Indira Gandhi Delhi Technical University for Women

(Established by Govt. of Delhi vide Act 09 of 2012)

Kashmere Gate, Delhi–110006

Scheme of Examination

&

Detailed Syllabus

(w.e.f. Academic Year 2013-2014 onwards)

for

Bachelor of Technology
(Information Technology)

Department of Information Technology

PROGRAMME OUTCOMES

Engineering Graduates will be able to:

PO1. Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2. Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4. Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5. Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.

PO6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9. Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10. Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11. Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change

PROGRAMME SPECIFIC OUTCOMES

Programme Specific Outcomes: Department of Information Technology

PSO1. The graduates shall have a scientific outlook with a wide spectrum fundamental knowledge of applied mathematics, basic engineering principles of physics and mechanics and their application as problem solving skills in the designing software applications.

PSO2. With a B.Tech degree in the field of Information Technology, graduates will be able to analyze and recommend the appropriate IT infrastructure needed to implement the project in the field of software application development. The graduates shall have thorough knowledge in design, develop, and testing the software systems to provide solutions to real-world problems.

PSO3. The graduates shall have demonstrable interpersonal and social communication skills along with team building, interpersonal relationship, group discussion, current affairs etc.

PSO4. The Graduates of Information technology will be able to use and implement core Information Technology concepts like human-computer interaction, information management, programming, and networking. The graduates of information technology can effectively integrate IT-based solutions into user-based environment.

THIRD SEMESTER

S.No.	Paper Code	Paper Title	L	P	Credits	Course Category
THEORY PAPERS						
1	BCS 201	Discrete Mathematics	4	-	4	DC
2	BIT 203	Database Management Systems	4	-	4	DC
3	BCS 207	Data Structure	4	-	4	DC
4	BIT 209	Object Oriented Programming	4	-	4	DC
5	BEC 211	Analog & Digital Electronics	4	-	4	ES
PRACTICAL/VIVA VOCE						
1	BIT 253	Database Management Systems Lab	0	2	1	DC
2	BCS 257	Data Structure Lab	0	4	2	DC
3	BIT 259	Object Oriented Programming using C++ and JAVA Lab	0	2	1	DC
4	BEC 261	Analog & Digital Electronics Lab	0	2	1	ES
		TOTAL	**20**	**10**	**25**	

FOURTH SEMESTER

S. No.	Paper Code	Paper Title	L	P	Credits	Course Category
THEORY PAPERS						
1	BAS 202	Advanced Engineering Mathematics	4	-	4	ES
2	BCS 204	Computer Organization & Architecture	4	-	4	DC
3	BCS 206	Analysis & Design of Algorithms	4	-	4	DC
4	BIT 208	Operating System	4	-	4	DC
5	BIT 210	Object Oriented Software Engineering	4	-	4	DC
PRACTICAL/VIVA VOCE						
1	BCS 254	Computer Organization & Architecture Lab	0	2	1	DC
2	BCS 256	Analysis & Design of Algorithms Lab	0	4	2	DC
3	BIT 258	Operating System Lab (using LINUX as Case Study)	0	2	1	DC
4	BIT 260	Object Oriented Software Engineering Lab	0	2	1	DC
		TOTAL	**20**	**10**	**25**	

FIFTH SEMESTER

S.No.	Paper Code	Paper Title	L	P	Credits	Course Category
THEORY PAPERS						
1	BCS 301	Theory of Computation	4	-	4	DC
2	BIT 303	Computer Graphics & Multimedia	4	-	4	DC
3	BIT 305	Requirement & Estimation Techniques	4	-	4	DC
4	BIT 307	Data Communication & Computer Networks	4	-	4	DC
5	BIT 309	Data Warehousing & Data Mining	4	-	4	DC
6	BAS 311	Human Values & Professional Ethics	3	-	3	HS
PRACTICAL/VIVA VOCE						
1	BIT 353	Computer Graphics & Multimedia Lab	0	4	2	DC
2	BIT 355	Requirement & Estimation Techniques Lab	0	2	1	DC
3	BIT 357	Data Communication & Computer Networks Lab	0	2	1	DC
4	BIT 359	Data Warehousing & Data Mining Lab	0	2	1	DC
		TOTAL	**23**	**10**	**28**	

SIXTH SEMESTER

S.No.	Paper Code	Paper Title	L	P	Credits	Course Category
THEORY PAPERS						
1	BIT 302	Web & Mobile Technologies	4	-	4	DC
2	BCS 304	Compiler Design	4	-	4	DC
3	BCS 306	Network Programming	4	-	4	DC
4	BCS 308	Cloud Computing	4	-	4	DC
5	BIT 310	Artificial Intelligence	4	-	4	DC
6	BAS 312	Engineering Economics	3	-	3	HS
PRACTICAL/VIVA VOCE						
1	BIT 352	Web & Mobile Technologies Lab	0	2	1	DC
2	BCS 354	Compiler Design Lab	0	2	1	DC
3	BCS 356	Network Programming Lab	0	2	1	DC
4	BCS 358	Cloud Computing Lab	0	2	1	DC
5	BIT 360	Artificial Intelligence Lab	0	2	1	DC
		TOTAL	**23**	**10**	**28**	

SEVENTH SEMESTER

S.No.	Paper Code	Paper Title	L	P	Credits	Course Category
THEORY PAPERS						
1	BIT 401	Software Testing	4	-	4	DC
2	BIT 403	Big Data Analytics	4	-	4	DC

EIGHTH SEMESTER

S.No.	Paper Code	Paper Title	L	P	Credits	Course Category
THEORY PAPERS						
1	BIT 402	Software Project Management	4	-	4	DC
2	BIT 404	Cyber Security Management	4	-	4	DC

Paper Code: BCS 201	**L**	**P**	**C**
Paper Title: Discrete Mathematics	**4**	**0**	**4**

Introduction: The discrete structures/mathematics subject introduces Propositional logic, Sets, Relations, and Functions, Algebraic structures, Graphs and Trees required for building mathematical foundation of computer science

Course Objectives:

- To introduce and understand the fundamental notions in discrete mathematics.
- To understand the basic concept of an algorithm and its application in combinatorial Mathematics.
- To introduce the basic properties of graphs and trees and model simple applications.
- To learn concepts of discrete mathematics

Pre-requisites: Basic concepts of set theory.

Course Outcomes: After completion of the course, the students will be able:

CO1: To convert a logic sentence in terms of predicates, quantifiers, and logical connectives and their validation.
CO2: Able to use logical notations to define and reason about fundamental mathematical concepts such as sets relations, functions and combinatorics.
CO3: Able to use logical notations to define and reason about fundamental mathematical concepts of abstract algebra.
CO4: Apply algorithms and use graphs and trees as tools to analyze and simplify Problems.

Pedagogy:
This course is structured around continuous progress. It will include a combination of lectures, and group activities focused on experiential learning, in-class discussions, regular assessments, and case studies. The topics will be presented to students using real-world scenarios and problem-solving activities

Unit-I

Set Theory: Introduction, Combination of sets, Multisets, Ordered pairs, Proofs of some general identities on sets. **Relations:** Definition, Operations on relations, Properties of relations, Composite Relations, Equality of relations, Recursive definition of relation, Order of relations. **Functions:** Definition, Classification of functions, Operations on functions, Recursively defined functions. Growth of Functions. **Natural Numbers:** Introduction, Mathematical Induction, Variants of Induction, Induction with Nonzero Base cases, Proof Methods, Proof by counter – example, Proof by contradiction.

[10 Hrs]

Unit-II

Algebraic Structures: Definition, Groups, Subgroups and order, Cyclic Groups, Cosets, Lagrange's theorem, Normal Subgroups, Permutation and Symmetric groups, Group Homomorphisms, Definition and elementary properties of Rings and Fields, Integers Modulo n. **Partial order sets:** Definition, Partial order sets, Combination of partial order sets, Hasse diagram. **Lattices:** Definition, Properties of lattices – Bounded, Complemented, Modular and Complete lattice. **[10 Hrs]**

Unit-III

Binary Logic: Proposition, well-formed formula, Truth tables, Tautology, Satisfiability, Contradiction, Algebra of proposition, Theory of Inference. **Predicate Logic:** First order predicate, well-formed formula of predicate, quantifiers, Inference theory of predicate logic. **Multivalued Logic:** Fuzzy Logic, Introduction to fuzzy sets, Operations on fuzzy sets.

[10 Hrs]

Unit-IV

Trees: Definition, Binary tree, Binary tree traversal, Binary search tree. **Graphs:** Definition and terminology, Representation of graphs, Multigraphs, Bipartite graphs, Planar graphs, Isomorphism and Homeomorphism of graphs, Euler and Hamiltonian paths, Graph coloring. **Recurrence Relation & Generating function:** Recursive definition of functions, Recursive algorithms, Method of solving recurrences. **Combinatorics:** Introduction, Counting Techniques, Pigeonhole Principle, Countability, Dovetailling**.** **[10 Hrs]**

TEXT BOOKS:

1. Kenneth H. Rosen, "Discrete Mathematics and Its Applications", 6/e, McGraw-Hill, 2006.
2. B. Kolman, R.C. Busby, and S.C. Ross, "Discrete Mathematical Structures", 5/e, Prentice Hall, 2004.
3. C.L. Liu, "Elements of Discrete Mathematics", TMH, 2000.

REFERENCE BOOKS:

1. Koshy, “Discrete Structures”, Elsevier Pub. 2008.
2. E.R. Scheinerman, “Mathematics: A Discrete Introduction”, Brooks/Cole,2000.
3. R.P. Grimaldi, “Discrete and Combinatorial Mathematics”, 5/e, Addison Wesley, 2004.
4. Jean Paul Trembley, R Manohar, “Discrete Mathematical Structures with Application to Computer Science”, McGraw-Hill, Inc. New York, NY, 1975.
5. John Yen & Reza Langari ,”Fuzzy logic intelligence control and information”, Prentice Hall, 1999.

Paper Code: BIT 203	**L**	**P**	**C**
Paper Title: Database Management Systems	**4**	**0**	**4**

Introduction:
Database Management System (DBMS) is used to create and manage databases. The main aim of a DBMS is to supply a way to store-up and retrieve the desired database information as per the application requirement, which is both convenient and efficient.

Course Objectives:
- To introduce the concepts of database management systems
- To design relational databases by applying normalization techniques to normalize the database
- Strong practice in SQL programming through a variety of database problems.
- Understand the needs of database processing and learn techniques for controlling the consequences of concurrent data access.

Pre-requisites: Basic concepts of set theory

Course Outcomes: After completion of the course, the students will be able:

CO1: To have a high-level understanding of major DBMS components and their function.

CO2: To model an application's data requirements using conceptual modeling tools like ER diagrams and design database schemas based on the conceptual model.

CO3: To write SQL commands to create tables and indexes, insert/update/delete data, and query data in a relational DBMS

CO4: To understand the concept of Transaction, concurrency, and Query processing.

Pedagogy:
Lecture delivery via discussions, whiteboard, slideshows, online learning material, Lab work with exercises on SQL.

UNIT – I

Overview of Concepts and Conceptual Database Design: Database administrator & Database Users, Characteristics of the Database, Database Systems, Concepts and Architecture, Data Models, Schemes & Instances, DBMS Architecture & Data Independence, Database Languages & Interfaces, Overview of Hierarchical, Network & Relational Database Management Systems, Data Modeling Using The Entity-Relationship Model – Entities, Attributes and Relationships, Cardinality of Relationships, Strong and Weak Entity Sets, Generalization, Specialization, and Aggregation. **[10 Hrs]**

UNIT – II

Relational Model, Languages & Systems: Relational Model Concepts, Relational Model Constraints, Translating your ER Model into Relational Model, Relational Algebra, Relational Calculus (tuple calculus)
SQL: A Relational Database Language, Data Definition in SQL, View and Queries in SQL, Specifying Constraints and Indexes in SQL, Practicing SQL commands using ORACLE
[10 Hrs]

UNIT – III

Relational Database Design: Functional Dependencies & Normalization for Relational Databases, Functional Dependencies, Normal Forms (1NF, 2NF, 3NF , BCNF,4NF,5NF), Lossless Join and Dependency Preserving Decomposition, Multivalued Dependency, Join dependency.
Transaction Management: Transaction Concept and State, Implementation of Atomicity and Durability, Serializability, Recoverability, Implementation of Isolation **[10 Hrs]**

UNIT – IV

Concurrency Control: Lock-Based Protocols, Timestamp-based Protocols, Deadlock Handling, Recovery System, Failure Classification, Storage Structure, Recovery and Atomicity, Log-based Recovery.
Query Processing: Query Processing Overview, Measures of Query Cost, Selection Operation, Sorting, Join Operation, Other Operations, Evaluation of Expressions.
Framework of Distributed Database Management Systems, **Introduction to Enhanced Databases:** Multimedia Databases, Object Oriented Databases, Mobile Databases and Case study on various commercially available DBMS**.** **[10 Hrs]**

TEXTBOOKS:

1. Korth, Silberschatz, "Database System Concepts", 6th Ed., TMH, 2010.
2. Elmsari and Navathe, "Fundamentals of Database Systems", 6th Edition, Pearson, 2013.
3. C. J. Date and Kannan, "An Introduction to Database Systems", 8th Ed., Pearson, 2006.

REFERENCE BOOKS:

1. Ceri and Pelagatti , Distributed Databases : Principles & Systems, McGraw-Hill Computer Science Series, 2008.
2. J. D. Ullman, "Principles of Database Systems", 2nd Ed., Galgotia Publications, 1999.
3. Conolly & Begg, " Database Management Systems, 5th Edition, Pearson Education Asia,2010

Paper Code: BCS 207	**L**	**P**	**C**
Paper Title: Data Structure	**4**	**0**	**4**

Introduction: Data structure is a specific way to store and organize data in a computer's memory so that this data can be used efficiently later. This course introduces various data structures and their useful applications in the computer science domain.

Course Objectives:

- To impart the basic concepts of data structures and algorithms
- To understand concepts about searching and sorting techniques
- To Understand basic concepts about stacks, queues, lists, trees and graphs
- To understand writing algorithms and step by step approach in solving problems with the help of fundamental data structures.

Course Outcomes

CO1: Explain the concept of time and space complexity of the algorithm.
CO2: Understand the use of fundamental data structures and algorithm appropriately to solve a number of computational problems.
CO3: Apply various algorithms to solve the problems of searching and of data.
CO4: Design programs using a variety of data structures such as stacks, queues, hash tables binary trees, search trees, heaps, graphs, and B-trees.

Pedagogy:

This course is structured around continuous progress. It will include a combination of lectures, and group activities focused on experiential learning, in-class discussions, regular assessments and case studies. The topics will be presented to students using real-world scenarios and problem-solving activities.

UNIT–I

Fundamentals of algorithm analysis: Time and space complexity, Elementary data structures and their applications. **Arrays:** ordered lists, representation of arrays, sparse matrices. **Linked List**: singly and doubly linked lists, Circular Linked list **Stacks:** Primitive operations, Application of stacks, multiple stacks. **Queue:** Primitive operations, Application of queues, multiple queues. **[10 Hrs]**

UNIT–II

Recursion: Recursive definition and processes, Factorial function, Fibonacci series, Recursive binary search. **Trees:** Binary Trees; Definition, traversal, threaded binary Tree. **Graphs:** Representation, traversal, connected components, shortest path algorithms, topological sort, Minimum Spanning Tree; Definitions and algorithms. **[10 Hrs]**

UNIT–III

Searching: Sequential Search, Binary Search, Tree Searching, Binary Search Tree, Insertion & Deletion, AVL trees, Multi way search tree, B tree, B^+ Tree. **Hashing:** Hash Function**,** Hash Table, Hashing Techniques. **[10 Hrs]**

UNIT–IV

Sorting: Quick Sort, Merge Sort, Heap Sort and other sorting techniques, K-way Merge Sort. **Files**: Creation and Processing of files, File handling, Reading/ Writing of files, Operations of files, File Organization, Indexing, Error handling. **Storage Management:** Automatic List Management, Reference Count Method, Garbage Collection, Collection and Compaction. **[10 Hrs]**

TEXT BOOKS:

1. Y. Langsam et. al., "Data Structures using C and C++", PHI, 1999.
2. R. L. Kruse, B. P. Leung, C. L. Tondo, "Data Structures and program design in C", PHI, 2000.

REFERENCE BOOKS:

1. Schaum's outline series, "Data Structure", TMH, 2002
2. E. Horowitz and S. Sahani, "Fundamentals of Data Structures", Galgotia Booksource Pvt. Ltd, 1999
3. Yashwant Kanetkar, "Data Structure through C", BPB, 2005

Paper Code: BIT 209	**L**	**P**	**C**
Paper Title: Object-Oriented Programming	**4**	**0**	**4**

Introduction:

This course provides in-depth coverage of object-oriented programming principles and techniques. Topics include classes, objects, overloading, data abstraction, information hiding, encapsulation, inheritance, polymorphism, file processing, templates, exceptions, container classes etc. The course material embraces the C++11 language standard/ Python with numerous examples demonstrating the benefits of C++11/Python. In the end some basics of Java will be covered.

Course Objective:

- To learn the syntax and semantics of the C++/java/python programming language.
- To understand object-oriented programming concepts, and apply them in solving problem
- To understand and design efficient programming.
- To demonstrate skills in writing programs using Java programming.

Pre-requisite: Basics of C Programming.

Course Outcomes: After completion of the course, the students will be able to:

CO1: Understand fundamentals syntax and their use to develop Object Oriented
CO2: Java/Python program to express proficiency and improve effective programming skills
CO3: Understand commonly used operations for file system, exception handling and create namespace solutions.
CO4: Implement Java based program and make effective use of Tools

Pedagogy:
This course is structured around continuous progress. It will include a combination of lectures, and group activities focused on experiential learning, in-class discussions, regular assessments and case studies. The topics will be presented to students using real-world scenarios and problem-solving activities.

UNIT - I

Introduction: Need for Object Oriented Programming, Comparison of Programming paradigms, Characteristics of Object-Oriented Programming Languages, Introduction to Object Oriented concepts (classes, objects, encapsulation, inheritance, data hiding, abstraction, polymorph ism), Overview and characteristics of C++ , Fundamentals Data Types & Literals Variables, Arrays, Operators, Control of Flow in OOP, Compilation and Execution Process in C++, Reference vs. Pointer variable.**Classes and Objects:** C++ & Java class declaration, Role of private, public and protected access specifiers, Memory organization of class, inline function, friend function, static members , constructor and destructors, instantiation of objects, default parameter value, object types, C++ garbage collection, dynamic memory allocation, new and delete operator

[10 Hrs]

UNIT - II

Polymorphism in C++: Function overloading, Constructor overloading, Compile time polymorphism, Overloading Rules, Operator Overloading (Unary and Binary) as member function/friend function.

Inheritance in C++: Inheritance, Types of Inheritance, Use of protected access specifier, Virtual base class, Ambiguity resolution using scope resolution operator and Virtual base class, Overriding inheritance methods, Constructors and Destructor in derived classes. Runtime polymorphism, Pointer to objects, Virtual Functions (concept of virtual table), pure virtual functions, Abstract Class**.** **[10 Hrs]**

UNIT - III

Managing Input / Output in C++: Concept of streams, console I/O – formatted and unformatted, Manipulators, File I/O – Predefined classes, file opening & closing, file manipulation, read & write operations, sequential and random file access.

Exception Handling in C++: Basic mechanism, Throwing, Catching and Re-throwing.

Namespace: Basic concept, role of scope resolution operator and using keyword.

[10Hrs]

UNIT - IV

Introduction to Java- Overview and characteristics of Java, Data types, Organization of the Java Virtual Machine, Compilation and Execution Process in java, Classes: String and String Buffer classes, Wrapper classes, using super keyword, Multilevel hierarchy abstract and final classes, Object class, Packages and interfaces, Access protection, Exception Handling: Fundamentals exception types, uncaught exceptions, throw, throws, final, built in exception, creating your own exceptions, Multithreaded Programming: Fundamentals, Java thread model: priorities, synchronization, thread classes, Runnable interface, inter thread Communication, suspending, resuming and stopping threads

[10Hrs]

TEXT BOOKS:

1. E. Balaguruswamy, "Object Oriented Programming with C++", 4th Edition, TMH, 2011.
2. Bjarne Stroustrup , "The C++ Programming Language", Pearson, 3rd Edition, 200.
3. Patrick Naughton and Herbertz Schildt , "Java-2: The Complete Reference", TMH, 1999.

REFERENCE BOOKS:

1. Schildt Herbert, "C++: The Complete Reference", Tata McGraw Hill, 4th Ed., 1999.
2. R. Nageswara Rao/kogent Solutions," Core Java: An Integrated Approach: Covers Concepts, programs and Interview Questions", 2008.
3. Pandey, "JAVA Programming", Pearson, 2012

Paper Code: BEC 211	**L**	**P**	**C**
Paper Title: ANALOG AND DIGITAL ELECTRONICS	**4**	**0**	**4**

Introduction:

The course will introduce fundamental principles of analog and digital electronics. The course provides sufficient basic knowledge for the undergraduate to understand the design of diodes and transistor-based circuits, op-amps and their applications as well as the design of digital circuits.

Course Objective:

- Understand the design and analysis of various analog electronic circuits.
- Understand the fundamental concepts and techniques used in digital electronics.

Pre-requisite:

- Student should have the prior knowledge of semiconductor electronics.
- Basic concept of number system.

Course Outcome: After completion of the course, student will be able to:

CO1: Understand basic electronic devices such as diodes, BJT & FET transistors
CO2: Understand various applications of Op-Amp.
CO3: Analyse logic processes and implement logical operations using combinational logic Circuits.
CO4: Design sequential circuits.

Pedagogy:

Class room teaching, problem solving approach, practical based learning, tutorials.

UNIT-I **[12 Hours]**

Semiconductor diodes, Characteristics and operation, Applications of p-n junction diode. **Bipolar Junction Transistor:** Construction and Operation, Common base (CB) configuration, Transistor amplifying action, Common emitter (CE) and Common collector (CC) configurations, definition of α and β, saturation, regions of operation of transistor, biasing methods.

Amplifiers: CE, CC, CE amplifier circuits and their comparisons, RC coupled amplifier, Frequency response, Gain-bandwidth, and Darlington pair, Class B push pull amplifier.

Feedback: Concept of negative & positive feedback and their relative advantages & disadvantages, Sinusoidal oscillators.

UNIT-II **[10 Hours]**

Field Effect Transistor: Introduction, JFET characteristics, Depletion & enhancement MOSFET, CMOS. Operational amplifier: Characteristics of ideal Op-Amp, Inverting & non-inverting amplifier, Differential amplifier, Adder & Subtractor, Integrator, Differentiator, Instrumentation amplifier, Schmitt trigger, Astable multivibrator.

UNIT-III **[10 Hours]**

Digital electronics: Analog & digital signals, Logic gates, Boolean algebra. Standard representation of logical functions, K-map representation and simplification of logical functions, Don't care conditions, X-OR & X-NOR simplification of K-maps.

Combinational circuits: Multiplexers, Demultiplexers, Decoders & Encoders, Adders & Subtractor, Code converters, Comparators, Decoder/drivers for display devices, A/D and D/A converters.

UNIT-IV **[10 Hours]**

Flip Flops: S-R, J-K, D & T Flip-flops, Excitation table of a flip-flop, Race around condition

Sequential circuits: Shift registers, Ripple counter, Design of synchronous counters and Sequence detectors, Sequence generators.

Text Books:

1. Morris Mano, "Digital Design", PHI, 5th edition, 2013.
2. Millman and Halkias, "Electronic Devices and Circuits" TMH, 4th Edition, 2015.
3. Salivahanan, Suresh Kumar, Vallavaraj, "Electronic Devices and Circuits" TMH, 4th Edition, 2016.

Reference Books:

1. Balbir Kumar and S. B. Jain, "Electronic Devices and Circuits" PHI, 2nd Edition 2014.
2. R.P. Jain, "Modern Digital Electronics", TMH, 4th Edtion, 2010
3. Roy Choudhury and Jain, "Linear Integrated Circuits", New Age Publishers, 4th Edition, 2017.

Paper Code: BCS 204	**L**	**P**	**C**
Paper Title: Computer Organization and Architecture	**4**	**0**	**4**

Introduction:

In order to achieve complete understanding of computer systems, it is always important to consider both hardware and software design of various computer components. In other words, every functionality of the computer has to be studied to increase the performance of the computer. Computer organization and architecture mainly focuses on various parts of the computer in order to reduce the execution time of the program, improve the performance of each part.

Course Objective:

- Understand the basics of computer organization: structure and operation of computers and their peripherals.
- Understand basic processing units and organization of simple processors.
- Expose different ways of communicating with I/O devices and standard I/O interfaces.
- Understand the concept of pipelining and other large computing systems.

Pre-requisite: Fundamentals of computers and digital logic.

Course Outcome:

CO1: Ability to Demonstrate an understanding of the design of the functional units of a digital computer system.

CO2: Explain the instruction set, instruction formats and Addressing modes of CPU

CO3: Ability to Recognize and manipulate representations of numbers stored in digital computers and perform Basic arithmetic Operations

CO4: Ability to analyze memory hierarchy and its impact on computer Cost/performance.

Pedagogy:

This course is structured around continuous progress. It will include a combination of lectures, and group activities focused on experiential learning, in-class discussions, regular assessments and case studies. The topics will be presented to students using real-world scenarios and problem-solving activities.

UNIT-I

Basic functional blocks of a computer and its Representation: Functional units, Basic operational concepts, Bus structures, Performance and metrics, Instructions and instruction sequencing, Hardware–Software Interface, Instruction set architecture, Addressing modes, RISC, CISC, ALU design, Fixed point and floating point operations, Case study of a CPU (Intel Atom Board). **[10 Hrs.]**

UNIT-II

CPU Control Unit Design: Execution of a complete instruction, Multiple bus organization, Hardwired control, Micro programmed control, Computer arithmetic, Integer addition and subtraction, ripple carry adder, carry look-ahead adder, etc. multiplication - shift-and-add, Booth multiplier, carry save multiplier etc. **Pipeline-** Basic concepts, Data hazards, Instruction hazards, Influence on instruction sets, Data path and control considerations, Performance considerations, Exception handling. Case Study of Intel Atom Board. **[10 Hrs.]**

UNIT-III

Memory system design: Basic concepts, Semiconductor RAM – ROM, Speed, Size and cost, Cache memories, Improving cache performance, Virtual memory, Memory management requirements, Associative memories, Secondary storage devices. Case study of Intel Atom Board. **[10 Hrs.]**

UNIT-IV

I/O Organization: Accessing I/O devices, Programmed Input/Output, Interrupts, Direct Memory Access, Buses, Interface circuits, Standard I/O Interfaces (PCI, SCSI, USB), I/O devices and processors. **[10 Hrs.]**

TEXT BOOKS:

1. John P. Hayes, "Computer Architecture and Organization", McGraw-Hill, 1998.
2. William Stallings, "Computer Organization and Architecture: Designing for Performance", Pearson Education, 2010.
3. M.Morris Mano, "Computer System Architecture", PHI, 2nd Edition.

REFERENCE BOOKS:

1. David A. Patterson and John L. Hennessy, "Computer Organization and Design: The Hardware/Software Interface", Elsevier, 2012.
2. Carl Hamachar, Zvonco Vranesic and Safwat Zaky, "Computer Organization", McGraw Hill, 1990.
3. Vincent P. Heuring and Harry F. Jordan, "Computer Systems Design and Architecture", Pearson Education, 2nd Edition, 1996

Paper Code: BCS 206	**L**	**P**	**C**
Paper Title: Analysis & Design of Algorithms	**4**	**0**	**4**

Introduction:
This course deals with teaching different methodologies of designing algorithms. There are certain standard approaches to analyzing the algorithms. This course deals with all aspects of this analysis. It teaches the concepts of Dynamic programming, different approaches of algorithm design like Greedy approach etc.

Course Objectives:

- Introduction, learning and analysis of performances of algorithmic efficiency of approaches such as searching, sorting etc.
- Introduction, learning and analysis of greedy paradigms.
- Introduction, learning and analysis of dynamic programming and backtracking.
- Introduction, learning and analysis of computational complexity and branch & bound.

Pre-requisites: Data structures

Course Outcomes: After completion of the course, the students will be able to:
CO1: Understand asymptotic complexities of the algorithms and design algorithms using Divide and conquer approach
CO2: Understand and apply greedy and dynamic programming approaches for design in algorithms
CO3: Understand, analyse and implement various graph algorithms and the backtracking approach of algorithm design.
CO4: Understand and implement different string-matching algorithms and NP-Complete Problems

Pedagogy:
This course is structured around continuous progress. It will include a combination of lectures, and group activities focused on experiential learning, in-class discussions, regular assessments and case studies. The topics will be presented to students using real-world scenarios and problem-solving activities.

UNIT–I

Preliminaries: Review of growth of function, Recurrences: Substitution method, Iteration method, Master method. **Divide and Conquer Approach:** Merge Sort, Quick sort, Simultaneous Max and Min Problem, Strassen's algorithm for Matrix Multiplications. **[10 Hrs.]**

UNIT–II

Greedy Algorithms: Elements of Greedy strategy, knapsack problem, job sequencing with deadlines, minimum spanning trees, Activity selection problem, Huffman Codes. **Dynamic Programming:** Elements of Dynamic Programming, Matrix Chain Multiplication, Longest common subsequence and optimal binary search trees problems. **[10 Hrs.]**

UNIT–III

Graph Algorithms: DFS, BFS, Topological Sort, Strongly Connected Components, Kruskal's and Prim's algorithm for MST, Dijkstra's and Bellman Ford Algorithm, All pair shortest paths Algorithm. **Back Tracking:** General method, 8 queen's problem, **Branch and Bound:** General Method, 0/1 knapsack. **[10 Hrs.]**

UNIT–IV

String matching: Naïve String Matching algorithm, Rabin-Karp Algorithm, String Matching with finite automata, The Knuth-Morris Pratt algorithm. **NP-Complete Problem:** Polynomial- time verification, NP-Completeness and Reducibility, NP-Completeness Proof, NP-Complete problems. **[10 Hrs.]**

TEXT BOOKS:

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest, Clifford Stein, "Introduction to Algorithms", 2nd Ed., PHI, 2004.
2. Ellis Horowitz and Sartaz Sahani, "Fundamental of Computer Algorithms", Galgotia Publications, 1999.

REFERENCE BOOKS:

1. A. V. Aho, J. E. Hopcroft, J. D. Ullman, "The Design and Analysis of Computer Algorithms", Addition Wesley, 1998.
2. D. E. Knuth, "The Art of Computer Programming", 2nd Ed., Addison Wesley, 1998
3. Jean Paul Trembley, Richard B.Bunt, "Introduction to Computer Science- An algorithmic approach", T.M.H, 2002.

Paper Code: BIT 208	**L**	**P**	**C**
Paper Title: Operating System	**4**	**0**	**4**

Introduction:

This course will aim at introducing classical internal algorithms and structures of modern operating systems including CPU scheduling, memory management, and device management. Topics including file systems, virtual memory, disk request scheduling, concurrent processes, deadlocks, security, and integrity will be covered.

Course Objectives:

- To learn the fundamentals of Operating Systems & the mechanisms of OS to handle processes and their communication.
- To learn the mechanisms involved in memory management.
- To gain knowledge on OS architecture, mutual exclusion algorithms, deadlock detection algorithms etc.

Pre-requisite: Basic programming knowledge in C or C++.

Course Outcome: After studying this course, students will be able:

CO1: To understand various types of OS, basic concepts, various functions of different OS, process management & CPU scheduling.
CO2: To compare and contrast various memory management schemes like paging, segmentation and to apply different deadlock handling algorithms.
CO3: To implement different disk scheduling algorithms, to apply and use various process synchronization techniques and device management strategies.
CO4: To analyse management of I/O and different file handling & directory implementation schemes OS.

Pedagogy:
This course is structured around continuous progress. It will include a combination of lectures, and group activities focused on experiential learning, in-class discussions, regular assessments and case studies. The topics will be presented to students using real-world scenarios and problem-solving activities.

UNIT–I

Introduction: What is an Operating System, **Types of O.S:** Simple Batch, Multi-programmed Batched, Time-Sharing, Personal-computer, Parallel, Distributed, Real-Time, Mobile
Operating-System Structures: Layered Architecture, System Calls, System Programs, System Structure, Virtual Machines.**Processes**: Process Concept, Process Scheduling, Operation on Processes, Cooperating Processes, Inter-process Communication, Threads, Multithreaded Programming. **CPU Scheduling:** Basic Concepts, Scheduling Criteria, Scheduling Algorithms, Multiple-Processor Scheduling, Real-Time Scheduling

[10 Hrs]

UNIT–II

Process Synchronization: Background, Critical-Section Problem, Synchronization Hardware, Semaphores, Classical Problems of Synchronization, Critical Regions, Monitors, Atomic Transactions. **Memory Management**: Background, Logical versus Physical Address space, Swapping, Contiguous allocation, Fragmentation, Paging, Segmentation, Segmentation with Paging. Virtual Memory: Demand Paging, Page Replacement, Page-replacement Algorithms, Performance of Demand Paging, Allocation of Frames, Thrashing.
Deadlocks: System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock

[10 Hrs]

UNIT–III

Device Management: Techniques for Device Management, Dedicated Devices, Shared Devices, Virtual Devices
Secondary-Storage Structure: Disk Structure, Disk Scheduling, Disk Management, Swap-Space Management, Disk Reliability, Stable-Storage Implementation
Information Management: Introduction, Simple File System, General Model of a File System, Symbolic File System, Basic File System, Access Control Verification, Logical File System, Physical File System

[10 Hrs]

UNIT–IV

File-System Interface: File Concept, Access Methods, Directory Structure, Protection, Consistency Semantics File-System Implementation: File-System Structure, Allocation Methods, Free-Space Management, Directory Implementation, Efficiency and Performance, Recovery.

[10 Hrs]

Note: Case Study of Linux & Windows along with O.S concepts to be taught

TEXT BOOKS:

1. Silbersachatz and Galvin, "Operating System Concepts", Pearson, 8th Ed., 2008.
2. R. C. Joshi, "Operating Systems", Wiley Dreamtech, 2005.

REFERENCE BOOKS:

1. Tannenbaum, "Operating Systems", PHI, 4th Edition, 2006.
2. E. Madnick, J. Donovan, "Operating Systems", Tata McGrawhill

Paper Code: BIT 210	**L**	**P**	**C**
Paper Title: Object Oriented Software Engineering	**4**	**0**	**4**

Introduction

This course introduces students to the different software development lifecycle (SDLC) phases used in developing, delivering, and maintaining software products. Students will also acquire basic software development skills and understand common terminology used in the software engineering profession. The aim of the course is to provide an understanding of the working knowledge of the techniques for estimation, design, testing and quality management of software development projects.

Course Objectives:

- To introduce the concepts of database management systems
- To introduce the concepts of software engineering, software processes and its models.
- To understand the software requirements analysis, transform the requirements using DFD, create software requirement specification document and validation of the software requirements.
- To understand fundamentals of software design, software quality and software maintenance.
- To understand the project planning process, size and cost estimation techniques further development of software.

Pre-requisites: Basic knowledge of Programming Languages.

Course Outcomes: After completion of the course, the students will be able:

CO1: Understand the concepts of Object Software engineering, Software process and its models.
CO2: Evaluate the Software Requirements, interpret and structure the requirements in the Software Required document.
CO3: Apply appropriate software architectures and patterns to carry out high level design of a system and be able to critically compare alternative choices, evaluate the quality and maintenance of the software through software testing.
CO4: Create the software project plan for size and cost estimation including risk analysis.

Pedagogy:
This course is structured around continuous progress. It will include a combination of lectures, and group activities focused on experiential learning, in-class discussions, regular assessments and case studies. The topics will be presented to students using real-world scenarios and problem-solving activities.

UNIT I

Introduction: Software Crisis, Software Processes, **Software Process Life cycle models:** Waterfall Model, Prototyping Model, Iterative Enhancement Model, Evolutionary Development and Spiral Model

Object Oriented Methodology: Object Oriented Concepts, Object Oriented Process and Models

[10 Hrs]

UNIT II

Software Requirements Analysis & Specifications: Requirement Elicitation Concepts, Managing Requirements Elicitation, Software Requirement Specification (SRS) Standards.

Analysis and Modeling: Analysis concepts, Data Flow Diagrams, Analysis Activities, Unified Modeling Language (Use cases, Class Diagram, Interaction diagrams, Activity diagram, object models) Modeling Interactions among Objects.

[10 Hrs]

UNIT III

Software System Design: Design standards, design issues: cohesion and coupling, object oriented design, Detailed class diagram, Reuse Concepts-Solution Objects, Inheritance and Design Patterns, Reuse Activities- Selecting Design Patterns and Components, Managing Reuse **Software Project Planning:** Function Point Model, Cost estimation, COCOMO model, Putnam Resource Allocation Model **Software metrics:** Function Count, Data Structure Metrics, Information Flow Metric, and Object oriented metrics.

[10 Hrs]

UNIT IV

Software Testing: Introduction to Functional testing and Structural Testing, Unit testing, integration and system testing, Testing Tools & Standards.

Software Maintenance: Management of Maintenance, Maintenance Process, Maintenance Models, Reverse Engineering, Software Re-engineering, Configuration Management, Documentation.

TEXT BOOKS:

1. Bruegge and Dutoit, "Object-Oriented Software Engineering- Using UML, Patterns and Hill Int. Ed., 2010
2. R. S. Pressman, "Software Engineering – A practitioner's approach", 7th ed,McGraw Hill Int. Ed., 2010.
3. K.K. Aggarwal & Yogesh Singh, "Software Engineering", New Age International, 3rd Edition, 2007
4. G. Booch, J Rumbaugh, I Jacobson, "The Unified Modeling Language User Guide" 11th Ed., Pearson Education, 2003.

REFERENCE BOOKS:

1. Timothy C. Lethbridge, Robert Laganiere "Object oriented Software Engineering: Practical Software development using UML and Java" 2nd Ed. McGraw Hill, 2005.
2. Jacobson, "Object-Oriented Software Engineering: A Use Case Driven Approach", Pearson 1991

Paper Code:BCS 301	**L**	**P**	**C**
Paper Title: Theory of Computation	**4**	**0**	**4**

Introduction: The study of automata and the theory of computation deal with the concepts of working of automatic machines and processing of input formal language data. This subject provides an important background material to students involved in understanding the basic functionalities of automata theory.

Course Objectives:

- Introduce concepts in Automata theory and theory of computation
- Identify different formal language classes and their relationships
- Design grammars and recognizers for different formal languages

Pre-requisite: Basic concepts of mathematics

Course Outcome: Upon successful completion of this course, students will be able to:

CO1: Understand the basics of automata and its fundamentals.

CO2: Understand theory of computation and concepts of formal languages

CO3: Design grammars and recognizers for different formal languages

CO4: Analyze the finite automata and regular expressions for accepting the language.

Pedagogy:
This course is structured around continuous progress. It will include a combination of lectures, and group activities focused on experiential learning, in-class discussions, regular assessments and case studies. The topics will be presented to students using real-world scenarios and problem-solving activities.

UNIT I

Introduction Alphabets, Strings And Languages, Automata And Grammars, Deterministic Finite Automata (Dfa)- Formal Definition, Simplified Notation: State Transition Graph, Transition Table, Language Of Dfa, Nondeterministic Finite Automata (Nfa), Nfa With Epsilon Transition, Language Of Nfa, Equivalence Of Nfa And Dfa, Minimization Of Finite Automata, Distinguishing One String From Other, Myhill-Nerode Theorem

[10 Hrs]

UNIT II

Regular Expression (RE) : Definition, Operators Of Regular Expression And There Are Precedence, Algebraic Laws For Regular Expression, Kleen's Theorem, Regular Expression To FA, DFA To Regular Expression, Arden Theorem, Non Regular Languages, Pumping Lemma For Regular Languages, Application Of Pumping Lemma, Closure Properties Of Regular Languages, Decision Properties Of Regular Languages, FA With Output : Moore And Mealy Machine, Equivalence Of Moore And Mealy Machine, Applications And Limitations Of FA.

[10 Hrs]

UNIT III

Context Free Grammar (CFG) And Context Free Languages (CFL) , Examples, Derivation Trees, Ambiguity In Grammar, Inherent Ambiguity, Ambiguous To Unambiguous CFG, Useless Symbols, Simplification Of Cfgs, Normal Forms For Cfgs: CNF And GNF , Closure Properties Of Cfls, Decision Properties Of Cfls: Emptiness, Finiteness And And Membership, Pumping Lemma For Cfls

Push Down Automata (PDA) Description And Definition, Instantaneous Description, Language Of PDA, Acceptance By Final State, Acceptance By Empty Stack, Deterministic PDA, Equivalence Of PDA And CFG, CFG To PDA And PDA To CFG, Two Stack PDA

[10 Hrs]

UNIT IV

Turing Machines (TM): Basic Model, Definition And Representation, Instantaneous Description, Language Acceptance By TM, Variants Of Turing Machine, TM As Computer Off Integer Functions, Universal TM, Church's Thesis, Recursive And Recursively Enumerable Languages, Halting Problem, Introduction To Undecidability, Undecidable Problems About Tms Applications Of TOC, Introduction To Recursive Function Theory.

[10 Hrs]

Text Books:

1. Peter Linz, "An Introduction To Formal Language And Automata", Fourth Tradition 2010
2. Hopcroft Ullman, " Introduction To Automata Theory Languages And Computation, Pearson Education 2nd Edition Copyright Year 2021

Reference Books:

1. K. L.P Mishra And N.Chandrasekaran," Theory Of Computer Science Automata Languages And Computation 3rd Edition Year 2008
2. Martin J.C, " Introduction To Languages And Theory Of Computation 4th Edition Year 2011.
3. Papadimitrou,C And Lewis C.Ll, " Elements Of The Theory Of Computation Phi Second Edition Year 2008

Paper Code BIT 303 **L P C**

Paper Title: Computer Graphics and Multimedia Technologies **4 0 4**

Introduction: The subject Computer Graphics introduces basic concepts of graphics, output primitives, transformations, projections, curve and surface generation methods and shading algorithms.

Course Objective:

- To introduce the basic concepts of computer graphics
- To introduce the concepts of 2D/3D transformations
- To introduce the concepts of curve generation and hidden surface detection.

Pre-requisite: Basic mathematics.

Course Outcome: Upon successful completion of this course, students will be able to:

CO1: Understand basic concepts of computer graphics and its applications.
CO2: Use the 2D/3D transformation and projection concepts in various projects
CO3: Understand concepts of curve generation and hidden surface detection.
CO4: Develop various applications of computer graphics

Pedagogy:
This course is structured around continuous progress. It will include a combination of lectures, and group activities focused on experiential learning, in-class discussions, regular assessments and case studies. The topics will be presented to students using real-world scenarios and problem-solving activities.

UNIT I

Scan Conversion Algorithms: Scan Converting Lines (DDA, Bresenham), Scan Converting Circles (Mid-point, Bresenham), Scan Converting Ellipses (Midpoint). Clipping: Two-Dimensional Clipping, Sutherland-Cohen Subdivision Line-Clipping Algorithm 2D- Transformation: Representation of Points, Transformations and Matrix, Transformation of Straight Line, 2-D - Rotation, Reflection, Scaling, Combined Transformations, Translation and Homogeneous Coordinates, Translation, Rotation about an Arbitrary Point, Reflection through an Arbitrary Line, window-to-viewport transformation

[10 Hrs]

UNIT II

3D-Transformation: Representation of Points, 3D- Scaling, 3D- Shearing, 3D- Rotation, Three Dimensional Translation, 3D- Reflection, Multiple Transformations, Rotation about an Axis Parallel to a Coordinate Axis, Rotation about an Arbitrary Axis in Space. The Dimensional Perspective Geometry: Geometric Projection, Orthographic Projections, Oblique Projections, Perspective Transformations, Single-Point Perspective Transformation, Two-Point Perspective Transformation, Three-Point Perspective Transformation. Solid Modeling: Representing Solids, Regularized Boolean Set Operation primitive Instancing Sweep Representations, Boundary Representations, Spatial Partitioning Representations, Constructive Solid Geometry, Comparison of Representations.

[10 Hrs]

UNIT III

Representing Curves & Surfaces: Polygon meshes, parametric, Cubic Curves, geometric and parametric continuities, Hermite, Bezier (4-point, 5-point, general), B-Spline, Quadric Surface Illumination and Shading: Modeling light intensities, ambient light, diffused light, specular reflection, attenuation factor, Reflection vector, Shading Models: constant shading, flat shading, gouraud shading, phong shading. Hidden-Surface Removal: Hidden Surfaces and Lines, Back-Face Detection, A-buffer, ZBuffers Algorithm, Scan-line Algorithm, The Painter's Algorithm, Area subdivision Introduction to Multimedia: Multimedia, Multimedia Terms, Introduction to making multimedia – The Stages of project, the requirements to make good multimedia, Multimedia Applications. **[10 Hrs]**

UNIT IV

IV Multimedia – making it work – Multimedia Hardware, Software and Authoring Tools, Graphics File Formats: TIFF, MIDI, JPEG, MPEG, RTF. Multimedia building blocks – Text, Sound, Images, Animation and Video, Digitization of Audio and Video objects, Data Compression: Different Compression algorithms concern to text, audio, video and images etc.

[10 Hrs]

Text Books

1. Steve Marschner, Peter Shirley, Fundamentals of Computer Graphics, CRC Press, 4th Ed. (2015)
2. D.Hearn & Baker: Computer Graphics, Prentice Hall of India
3. Foley, Van Dam, Feiner, Hughes, "Computer Graphics Principles & Practice"
4. Tay Vaughan, "Multimedia: Making it Work", TMH

Reference Books

1. K. Andleigh and K. Thakkar, "Multimedia System Design", PHI, PTR
2. Rogers & Adams, "Mathematical Elements for Computer Graphics", McGraw Hill

Paper Code: BIT 305	**L**	**P**	**C**
Paper Title: Requirement and Estimation Techniques	**4**	**0**	**4**

Introduction

A requirement gathering is the cornerstone of any software development project. In this course, students will gain the knowledge and skills needed to capture software requirements using clearly defined processes. They will learn to specify user and system requirements, match the process to the size of the software project, and apply quality and consistency tests to the requirements model. It will equip the students with skills and knowledge in developing, leading, designing, testing or managing a requirements initiative for a software system.

Course objective:

- To introduce the essential aspects of software requirements; elicitation technique, requirements analysis; software quality attributes
- Understand the software requirements management principles and practices.
- Learn the fundamentals of Software estimation components, size estimation. Effort, schedule and Cost Estimation models
- Demonstrate the techniques learned for requirements Management and estimation requirements Management for size estimation and cost estimation through case studies

Course Outcome: At the end of the course, the students will be able to:

CO1: Understand and demonstrate essential software requirements

CO2: Describe requirement analysis process of software from engineering perspective

CO3: Perform cost estimation using estimation models like Function Point Analysis and COCOMO.

CO4: Apply different Techniques for software management and estimation.

Pedagogy:

This course is structured around continuous progress. It will include a combination of lectures, and group activities focused on experiential learning, in-class discussions, regular assessments and case studies. The topics will be presented to students using real-world scenarios and problem-solving activities

UNIT I

Software Requirements**:** Why has Requirement Engineering Become so important Industrila challenges in Requirements, Requirement Engineering and Artifact Modelling, Eliciting Requirements, Interview, IBIS, CORE, FODA, SSM, Model Driven Requirements Engineering, MDRE Process, Elicitation and Analysis Model Heuristics, Determining Model Completeness, Quality Attribute Requirements.

[10 Hrs]

UNIT II

Requirement Management, Change Management, Requirements Management Activities, Traceability, Creation of Requirements Management, Requirement-Driven System Testing, Process, Software Measurement, Why Measurement, Measurement Foundations, Making Measurement a success, Simple effective Measurement Process, Planning the Measurement Process, Planning with Measurement Frameworks, ISO 15939, CMMI, GQM Approach, CAME Approach..

[10 Hrs]

UNIT III

Software Estimation techniques and Estimate Planning, Executing the estimate, Software sizing, Planning and controlling the project via the Estimate, SLOC, Logical SLOC counting Details, Function Point Sizing, International Function Point User Group Counting Standards, Basic Process, SEER-Function Based Sizing, COSMIC Full Function Point Approach

[10 Hrs]

UNIT IV

Software Cost Estimation Methods**:** heuristic approach, parametric approach COCOMO, COCOMO II, strategy and rationale, Development Effort Estimates, Software Economies and Diseconomies of Scale, Cost Factors, Application Composition Model, Early Design Model, Post-Architecture Model, Case study, Software Cost-Estimating Research Issues.

[10 Hrs]

Text Books:

1. Brian Berenbach, Daniel Paulish, Juergen Kazmeier, Arnold Rudorfer : Software & Systems Requirements Engineering: In Practice Hardcover– March 26, 2009.
2. M. A. Parthasarathy: Practical Software Estimation: Function Point Methods for Insourced and Outsourced Projects, Addison-Wesley Professional,2007

Reference Books:

1. Daniel D. Galorath and Michael W. Evans, Software Sizing, Estimation and Risk Management, Auerbach Publications, 2006
2. Christof Ebert and Reiner Dumke, Software Measurement: Establish, Extract, Evaluate, Execute, Springer, 2007

Paper Code: BIT 307	L	P	C
Paper Title: Data Communications & Computer Networks	**4**	**0**	**4**

Introduction:

Data communications refers to the transmission of this digital data between two or more computers and a computer network or data network is a telecommunications network that allows computers to exchange data. The physical connection between networked computing devices is established using either cable media or wireless media. The best-known computer network is the Internet.

Course Objectives:

- The students should understand the layers of networking devices.
- They should be familiar with a few networking protocols.
- They should study the different types of networks and topologies of networks.

Pre-requisite: Data Structures and Algorithms.

Course Outcomes: Upon successful completion of this course, students will be able to:

CO1: Describe the fundamental concepts and layered architecture of computer networking.
CO2: Explain the basic concepts of link layer properties to detect error and develop the solution for error control and flow control. Design, calculate, and apply subnet masks and addresses to fulfill networking requirements. Also, compare various routing protocols.
CO3: Comprehend the duties of transport layer and congestion control techniques.
CO4: Illustrate the features and operations of various application layer protocols such as DNS, HTTP, FTP, e-mail protocols and other applications; and focus on network security issues to secure communication towards society.

Pedagogy:

The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Students would be encouraged to develop an understanding of the existing real life cyber security issues and how they are solved. Emphasis would be given on assignments where students will be given numerical/ programming assignments based on topics studied in previous lectures. Course will have a blend of theory and practice for the benefit of students. Use of ICT, web-based sources as well as blackboard teaching will be adopted.

UNIT – I

Introduction: Goals and Applications of Networks, Layering Concept, OSI Reference Model vs TCP/IP Protocol Suite, Networks Topology.

Physical Layer: Signals, Digital Transmission – Analog to Digital & Digital to Digital, Analog Transmission – Digital to Analog & Analog to Analog, Multiplexing – FDM & TDM, Media – Guided and Unguided, Switching – Packet based & Circuit based. Hub & Repeater. Sampling Theorem (Nyquist-Shannon Theorem)**Network Traffic Capturing:** Wireshark (windows) and tcpdump (linux).

[10Hrs]

UNIT – II

Data Link Layer: Addressing; Error Detection & Correction – General concepts, Checksum & CRC; Medium Access – Aloha, CSMA, CSMA/CD & CA; Protocols – Ethernet, ARP & RARP; Switch – Learning & Filtering Mechanism, Wireless Access (Bluetooth & Wi-Fi)

Network Layer: IP Addressing & Subnets; Basic Routing (or Forwarding) Mechanism; IPv4 frame format and functions; Routing protocols – RIP, OSPF & BGP and algorithms – Distance Vector & Link State.

Linux Network Commands: arp, route, ifconfig, netstat, traceroute, ping.

[10Hrs]

UNIT – III

Transport Layer: Port Addresses; Protocols - Simple, Stop n Wait, Go Back N & Selective Repeat; UDP – Services & Applications; TCP – header format, connection setup & termination, state transition diagram, flow control, error control, congestion control & timers.

[10Hrs]

UNIT – IV

Application Layer: Web & HTTP, FTP, Email, Telnet, SSH, DNS.

Advanced Protocols: SNMP, RTP, SIP, BitTorrent, Wireshark (Case Studies) **[10Hrs]**

TEXT BOOKS:

1. Forouzan, "Data Communication and Networking", TMH, 5th Edition, 2013.
2. A.S. Tanenbaum, "Computer Networks", PHI, 4th Edition, 2002.
3. W. Stallings, "Data and Computer Communication", Macmillan Press, 2013.
4. Comer, "Computer Networks and Internet", PHI, 2008
5. Comer, "Internetworking with TCP/IP", PHI, 2008.

REFERENCE BOOKS:

1. W. Stallings, "Data and Computer Communication", McMillan, 2010
2. J. Martin, "Computer Network and Distributed Data Processing", PHI,2008.
3. W. Stallings, "Local Networks", McMillan, 2013.
4. M.Schwertz, "Computer Communication Network Design and Analysis", PHI, 1977.
5. S. Keshav, "An Engineering Approach to Computer Networking, Pearson", 2001.

	L	P	C
Paper Code: BIT 309			
Paper Title: Data Warehouse and Data Mining	**4**	**0**	**4**

Course Objective:

- Be familiar with mathematical foundations of data mining tools. Understand and implement classical models and algorithms in data warehouses and data mining.
- Characterize the kinds of patterns that can be discovered by association rule mining. classification and clustering.
- Master data mining techniques in various applications like social, scientific and environmental context.
- Develop skill in selecting the appropriate data mining algorithm for solving practical problems.

Course Outcome: At the end of the course, the students will be able to:

CO1: Understand the functionality of the various data mining and data warehousing components.
CO2: Appreciate the strengths and limitations of various data mining and data warehousing models.
CO3: Explain the analyzing techniques of various data.
CO4: Describe different methodologies used in data mining and data warehousing.
CO5: Compare different approaches of data warehousing and data mining with various technologies.

UNIT-I

The Compelling Need for data warehousing. Data warehouse - The building Blocks Defining the business requirements, Requirements definition (scope and content), Principles of dimensional modeling, Dimensional Modeling.

OLAP in the Data Warehouse: Demand for Online analytical processing, need for multidimensional analysis, OLAP characteristics, features and functions, dimensional analysis, hyper cubes, Drill-down and roll-up, slice-and-dice, rotation.

OLAP models: MOLAP model, ROLAP model, HOLAP model, ROLAP versus MOLAP OLAP implementation considerations. **[10 Hrs]**

UNIT-II

Data Mining Basics: Data Mining Definition. The knowledge discovery process, OLAP versus data mining, data mining and the data warehouse, Process of data mining. Data Mining Applications. Benefits of data mining.

Associations and Correlations: Association rule mining, Apriori algorithm, improving efficiency, kinds of association rules, multilevel, multi-dimensional. **[10 Hrs]**

UNIT-III

Classification and Regression: types of classification algorithm, Bayesian, rule based, decision tree, KNN.

Cluster analysis: Overview of grid based, model based. density based, partitioning based, hierarchical based clustering methods. **[10 Hrs]**

UNIT-IV

Major Data Mining Techniques: Cluster detection, K-means algorithm, link analysis, neural networks, genetic algorithms, fuzzy logic. Web Mining, Sentiment Analysis, Opinion Mining **[10 Hrs]**

TEXT BOOKS:

1. M.H. Dunham, "Data Mining Introductory and Advanced Topics", Pearson Education,2008

2. Jawei Han, Micheline Kamber, and Jian Pei, Data Mining: Concepts and Techniques, 3rd Edition, Morgan Kaufmann, 2011.

REFERENCE BOOKS:

1. Pieter Adriaans, Dolf Zantinge, "Data Mining", Pearson Education Asia, 2001

2. Ralph Kimball. "The Data Warehouse Lifecycle toolkit", John Wiley, 2 edition, 2007

3. M Berry and G. Linoff, "Mastering Data Mining", John Wiley, 3 edition, 2011

4. Paul Raj Poonia, "Fundamentals of Data Warehousing", John Wiley & Sons, 2004

Paper Code: BAS 311	**L**	**P**	**C**
Paper Title: Human Values and Professional Ethics	**3**	**0**	**3**

Introduction:

Values and Ethics are very relevant in today's environment of conflicts and stress in every profession, with obligations to be met by one person in many directions. A formal study will certainly improve one's ability and judgment and refine one's behavior, decisions, and actions in performing the duty to the family, organization, and to the society.

Course Objectives:

To facilitate the development of a Holistic perspective among students towards life, profession and happiness, based on a correct understanding of the Human reality and the rest of Existence. Such a holistic perspective forms the basis of Value based living in a natural way. To inculcate Ethics and Human Values into the young minds and develop moral responsibility and mould them as best professional which will create ethical vision and achieve harmony in life.

Pre-requisite: High school level moral studies

Course Outcomes: After completion of the course, the students should be able to:

CO1: Develop the capability of shaping themselves into outstanding personalities, through a value-based life.
CO2: Students turn themselves into champions of their lives.
CO3: Students take things positively, convert everything into happiness and contribute for the happiness of others.
CO4: Students become potential sources for contributing to the development of the society around them and institutions / organizations they work in.
CO5: Students shape themselves into valuable professionals, follow professional ethics and are able to solve their ethical dilemmas.

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, web-based resources as well as flipped class room teaching will be adopted.

Paper Code: BAS 311 | L | P | C
Paper Title: Human Values and Professional Ethics | 3 | 0 | 3

INSTRUCTIONS TO PAPER SETTERS: **MAXIMUM MARKS: 60**

1. **Question No. 1 should be compulsory and cover the entire syllabus. This question should have objective or short answer type questions. It should be of 20 marks.**
2. **Apart from Question No. 1, rest of the paper shall consist of four units as per the syllabus. Every unit should have two questions. However, student may be asked to attempt only 1 question from each unit. Each question should be 10 marks**

UNIT–I

Human Values: Morals, Values and Ethics, Integrity, Work Ethic, Respect for Others, Living Peacefully, Caring, Sharing, Honesty, Valuing Time, Co-operation, Commitment, Empathy, Self-Confidence, Character, Spirituality. **Indian values (on the conceptual framework of Vedas):** Purusharth, Niskama karma, Religion and Human Values, Towards a World Religion, Ethical Living and Harmony in Life.

[8 Hrs]

UNIT–II

Ethics and Engineering Profession: Profession and Professionalism, Ethical Theories: Kohlberg's Theory, Gilligan's Theory, Feminist Consequentialism, Moral Dilemmas, Types of Enquiry, Uses of Ethical Theories, Engineering Profession, **Engineering Professionals:** Training, Skill Set, Life Skills. **Engineering Ethics:** Making Senses and Issues, Ethical Obligations of Engineers, Ethical Codes for Engineers. **[7 Hrs]**

UNIT-III

Engineering as a Social Experimentation, Safety Responsibility and Rights: Engineering as experimentation, Engineers as responsible Experimenters, Concept of Safety and Risk, Engineer's Responsibility for Safety, **Risk :** Benefit Analysis, **Case Studies:** The challenger case study, The Three Mile Island, Fukushima Nuclear Disaster, Bhopal Gas Tragedy, Disaster Management, Professional Rights, Employee Rights, Intellectual Property Rights (IPRs), Human Rights and Human Responsibilities. Major Ethical Issues.

[8 Hrs]

UNIT–IV

Ethics and Global Issues: Ethics in Global Scenario, Multinational corporations, Environmental ethics, computer ethics, Business Ethics, Corporate Social responsibility, Weapons Development, Research Ethics.

[7 Hrs]

TEXT BOOKS:

1. Govindarajan M., Natarajan S., Senthil Kumar V. S., "Engineering Ethics", Prentice Hall, New Delhi, 2004.
2. Subramaniam R., "Professional Ethics", Oxford University Press, New Delhi, 2013.

3. Mike Martin and Roland Schinzinger, "Ethics in engineering", McGraw-Hill, New York 1996.
4. RR Gaur, R Sangal, GP Bagaria, "A Foundation Course in Human values and Professional Ethics", Excel Books Pvt. Ltd, New Delhi 2009.
5. A.N.Tripathi, "Human Values", New Age International Publishers, New Delhi, 2nd Edition, 2004.

REFERENCE BOOKS:

1. B.P. Banerjee, "Foundation of Ethics and Management", Excel Books, 2005.
2. Fleddermann, Charles D., "Engineering Ethics", Pearson Education. 2004.
3. Harris, Charles E., Protchard, Michael S. And Rabins, Michael, J., Wadsworth, "Engineering Ethics- Concepts and Cases", Thompson Learning, 2000
4. Boatright, John R., "Ethics and the Conduct of Business", Pearson Education, New Delhi, 2003.
5. Swami Ranganathananda, "Universal Message of the Bhagavad Gita: An exposition of the Gita in the light of modern thought and modern needs", Vol. I – III, Advaita Ashrama (Publication Department), Kolkata. 2000.
6. Peter Singer, "Practical Ethics", Oxford University Press, 1993.

Paper Code: BIT 302	**L**	**P**	**C**
Paper Title: Web and Mobile Technologies	**4**	**0**	**4**

Introduction: Web and Mobile Technologies refers a technology that allows transmission of data, voice and video via a computer or any other wireless enabled device. It is free from having a connection with a fixed physical link. It facilitates the users to move from one physical location to another during communication.

Course Objective:

- To introduce the basic concepts and principles in mobile computing. This includes major techniques involved, and networks as well as systems issues for the design and implementation of mobile computing systems and applications.
- To understand the basic concepts of mobile communication and computing.
- To understand telecommunication systems and gain knowledge about different mobile platforms and application development.

Course Outcomes: Upon successful completion of this course, students will be able to:

CO1: Understand the basic terminology of web and to implement CSS, HTML and XML in web development.
CO2: Understand the concepts and use of JavaScript and PHP in web development.
CO3: Implement DHTML and to understand the web services and the concepts of mobile technologies.
CO4: Understand mobile ad-hoc network (MANET), mobile commerce and various mobile operating systems

Pre-requisite: Computer Networks

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, webbased resources as well as flipped classroom teaching will be adopted.

Paper Code: BIT 302	L	P	C
Paper Title: Web and Mobile Technologies	4	0	4

INSTRUCTIONS TO PAPER SETTERS: MAXIMUM MARKS: 60

1. Question No. 1 should be compulsory and cover the entire syllabus. This question should have objective or short answer type questions. It should be of 20 marks.
2. Apart from question no. 1, rest of the paper shall consist of four units as per the syllabus. Every unit should have two questions. However, student may be asked to attempt only 1 question from each unit. Each question should be of 10 marks.

UNIT I

Introduction to the Internet, the World Wide Web: the idea of Hypertext and Hypermedia, how the web works, browser working, MIME types, Plug-ins and helper applications.

Introduction to HTML 5.0: basic tags of HTML, tables, frames, forms, separating style from structure with Style Sheets, inline style and internal style specifications within HTML, external Linked style specification using CSS.

Introduction to XML: XML vs. HTML, uses of XML, simple XML, XML key components, DTD and schemas, using XML with application.

[10 Hrs]

UNIT II

Client Side Programming: introduction to JavaScript, JavaScript programming, variables, functions, conditions, loops, JavaScript object model, event handling, forms handling, Cookies, hidden fields, images, applications.

Server Side Programming: introduction to PHP, basics of PHP, PHP file handling, PHP file upload, PHP sessions, PHP cookies, PHP error handling, PHP MySQL introduction, PHP MySQL insert into, PHP MySQL select, PHP MySQL, where clause, PHP MySQL update, PHP MySQL delete.

[10 Hrs]

UNIT III

DHTML: combining HTML, CSS and JavaScript, DHTML, Document Object Model (DOM).

Web Services: components and working of Web Services, Web Services architecture, introduction to Service Oriented architecture, SOAP, WSDL, UDDI, AJAX, overview of Mobile Computing, overview of Cloud Computing.

Mobile Technologies: introduction, applications of Mobile Technology, concepts of Multiplexing and Modulation, Spread Spectrum, SDMA, TDMA, FDMA, CDMA, GSM, Bluetooth, EDGE, UMTS, 4G Networking.

[10 Hrs]

UNIT IV

Mobile Operating Systems: introduction to various Mobile OS, features of a Mobile OS, history of Mobile OS, Mobile OS structure, Mobile OS platforms, future of Mobile OS, case study of Android.

Mobile Commerce: history of Mobile Commerce, services and applications, Mobile marketing and advertising, Mobile Commerce life-cycle, Mobile Entertainment Services, SyncML.

MANET: types of MANET's, security in MANET, Smart Cards, future of Mobile Technologies.

[10 Hrs]

TEXT BOOKS:

1. Deitel, "Internet and World Wide Web, How to Program", PHI, 2008.
2. Ivan Bay Ross, "HTML, DHTML, JavaScript, Perl CGI ",PBP, 3rd Edition,2005.
3. Ethan Cerami, "Web Services", O'Reilly Media, 2002.
4. Ron Schneider man, "The Mobile Technology Question and Answer Book: A Survival Guide for Business Managers", 2002.

REFERENCE BOOKS:

1. Rick Dranell, "HTML4 unleashed", Techmedia Publication, 2000.
2. Jochen Schiller, "Mobile Communications", Addison-Wesley, 2004.

Paper Code: BCS 304	**L**	**P**	**C**
Paper Title: Compiler Design	**4**	**0**	**4**

Introduction: This course provides the complete description about inner working of a compiler. This course focuses mainly on the design of compilers and optimization techniques. It also includes the design of Compiler writing tools. This course also aims to convey the language specifications, use of regular expressions and context free grammars behind the design of compiler.

Course Objectives:

- Introduce major concepts of language translation and compiler design.
- Impart the knowledge of practical skills necessary for constructing a compiler.

Prerequisite: Basic programming skills

Course Outcomes: The students will be able to

CO1: Explain the compiler architecture and different phases of compilation with compile time error handling.
CO2: Compare top down with bottom-up parsers, and develop appropriate parser to produce parse tree representation of the input
CO3: Illustrate language tokens using regular expressions, context free grammar and finite automata and design lexical analyzer for a language.
CO4: Design a compiler for a small subset of C language.

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, web-based resources as well as flipped class room teaching will be adopted.

UNIT 1

Introduction to compilers – Analysis of the source program, Phases of a compiler, grouping of phases, compiler writing tools– bootstrapping. Case study: MiniC (A small subset of C language)

Lexical Analysis-The role of Lexical Analyzer, Input Buffering, Specification of Tokens using Regular Expressions, Review of Finite Automata, Recognition of Tokens, Case study: Lexical Analysis for MiniC, Syntax Analysis**:** Review of Context-Free Grammars – Derivation trees and Parse Trees, Ambiguity.

[10Hrs]

UNIT 2

Top-Down Parsing: Recursive Descent parsing, Predictive parsing, LL (1) Grammars**.** Bottom-Up Parsing: Shift Reduce parsing – Operator precedence parsing (Concepts only). LR parsing – Constructing SLR parsing tables, Constructing Canonical LR parsing tables and Constructing LALR parsing tables. Case study: Syntax analysis for MiniC. **[10 Hrs]**

UNIT 3

Syntax directed translation: Syntax directed definitions, Bottom- up evaluation of S- attributed definitions, L- attributed definitions, Top-down translation, Bottom-up evaluation of inherited attributes. Type Checking: Type systems, Specification of a simple type checker. Run-Time Environments: Source Language issues, Storage organization, Storage allocation strategies. **[10 Hrs]**

UNIT 4

Intermediate Code Generation (ICG): Intermediate languages – Graphical representations, Three Address code, Quadruples, Triples. Assignment statements, Boolean expressions. Code Optimization: Principal sources of optimization, Optimization of Basic blocks, Code generation: Issues in the design of a code generator. A simple code generator. Case study: MiniC Code Generator for the MiniC Architecture

[**10Hrs]**

TEXT BOOKS

1. A. Monica, S. Lam, R. Sethi and D. Ullman, "Compilers – Principles Techniques and Tools", Pearson Education India; 2nd edition, 2013/ Latest Edition .
2. K. C. Louden, "Compiler Construction – Principles and Practice", Cengage Learning Indian Edition 2006/ Latest Edition.

REFERENCE BOOKS

1. A. I. Hollub, "Compiler Design in C", Pearson Education India; 1st edition, 2015/ Latest Edition..
2. A.W. Appel, M. Ginsburg, "Modern Compiler Implementation in C", Cambridge University Press, 2004/ Latest Edition

Paper Code: BCS 308	L	P	C
Paper Title: Cloud Computing	4	0	4

Introduction: Cloud computing is a scalable service provider platform that provides on-demand and pay per use computing service for various types of shared pool of resources such as memory, servers, storage, networking, software, database, applications designing etc., with the help of the internet. This course will introduce various aspects of cloud computing including fundamentals of cloud computing, load balancing techniques, security challenges, case studies and industrial applications of cloud computing. This will help students to use and explore the cloud computing platforms.

Course Objectives:

- To learn the use of various cloud computing services and cloud deployment models.
- Understand the concept of virtualization in cloud computing.
- To apply the concepts of cloud computing for designing, evaluating, simulating and comparing various applications in a cloud computing environment.
- To gain the confidence in resource management and load balancing algorithms in a cloud computing environment.
- To gain the confidence of security attacks and their provisions at various levels of cloud computing.

Prerequisite: Basic understanding of Operating System, Internet, Parallel and Distributed Computing.

Course Outcomes:

CO1: To articulate key concepts of cloud computing and computing techniques, strength and limitations of cloud computing with possible application domains.
CO2: To identify the architecture and infrastructure of cloud computing including SaaS, PaaS, IaaS, public cloud, private cloud and hybrid cloud.
CO3: To interpret various data, scalability and cloud services to acquire efficient database for cloud storage.
CO4: To explain the core issues of cloud computing such as security, privacy and interoperability and deal with controlling mechanism for accessing sage cloud service.

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, web-based resources as well as flipped class room teaching will be adopted.

UNIT- I

Cloud Computing Fundamentals: Introduction of cloud computing, History of cloud computing, Trends in computing, Grid computing, Cluster computing, Distributed computing, Utility computing, Fog computing, NIST definition and characteristics of cloud computing, Cloud as green and smart, Cloud as IaaS, PaaS, SaaS, BPaaS and HaaS, SPI framework, SPI vs. traditional IT Model, Cloud deployment models, Benefits and challenges**.** **[10 Hrs]**

UNIT- II

Virtualization and Cloud Architecture: Virtualization concept, Resource virtualization, Server virtualization, Storage virtualization and Network virtualization, Storage Network Design: Architecture of storage, Analysis and planning, Storage models, Cloud optimized storage, Virtual Box and Microsoft Hyper-V**.** **[10 Hrs]**

UNIT- III

Cloud Security: Web services, Web 2.0, Web OS, Security challenges and preventive measures: Infrastructure layer, Network layer and Application layer of cloud computing architecture, Security models in cloud, Resource management in cloud computing, Static and dynamic load balancing in cloud computing, Identity access management and Trust in cloud computing, Thin client**.** **[10 Hrs]**

UNIT- IV

Cloud providers and case studies: Amazon EC2, Amazon EC service level agreement and recent developments, GoGrid, Salesforce.com, Force.com, Google App Engine, Rackspace, Government of India Cloud, IBM cloud, Eucalyptus cloud, Analysis of Case Studies when deciding to adopt cloud computing architecture **[10 Hrs]**

TEXT BOOKS:

1. B. Sosinsky, "Cloud Computing Bible", 1st Edition, Wiley-India, 2011/ Latest Edition.

2. R. Buyya, C. Vecchiola, and S. T. Selvi, "Mastering cloud computing: foundations and applications programming", 1st Edition, Newnes, 2013/ Latest Edition.

REFERENCE BOOKS:

1. A. Shawish and M. Salama, "Cloud computing: paradigms and technologies." In Inter-cooperative collective intelligence: Techniques and applications, Springer, 2014/ Latest Edition.

Paper Code: BIT 310	**L**	**P**	**C**
Paper Title: Artificial Intelligence	**4**	**0**	**4**

Introduction: This course is an introduction to the basic Knowledge representation, problem solving and learning methods of artificial intelligence. After learning this course, the student should be able to understand the basic concepts of problem solving and learning in intelligent system engineering.

Course Objective:

- To Introduce the basic concepts of artificial intelligence, problem solving, knowledge representation and reasoning.
- To introduce the basic concepts of handling uncertainty
- To help the students to applications of AI in different fields

Pre-requisite: Discrete Mathematics, Programming Concepts.

Course Outcome: Upon successful completion of this course, students will be able to:

CO1: Apply the concepts of artificial intelligence for real-world problem solving.

CO2: Work in programming languages like Java or Python.

CO3: Apply the concepts of handling uncertainty in various applications.

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, web-based resources as well as flipped class room teaching will be adopted.

UNIT- I

Introduction to AI: Brief introduction about Intelligent agents and Problem Solving. Turing Test. Uninformed Search Strategies, Informed Search Strategies, Heuristics. Solving problems by searching, BFS, DFS, Issues in design of Intelligent Search Algorithms. **[10 Hrs]**

UNIT- II

Knowledge Representation: Knowledge Representation using predicate logic, Rule Based Systems, Ontology, WordNet and Concept Net as Knowledge representation tools. Programming with Python/Java. Text Feature Extraction - BoW Model, TF-IDF. Word Embeddings - Word2Vec, GloVe, stemming, lemmatization **[10 Hrs]**

UNIT- III

Decision Making in Uncertainty: Handling Uncertainty, Probabilistic Reasoning, Fuzzy Logic, Learning by induction, Introduction to Neural Network Genetic Algorithms basics. Rough Sets.

Case Studies of Applications of Uncertainty **[10 Hrs]**

UNIT- IV

Real World Applications of AI: Real World Applications of AI: Expert System Architecture, Case Studies: MYCIN, Applications in NLP, Medical Sciences, Agriculture, education, Social Network Analysis, Information Retrieval from Search Engines and Metasearch Engines, IoT Applications & Big Data Analytics Applications,Ethics in AI**.** **[10 Hrs]**

TEXT BOOKS

1. S.J. Russell and P. Norvig, "Artificial Intelligence- A Modern Approach", Pearson 3rd Edition, 2010 Latest Edition

REFERENCE BOOKS

1. E. Rich and K. Knight, "Artificial Intelligence", McGraw Hill Education; 3rd Edition 2017, Latest Edition.

Paper Code: BIT 401	**L**	**P**	**C**
Paper Title: Software Testing	**4**	**0**	**4**

Introduction: Software testing helps in finalizing the software application or product against business and user requirements. It is very important to have good test coverage in order to test the software application completely and make sure that it's performing well and as per the specifications. Software testing makes sure that the testing is being done properly and hence the system is ready for use.

Course Objectives:

- The students should understand software testing as a fundamental component of the software lifecycle.
- Finding defects which may get created by the programmer while developing the software.
- Gaining confidence in and providing information about the level of quality.
- To make sure that the end result meets the business and user requirements.

Prerequisite: Software Engineering, Programming Skills and Database Management System.

Course Outcome:
CO1: Understand the process of applying tests to software and the fundamental components of a test case.
CO2: Derive test cases from software requirement specifications - including being able to partition input and output domains, form test specifications, and identify valid combinations of input.
CO3: Make logical arguments that prove the correctness of program implementations.
CO4: Write code to automate test execution and analysis.

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, web-based resources as well as flipped classroom teaching will be adopted.

UNIT- I

Introduction: Testing Objectives, Software Testing Process, Software Testing Principles, Tester Role in Software Development Organization, Test Case Implementation and Execution. **Testing Concepts**: Levels of Testing, Test Cases Design and Strategy, Test Suit, Test Plan, Testing as a Process, Testing and Debugging, Limitations of Testing, Software Testing Tools: Characteristics of Modern Tools, Static Testing Tools, Dynamic Testing Tools, Process Management Tools**.** **[10Hrs]**

UNIT- II

Functional Testing: Boundary Value Analysis, Robustness Testing, Worst case testing, Special Value Testing, Equivalence Class Testing-Weak normal, Strong normal, weak robust and Strong Robust, Decision Table Based Testing, Cause Effect Graphing Technique **Structural Testing**: Control flow testing-Statement, Branch, Condition and Path coverage, Data Flow Testing, Testing strategies, Generation of test cases, Slice-based Testing, Mutation Testing, Integration Testing, Decomposition based Integration, Call Graph based Integration, **System Testing**: Thread Testing.

[10Hrs]

UNIT- III

Introduction to Object Oriented Testing, State Based Testing, Class Testing, Web Testing, Issues in Object Oriented Testing, Regression testing, Selection of test cases, reducing the number of test cases, Prioritization guidelines**.** **[10Hrs]**

UNIT- IV

Software Verification Methods, SRS Verification, SDD Verification, Source Code Reviews, Software Project Audit, Debugging Process and Approaches, Software Testing Metrics, Metric used in Testing, Software Quality and Quality Models**.** **[10Hrs]**

TEXT BOOKS

1.Yogesh Singh, "Software Testing", Cambridge University Press, 2011
2. Paul C. Jorgensen, "Software Testing: A Craftsman's Approach", Auerbach Publications; 3rd Edition, 2013

REFERENCE BOOKS

1.Ilene Burnstein, "Practical Software Testing: A Process-Oriented Approach", Springer, 2003.
2.Aditya P. Mathur, "Foundations of Software Testing", Prentice Hall 2008

Paper Code: BIT 403
Paper Title: Big Data Analytics

L	P	C
4	0	4

Introduction:

Our ability to handle Big Data has increased the strategic value of data. Companies employ Big Data technologies for a wide range of analytics, descriptive, predictive and prescriptive, based on their data assets. Collection, storage and retrieval of data assets and processing them in reasonable response time is crucial today. This course deals with volume, variety and velocity aspects of Big Data. It exposes students to basic techniques for managing and processing such data.

Course Objectives:

At the end of the course students should demonstrate the ability to manage big data and process it.

Pre-Requisites:

Essential: Distributed Systems, Data warehouse

Desirable: NoSQL Databases

Course Outcomes:

CO1: Perform data gathering of large data from a range of data sources.
CO2: Critically analyse existing Big Data datasets and implementations, taking practicality, and usefulness metrics into consideration.
CO3: Understand the role of statistics in the analysis of large of datasets.
CO4: Apply suitable statistical measures and analyses techniques for data of various structure and content and present summary statistics.

Pedagogy:

The course will be delivered in workshop mode with lecture material and problem-solving exercises suitably interspersed during lecture contact hours. Tutorial work shall be pen and paper problem solving as well as coding exercises. Take homework shall be oriented to use of tools based on lecture content. Students shall install and learn to use these independently. There shall be about 5 hours per week of take-home work.

Unit I

Introduction – RDBMS Overview, Challenges of Conventional Systems, Intelligent Data Analysis, Nature of Data, Analytic Processes and Tools, Analysis vs Reporting, Modern Data Analytic Tools, Statistical Concepts: Sampling Distributions, Re-Sampling, Statistical Inference, Prediction Error, Accuracy measures, Cutoff, Oversampling & Asymmetric Costs.

Big Data - Volume, Velocity, Variety, Veracity, types & sources of Big Data OLAP & RTAP.

[10 Hrs]

Unit II

Data Exploration & Dimension Reduction: Data Summaries, Data Visualization, Correlation Analysis, Reducing no of categories in Categorical variables, Principal Component Analysis for Classification & Prediction, Multi Variate Regression Analysis, Bayesian Modeling, Support Vector Method, Time Series Analysis. **[10 Hrs]**

Unit III

Introduction to Streams Concepts, Stream Data Model and Architecture, Stream Computing, Sampling Data in a Stream, Filtering Streams, Counting Distinct Elements in a Stream, Case Study: Real Time Sentiment Analysis/Stock Market Prediction. **[10 Hrs]**

Unit IV

Hadoop - The Hadoop Distributed File System – Components of Hadoop, Analyzing the Data with Hadoop, Map Reduce, Map Reduce Types and Formats, Map Reduce Features, NoSQL Database.

Applications on Big Data Using Pig and Hive, Querying Data in Hive through HiveQL **[10 Hrs]**

TEXT BOOKS

1. Chris Eaton, Dirk De Roos, Tom Deutsch, George Lapis, Paul Z., "Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data", McGraw Hill Publishing, 2011
2. Shmueli, Patel & Bruce, "Data Mining for Business Intelligence", 2nd Edition, Wiley Interscience Publications, 2010.

REFERENCE BOOKS

1. Jiawei Han, Micheline Kamber "Data Mining Concepts and Techniques", Second Edition Elsevier, Reprint

Paper Code: BIT 402	L	P	C
Paper Title: **Software Project Management**	4	0	4

Introduction: This course is aimed at introducing the primary important concepts of project management related to managing software development projects. The main objective of this course is to help the students to learn how to successfully plan and implement a software project management activity, and to complete a specific project in time with the available budget.

Course Objective:

- To learn software project management phases.
- To establish a project plan and then execute that plan to accomplish the project objective.
- To create a work breakdown structure, assign responsibility, define specific activities and sequencing them for a software project.
- To learn planning and estimation and scheduling of software project activity components, resources and durations.

Prerequisite: Knowledge of Software Engineering, Basic Programming Course

Course Outcome: Upon successful completion of this course, students will be able to:

CO1: Apply techniques for controlling and enhancing the software development process.
CO2: Understand the essential project management stages and problems that could make an IT project successful or unsuccessful.
CO3: Understand project management principles and methods in an IT project.
CO4: Understand the project's business context and extent, choose the best project management strategy.

Pedagogy:

The course will be delivered in workshop mode with lecture material and problem-solving exercises suitably interspersed during lecture contact hours. Tutorial work shall be pen and paper problem solving as well as coding exercises. Take home work shall be oriented to use of tools based on lecture content. Students shall install and learn to use these independently. There shall be about 5 hours per week of take-home work.

UNIT - I

Introduction: Introduction to software project management activities, Attributes of a project, Project life cycle, Project Management process, Project selection, Preparing a request for proposal, Soliciting proposals, Proposal preparation, Pricing considerations, Proposal submission and follow up, Customer evaluation of proposals. **[10Hrs]**

UNIT - II

Project Management Organizational Structures - Functional type organization, Project type organizations, Matrix-type organization, Project Planning - Project objective, Work breakdown structure, Developing the network plan, Network principles, Preparing the network diagram, Critical path analysis, PERT ,Project Scheduling- Activity duration estimates, Project schedule calculations. **[10Hrs]**

Unit - III

Schedule Control- Project control process, Effects of actual schedule performance, Incorporating project changes into the schedule, Updating the project schedule, Approaches to schedule control, Resource Considerations- Resource constrained planning, Planned resource utilization, Resource Leveling, Resource limited scheduling. **[10Hrs]**

Unit - IV

Risk Management – Risk, Categories of risk, A framework for dealing with risk, Evaluating risks to the schedule, Monte Carlo simulation and critical chain concepts. Project Cost Planning and Performance – Project cost estimates, Project budgeting, Determining the actual cost, Determining the value of work performed, Cost performance analysis, Cost forecasting, Cost control, Software project metrics, Project control and closure, Project Management Issues with regard to New Technologies, Case Study & use of software project management tool.

[10Hrs]

TEXT BOOKS:

1. Clements and Gido, Effective Project Management, Cengage Learning, 2012
2. Bob Hughes, Mike Cotterell, Rajib Mall"Software Project Management", Fifth Edition, McGraw Hill, 2013

REFERENCE BOOKS:

1. A Guide to the Project Management Body of Knowledge (PMBOK Guide), 5th Edition, Project Management Institute, 2013.
2. Samuel J. Mantel, Jr. et al, Project Management, Wiley India, Eighth Edition, 2012.
3. Jeffrey K. Pinto, Project Management – Achieving Competitive Advantage, 3rd Edition Pearson Education, 2013

Paper Code: BIT 404 **L P C**

Paper Title: Cyber Security Management **4 0 4**

Introduction:

Cyber Security and Forensics is the application of investigation and analysis techniques to gather and preserve evidence from a particular computing device in a way that is suitable for presentation in a court of law. This course provides for a broad introduction of cyber security and forensics concepts, industry best practices for information security and key security concepts that will protect an organization against fraud, data breaches and other vulnerabilities. It enables the students to gain in-depth knowledge in the field of Computer forensics & Cyber Crime.

Course Objectives:

- To maintain an appropriate level of awareness, knowledge and skill to allow students to minimize the occurrence and severity of information security incidents.
- To learn techniques used to detect, respond and prevent network intrusions.
- To identify and apply appropriate forensics tools to acquire, preserve and analyse system image.
- To protect information and information systems from unauthorized access, use, disclosure, disruption, modification or destruction in order to provide confidentiality, integrity and availability.
- Identify sources of evidentiary value in various evidence sources including network logs, network traffic, volatile data.

Pre-requisites: Knowledge of Computer Networking, Linux, UNIX, Understanding of Web Application Architecture and HTTP/HTTPS communication.

Course Outcomes:

CO1: Analyze and evaluate the cyber security needs of an organization.
CO2: Determine and analyze software vulnerabilities and security solutions to reduce the risk of exploitation.
CO3: Measure the performance and troubleshoot cyber security systems.
CO4: Implement cyber security solutions and use of cyber security, information assurance, and cyber/computer forensics.

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Students would be encouraged to develop an understanding of the existing real life cyber security issues and how they are solved. Emphasis would be given on assignments where students will be given numerical/ programming assignments based on topics studied in previous lectures. Course will have a blend of theory and practice for the benefit of students. Use of ICT, web-based sources as well as blackboard teaching will be adopted.

UNIT- I

Introduction to Intellectual Property Rights, Evolution of Intellectual Property Laws, Standards and Concepts in Intellectual Property, Introduction and need for intellectual property right (IPR), Types of IPR, Legislation covering IPRS in India, Patent and kind of inventions protected by a patent. Understanding Cyber Laws, Scope of cyber laws, Need for Cyber Laws, law and legal system, Jurisprudence of Indian Cyber Law, Security threat to cyber space and e-commerce**.**

[10 Hrs]

UNIT- II

Introduction to Geographical Indications, New Plant Varieties, Unfair Competitions. Plant Breeder and TRIPS agreement, Copy Rights, Rights covered by copyright, Protection of copyright, Trademarks, Rights of trademark, signs used in trademarks, Types of trademark function, Protection of trademark, Registration of trademark, Domain name and how does it relate to trademarks, cases related to IPR Infringement **[10 Hrs]**

UNIT- III

Components of Cyber Law, Introduction of relevant provisions of Indian Penal Code, Indian Evidence Act, Bankers Book Evidence Act, Reserve Bank of India Act, etc. related to cyber security.

Obscenity and pornography on Cyber space, hacking, punishment for violation of Privacy under IT Act, Ministerial Order on blocking of websites, Cyber laws in US, Cyber laws in Global Prospective, MLAT (mutual legal assistance treaty) International Treaty for Cyber laws.

[10 Hrs]

UNIT- IV

Information Technology Act – a brief overview, Indian IT ACT, 2000 and its amendments, Legal issues pertaining to Device, Mobile Apps and Social Media, IT Act Grey Areas, Protection of IPR in Indian Cyber Space, Plagiarism Issues. **[10 Hrs]**

TEXT BOOKS:

1. Harish Chander, "Cyber Laws and IT Protection", 1st edition, PHI Learning, 2012.
2. Pavan Duggal, "Law Relating to iPads, Tablets, Smartphones & Smart Devices", Universal Law Publishing, 2013.