students to design projects using object oriented approach

2. OBJECTIVES:
At the end of this course, the students will be able to
✓ Discuss various OOA approached
✓ Describe the concept of unified modelling language
✓ Explain different diagrams used

3. SYLLABUS

Module I: Object Oriented Concepts, class, object, defining a class, Comparison between Algorithmic Decomposition and Object Oriented Decomposition, Object Oriented Themes: Abstraction, Encapsulation, Inheritance, polymorphism.

Module II: Concept of unified modelling language, Object Oriented Analysis and Design using UML, UML Diagrams: Class Diagram, representing various features of a class, messages, use case diagram, identifying use cases, examples.

Module III: interaction diagrams: Sequence Diagram, examples, elements of a sequence diagram, system-level and service level diagrams, benefits of sequence diagram, Collaboration Diagram, elements examples

Module IV: Activity Diagram, State Chart Diagram, Component Diagram, modelling interfaces, Deployment Diagram, elements, examples.

Assignment & Activities Case study: generating UML diagrams for a system.

4. REFERENCES:
4.1 Core
❖ Ali Bahrami - *Object Oriented Systems Development*, McGrawHill
4.2 Additional
❖ Grady Booch, *Object Oriented Analysis and Design*, Addison Wesley, Pearson
❖ Edward Yourdon , Carl Argila, *Case Studies in Object-Oriented Analysis and Design*
❖ Joey F. George, Dinesh Batra, Joseph S. Valacich , Jeffrey A. Hoffer, *Object-Oriented Systems Analysis and Design*

NB:- Activities and assignments are not meant for End_Semester_Examination

CP1643 DATA MINING & DATA WAREHOUSING

1. AIM:
- ✓ To get an entry-level understanding of the concepts of Data Mining

2. OBJECTIVES:
- ✓ To get an understanding of the general properties of data in large databases
- ✓ Understand a variety of real-world applications that require mining
- ✓ To get an overview of data warehousing and different data mining techniques
- ✓ How to discover useful patterns and associations in huge quantities of data

3. SYLLABUS

Module I: Overview: Data, Information, Knowledge; Knowledge Discovery; Types of data for Mining; Application Domains; Data Mining Functionalities; Data Processing – Understanding Data, Pre-processing Data –Forms of Data Pre-processing, Data Cleaning (Definition and Phases only), Need of Data Integration, Steps in Data Transformation, Need of Data Reduction;

Module II: Data Warehouse: Database Systems & Data Warehouses – Difference; Data Warehouse – Definition & Features; Multidimensional Data Model – Data Cubes ; OLAP (Definition and Functions only); Market Basket Analysis; Association Rule – Overview; Criteria for classifying Frequent Pattern Mining ; Mining Single Dimensional Boolean Association Rule – Apriori Algorithm;

Module III: Classification: Classification vs Prediction; Issues; **Use of** Decision Trees for Classification; Bayesian Classification – Bayes' Theorem, Naïve Bayesian Classifier; Lazy Learners - k–Nearest Neighbour Method; Rule-Based Classification – Using IF-THEN rules for classification;

Module IV: Cluster Analysis: Introduction & Requirements; Characteristics of Clustering Techniques; Types of Data in Cluster Analysis; Categories of Clustering - Partitioning Methods; Outlier Detection in Clustering;

Activities and Assignments: Mining Web, Temporal, Text, Multimedia, Medical data and other Applied Data Mining areas; OLAP tools; Introduction to RapidMiner and other free and open-source data mining tools;

4. REFERENCES

4.1 Core
- ❖ Sunitha Tiwari & Neha Chaudhary, *Data Mining And Warehousing*, Dhanpat Rai & Co

4.2 Additional
- ▪ Jiawei Han And Micheline Kamber, *Data Mining Concepts And Techniques*, Elsevier
- ▪ Arun K Pujari, *Data Mining Techniques*, Universities Press
- ▪ G.K Gupta, *Introduction To Data Mining With Case Studies*, PHI

NB:- Activities and assignments are not meant for End_Semester_Examination

CP 1661 ELECTIVE:
CP 1661.1 BIOINFORMATICS

1. AIM:
- ✓ To motivate students towards the field of Biology where the service of IT professionals are much awaited.

2. OBJECTIVES:
On completion this course, the student should:
- ✓ Refresh the knowledge in Biology
- ✓ Develop ideas on representing the biological terms in Computer Science.
- ✓ Be aware of the developments in the emerging field of Bioinformatics.

3. SYLLABUS

Module I: Introduction: Aim & Scope of Bioinformatics; Biological foundations of Bioinformatics – Cell, Gene, Nucleic acids, Proteins, Structure of DNA , RNA and Proteins; Storage of Genetic Information; Central Dogma of Molecular Biology; Branches of Bioinformatics;

Module II: Biological Databases: *(Overview of databases only)* Primary Databases – Nucleotide Sequence databases (GenBank, DDBJ, EMBL); Protein Sequence databases (SWISS-PROT, PIR); Secondary Databases – PROSITE, PRINTS, BLOCKS ; Structure databases – PDB, SCOP, CATH; Metabolite database – KEGG; Literature database – PubMed; Data storage and Retrieval Tools – Entrez, SRS;

Module III: Sequence Alignment: *(Basics of sequence alignment and tools)* Introduction to Sequence Comparison - Pairwise Alignment and Multiple Sequence Alignment; Global & Local Alignments, , Gaps, Patterns of Substitution; Scoring Matrices – PAM, BLOSUM; Sequence comparison Tools – BLAST, FASTA; Prediction Tools – GENSCAN, SNP; Visualization Tools – RasMol, PyMol, SWISS-PDBViewer;;

Module IV: Related areas: Understanding Genomics, Proteomics, Pharmacogenomics, DNA Microarray; DNA Fingerprinting; Application of Bioinformatics in Computer-Aided Drug Design; Importance of Perl language in Bioinformatics;

Activities and Assignments: Search the web using PubMed, Retrieving DNA and Protein Sequences, Simple programs in Perl, Open-Source Bioinformatics Software;

4. REFERENCES
4.1 Core
- ❖ Selzer-Marhofer-Rohwer, *Applied Bioinformatics – an introduction*, Springer

4.2 Additional
- ▪ Dan-E-Krane, Michael.L.Raymer, *Fundamental Concepts of Bioinformatics*, Pearson Education
- ▪ Zhumur Ghosh, Bibekanand Mallick, *Bioinformatics – Principles and Applications*, Oxford Higher Education

NB:- Activities and assignments are not meant for End_Semester_Examination

CP 1661.2 TRENDS IN COMPUTING

1. AIM:

➢ Introduce advanced computing technologies and their application areas

2. OBJECTIVES:

✓ Understand the concepts of grid computing
✓ Basic idea on how users can log into different systems in the cloud and access software and hardware resources
✓ How problems with uncertainty, imprecision and partial truth could be solved using soft computing techniques

3. SYLLABUS

Module I: Grid Computing: *Basic Concepts*: Application areas; Grid Layered Architecture; Distributed Computing; Data Grids – Resource Sharing; Pathway to Grid Computing; **Cloud Computing –** Overview, Web 2.0 and the cloud, Cloud Types, Uses of Cloud; Components of Cloud Computing - Software as a Service, Platform as a Service, Infrastructure as a Service, Identity as a Service (**Concepts only**) ;

Module II: Data storage in the cloud: Understanding, Advantages and Disadvantages of Cloud-Based Data Storage; Disaster Recovery – understanding threats; Service-Oriented Architecture – understanding SOA, Web services;

Module III: Soft Computing: Soft Computing VS Hard Computing; Introduction to Neural Networks – Intelligence, Neurons, Artificial Neural Networks, Application Scope of Neural Network, Brain VS Computer, Problem areas, Training of Artificial Neural Networks – Supervised and Unsupervised; From ordinary sets to Fuzzy sets – Basics of Fuzzy Logic Theory, Foundations of fuzzy logic – Fuzzy Sets , Membership Functions;

Module IV: Evolutionary Algorithm: Traditional Algorithm VS Genetic Algorithm; Genetic Algorithm Operators – Reproduction (Roulette Wheel Selection, Tournament Selection), Crossover (one point crossover, two point crossover, uniform crossover), Mutation; Comparison of Operators; Genetic Algorithm Cycle; Applications;

Activities and Assignments: *Study of different Grid Projects, Migrating to Cloud, Mobile Cloud Computing, Cloud-based applications, Engineering and Industrial applications of Soft Computing, Support Vector Machine*

4. REFERENCES

4.1 Core

❖ Venkatakrishna& etal,*Principles of Grid Computing – Concepts And Applications*, Ane Books
❖ Kris Jamsa,*Cloud Computing*, Jones &Bartlett Learning.
❖ Rahul Deva & Garima Kulshreshtha, *Soft Computing*, Shrof Publishers & Distributors Pvt.Ltd.

4.2 Additional

▪ Rajkumar Buya and etal,*Cloud Computing – Principles And Paradigms*, Wiley Publishers.
▪ S.Rajasekharan & G.A.Vijayalakshmi Pai, *Neural Networks, Fuzzy Logic and Genetic Algorithms Synthesis and Applications*, PHI Learning Private Limited
▪ Manish Mahajan & Rajdev Tiwari,*Introduction to Soft Computing*, Acme Learning.
▪ Dilip K Pratihar,*Soft Computing – Fundamentals & Applications*, Narosa.
▪ S.N.Sivanandam, S.N.Deepa,*Principles of Soft Computing*, Wiley India.

NB:- Activities and assignments are not meant for End_Semester_Examination

CP 1661.3 SOFTWARE TESTING

1. AIM:
➢ To provide students the knowledge of testing softwares

2. OBJECTIVES:
At the end of this course, the students will be able to
- ✓ Discuss the basic concept of testing
- ✓ Explain different types of testing
- ✓ Describe the tools used for testing

3. SYLLABUS

Module-I: Introduction:- Purpose of testing, testing and debugging, model for testing, types of testing, types of bugs Flow graphs and Path testing:- Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation,

Module-II: Transaction Flow Testing:-transaction flows, example of transaction testing. Dataflow testing:- Basics of dataflow testing, example of dataflow testing, Domain Testing:- domains and paths, Nice & ugly domains, Interfaces testing

Module-III: Paths, Path products and Regular expressions:- path products & path expression, reduction procedure, regular expressions & flow anomaly detection

Module-IV: Logic Based Testing:- overview, decision tables, path expressions, kv charts, State Graphs and Transition testing:- good & bad state graphs, state testing,

Activity and Assignment: Case study: software testing techniques and examples. (JMeter or Win-runner).

4. REFERENCES:
4.1 Core
- ❖ Baris Beizer, *Software Testing techniques,* Dreamtech, second edition.
4.2 Additional
- ▪ Edward Kit, *Software Testing in the Real World*, Pearson Education of India.
- ▪ K.K.Prasad, *Software Testing Tools*, Dreamtech

NB:- Activities and assignments are not meant for End_Semester_Examination

CP 1644 FREE and OPEN SOURCE SOFTWARE (FOSS) LAB

PART A
Installing and Configuring Linux, Partition Creation, Familiarization of using basic Linux commands - cat with options, ls with options, mkdir, cd, rmdir, cp, mv, cal, pwd, wc, grep with options, I/O redirection using >,>>,<,| etc. Using Desktop-GNOME-KDE- Linux Commands Accessing and Running Applications.

Introduction to vi editor- Three modes in which vi editor works, Commands in vi input mode for inserting, replacing, saving and quitting, Commands in vi for deleting, paging and scrolling, Undoing last editing instructions, search and replace, Emacs Editor

PART B
Understand shell programming and use of different conditional statements in shell programming- Basic Shell Commands - Shell programming statements, operators and conditional statements

Administration: Understanding System Administration: Root login-super user-GUI tools, commands and Log files-Configuring Hardware-File System and Disk Management- Monitoring performances.

Any one application as case study (GiMP, SciLab, Moodle, Joomla etc.) (**for lab records only, not for ESE**)

REFERENCES
❖ Christopher Negus, *Red Hat Linux 9 Bible*, WILEY- Dreamtech, New Delhi,
❖ Thomas Schenk, *Red Hat Linux System Administration*, Techmedia, New Delhi, 2003.

CP1645 MAJOR PROJECT & VIVA

1. AIM:
 - To expose student to industry-standard project practices, through a real-life project work under time and deliverable constraints, applying the knowledge acquired through various courses.

2. OBJECTIVES:
 - To provide an opportunity to apply the knowledge gained through various courses in solving a real life problem
 - To provide an opportunity to practice different phases of software/system development life cycle
 - To introduce the student to a professional environment and/or style typical of a global IT industry
 - To provide an opportunity for structured team work and project management
 - To provide an opportunity for effective, real-life, technical documentation
 - To provide an opportunity to practice time, resource and person management.

3. PROJECT GUIDELINES

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

The minimal phases for the project are: Project search, finalization and allocation, Investigation of system requirements, Data and Process Modelling, System Design, Program design, Program coding and unit testing, System integration, System implementation and acceptance testing.

3.1 Planning the Project: The Major Project is an involved Exercise which has to be planned well in advance. The topic should be chosen in Semester 4 itself and the case study of Course CS1302 should as far as possible, be based on the project topic, though on Exceptional cases, for valid reasons, the project guide may waive this condition. Related reading, training and discussions should start from semester 5 itself.

3.2 Selection of project work: Project work could be of 3 types:
a) Developing solution for a real-life problem: In this case, a requirement for developing a computer based solution already Exists and the different stages of system development life cycle is to be implemented successfully. Examples are Accounting Software Package for a particular organization, Computerisation of administrative functions of an organization, Web Based Commerce, etc. The scope for creativity and Exploration in such projects is limited, but if done meticulously, valuable Experience in the industrial context can be gained.
(b) Innovative Product development: These are projects where a clear-cut requirement for developing a computer based solution may not be Existing, but a possible utility for the same is conceived by the proposer. An Example is a Malayalam Language Editor with Spell Checker, Computer Music Software for Indian Music, Heat Engines Simulation Software for eLearning, Digital Water Marking Software,
(c) Research level project: These are projects which involve research and development and may not be as structured and clear cut as in the above case. Examples are Malayalam Character Recognition, Neural Net Based Speech Recogniser, Biometric Systems, Machine Translation System etc. These projects provide more challenging opportunities to students, but at EX level is

a difficult choice. If any student identifies proper support in terms of guidance, technology and references from External organizations and also the supervisors are convinced of the ability of the student(s) to take up the project, it shall be permitted. The methodology and reporting of such projects could be markedly different from type (a) and is left to the proposer/external supervisor of the projects.

3.3 Selection of Team: To meet the stated objectives, it is imperative that Major Project is done through a team effort**.** Though it would be ideal to select the team members at random (drawing lots) and this should be strongly recommended, due to practical considerations, students may also be given the choice of forming themselves into teams with 3 to 5 members (teams less than 3 members may be permitted in Exceptional cases, for valid reasons). A gender mix should also be strongly suggested. A team leader shall be elected through drawing lots. Teams shall maintain team meeting minutes and ensure that every team member has tasks assigned in writing. Team meeting minutes shall form a part of the Project Report. Even if students are doing projects as groups, each one must independently take up different modules of the work and must submit the reports also independently (though, in such cases, some common materials is permissible). Evaluation will also be done independently.

3.4 Selection of Tools: No restrictions shall be placed on the students in the choice of platforms/tools/languages to be utilized for their project work, though open source is strongly recommended, wherever possible. No value shall be placed on the use of tools in the evaluation of the project.

3.5 Selection of Organisation & Guide: No restrictions shall be placed on the students in the choice of organization where project work may be done, in terms of locality, type (public/private) etc. It is the duty of the Head of Institute/Principal of College to ensure that the Aim, Objectives and full project guidelines are communicated to the external organization. The guide should ideally be a post-graduate with minimum 2 years of work experience.

Students may also choose to do project in the college/institute (or partially in the college/institute and partially in an external organization), especially product-based work, but in such cases the supervisors must ensure that (i) industry practices are followed (ii) the students undertake a planned visit to an IT industry with international operations to make up for the loss of experience and (iii) the services of an external guide with industry experience is obtained.

3.6 Project Management: Head of Institute/Principal of College should publish a list of students, projects topics, internal guide and external organization (if any) and teams agreed, before the end of semester 5. Changes in this list may be permitted for valid reasons and shall be considered favourably by Head of Institute/Principal of College any time before commencement of the project. Any request for change after commencement should considered by a committee of 3 teachers and their recommendation shall be accepted by Head of Institute/Principal of College.

Gantt-chart of proposed activities and a draft statement of project deliverables (which may subsequently be altered if justified) should be prepared before the commencement of the project. The actual completion of each phase should be noted on the chart in the course of the project work. Students should submit a fortnightly report of progress which could be indication of percentage of completion marked on the orginal Gantt-chart, with any notes attached. Students should ideally keep a daily activity log sheet. Team meetings should be documented in the format given at the end. Changes in the submitted documents are possible, as project development is essentially an evolutionary process. The project guide must ensure that changes are necessary due to the knowledge gained in succeeding phases of the project. The date of

completion of a phase should be brought forward if the changes made are deemed to be errors and not due to additional knowledge gained from a succeeding phase.

3.7 Documentation:

The following are the major guidelines: The final outer dimensions of the report shall be 21 cm X 30 cm. The colour of the flap cover shall be light green. Only hard binding should be done, with title of the thesis and the words "<BRIEF TITLE> BSc (CS) Project Report 200…" displayed on the spine in 20 point, Bold, Times New Roman, as in example below. In case the title is too long, a shorter version of it may be used (Like "Image Pro" instead of "Image Pro – An Interactive Image Processing package"). It is highly recommended that Latex be used for documentation.

• The text of the report should be set in 12 pt, Times New Roman, Single Spaced.

• Headings should be set as follows: CHAPTER HEADINGS 20 pt, Times New Roman, Bold, All Caps, Centered.

<div align="center">

WEB BASED BILLING SOFTWARE: BSC(CS) PROJECT 2009
</div>

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

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

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

<div align="center">

<PROJECT TITLE>

<STUDENT'S NAME>

<COLLEGE NAME>

PROJECT REPORT
Submitted in partial fulfilment of the
Requirements for the award of
Bsc (computer science) degree of
University of kerala
2014
</div>

Some general guidelines on documentation stylistics are:

• Double quotes and single quotes ("", ') should be used only when essential. In most cases words put in quotes are better highlighted by setting them in italics. Eg: This process is known as "morphing". This process is known as *morphing*.

• Page numbers shall be set at right hand top corner, paragraph indent shall be set as 3.

• Only single space need be left above a section or sub-section heading and no space may be left after them.

• Certificate should be in the format: "Certified that this report titled....................... is a bonafide record of the project work done by Sri/Kum...................... under our supervision and guidance, towards partial fulfillment of the requirements for the award of the Degree of BSC (Computer Science) of the University of Kerala" with dated signatures of Internal; Guide, external guide and also Head of Institute/College.

• If the project is done in an external organization, another certificates on the letterhead of the organization is required: "Certified that his report titled.............................. is a bonafide record of the project work done by Sri/Kum............................ under any supervision and guidance, at the ....Department of.................... (Organization) towards partial fulfilment of the requirements for the award of the Degree of BSC (Computer Science) of the University of Kerala".

• References shall be IEEE format (see any IEEE magazine or transaction). Take care in use of italics and punctuation. While doing the project, keep note of all books you refer, in the correct format, and include them in alphabetical order in your reference list. Eg: A book is cited as: Kartalopoulos, S V Understanding Neural Networks and Fuzzy Logic, BPB Publishers, 1996, pp. 21-27. (pp.21-27 indicates that pages 21-27 have been referred. If the whole book is being referred, this may be omitted. If a single page is referred, say 7, it may be cited as p.7 Report writing is NOT a hasty activity done after finishing the project. Students must try to develop the report along with the work, so as to give it flesh and blood. Drafts should be read, modified, spell checked and grammar checked at least thrice during the course of the project and before a final printout is taken, the same may be got approved from the internal guide. The students should send two interim reports to internal guides. This will also help the students in their report writing.

The Gantt chart, fortnightly progress reports, and team meeting minutes mentioned in section 3.5 should appear as appendix to the project report. Regarding the body of the report, as an indicative Example, the following is given (though students should not attempt to fit every kind of project report into this format):

– Organizational overview (of the client organization, where applicable)
– Description of the present system
– Limitations of the present system
– The Proposed system- Its advantages and features
– Context diagram of the proposed system.
– Top level DFD of the proposed system with at least one additional level of Expansion
– Structure Chart of the System
– System flowchart
– Menu Tree
– Program List
– Files or tables (for DBMS projects) list. Class names to be entered for each file in OO systems.
– List of fields or attributes (for DBMS projects) in each file or table.
– Program – File table that shows the files/tables used by each program and the files are read, written to, updated, queried or reports were produced from them.
– Reports List with column headings and summary information for each report.
– System Coding and variable/file/table naming conventions
– System controls and standards
– Screen layouts for each data entry screen.
– Report formats for each report.

Program documentation is suggested on the following lines:
– Program id
– Program level run chart
– Program function Explanation
– Data entry screen (reproduced from system documentation).
– Report layout (reproduced from system documentations)
– Program level pseudo code or flowchart.
– Decision tables, decision trees, with English Explanation where necessary.
– Program listing
– Test data

– Test results.

3.8 Methodology:
Wherever applicable, object oriented approach should be used for software development. The project report should generally contain details of the following steps *(though students should not attempt to fit every kind of project into this format):*

(a) Analysis
- Study of existing systems and its drawbacks (general)
- Understanding the functionalities of the system (detailed)
- Preparation of requirement
- Conduct of Feasibility study
- Identification of relevant Objects
- Abstraction of each object (attributed and methods)
- Relationship between objects

(b) Design
- Design of each subsystems
- Design of each classes
- Design of communications between objects
- Design of Algorithms for problem solving
- User interface Design
- Any other steps if necessary

(c) Coding and Impletion
(d) Testing
(e) Security, Backup and Recovery Mechanisms
(f) On line help and User Manuals
(g) Upgradability Possibilities

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

4. REFERENCES
4.1 Core
- ❖ S A Kelkar, *Software Project Management*, Prentice Hall of India
- ❖ W Alan Randolph, Barry Z. Posner, *Effective project planning and management*, PHI

4.2 Additional
- ▪ Greg Mandanis, *Software Project Management Kit for Dummies*, IDG Books
- ▪ Joel Henry, *Software Project management*
- ▪ Frederic P B, Mythical Man-month, *Essays on Software Engineering*, Addison Wesley
- ▪ David Lamport, *Latex: A document Preparation System*, 2/e, Pearson Education