Assam Don Bosco University
Programme Project Report (PPR)

Master of Computer Application

i. Programme's Mission and Objectives

Creating an intense teaching and research environment that moulds individuals into competent professionals who are innovative and committed to meet real world challenges.

Mission

1. To produce competent Computer Science professionals by promoting excellence in education and training.
2. To inculcate the spirit of self-sustainability through research, consultancy, development activities and lifelong learning.
3. To extend technical expertise to meet real world challenges and play a leading role in technical innovation, creativity and application development.
4. To infuse a sense of commitment in individuals for the betterment of the society through technology.

Programme Educational Objectives (PEOs)

1. To prepare the students to have strong foundation in computer science engineering with impetus to higher studies, consultancy, research and development.
2. To prepare the students to be self sustainable and proficient to meet the real world challenges ethically and responsibly, in service to socio-economic development of the society.
3.To inculcate the spirit of life-long learning, understanding, and applying new ideas and technologies to provide novel engineering solutions in the rapidly changing environment.

ii. ***Relevance of Program with the Assam Don Bosco University Mission & Goals***

The programmes mission and objectives are in alignment with the University's mission, vision and goals, as detailed below.

Vision

The vision of Don Bosco University is:

'To mould young persons into intellectually competent, morally upright, socially committed and spiritually inspired citizens at the service of India and the world of today and tomorrow, by imparting holistic and personalized education.'

Guided by this vision and leveraging its century-old expertise in education in India and abroad, Don Bosco University is envisaged to be a centre of excellence in study and research focusing upon the following:

- Providing easier access to higher education for the under-privileged.
- Harmonizing technical excellence with human and religious values.
- Employment-oriented courses in emerging areas of contemporary technology and service.
- 'Teacher Education' as a privileged area of interest to accelerate the pace, reach and quality of education.
- Impetus to research initiatives with practical and social relevance.
- Providing a forum for debate and research on key human issues like religion & culture, peace & justice.
- Contributing to the socio-economic development of North-East India.
- Boosting international linkages and collaboration in university education.

The spirit of the university has been encapsulated in its emblem which has five distinct elements:

- The Cross' signifies its underlying Christian inspiration;
- 'The Profile of St. John Bosco' denotes its distinctive educational philosophy;
- 'The book emphasizes its commitment to academic excellence;
- 'The rays of the Rising Sun' stand for its dynamism and commitment to society; and,
- The motto, 'Carpe Diem' which meaning 'Seize the Day' challenging every Bosconian to grasp the opportunities presented by each new day and attain 'life in its fullness.'

Mission

Built on a great legacy inherited from our founding fathers, our mission is to create an environment of stimulating intellectual dialogue across disciplines and harvest ***knowledge with a cutting-edge through high quality teaching, research, and extension activities*** *leading to the generation of students who would provide leadership, vision and direction to society.*

Goal

Our goal is to realize this vision by 2025. ***Our strategy is to develop innovative programmes in basic and emerging disciplines*** *in a phased manner and to update them periodically so as to keep ourselves on track and on time. Our commitment is to involve the faculty and students in* ***interactive learning environment both within and outside the University through contextual and experiential programmes*** *so that they would be builders of a* ***worldwide-network*** *of* ***knowledge-sharin****g and excel in their performance with a winning edge in the wider context of globalization.*

iii. Nature of Prospective Target Group of Learners

This Program is designed to target working individuals who wish to further their

professional and academic qualifications, or wish to acquire domain specific knowledge and skills in their chosen profession or industry, in the field of IT and Software.

iv. Appropriateness of Programme to be conducted in Online Learning mode to acquire specific skills and competence

A IT program of this nature is apt for delivery in the Online mode, given its practical nature. It's students, who would already be working in the Software/IT industry, shall benefit from putting into practice their learnings on an ongoing basis, and shall be able to recognize aspects and applicability of their studies in their everyday work life.

v. Instructional Design

a. Curriculum Design

As per University Curriculum in place for campus programs.(Attached)

b. For Online Mode:

E-Learning Materials Quality Standard:

The online courses should comply with the following Quality standards, namely:-

(i) The courses should follow the following four quadrant approach, as per the SWAYAM Guidelines:-

(a) Quadrant-I is e-Tutorial; which shall contain: Video and Audio Content in an organized form, Animation, Simulations, video demonstrations, Virtual Labs, etc, along with the transcription of the video.

(b) Quadrant-II is e-Content; which shall contain; self-instructional material (digital Self Learning Material), e-Books, illustrations, case studies, presentations etc, and also contain Web Resources such as further references, Related Links, Open source Content on Internet, Video, Case Studies, books including e-books, research papers and journals, Anecdotal information, Historical development of the subject, Articles, etc.

(c) Quadrant-III is the Discussion forum; for raising of doubts and clarifying the same on real time basis by the Course Coordinator or his team.

(d) Quadrant-IV is Assessment; which shall contain; Problems and Solutions, which could be in the form of Multiple Choice Questions, Fill in the blanks, Matching Questions, Short Answer Questions, Long Answer Questions, Quizzes, Assignments and solutions, Discussion forum topics and setting up the FAQs, Clarifications on general misconceptions.

c. Duration of the Program

As per University duration policy for on campus programs, in conjunction with the UGC regulation in place.

d. Faculty and Support Staff Requirement

As per UGC Regulations

e. Instructional Delivery Mechanisms

Courses to be delivered in an 'online' mode with learning material in the form of E-SLM's, and Self Assessments being available for the students. Additionally, virtual live lectures and recorded lecture sessions to be provided as per a fixed schedule towards the end of each term.

f. Student Support Systems

Learner Support Service via Web, Chat, Call Support. Access to counsellors at Department on University campus. Access to E-Learning Library resources in the student portal.

vi. Procedure for admissions, curriculum transaction and evaluation

a. Admission Policy

Admissions to be conducted twice a year (January and July), and as per common regulation for online programs already in place and in effect at the time, in conjunction with UGC Regulations.

b. Minimum Eligibility

As per common regulation for online programs already in place and in effect at the time, in conjunction with UGC Regulations.

c. Fee Structure

As per fee structure in place for online programs, and as decided by fee committee from time to time.

d. Programme Delivery Methodology

Courses to be delivered under the 'online' mode of learning, with students being provided Self learning material in eBook format, along with access to online Self Assessment tools. Contact Programs Sessions to be conducted online

once per semester.

e. Web Based Tools

A Web based Portal which shall allow the student access to the following

- Admission & Enrolment Details
- Fee Details and Online Fee Payment Gateway
- Prospectus, Regulations & Syllabus
- Notifications (Admissions, fees, examinations etc)
- Course List, with completion status and scores / results
- eBook's of SLM's.
- Self Assessment Tests (unscored)
- Internal Assessments - IA1 & IA2
- Online PCP Lectures (Recorded or via Virtual Classroom session) as conducted each semester.
- Online Learners forum, for student to student interaction.
- Online Copy of the Grade sheet.
- Access to online support in the form of web chat, ticketed email support etc.

f. Evaluation Methodology – Tools & Methods

Internal Assessment marks to comprise the results of Internal Assessment Tests (IA1 & IA2) over the duration of the Semester/Term towards 30% of the final marks, whereas the End Term Assessment , conducted at Examination centres (as per UGC Regulations) to comprise of the remaining 70%.

End Term Exam Regulations to be as per regulations governing on campus programs.

Being an online program, there shall be no provision or requirement to repeat or drop a year within the program.

vii. Requirement of the laboratory support and Library Resources

Practical Subjects in this subject can mostly be covered with access to a computer, internet and software tools (available under FOSS/Open Source). Students may be asked to make arrangements for the same at their own cost.

As per the syllabus/curriculum, should there be need for special equipment/laboratory access (other than the above mentioned regular computer lab or internet connectivity), the same shall be arranged by student on their own which shall be notified by prior intimation.

Provisions also shall be made for access to virtual lab's and simulators via the students online study portal.

Library resources can be accessed online.

viii. Cost Estimate of the Programme and the provisions

Upon receipt of approval from the UGC, the Finance Committee may set aside budgetary provisions towards Programme Development, Programme Delivery, Programme Maintenance. Once the programs are operational, fee receipts from the program may be used to cover the same, as per the guidance of the Syndicate and Finance Committee.

ix. Quality Assurance mechanism and expected programme outcomes

The expected outcome from these programs is a measurable increase in the skills and knowledge of the student in his/her area of study, and that increase should be reflected in a proportional increase in available job opportunities / role or profile changes in his or her current job or industry, and a change / increment in earning capability .

A Centre for Internal Quality Assurance shall be setup within 1 year of the launch of these programs (as per UGC requirements). The CIQA, working in collaboration with the Department , shall put in place a continuous quality measurement and improvement framework, using both student – teacher feedback, and outcome measurement on data gathered from students during and after completion of their programs via surveys. The findings, action taken report and results shall be published and made available via the CIQA each year in the form of a report.

Master of Computer Application

Term	Course	Credits
1	Mathematical Foundation for Computer Science	4
1	Theory of Computation	4
1	Operating Systems	4
1	Data Structures and Algorithms	4
1	Programming Through Java	4
1	Programming & Problem Solving through C	4
2	Software Engineering	4
2	Advanced Database Management Systems	4
2	Data Communication and Computer Networks	4
2	Internet Technology and Applications	4
2	Sensor Networks and Internet of Things	3
2	Computer Graphics	4
3	Machine Learning	4
3	Enterprise Resource Planning	4
3	Cyber Law and IT Security	4
3	Research Methodology and IPR	2
3	Elective I- Artificial Intelligence	4
3	Elective II- Data Science	4
3	Web Programming	4
4	Project Work	18

Master of Computer Application
Syllabus – Semester I

Mathematical Foundation for Computer Science

Objective

- To introduce the concepts of mathematical logic.
- To introduce the concepts of sets, relations, and functions and relate practical examples to the appropriate set, function, or relation model, and interpret the associated operations and terminology in context.
- To perform the operations associated with sets, functions, and relations.
- To understand combinatorics and apply in solving problems.
- To use Graph Theory for solving problems

Module I

Mathematical Logic: Statements and notations, Connectives, Well-formed formulas, Truth Tables, tautology, Logical equivalence: The Laws of logic, Logical Implication, Normal forms, Quantifiers, universal quantifiers. Predicates: Predicative logic, Free & Bound variables, Rules of inference, Consistency, proof of contradiction.

Module II

Set theory: Introduction, Basic Concepts of Set Theory, Representation of Discrete Structures, Relations and Ordering, Matrix representation of relations and partial ordered sets, representation of relations by Graphs; Lattices as Partially Ordered Sets, Boolean algebra; Functions. Algebraic Structures: Introduction, Algebraic Systems, Semi groups and Monoids; Groups, Congruence Relation and Quotient Structures, permutation groups, Lagrange's Theorem; Normal subgroups. Algebraic Structures with two Binary Operation, Rings, Integral Domain and Fields. (Definition, basic properties and examples)

Module III

Elementary Combinatorics: Basics of counting, Combinations & Permutations with repetitions, Constrained repetitions, Binomial Coefficients, Binomial and Multinomial theorem, Pigeonhole Principle, principles of Inclusion – Exclusion.

Module IV

Graph Theory: Basic Concepts, Sub graphs, Multi graphs Representation of Graphs, Isomorphism, Paths and Circuits, Traversing a Graph, DFS, BFS, Eulerian and Hamiltonian graphs, Shortest path algorithms, Planar Graphs, Chromatic Numbers. Tree and Spanning Trees. Applications of Graph Theory.

Course/Learning Outcomes

CO1: Ability to apply mathematical logic to solve problems (Remembering, Understand)
CO2: Recall some basic concept of set theory and understand the concept of graph theory and Group theory. (Remembering)
CO3: Interpret logic sentence in terms of predicates, quantifiers, and logical Connectives (Understanding)
CO4: For a given a discrete problem, classify its algebraic structure (Analyzing)
CO5: Derive the solution of a problem using deductive logic and prove the solution based on logical inference (Applying)
CO6: Evaluate Boolean functions and simplify expressions using the properties of Boolean algebra (Evaluating)
CO7: Develop the given problem as graph networks and solve with techniques of graph theory. (Creating)

Suggested Readings

1. Discrete Mathematical Structures with Applications to Computer Science by J. P. Tremblay and R Manohar, Tata McGraw-Hill Publications, 1997.
2. Graph Theory by Narsingh Deo, Prentice-Hall of India publications, 2004.
3. Discrete Mathematical Structures, Theory and Applications. D.S. Malik, Thomson Learning, I Edn
4. Discrete Mathematics for Computer Science, Haggard, Thomson Learning, I Edn
5. Discrete Mathematics and Its Applications by Kenneth H Rosen. Tata McGraw-Hill Publications
6. Mathematical foundation of Computer Science by Y. N Sings. New Age international Publishers
7. Bernard Kolman, Robert. C.Busby & Sharon Ross, "Discrete Mathematical structures" Prentice Hall of India,2001.

Theory of Computation

Objective: The objective of the Theory of Computation is to introduce and study abstract, mathematical models of computation (such as finite state, pushdown and Turing machines), and to use the abstract machine models to study the ability to solve computational problems. At the complete course students will be able to use regular expressions effectively and appropriately, construct derivations and parse trees, write simple programs for a Turing machine, understand the equivalence of grammars, languages and automata and translate between grammars, languages and automata.

Module I - Theory of Automata

Definition of an Automaton, Description of a Finite Automaton, Transition Systems, Properties of Transition Functions, Acceptability of a String by a Finite Automaton, Nondeterministic Finite State Machines, The Equivalence of DFA and NDFA, Mealy and Moore Models, Minimization of Finite Automata.

Module II Formal Languages, Regular Sets and Regular Grammars

Definition of formal languages, Chomsky Classification of Languages, Languages and Their Relation, Recursive and Recursively Enumerable Sets, Operations on Languages, Languages and Automata; Regular Expressions, Finite Automata and Regular Expressions, Pumping Lemma for Regular Sets, Application of Pumping Lemma, Regular Sets and Regular Grammars.

Module III Context-free Languages

Context-free Languages and Derivation tree, Ambiguity in Context-free Grammars, Simplification of Context-free Grammars, Normal Forms for Context-free Grammars, Pumping Lemma for Context-free Languages, Decision Algorithms for Context-free Languages.

Module IV Pushdown Automata Turing Machines and Linear Bounded Automata

Basic Definitions, Acceptance by PDA, Pushdown Automata and Context-free Languages, Parsing and Pushdown Automata; Turing machine Model, Representation of Turing Machine, Language Acceptability by Turing Machines, Design of Turing Machines, Universal Turing Machine and Other Modification, The Model of Linear Bounded Automaton, Turing Machines and Type 0 Grammars, Linear Bounded Automata and Languages, Halting Problem of Turing Machines, NP-Completeness.

Course/Learning Outcomes

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

CO1: Define basic terminology like Deterministic and Non deterministic automata, Pushdown Automata, Parse Tree, Regular Languages, Turing Machines etc. (Remembering)

CO2: Explain the concepts, core terms and tools used in automata theory (Understanding)
CO3: Make use of techniques, components and tools of a typical automated machine and apply it in designing new machines (Applying)
CO4: Choose which input pattern would be accepted by a Turing Machine, Pushdown Automata, Finite Automata etc. (Applying)
CO5: compare and contrast various types of machines in Automata theory and relate it to everyday appliances like washing machines, fans, etc (Analysing)
CO6: design an automata and evaluate it in terms of correctness, computation cost and complexity. (Evaluating)
CO7: design new automata for given problems by using most appropriate algorithmic strategy considering the problem domain. (Creating)

Suggested Readings

1. K.L.P. Mishra, N. Chandrasekaran, Theory of Computer Science, BPB Publication, Prentice-Hall of India, Second Edition.
2. H.R. Lewis and C.H.Papadimitriou, Elements of the Theory of Computation, Second Edition, Prentice Hall of India.
3. H.E. Hopcraft and J.D. Ullamn, Introduction to Automata Theory, Languages and Computation, Narosa Publications.
4. J.C. Martin, Introduction to Languages and the Theory of Automata, Tata McGraw-Hill.
5. C.H. Papadimitriou, Computation Complexity, Addison-Wesley.

Operating Systems

Objective: The main objective of this course is to introduce the students to a layer of software called Operating Systems, whose job is to manage all the devices of a computer system and provide user programs with a simple interface to the hardware. This course will familiarize the students with the concepts of processes, memory management, file management, Input/Output management and the potential problem of deadlocks. The students will also learn about the Linux operating system, which is a full-blown Unix clone and is fast gaining popularity worldwide.

Module I: Concepts, Processes and Threads

Operating system as an Extended Machine and as a Resource Manager, Operating system concepts (Files, Deadlocks, Memory Management, Input/Output, Processes, The Shell, Security), The evolution of Operating Systems (Serial Processing, Simple Batch Systems, Multiprogrammed Batch Systems, Mainframe Operating Systems, Server Operating Systems, Time Sharing Systems, Multiprocessor Operating Systems, Real-Time Systems, Embedded Operating Systems, Smart Card Operating), System Calls (Process Management, File Management, Directory management), Introduction to Processes (The Process Model, Process Creation, Process Termination, Process Hierarchies, Process States, Implementation of Processes, Process Control Block), Threads (The Thread Model, Thread Usage, Implementing Threads(In User Space and Kernel), Scheduler Activation, Pop Up Threads, Interprocess Communication (Race conditions, Critical Sections, Mutual Exclusion with Busy Waiting, Sleep and wakeup, Semaphores, Mutexes, Monitors, Message Passing), Classical IPC problems (The Dining Philosophers Problem, The Sleeping Barber Problem), Process Scheduling (Scheduling in Batch Systems, Scheduling in Batch Systems, Scheduling in Interactive Systems, Scheduling in Real-Time Systems, Thread Scheduling)

Module II: Deadlocks and Memory Management

a) Resources, Deadlock (Conditions for Deadlock, Deadlock modeling), Deadlock detection and recovery, Deadlock avoidance, Deadlock prevention
b) Memory management without swapping or paging (Monoprogramming without swapping or paging,

Multiprogramming with fixed partitions, Relocation and Protection), Swapping, Virtual Memory (Paging, Page Tables), Page Replacement Algorithms (Not-recently-used, First in first out, Second Chance page replacement algorithm, The Clock Page Replacement Algorithm, Least Recently used page replacement algorithm, The Working Set Page Replacement Algorithm, Modeling Paging Algorithms (Belady's Anomaly, Stack Algorithms, Predicting page fault rates), Design issues for Paging Systems, Implementation issues, Segmentation (Implementation of pure segmentation, Segmentation with Paging: MULTICS)

Module III - Input/output and File Systems

a) Principles of I/O hardware (I/O devices, Device Controllers, Direct memory access), Principles of I/O software, I/O Software Layers, Disks (Disk hardware, disk formatting, disk arm scheduling algorithms, Error handling, Track-at-a- time caching, RAM disks) Clocks (Clock hardware, Clock software), Terminals (Terminal hardware, Input software, Output software)

b) Files (File Naming, File structure, File types, File access, File attributes, File operations, Memory mapped files), Directories, File System layout (Implementing files, Implementing directories, Shared files), Security (The security environment, Generic Security Attacks, Design Principles For Security, User Authentication), Protection mechanisms (Protection Domains, Access Control Lists, Capabilities, Multilevel Security, Covert Channels), Type of File Systems (FAT, VFAT, FAT32, NTFS)

Module IV - Introduction to Linux OS design – Case study

Overview of Unix, Processes in Unix (Fundamental Concepts, Process Management System Calls in Unix, Implementation of Processes in Unix), Memory Management in Unix, Input/Output in Unix, The Unix File System, Security in Unix

Course/Learning Outcomes

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

CO1: Elaborate what operating systems are, what they do and how they are designed and constructed. (Creating)

CO2: Define process concepts like process scheduling, inter-process communication, process synchronization and concurrency. (Remembering)

CO3: Explain different memory management schemes, relate various approaches to memory management and effectiveness of a particular algorithm. (Understanding)

CO4: Identify different page replacement algorithms to solve problems. (Applying)

CO5: Explain how the file system, mass storage and I/O are handled in a modern computer system. (Remembering, Understanding)

CO6: Analyse the mechanisms necessary for the protection and security of computer systems. (Analysing)

CO7: Determine the concepts learned with case studies of Linux and Windows. (Evaluating)

Suggested Readings

1. Andrew S Tanenbaum, Modern Operating Systems , (Second Ed.), Prentice Hall of India, New Delhi,
2. William Stallings, Operating Systems, Fourth Edition, Prentice Hall of India, New Delhi.
3. Silberschatz, Galvin, Operating System Concepts, Fifth Edition, John Wiley and Sons (Asia) Pte.
4. HM Deitel, Operating Systems, Second Edition, Pearson Education.
5. Pramod Chandra P. Bhatt, An Introduction to Operating Systems Concept, Prentice Hall of India.
6. Maurice J. Bach, The Design of the Unix Operating System, Prentice Hall of India, New Delhi.
7. Kernighan and Pike, The Unix Programming Environment, Prentice Hall of India, New Delhi.

Data Structures and Algorithms

Objectives

- To introduce first level topics covering basics in algorithms and data structures.

- To enable students to choose appropriate data structures, understand the ADT/libraries, and use of it to design algorithms for a specific problem.
- To understand the necessary mathematical abstraction to solve problems.
- To apply important algorithmic design paradigms and methods of analysis.

Module I

Introduction to Algorithms, Fundamentals Stages of Problem Solving, and Classification of Algorithms - Based on Implementation, Based on Design, Based on Area of Specialization, Based on Tractability, Basics of Algorithms Analysis, Asymptotic Analysis, Mathematical Analysis of Iterative and Recursive Algorithms, Empirical Analysis of Algorithms, Models of Computations - RAM model, Turing Machine.

Module II

Data Structures: Abstract Data Types (ADTs), Stacks, Queues, Circular Queues, Implementation of Stacks using Queues, Implementation of Queues using Stacks, Priority Queues, Heaps, Linked Lists, Search and Update Operations on Varieties of Linked Lists, Graphs, Binary Trees, Tree Traversals, Binary Search Trees (BSTs), AVL Trees, Red Black Trees, Splay Trees, B-Trees, Disjoint Sets.

Module III

Analysis of Sorting and Searching Algorithms: The sorting problem, Brute Force Approach - Sequential Search, Bubble Sort, Selection Sort, Decrease-and-Conquer Approach - Insertion Sort, Binary Search, Divide-and-Conquer Approach - Quick Sort, Merge Sort, Transform-and-Conquer Approach - Heap Sort, Linear Sorting Algorithms - Counting Sort, Radix Sort, Bucket Sort, Hashing - Hash Function, Collisions in Hashing, Separate Chaining, Open Addressing, Analysis of Search Operations.

Module IV

Graph Algorithms: Graphs and their Representations, Graph Traversal Techniques - Breadth First Search (BFS) and Depth First Search (DFS), Minimum Spanning Trees (MST), Greedy Techniques - Prim's and Kruskal's algorithms for MST, Dijkstra's Algorithm for Single Source Shortest Paths, Dynamic Programming - Warshall's Algorithm for finding Transitive Closure of a Graph, Floydd's Algorithm for All-Pairs Shortest Paths Problem. Algorithmic Design Techniques: Greedy Algorithms - Coin Change Problem, Scheduling Problem, Knapsack Problem, Huffman Trees, Divide-and-Conquer Approach - Strassen Matrix Multiplication, Closest-pair Problem, Tiling Problem, Dynamic Programming – Longest Common Subsequence (LCS) problem, Optimal Binary Search Trees, Travelling Salesperson Problem, Chain matrix multiplication.

Module V

Tractable and Intractable Problems: Computability of Algorithms, Computability Classes – P, NP, NP-Complete, NP-Hard. Basics of Backtracking, Branch-and-bound methodologies for Algorithm design, Approximation algorithms, Randomized algorithms.

Course/Learning Outcomes

At the end of the course, students would be able to:

CO1: Know the formal definition of algorithms, importance of analysis of an algorithm and get familiar with different types of problem and their solutions. (Remembering and Understanding)

CO2: Choose appropriate data structure as applied to specified problem definition. (Applying)

CO3: Understand different design strategies such as brute force, divide-and-conquer, dynamic programming, greedy technique and backtracking used for the design of algorithms. (Understanding)

CO4: To design and analyse algorithms for given problems. (Applying)

CO5: Compare and analyse different design strategies and assess an algorithm in terms of correctness, computation cost and memory space used. (Analysing and Evaluating) .

CO5: Design new algorithms for given problems by using most appropriate algorithmic strategy considering the problem domain. (Creating) .

Suggested Readings

1. Mark Allen Weiss, Data Structure and Algorithm Analysis in C++, Fourth Edition, Pearson, 2014.
2. S. Sridhar, Design and Analysis of Algorithms, Oxford University Press, 2014.
3. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, Introduction to Algorithms, Third Edition, The MIT
4. Press, 2009.
5. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, Fundamentals of Computer Algorithms, Universities Press, 2008.

Programming & Problem Solving through C

Unit- 1

Overview of C- Importance of C, Basic structure of C Programs, Basic programing constructs- character set, tokens, Constants, Variables, and Data Types, Keywords and identifiers, symbolic constants.

Operators and Expression – Arithmetic, relational, logical, increment and decrement, conditional, bitwise, Expression, precedence of operators, type conversion and casting, mathematical functions, Managing Input and Output Operations - Reading a character, writing a character, formatted input, formatted output.

Unit- 2

Decision Making and Branching – If statement – Different forms of if statement, switch, break and continue, Looping statements in C – For, while and do while, nested loops, Structured data types in C – Array – One dimensional and two dimensional array, String, Structures and union.

Unit- 3

User-Defined Functions – Need for user-defined functions, multi-function program, general form of C function, Category of functions, Handling of non-integer functions, Nesting of functions, Recursion, Functions with arrays and Structures, The scope and lifetime of variables in functions.

Unit- 4

Pointers and File Handling - Understanding pointers. Accessing the address of a variable, Declaring and initializing pointers, Accessing a variable through its pointer. File Handling– Definition and need of file. Defining, Opening, and Closing a file. Input and output operations on files. Random access to files with example programs.

Suggested Readings:

1. Problem Solving with C, Somashekara M. T, PHI.
2. Programming in ANSI by E. Balagurusamy, TMH.

Programming Through Java

Objective: The course is designed to impart the knowledge and skill required to solve real world problems using an object- oriented approach utilizing Java language constructs. This course covers the two main parts of Java i.e. Java Language and Java Library (JDK 5). After completion of the course, a student is expected to be able to

- Do Object Oriented Programming using Java
- Implement Exception handling and Multithreading in Java.
- Create Java I/O Applications and Applets.

- Set up a GUI using Swing components
- Do Network Programming in Java.
- Access relational databases from the Java program and use Java Beans and Servlets.

Module I: Core Java Programming

a) Java Overview: Genesis, Java Philosophy, Java and Internet, Object-Oriented Programming features, Java Applet and Application, Java Environment and Java Development Kit (JDK) and Java Standard Library (JSL),

b) Java language fundamentals: The scope and lifetime of variable, Type conversion and casting, Control statements, Arrays

c) Classes and objects: The this keyword, Garbage collection, Overloading constructor, Using object as parameters, Argument passing, Returning objects, Recursion, Introducing Access control (public, private and protected), static, final, nested classes, String class, Command- line argument

Module II: Inheritance, Exception handling, Multithread and Applets

a) Inheritance: Member access and inheritance, method overriding, dynamic method dispatch, using abstract classes, using final with inheritance, the Object class; Packages, Interface, classpath

b) Exception handling: Fundamentals, Exception types, Java's built-in exceptions, user defined exceptions.

c) Multithreaded Programming: The Java thread model (thread priorities, synchronization and inter-thread communication); Deadlock, Thread Group

d) I/O Basics: Streams, the stream classes, the predefined streams, reading console input, writing console output, the transient and volatile modifiers, using instance of native methods

Module III String handling, Utility classes, java.lang and java.io

a) String handling: String constructors, methods for character extraction, string searching and comparison, data conversion using valueof (), StringBuffer.

b) Exploring java.lang: Simple type wrappers, System class, class Class, Math functions

c) The utility classes: Vector, Stack, HashTable, StringTokenizer, Bitset, Date, Calendar, Gregorian Calendar, Random, Observable

d) Input/Output - Exploring java.io: The java.io classes and interface, File class and methods for creating, renaming, listing and deleting files and directories, I/O stream classes (FileInputSream, FileOutputStream, BufferedInputStream, BufferedOutputStream, PushBackInputStream, InputStreamReader, BufferedReader, BufferedWriter, PrintStream, RandomAccessFile)

Module IV: Networking, Images, Applet class and Swing

a) Networking: Socket overview, Stream Sockets, Datagram sockets, Manipulating URLs, establishing a simple Server/Client using Stream Sockets, Connectionless Client/Server Interaction with Datagrams

b) Images: File formats, image fundamentals, creating, loading and displaying images, ImageObserver, MediaTracker

c) The Applet class: applet architecture, passing parameters to applets, getDocumentBase, getCodeBase, and showDocument, AppletContext and AudioClip interfaces, Graphics class and methods for drawing lines, rectangles, polygons and ovals

i) Swing: Component and Container classes, Layout managers (FlowLayout, GridLayout, BorderLayout), Handling events, Adapter classes, Anonymous inner classes

ii) Swing GUI components: JLabel, JTextField, JTextArea, JButton, JCheckBox, JRadioButton, JList, JComboBox, JscrollBar, JScrollPane, JToolTip, JPanel, JFrame iii) Menus: JmenuBar, JMenu, JMenuItem, JSeparator

Module V: Java Beans, JDBC, Java Servlets

a) Java Beans: Introducing JavaBeans Concepts and Bean Development Kit (BDK), Using the Bean Box,

Writing a simple Bean, Bean Properties (simple properties), Manipulating events in the Bean Box
b) Java database connectivity (JDBC): Introduction to JDBC, type of JDBC connectivity, Establishing database connections, Accessing relational database from Java programs

Course/Learning Outcomes

At the end of this course students will be able to:
CO1: Recall the various features of Object Oriented programming by utilizing the JAVA language construct. (Remembering)
CO2: Explain the standard library, scope and lifetime of a variable and various control statements used in JAVA programs. (Understanding)
CO3: Interpret the concept of classes and object in JAVA and apply exception handling to solve various exceptions (Applying)
CO4: Contrast the different type of inheritance and polymorphism and Analyse it in resolving various problems (Analysing)
CO6: Select the appropriate GUI and will be able to justify their decision to use a particular GUI by evaluating the required parameters depending on the domain and requirement. (Evaluating)
CO5: Develop algorithms based on the knowledge they have gained to design cost effective and user friendly applications. (Creating)

Suggested Readings

1. Deitel, H. M.; P. J. Deitel, Java: How to Program (Sixth Edition), New Delhi: Prentice-Hall India, 2005
2. Schildt, H., The Complete Reference Java 2 (Fifth Edition), New Delhi: Tata McGraw-Hill, 2005
3. Moss, K., Java Servlets (Second Edition), New Delhi: Tata McGraw-Hill
4. Russel, Java Programming for the absolute Beginner, New Delhi: Prentice-Hall India
5. Hanagan D., Java Examples in a Nutshell (Third Edition), New Delhi: O' Reilly, 2001

Master of Computer Application
Syllabus – Semester II

Software Engineering

Objective: The field of software engineering aims to find answers to the many problems that software development projects are likely to meet when constructing large software systems. The objective of this paper is to make students aware of the problems incurred by large-scale software development and the solutions proposed. It covers a framework for studying and evaluating software tools, and stresses the importance of theory in the development of software.

Module I

a) The Product and The Process: The Product - Evolving Role of Software, Software (Characteristics, Components and Applications;

b) The Process – Software Engineering A Layered Technology, The Software Process, Software Process Models, The Linear Sequential Model, The Prototyping Model, The RAD Model, Evolutionary Process Models (The Incremental Model, The Spiral Model, The Component Assembly Model, The Concurrent Development Model), The Formal Methods Model, Fourth Generation Techniques;

c) Project Management Concepts – The Management Spectrum (People, The Problem, The Process and The Project);

d) Software Process and Project Metrics – Measures, Metrics and Indicators, Metrics in the Process and Project Domains, Software Measurement, Reconciling Different Metrics Approaches, Metrics for Software Quality;

e) Software Project Planning – Observation on Estimating, Project Planning Objectives, Software Scope, Resources, Project Estimation Technique – Empirical estimation techniques (Expert Judgement Technique, Delphi Cost Estimation), Heuristic estimation techniques (COCOMO Model), Halstead Software Science (An Analytical Technique), The Make-Buy Decision;

Module II

a) Project Scheduling and Tracking - Basic Concepts, The Relationship between People and Effort, Defining a Task set for the Software Project, Selecting Software Engineering Tasks, Defining a Task Network, Scheduling, The Project Plan;

b) Software Projects Risks, Quality Assurance and Configuration Management: RiskManagement- Reactive Vs. Proactive Risk Strategies, Software Risk, Risk Identification, Risk Projection, Risk (Mitigation, Monitoring and Management), Safety Risks and Hazards, The RMMM Plan;

c) Software Quality Assurance - Quality Concepts, The Quality Movement, Software Quality Assurance, Software Reviews, Formal Technical Reviews, Statistical Quality Assurance, Software Reliability, The SQA Plan, The ISO 9000 Quality Standards;

d) Software Configuration Management - Software Configuration Management, The SCM Process, Identification of Objects in the Software Configuration, Version Control, Change Control, Configuration Audit, Status Reporting;

e) System Engineering - Computer Based Systems, Product Engineering

Module III

a) Analysis and Design: Analysis Concepts and Principles - Requirements Analysis, Communication Techniques, Analysis Principles, Software Prototyping, Specification, Specification Review;

b) Analysis Modeling- The Elements of the Analysis Model, Data Modeling, Functional Modeling and Information Flow, Behavioral Modeling, The Mechanics of Structured Analysis, The Data Dictionary;

c) Design Concepts and Principles - Software Design and Software Engineering, The Design Process, Design Principles, Design Concepts, Effective Modular Design, Design Heuristic for Effective Modularity, The Design Model, Design Documentation;

d) Design Methods - Data Design, Architectural Design, The Architectural Design Process, Architectural Design Optimization, Interface Design, Human-Computer Interface Design, Interface Design Guidelines,

Procedural Design;
e) Design For Real Time systems - Real Time Systems;
f) Case studies on diagram - Use case, Class, Activity, Sequence

Module IV

a) Software Testing: Software Testing Methods - Software Testing Fundamentals, Test Case Design, White Box Testing, Basis Path Testing, Control Structure Testing, Black Box Testing, Testing for Specialized Environments;
b) Software Testing Strategies - A Strategic Approach to Software Testing, Strategic Issues, Unit Testing, Integration Testing, Validation Testing, System Testing, The Art of Debugging;
c) Technical Metrics For Software - Software Quality, A Framework For Technical Software Metrics, Metrics for the Analysis Model, Metrics for the Design Model, Metrics for Source Code, Metrics for Testing, Metrics for Maintenance

Module V

a) Object Oriented Software Engineering: Object Oriented Concepts and Principles - The Object Oriented Paradigm, Object Oriented Concepts, Identifying the Elements of an Object Model, Management of Object Oriented Software Projects
b) Object Oriented Analysis - Object Oriented Analysis, Domain Analysis, Generic Components of the Object Oriented Analysis Model, The OOA Process, The Object Relationship Model, The Object Behavior Model
c) Object Oriented Design - Design for Object Oriented Systems, The Generic Components of the OO Design Model, The Systems Design Process, The Object Design Process, Design Patterns, Object Oriented Programming
d) Advanced Topics in Software Engineering: Cleanroom Software Engineering- The Cleanroom Approach, Functional Specification, Design Refinement and Verification, Cleanroom Testing
e) Software Reuse - Management Issues, The Reuse Process, Domain Engineering, Building Reusable Components, Classifying and Retrieving Components, Economics of Software Reuse
f) Reengineering - Software Reengineering, Reverse Engineering, Restructuring, Forward Engineering, Economics of Reengineering
g) Computer Aided Software Engineering - Case Definition, Building Blocks of Case, Taxonomy of Case Tools, Integrated Case Environments, The Integration Architecture, The Case Repository

Course/Learning Outcomes

At the end of this course students will be able to:
CO1: define the life cycle models of software. (Remembering)
CO2: explain, identify and differentiate various software life cycle models (Understanding)
CO3: experiment with different software architectures and identify the best feasible one (Applying)
CO4: maintain the software project by using a maintenance plan. (Applying)
CO5: analyse and design the software requirement specification (Analysing)
CO6: summarize, Evaluate and validate a practical solution towards a software application development and also deploy a product of their own. (Evaluating, Creating)
CO7: develop and create various design diagrams and find solutions to problems. (Creating)

Suggested Readings

1. Roger S. Pressman, Software Engineering A Practitioner’s Approach, Fourth Edition, Tata McGraw Hill.
2. Rajib Mall, Fundamentals of Software Engineering, Second Edition, Prentice Hall of India Private Limited.
3. Ian Sommerville, Software Engineering, Sixth Edition, Addison Wesley, Pearson Education.
4. Carlo Ghezzi, Mehdi Jazayeri, Dino Mandrioli, Fundamentals of Software Engineering, Second Edition,

Prentice Hall of
India Private Limited, New Delhi, 2002.
5. Jeffrey A. Hoffer, Joey F. George, Joseph S. Valacich, Modern Systems Analysis and Design, Second Edition, Pearson Education.
6. Richard E Fairley, Software Engineering Concepts, Tata McGraw Hill Publishing Company Limited, New Delhi, 1997.
7. Hans Van Vilet, Software Engineering Principles and Practice, Second Edition, John Wiley and Sons, Ltd.

Computer Graphics

Unit- 1

Overview of Graphics Systems- Video Display Devices, Refresh cathode-ray Tubes, Raster Scan Display, Random Scan Display Color CRT Monitor, Direct View Storage Tubes, Flat panel Display, Three Dimensional Viewing Devices, Stereoscopic and Virtual-Reality Systems, Raster Scan Systems Video Controller, Raster Scan Display Processor, Random-Scan Systems, Graphics Monitors and Workstations, Input Devices, Hard Copy Devices, Graphics Software, Coordinate Representations,Graphics Functions, Software Standards, PHIGS Workstations.

Unit- 2

Output Primitives- Points and Lines, Line Drawing Algorithms, Loading the Frame Buffer, Line Functions, Circle –generating Algorithms, Ellipse-generating Algorithms, Other Curves, Conic Sections, Polynomial and Spline Curves, Parallel Curve Algorithms, Curve Functions, Pixel Address and Object Geometry Screen Grid Coordinate, Maintaining Geometric Properties of Displayed Objects, Filled-Area Primitives, Scan-line polygon Fill Algorithm, Inside Outside Test, Scan –Line Fill of Curved Boundary Areas, Boundary Fill Algorithm, Flood Fill Algorithm, Fill-Area Functions, Cell Array, Character Generations.

Unit- 3

Two- Dimensional Geometric Transformations- Basic Transformations: Translations, Rotations, Scaling; Matrix Representations and Homogeneous Coordinates, Composite Transformations: Translations, Rotations, Scaling, General Pivot Point Rotations, General Fixed Point Scaling, General Scaling Directions, Concatenation Properties, General Composite Transformations and Computational Efficiency, Other Transformations: Reflections, Shear; Transformations Between Coordinate Systems, Affine Transformations, Transformation Functions, Raster Method for Transformations.

Unit- 4

Two- Dimensional Viewing- The Viewing Pipeline, Viewing Coordinate Reference Frame, Window-to-Viewport Coordinate Transformations. Two -Dimensional Viewing Functions, Clipping Operations, Point Clipping, Line Clipping: Cohen-Sutherland Line Clipping, Liang-Barsky Line Clipping, Nicholl-Lee-Nicholl Line Clipping; Polygon Clipping: Sutherland-Hodgeman Polygon Clipping, Weiler-Atherton Polygon Clipping; Curve Clipping, Text Clipping, Exterior Clipping.

Unit- 5

Three Dimensional Concept and Some Object Representation- Three-Dimensional Display Methods, Parallel Projections, Perspective Projections, Depth Cueing, Visible Line and Surface, Identification, Surface Rendering, Exploded and Cutway Views, Three-dimensional and Stereoscopic Views, Three Dimensional Graphic Packages, Polygon Surfaces, Polygon Tables, Place Equations, Polygon Meshes, Curved Line and Surfaces, Quadric Surfaces: Sphere, Ellipsoid, Torus, Superquadrics, Superellipse, Superellipsoid, Blobby Objects, Spline Representations, Interpolations and Approximations Splines, Parametric Continuity Conditions, Geometric Continuty Conditions, Spline Specifications, Cubic Spline Intepolation Methods, Natural Cubic Splines, Hermite Interpolations, Cardinal Splines, Kochanek-Bartels Splines, Bezier Curves, Properties of Bezier Curves, Design Technique using Bezier Curves, CubicBezier Curves, Bezier Surfaces.

Suggested Readings:

1. Computer Graphics, Donald Hearn and M Pauline Baker, PHI, India.

Data Communication and Computer Networks

Objective: This course introduces students to computer networks and concentrates on building a firm foundation for understanding Data Communications and Computer Networks. It deals with the Data link layer, the Network layer, the Transport layer and the Application Layer. This course also introduces the concepts of network security and cryptography.

Module I: Digital Communications
Signals, noise, Nyquist rate, Shannon capacity; Analog transmission: modulation techniques, FDM; Digital transmission: PCM, TDM, line coding, xDSL,; Transmission media: Guided (twisted pair, coaxial, fiber optic) and unguided media; Local area networks: Ethernet, Fast Ethernet, introduction to Gigabit Ethernet and WLANs; Repeater, Hubs, Bridges, Switches, Router and Gateway.

Module II: Media Access Control and Data Link Layer
Data Link Layer Fundamentals: Framing, Error Control, Flow Control, Error Detection and Correction; Data link protocols: Stop- &-Wait ARQ, Go-Back-NARQ, Selective Repeat ARQ, Piggybacking Multiple Access Protocols: Advantages of Multiple-Access Sharing of Channel Resource, Pure ALOHA, Slotted ALOHA, Carrier Sense Multiple Access (CSMA), CSMA with Collision Detection (CSMA/CD), Asynchronous Transfer Mode (ATM)

Module III: Network Layer
IPv4 Addresses: Address space, Notations, Classful addressing, classless addressing, NAT; IPv6 Addresses: advantages, structure, address space, packet format, extension header; Transition from IPv4 to IPv6; Address Mapping, Delivery, Forwarding, Unicast Routing Protocols, Multicast Routing Protocols.

Module IV: Transport Layer and Application Layer
Process to Process Delivery: Client Server paradigm, Connectionless vs Connection Oriented Service, Services provided to upper layers, Transport Service primitives. UDP: Introduction, User Datagram, Checksum, UDP operations, use of UDP, Remote Procedure call TCP: Introduction, TCP Service Model, TCP Protocol, segment header, Connection Establishment and release, Transmission Policy, Congestion Control, Timer Management, Wireless TCP and UDP . Application Layer: Domain Name System, Simple Mail Transfer Protocol (SMTP), POP3, IMAP, File Transfer Protocol (FTP) Network Security: Cryptography, Symmetric Key Algorithms, Public Key Algorithms, Digital Signatures, Communication Security, Web Security

Course/Learning Outcomes
At the end of the course the students will be able to:
CO1: Understand and explain Data Communications System and its components (Understand, Explain)
CO2: Understand and identify different networking terminologies and network architecture. Design issues in network and network transition. (understand)
CO3: Students would be able to distinguish between IPV4 and IPV6 network together with MAC layer transmission and modulation schemes. (Analyze)
CO4: Students would be able to understand and analyze what type of network to implement and decide what protocols to configure (Analyze)
CO5: Students would be able to know why different layers are embodied with different protocols and different network architecture for different network needs. (Evaluate)
CO6: Have a basic knowledge of the use of cryptography and network security; (Apply)

Suggested Readings

1. Andrew S. Tenenbaum, Computer Networks (Fourth Ed.), Prentice Hall of India, 2002
2. W Richard Stevens, UNIX Network Programming – Volume I (2nd Ed.), Prentice Hall of India, 2002
3. William Stallings, Data and Computer Communications (Sixth Ed.), Prentice Hall of India, 2000
4. Fred Halsall, Data Communication, Computer Networks and Opens Systems, (4th Ed.), Pearson Education, 2000
5. William Stallings, Cryptography and Networking Security - Principles and Practice, Pearson

Advanced Database Management Systems

Objective: The objective of this course is to introduce the basic conceptual background necessary to design and develop simple database systems. The course stresses on database modelling and design, physical file storage techniques and SQL query language facilities provided by database management systems. The course also presents some advanced database management concepts like query processing and optimization, transaction processing, concurrency control, recovery and security issues in database management systems.

Module I: Introduction
Introduction: Introduction to databases, characteristics of the database approach, database users and designers, role of a DBA, advantages of using a DBMS, data models, schemas, instances, DBMS architecture (Three-Schema Architecture), Database systems- Network, Hierarchical, Relational, Data Independence

Module II: Relational Data Model and ER Models
Relational Model: Domains, Attributes, Tuple and Relation; Super keys Candidate keys and Primary keys for the Relations. Relational Constraints: Domain Constraint, Key Constraint, Integrity Constraint. Relational Algebra: basic relational algebra operations-SELECT, PROJECT, UNION, INTERSECTION, SET DIFFERENCE, Cartesian PRODUCT, JOIN, Aggregate functions. Entity Relationship (ER) Model: Entities, Attributes, Relationships. More about Entities and Relationships, E-R Diagram, Conversion of E-R Diagram to Relational Database, Case Study.

Module III: Functional Dependencies and Normalization
Functional Dependencies, First Normal Form, Second Normal Form, Third Normal Form, Boyce-Codd Normal Form, Multivalued Dependencies.

Module IV: Data Storage, Indexing, Query Processing and Query Optimization
Overview of physical storage media, Magnetic Disks, RAID, File Organization: Fixed-length records, variable-length records. Organization of records in Files - Heap Files, Sequential File, Hashed Files. Indexing: Types of Single-level Ordered Indexes (Primary Indexes, Clustering Indexes, Secondary Indexes), Multilevel Indexes, Multilevel indexing using B tree and B+ tree, Indexing on multiple keys. Query Processing: Overview of query processing, Algorithms for query processing, Query Optimization

Module V: Transaction Processing and Concurrency Control
Transaction Processing: Transaction, ACID properties of transaction, transaction states, schedules, serializability, tests for serializability, recoverability, transaction definition in SQL. Concurrency Control: Concurrent execution of transaction, Lock- based techniques for concurrency control, Graph-based protocol, Timestamp based protocol, Deadlock, Deadlock prevention
methods, Deadlock detection Deadlock recovery.

Module VI: Recovery and Security
Recovery system: Types of failure, types of storage, recovery and Atomicity, Log-based recovery, shadow

paging, recovery with concurrent transactions, buffer management, logical undo logging, transaction rollback, checkpoints, restart recovery, fuzzy checkpointing.
Security: Security and Integrity-security violations, authorization and views, granting of privileges, security specifications in SQL, encryption, and statistical databases.

Course/Learning Outcomes
At the end of this course students will be able to:
CO1: Describe the fundamental concepts necessary for designing and implementing database systems and applications (knowledge)
CO2: Explain the core terms, concepts, and tools of relational database management systems (comprehension)
CO3: Design ER-diagrams and corresponding schema diagrams for handling database projects (synthesis)
CO4: Recall and identify the techniques used by a DBMS to process, optimize and execute high level queries. (Remembering)
CO5: Describe fundamentals of transaction processing system, including ACID properties of a transaction. (Understanding)
CO6: Illustrate concurrency control & analyze several concurrency control techniques for ensuring serializability, locking, timestamping. (Analysing)
CO7: Discuss some of the techniques that can be used for database recovery from failures. (Understanding)
CO8: classify security issues and threats to databases and summarize the control measure for securing databases against a variety of threats. (Creating, Understanding)

Suggested Readings

1. Silberschatz, HF Korth, S Sudarshan, Database System Concepts, Tata- McGraw Hill, 1997.
2. R Elmasri, SB Navathe, Fundamentals of Database Systems, Addisson, Wesley (Third Edition) 2000
3. DM Kroenke, Database Processing: Fundamentals, Design and Implementation, Prentice-Hall of India, (Eighth Edition) 2002.
4. GW Hansen, JV Hansen, Database Management and Design, Prentice-Hall of India, (2nd Edition) 2001.
5. Thomas M Connolly, Carolyn E Begg, Database Systems, A Practical Approach to Design, Implementation and Management, Addison Wesley Longman Ltd. 1999.

Internet Technology and Applications

Objective: The objective of the course is to familiarize the students with a discussion on Internet and its growth. It also provides the students a study on the basic services provided by the Internet. A familiarization on the markup languages, scripting languages and web application development are also being discussed to make the student competent to design websites. It has been taken into consideration that this paper assumes that the students must know well in advance about the various protocols of the Internet and the knowledge of HTML and databases.

Module I: Introduction to Internet
History of the Internet; History of the World Wide Web; W3C (World Wide Web Consortium); Levels of Internet Connectivity (Dial-up, Leased Line, DSL, VSAT); Requirements for Internet connectivity; Use of Browsers; Different types of browsers (IE, Opera, Netscape, Firefox); Search engines; FTP; Electronic Mail; Instant Messaging; DHCP; DNS; HTTP; URL; Proxy Servers.

Module II: Internet Markup Languages
a) XHTML: What is XHTML?; Components of XHTML; Elements of XHTML (Headers, Paragraphs,

b) Linking, Images, Special Characters, Lists, Tables, Forms, Framesets)
c) Cascading Style Sheets: Inline Styles; Embedded Style; Conflicting Style; Linking External Styles; W3C CSS Validation Service; Use of CSS (Positioning Elements, Backgrounds, Text flow)
d) XML: What is XML? Structuring Data; XML Namespaces; Document Type Definitions and Schemas; XML Vocabularies; Document Object Model (DOM and its methods); Extensible StyleSheet Language (XSL)

Module III: Web servers, Databases and Scripting Languages
a) Web servers: What is a web server; HTTP Request Types; System Architecture of a Web server; Client-side Scripting versus Server-side Scripting; Accessing Web servers; Apache Web Server.
b) Databases: Introduction to each one of the following: SQL, MYSQL, DBI
c) Scripting Languages: Javascript: Operators, Data Types, Control Structures, Functions, Arrays, String Manipulation. JQuery, ASP.NET. Introduction to Perl and CGI (Common Gateway Interface). JSP: Introduction; JSP Overview; Scripting; Standard Actions; Directives
d) Java Servlets: Servlet overview and architecture, Servlet Interface and Servlet life cycle, HttpServlet Class, HttpServletRequest Interface, HttpServletResponse Interface, Handling HTTP get Requests, Deploying a web application, Handling HTTP get requests containing data, Handling HTTP post requests.

Module IV: Web Application Development Using PHP
a) Web Site Design Considerations: Using Logical Design: Planning your website, drawing a map, using a top-down approach, flexibility, other web design metaphors. Creating templates. Creating a Compatible Design: Designing for different color depths, resolutions, different browser considerations, accommodating limited bandwidth. Validating your work.
b) PHP: Introduction to PHP; Data Types; Control Structures; Functions; Strings; Arrays; Querying Web Databases using PHP; Writing to Web Databases; Errors, Debugging and Deployment; Reporting in PHP; Validation Techniques in PHP.

Course/Learning Outcomes
At the end of this course students will be able to:
CO1: Recall and examine the growth of Internet and identify the history behind it. (Remembering)
CO2: identify and differentiate the various services provided by the internet. (Understanding)
CO3: experiment with various mark-up languages and scripting languages. (Applying)
CO4: Analyse and design a website of their own and can also identify the faults in the design. (Analysing)
CO5: develop and create a website of their own. (Creating)
CO6: summarize and validate a practical solution towards a web application development and also deploy a website of their own. (Evaluating)

Suggested Readings
1. Delhi Deitel and Deitel, Internet and World Wide Web: How to Program, 2nd Edition, Prentice Hall of India Pvt. Ltd., New
2. Hugh E. Williams and David Lane, PHP and MySQL, 2nd Edition, O'Reilly, Shroff Publishers and Distributors Pvt. Ltd.
3. Moss, K., Java Servlets (Second Edition), New Delhi: Tata McGraw-Hill
4. Internet Complete, 2nd Edition, BPB Publications., New Delhi
5. Douglas E. Comer, The Internet Book: Everything you need to know about Computer Networking and how the Internet works, 3rd Edition, Prentice Hall of India Pvt. Ltd., New Delhi

Sensor Network and Internet of Things

Objective: This course will introduce the students to the Internet of Things(IoT) and basic structure of communication protocols in sensor networks. The course is designed to build up basic understanding of how to set up an application specific IoT network with better orientation and representation of sensor nodes.

Module I

Introduction to IoT: Sensing, Actuation, Networking basics, Communication Protocols, Sensor Networks, Machine-to-Machine Communications, IoT Definition, Characteristics; IoT Functional Blocks, Physical design of IoT, Logical design of IoT, Communication models & APIs.

Module II

M2M to IoT-The Vision-Introduction, From M2M to IoT, M2M towards IoT-the global context, a use case example, Differing Characteristics. Definitions, M2M Value Chains, IoT Value Chains, An emerging industrial structure for IoT M2M vs IoT an Architectural Overview–Building architecture, Main design principles and needed capabilities, An IoT architecture outline, standards considerations. Reference Architecture and Reference Model of IoT.

Module III

IoT Architecture: Introduction, Reference Model, Functional View, Information View, Deployment and Operational View, Other Relevant architectural views. IoT with Arduino: Hands on Experience with Arduino (Firmware development & RTOS) - General Purpose I/O(GPIO), Serial Communication Interfaces: RS-232/485, Synchronous Peripheral Interfaces: I2C, SPI Sensors interfacing with Arduino. IoT with Raspberry PI: Hands on Experience with Raspberry PI (Firmware development) - Setting up Raspberry PI SD Card, Raspberry PI booting up & Initialization; General Purpose I/O(GPIO), Serial Communication Interfaces: RS-232/485, Synchronous Peripheral Interfaces: I2C, SPI; Sensors Interfacing with Raspberry PI Domain specific applications of IoT: Home automation, Industry applications, Surveillance, applications, Other IoT applications.

Module IV

Wireless Sensor Networks & Protocols: Wireless Sensor Networks (WSNs), Introduction to WSNs Topologies in WSNs; Wired Communication Protocols – Ethernet, Serial Communications; Wireless Communication protocols Wifi, RF, IPV4/V6, 6LOWPAN, ZigBee(IEEE802.15.4), BLE, GSM(2G/3G/LTE)
.

Course/Learning Outcomes

CO1: Explain the definition and usage of the term “Internet of Things” in different contexts and understand the key components that make up an IoT system (Remembering))
CO2: Understand why it is necessary to build a separate model for IoT and what parameters influences the operation of IoT network. (Understanding)
CO3: Apply the knowledge in designing IoT network for addressing real life issues for easing the day to day life activities. (Applying))
CO4: Apply the knowledge and skills acquired during the course to build and test a complete, working IoT system involving prototyping, programming and data analysis. (Applying)
CO5: Know what type of sensor protocols and architecture to adopt for efficient communication and what services offline and online to be used for problem solving. (Analyzing)

Suggested Readings

- Adrian McEwen and Hakim Cassimally, Designing the Internet of Things, Wiley, 2013.
- Qusay F. Hassan, Internet of Things A to Z: Technologies and Applications, Wiley-Blackwell, 2018.
- Peter Waher, Mastering Internet of Things: Design and Create Your Own IoT Applications Using Raspberry Pi 3, Packt Publishing, 2018.

- Kazem Sohraby, Minoli Daniel and Znati Taieb, Wireless sensor networks: technology, protocols, and applications. John Wiley & Sons, 2007.
- Waltenegus Dargie, Christian Poellabauer, Fundamentals of Wireless Sensor Networks: Theory and Practice (Wireless Communications and Mobile Computing), Wiley-Blackwell, 2010.

Master of Computer Application
Syllabus – Semester III

Enterprise Resource Planning

Objective: To help the student understand the conceptual elements of ERP and its theory and implementation. This is especially poignant in view of large number of organizations implementing ERP applications in recent years. The student will appreciate the impact that ERP brings into the daily operations of firms with respect to their productivity, integration, communication, etc.

Module I: ERP Basics

a) Evolution and structure of ERP, ERP concepts, growth of the ERP market, conceptual model of ERP, 2-tier and 3-tier architecture, elements in ERP architecture, advantages/benefits of ERP, overview of an enterprise, integrated management information, business modelling, integrated data model

b) ERP and related technologies: Business Process Reengineering (BPR), Management Information Systems (MIS), Decision Support Systems (DSS), Data Warehousing, Data Mining, Online Analytical Processing (OLAP), Supply Chain Management.

Module II: ERP Modules

Item types in ERP, Manufacturing, distribution and Financial requirements, item control module in ERP, Finance module, Manufacturing and Production Planning module, Sales and Distribution module, Plant Maintenance module, Quality Management module, Materials Management module, Capital Requirement Planning module, Purchase Control module, Human Resources modules; concept of Bill of materials, concept of formula management.

Module III: Profiling ERP Vendors

SAP AG: R/3 –, overview of R/3 system, R/3 modules, R/3 and the internet BAAN: Baan ERP modules, Baan ERP Tools Oracle: Oracle modules – Financials, Human Resources, Projects, Manufacturing, Supply chain. PeopleSoft: Accounting and control, Treasury Management, Performance Management, Sales and Logistics, Procurement.

Module IV: ERP Implementation Lifecycle

Elements of implementation methodology, Pre-evaluation Screening, Package evaluation, project planning phase, Gap Analysis, Business Process Re-engineering, configuration, Implementation team training, testing, product migration and support, Problems in ERP implementation, cost of ERP.

Module V Best Practices in ERP

a) Concept of Best Practices, concept of Customer Order Decoupling Point(CODP), Demand Management – Sales and Operations Planning, ERP scenario in India, future directions in ERP.

b) Case studies should also be introduced to highlight situations where ERP projects are implemented, and the success stories/benefits/difficulties of these implementations.

Course/Learning Outcomes

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

CO1: Recall the conceptual elements of ERP. (Remembering)

CO2: Demonstrate the Influence of ERP in Large Organizations. (Understanding)

CO3: Identify the impact of ERP into the daily operations of firms with respect to their productivity, integration, communication etc. (Applying)

CO4: Analyse the practical side of ERP implementation with different vendors. (Analysing)

CO5: Discuss and evaluate the best practices of ERP with various case studies and real time examples.

(Creating, Evaluating)

Suggested Readings

1. O'Leary, Daniel E, Enterprise Resource Planning Systems: systems, life cycle, electronic commerce and risk, Cambridge University Press.
2. Alexis Leon, Enterprise Resource Planning, 14th reprint, Tata McGraw Hill, New Delhi 2005
3. Rahul V Altekar, Enterprise Resource Planning (Theory and Practice), Prentice Hall India, New Delhi 2004
4. Alexis Leon, ERP Demystified, Tata McGraw Hill Pub. Co. Ltd, 2000
5. Kent Sandoe, Enterprise Integration, John Wiley and Sons
6. Garg and Venkitakrishnan, Enterprise Resource Planning: Concepts and Practice, 2nd edition, Prentice Hall India
7. Garg and Venkitakrishnan, ERPWARE: ERP Implementation Framework, Prentice Hall India
8. ERP – Concepts and Cases, ICFAI University Press, 2004

Cyber Law and IT Security

Course/Learning Outcomes:

- Apply fundamental concepts of Information Security threats and vulnerabilities to adopt right security measures and design real time scenarios. (Applying)
- Determine and analyze software vulnerabilities and security solutions to reduce the risk of exploitation. (Analyzing)
- Analyze and evaluate the cyber security needs of an individual/organization. (Analyzing, Evaluating)
- Design operational and strategic cyber security strategies and policies. (Creating)
- Analyze various types of cybercrime and formulate procedures for real world cybercrime Investigations. (Analyzing)

Module I

Object and Scope of the IT Act - Genesis, Object, Scope of the Act. Encryption -Symmetric Cryptography, Asymmetric Cryptography, RSA Algorithm, Public Key Encryption

Module II

Digital Signature- Technology behind Digital Signature, creating a Digital Signature, Verifying a Digital Signature, Digital Signature and PKI, Digital Signature and the Law. E-Governance and IT Act 2000- Legal recognition of electronic records, Legal recognition of digital signature, Use of electronic records and digital signatures in Government and its agencies, Certifying Authorities. Need of Certifying Authority and Power. Appointment, function of Controller. Who can be a Certifying Authority? Digital Signature Certifications. Generation, Suspension and Revocation of Digital Signature Certificate.

Module III

Domain Name Disputes and Trademark Law: Concept of Domain Names, New Concepts in Trademark, Jurisprudence, Cyber- squatting, Reverse Hijacking, Meta tags, Framing, Spamming, Jurisdiction in Trademark Dispute

Module IV

Cyber Regulations Appellate Tribunal: Establishment & Composition of Appellate Tribunal, Powers of Adjudicating officer to Award Compensation, Powers of Adjudicating officer to impose Penalty.

Module V
The Cyber Crimes (S-65 to S-74): Tampering with Computer Source Documents(S-65), Hacking with Computer System(S-66), Publishing of Information Which is Obscene in Electronic Form(s-67), Offences: Breach of Confidentiality & Privacy (S-72), Offences: Related to Digital Signature Certificate (S-73 & S-74)

Suggested Readings

- Farooq Ahmad, Cyber Law in India, Pioneer Books
- Vakul Sharma, Information Technology Law and Practice, Universal Law Publishing Co. Pvt. Ltd.
- Suresh T Vishwanathan, The Indian Cyber Law, Bharat Law house New Delhi.
- P.M. Bakshi and R.K.Suri, Hand book of Cyber and E-commerce Laws, Bharat Law house New Delhi.
- Rodney D. Ryder, Guide to Cyber Laws, Wadhwa and Company Nagpur.
- The Information Technology Act, 2000, Bare Act, Professional Book Publishers, New Delhi.

Machine Learning

Course/Learning Outcomes

- Learn mathematical principles used in learning algorithms and relate them to learning principles. (Understanding)
- Construct and classify learning algorithms used in different problems. (Applying)
- Know what and how to perform pre-processing to make dataset ready for learning algorithms(Analysing)
- Create learning models and evaluate the effect of it in a given problem domain. (Evaluation)

Module I
Mathematical Foundations of Machine Learning: Linear Algebra-Linear equations & vector spaces,matrix decomposition: determinant, Eigen values, singular value decomposition, matrix approximation, Calculus: Differentiation & partial differentiation, gradient of matrices, Probability & Distributions: Discrete and Continous probabilities, Gaussian distribution, Continous optimization: Gradient descent, Lagrange Multiplier.

Module II
Learning: supervised and unsupervised learning, necessary of supervised learning, KNN, regression models, Naive Bayes' classifier, decision trees, random forest classifier, SVM: linear, non-linear.

Module III
Data preprocessing & Scaling: Different kinds of preprocessing, Data transformations, Scaling: training data & testing data, Types of unsupervised learning, dimensionality reduction, clustering: k-Means, Fuzzy C -Means, DBSCAN, Comparing and evaluating clustering.

Module IV
Neural Network: Biological to Artificial neurons, Logical computations with neurons, perceptron, MLP & backpropagation, Tuning neural network hyperparameters, vanishing and exploding gradient problems, momentum optimization: AdaGrad, Adam optimization, Regularization: L1 & L2, Convolutional property of neural network.

Module V
Deep learning: Convolutional Neural Network: AlexNet, GoogleNet, ResNet, RNN: Training a RNN, deep

RNN, LSTM cell, Autoencoders: stacked autoencoders, sparse autoencoder.

Suggested readings:

- Hands-On Machine Learning with Scikit-Learn & TensorFLow. Second edition, Aurelien Geron,2017, O' Reilly Media.
- Machine Learning-An Algorithmic Perspective, second edition, Stephen marshland,2015, CRC Press.
- Deep Learning, John D. Kelleher,2019, Massachusetts Institute of Technology(MIT).

Web Programming

UNIT - 1

Fundamentals of Web: Internet, WWW, Web Browsers, and Web Servers; URLs; MIME; HTTP; Security; the Web Programmers Toolbox. HTML: Origins and evolution of HTML; Basic syntax; Standard HTML document structure; Basic text markup, Hypertext Links; Lists; Tables; Forms; Frames.

UNIT - 2

Cascading Style Sheets (CSS): Introduction; Levels of style sheets; Style specification formats; Selector forms; Property value forms; Font properties; List properties; Color; Alignment of text; The Box model; Background images; The and tags;

UNIT - 3

JavaScript: Overview of JavaScript; Object orientation and JavaScript; General syntactic characteristics; Primitives, operations, and expressions; Screen output and keyboard input; Control statements; Object creation and modification; Arrays; Functions; Constructor; Pattern matching using regular expressions; Errors in scripts; Examples.

UNIT -4

CGI Programming: The Common Gateway Interface; CGI linkage; Query string format; CGI.pm module; A survey example; Cookies.

Reference Books:

1. Programming the World Wide Web – Robert W. Sebesta, 4 th Edition, Pearson Education, 2008.
2. Internet & World Wide Web How to H program – M. Deitel, P.J. Deitel, A. B. Goldberg, 3 rd Edition, Pearson Education / PHI, 2004.
3. Web Programming Building Internet Applications – Chris Bates, 3 rd Edition, Wiley India, 2006.
4. The Web Warrior Guide to Web Programming – Xue Bai et al, Thomson, 2003.

Registrar
Assam Don Bosco University
Tapesia Gardens, Sonapur
Assam-782402, INDIA

ASSAM DON BOSCO UNIVERSITY
SONAPUR
ASSAM-782402
INDIA
GUWAHATI