COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY
DEPARTMENT OF COMPUTER APPLICATIONS
KOCHI – 682 022, KERALA, INDIA

MASTER OF COMPUTER APPLICATIONS
SYLLABUS
(2019 ADMISSION)

Master of Computer Applications (MCA) Course Structure
(2019 Admission onwards)

Semester I

Course Code	Paper	Hours			Marks		Credit
		L	T	P	Sessional	Final	
19-381-0101	Programming in C	3	1		50	50	3
19-381-0102	Mathematical Foundations for Computer Applications	3	1	2	50	50	3
19-381-0103	Computer Organization and Architecture	3	1	2	50	50	3
19-381-0104	Database Management System	3	1		50	50	3
19-381-0105	Digital Electronics and Microprocessors	3	1	2	50	50	3
19-381-0106	C Programming LAB			4	50	50	2
19-381-0107	DBMS LAB			4	50	50	2
					Total		19

Semester II

Course Code	Paper	Hours			Marks		Credit
		L	T	P	Sessional	Final	
19-381-0201	Data Structures.	3	1		50	50	3
19-381-0202	Object Oriented Programming with C++.	3	1		50	50	3
19-381-0203	Operating Systems.	3	1	2	50	50	3
19-381-0204	Data Communication and Networks.	3	1	2	50	50	3
19-381-0205	Software Engineering.	3	1	2	50	50	3
19-381-0206	C++ Programming LAB.			4	50	50	2
19-381-0207	Data Structures LAB.			4	50	50	2
					Total		19

Semester III

Course Code	Paper	Hours			Marks		Credit
		L	T	P	Sessional	Final	
19-381-0301	Design and Analysis of Algorithms	3	1	2	50	50	3
19-381-0302	Applied Probability and Statistics.	3	1	1	50	50	3
19-381-0303	JAVA Programming	3	1		50	50	3
	Elective I.	3	1	2	50	50	3
	Interdisciplinary Elective/ Elective II (for other campuses)	3		2	50	50	3
19-381-0306	Mini Project			2	50		1
19-381-0307	JAVA LAB			4	50	50	2
					Total		18

Semester IV

Course Code	Paper	Hours			Marks		Credit
		L	T	P	Sessional	Final	
19-381-0401	Technical Communications	3	1	2	50	50	3
19-381-0402	Data Mining and Machine Learning	3	1	2	50	50	3
19-381-0403	Web Technology and Internet Programming.	3	1		50	50	3
19-381-0404	Python Programming.	3	1	2	50	50	3
19-381-0405	Information Security.	3	1	2	50	50	3
19-381-0406	Web and Internet programming LAB			4	50	50	2
19-381-0407	Mini Project			4	100		2
19-381-0408	Seminar.	50					1
		Total					20

Semester V

Course Code	Paper				Marks		Credit
		L	T	P	Sessional	Final	
19-381-0501	Applied Artificial Intelligence.	3	1	2	50	50	3
	Elective III	3	1	2	50	50	3
	Elective IV	3	1	2	50	50	3
	Elective V	3	1	2	50	50	3
	Elective VI	3	1	2	50	50	3
19-381-0506	Python Programming LAB			4	100		2
		Total					17

Semester VI

Course Code	Paper	Marks		Credit
		Internal	External	
19-381-0601	Project Work and Course Viva Voce.	200	200	16
Total				16

LIST OF ELECTIVES

Elective I & Elective II

19-381-0311 Computer Graphics
19-381-0312 Theory of Computation
19-381-0313 Mobile Computing
19-381-0314 Embedded System
19-381-0315 Real Time Systems
19-381-0316 Multicore Processing
19-381-0317 Introduction to Cryptography
19-381-0318 Object Oriented Analysis and Design

Elective III

19-381-0511 Web Enabled JAVA Programming
19-381-0512 Visual Programming VB.NET
19-381-0513 Android Application Development
19-381-0514 Web Application Design using PHP
19-381-0515 Linux and Shell Programming

Elective IV

19-381-0521 Software Testing
19-381-0522 Distributed and Cloud Computing
19-381-0523 Software Project Management
19-381-0524 Business Analytics
19-381-0525 Software Quality
19-381-0526 Design Patterns

Elective V

19-381-0531 System Software and Compiler Design
19-381-0532 Network Security and Wireless Security
19-381-0533 Wireless Sensor Networks
19-381-0534 Software Defined Networks
19-381-0535 Security Threats and Vulnerabilities
19-381-0536 Block Chain Technology

Elective VI

19-381-0541 Big Data Analytics
19-381-0542 Natural Language Processing
19-381-0543 Digital Image Processing
19-381-0544 Deep Learning
19-381-0545 Bio Informatics
19-381-0546 Internet of Things
19-381-0547 Data Science and Analytics

19-381-0101 Programming in C

Course Objective

1. To learn basic principles of Problem solving.
2. To understand techniques for specifying data and operations on data using a programming Language.
3. To understand and explore procedure oriented techniques and approaches for constructing programs.

Learning Outcome

1. Student will be able to write algorithms and draw flow charts
2. Will be able to write programs for solving a given problem
3. Will Learn how to handle files
4. Will learn memory allocation and deallocation.

UNIT I Introduction: Programming as a tool for problem solving, Algorithms, Flowchart, Compilers, Operating System, Preprocessor, The GNU C Compiler. Introduction to C programming - Character Set, Identifiers and Keywords, Variables – Declaration of variables, initialization, Data Types- Basic data types, Qualifiers (short, long, unsigned etc), Renaming datatypes using typedef, Arrays, Structure and Union as user defined types (brief idea only), Promotion and type casting. Constants, Operators and expressions, Operator precedence and associativity.

UNIT II Basic Input – Output-Single character input output, string input output, general input output, types of characters in format string, scanf width specifier. Control Structures – Branching and Looping: if-elseif-else, multiway decision-Switch, Loops –while, for, do while. Break, Continue statements in loops and conditional statements. Goto-labels.

UNIT III Functions – Advantages of functions, main() function, function parameters, return value, passing arrays to functions, Recursion, Comparison of Iteration and recursion, Scope and Extend.Arrays and strings revisited- Built-in string functions, Multi-dimensional arrays, arrays of strings

UNIT IV Structures and Union- Introduction, Declaring and using structures, structure initialization, structure within a structure, operations on structures, array of structures. Unions, operations on union, difference between structures and union.

UNIT V Pointers- Introduction, Definition and uses of Pointers Address Operator &, Pointer Variables, Dereferencing Pointers ,Void Pointers, Pointer Arithmetic, Pointer to pointers, Pointers and arrays, Pointers and functions, Array of Pointers, Pointers and strings, pointers to functions, pointers to structures, Dynamic Memory Allocation- Demonstration of malloc(), calloc(), realloc() and free() functions. Files - File structure, File handling functions, file types, Error Handling

TEXT BOOKS:

1. K.R .Venugopal, S.R Prasad,"Mastering C", 11th Reprint, Tata McGraw-Hill, 2011.

REFERENCES:

2. Kernighan, Brain W and Ritchie, Dennis M, 'The C Programming Language'. 2 nd Ed. Prentice Hall, 2007.
3. Byron Gottfried, 'Programming with C', 2 nd Ed. Schaum's outline series, 2002.
4. Les Hancock and Morris Krieger, 'The C Primer', McGraw-Hill, 1987

****

19-381-0102 Mathematical Foundations for Computer Applications

Course Objective

1. To learn the mathematical foundations applicable to computing.

Learning Outcome

Students will be able to

1. Explain sets, relation and functions and illustrate its direct application in Computer languages
2. Solve the problems using the concepts of Graphs, Trees
3. Deduce complex task by various Mathematical logic
4. Solve recurrence relations for a given problem especially for complicated calculations
5. To handle problems from statement calculus

UNIT I Sets - Relations – Functions - Mathematical Inductions (Simple and strong) – Principles of Counting (Addition & Multiplication).Mathematical logic : logical identities rules of inference predicates and quantifiers

UNIT II Introduction to Vector Space, Matrices ,determinants and inverse matrix ,System of equations, Linear transformation - rank and nullity , Consistency and inconsistency of linear system of equations, rank nullity theorem , Echelon form of a matrix and Row reduced echelon form of matrix. Eigen values and Eigen vectors

UNIT III Recurrence Relations and Generating Functions - Homogeneous and non-homogeneous recurrences and their solutions - solving recurrences using generating functions.

UNIT IV Finite Automata – Context-Free Grammars – Chomsky's Normal form -Griebach Normal Form.

UNIT V Push-down Automata - Equivalence of CFL's and PDA's - Non-context free languages.

TEXT BOOK:

1. Kenneth H. Rosen, "Discrete Mathematics And Its Applications", 7th Ed,McGrawHill, 2012.
2. Erwin Kreyszig, "Advanced Engineering Mathematics", Wiley India, 9th Edition 2011.
3. Bernard Kolman, RobertBusby and Sharon Cutler Ross, "Discrete Mathematical Structures for Computer Science", 6 th Ed, PHI , 2013.

REFERENCES:

1. Eric Lehman, F. Thomson Leighton, Albert R. Meyer, "Mathematics for Computer Science", MIT 7th Ed, 2015
2. William Stein, "Elementary Number Theory: Primes, Congruences, and Secrets": A Computational Approach Springer, 2008.
3. Sipser, "Introduction to the Theory of Computation, CENGAGE Learning, 2014. 4. Ernest Davis, "Linear Algebra and Probability for Computer Science Applications ", 1st Edition, CRC Press 2012.
4. Tom M. Apostol, "Introduction to Analytic Number Theory", Springer, 1998.

19-381-0103 Computer Organization and Architecture

Course Objective

1. To conceptualize the basics of organizational and architectural issues of a digital computer.
2. To analyze performance issues in processor and memory design of a digital computer.
3. To understand various data transfer techniques in digital computer.
4. To analyze processor performance improvement using instruction level parallelism.

Learning Outcome

1. To understand the basics of computer hardware and how software interacts with computer hardware
2. To analyze and evaluate computer performance
3. To deal with different types of computers
4. To understand how computers represent and manipulate data
5. To understand computer arithmetic and convert between different number systems
6. To identify high performance architecture design
7. To develop independent learning skills and be able to learn more about different computer architectures and hardware.

UNIT I Basic Structure of Computers - Computer Types, Functional Units, Basic Operational Concepts, Number Representation and Arithmetic Operations, Character Representation, Performance

UNIT II Instruction Set Architecture - Memory Locations and Addresses, Memory Operations, Instructions and Instruction Sequencing , Addressing Modes , Stacks, Subroutines, Additional Instructions, Condition Codes, Encoding of Machine Instructions Input /Output Organization - Accessing I/O Devices, Interrupts, Direct Memory Access, Bus Structure , Bus Operation , Bus Arbitration Interface Circuits, Interconnection Standards

UNIT III The Memory System - Basic Concept , Semiconductor RAM Memories Read-only Memories Memory Hierarchy , Cache Memories - Mapping Functions , Virtual Memory - Address Translation Memory Management

Computer Arithmetic - Addition and Subtraction of Signed Numbers, Multiplication of Unsigned Numbers , Multiplication of Signed Numbers, Integer Division

UNIT IV Basic Processing Unit - Some Fundamental Concepts , Instruction Execution, Hardware Components, Instruction Fetch and Execution, Control Signals, Hardwired Control Microprogrammed Control

Pipelining : Basic Concept, Pipeline Organization , Pipelining Issues Data Dependencies, Memory Delays , Branch Delays, Superscalar Operation

UNIT V Parallel Processing and Performance: Hardware Multithreading , Vector (SIMD) Processing - Graphics Processing Units, Shared-Memory Multiprocessors, Cache Coherence - Write-Through Protocol, Write-Back protocol, Snoopy Caches, Directory-Based Cache Coherence , Message-Passing Multicomputer

TEXT BOOKS

1. Computer organization And Embedded Systems, Hamacher, Vranesic, Zaky, Manjikian, 6Ed, McGraw-Hill , 2012
2. Manish Saraswat, 'Computer Architecture And Organisation', 1st Ed. Vayu Education Of India, 2011.

REFERENCES

1. Tanenbaum A.S, 'Structured Computer Organization'. 5/e, Prentice Hall of India 2006
2. Mano, M M, 'Computer System Architecture'. 3rd Ed. Prentice Hall of India, 2007.
3. Hayes, 'Computer Architecture and Organization', 2nd Ed. McGraw Hill, 1998.

****

19-381-0104 Database Management Systems

Course Objective

1. Understand the role of a database management system in an organization.
2. Understand basic database concepts, including the structure and operation of the relational data model.
3. Construct simple and moderately advanced database queries using Structured Query Language (SQL).
4. Understand and successfully apply logical database design principles, including E-R diagrams and database normalization.
5. Understand the concept of a database transaction and related database facilities, including concurrency control, backup and recovery, and data object locking and protocols.
6. Understand the role of the database administrator.

Learning Outcome

1. To describe data models and schemas in DBMS
2. To understand the features of database management systems and Relational database.
3. To use SQL- the standard language of relational databases.
4. To understand the functional dependencies and design of the database.
5. To understand the concept of Transaction and Query processing.

UNIT I Introduction to File and Database systems- History- Advantages, disadvantages- Data views – Database Languages – DBA – Database Architecture – Data Models - Keys – Mapping Cardinalities.

UNIT II Relational Algebra and calculus – Query languages – SQL – Data definition – Queries in SQL – Updates – Views – Integrity and Security – triggers, cursor, functions, procedure – Embedded SQL – overview of QUEL, QBE.

UNIT III Design Phases – Pitfalls in Design – Attribute types –ER diagram – Database Design for Banking Enterprise – Functional Dependence – Normalization (1NF, 2NF, 3NF, BCNF, 4NF, 5NF).Transaction concept – state - Serializability – Recoverability- Concurrency Control –Locks- Two Phase locking – Deadlock handling – Transaction Management in Multidatabases.

UNIT IV Overview of Storage and Indexing: Data on External Storage, File Organization and Indexing - Clustered Indexes, Primary and Secondary Indexes, Index data Structures - Hash Based Indexing, Tree based Indexing, Comparison of File Organizations.

UNIT V Object-Oriented Databases- OODBMS- rules – ORDBMS- Complex Data types – Distributed databases – characteristics, advantages, disadvantages, rules- Homogenous and Heterogenous- Distributed data Storage – XML – Structure of XML Data – XML Document. Introduction to MongoDB , Overview of NoSQL.

TEXT BOOK

Abraham Silberschatz, Henry F. Korth and ASudarshan, 'Database System Concepts". 6th Ed., McGraw Hill International Edition, 2010.

REFERENCES

1. Philip J. Pratt, Joseph J Adamski, 'Database Management Systems', Cengage Learning, 2009
2. RameezElmasri, Shamkant B. Navathe, 'Fundamentals of Database Systems', 5th Ed., Pearson
3. Education, 2009
4. Arun K Majumdar, Pritimoy Bhattacharyya, 'Database Management Systems', TMH, 2009
5. ISRD group, 'Introduction to Database Management Systems', TMH, 2008
6. Raghu Ramakrishnan, Johannes Gehrke 'Database Management Systems', McGraw Hill International Edition, 2003

****

19-381-0105 Digital Electronics and Micro Processors

Course Objective

1. To have a thorough knowledge of sequential circuits and combinational circuits along with the basics of Verilog coding.
2. To familiar with microprocessors.

Learning Outcome

1. Student will be able to develop programs using Verilog coding.
2. Understand the evolution of microprocessors.

UNIT I Brief review of number systems:Binary, Octal, Hexadecimal.Definition of Boolean Algebra - Basic Theorems and Properties ofBoolean Algebra - Boolean Functions - Canonical and Standard Forms - Digital Logic Gates and timing concepts.The Map Method - K-map 4 variable – Product of Sums Simplification.

UNIT II Binary Adder-Subtractor, Carry look Ahead Adder, Code Converters-Binary to Gray, Gray to Binary, BCD to Excess-3 Code Conversion and vice versa, Magnitude Comparator-4 bit, Decoders, Encoders, Multiplexers, De-multiplexer.

UNIT III Latches, Flip-Flops-SR, D, JK & T, realization of FFs, synchronous and asynchronous sequential circuits- Shift Registers-SISO, SIPO, PISO,PIPO, Design of counters-Modulo-n, Johnson, Ring, Up/Down

UNIT IV Basic Verilog coding: Lexical Conventions - Ports and Modules-Operators -Gate Level Modeling-System Tasks & Compiler Directives - Test Bench- Data Flow Modeling - Behavioral level Modeling -Tasks & Functions

UNIT V Evolution of microprocessors, 8086 Microprocessor – Architecture and signals, Memory organisation, Minimum and maximum mode of operation, Minimum mode Timing Diagram. 8086 Addressing Modes, 8086 Instruction set and Assembler Directives – Assembly Language Programming with Subroutines, Macros, Passing Parameters, Use of stack.Interrupts - Types of Interrupts and Interrupt Service Routine. Handling Interrupts in 8086, Interrupt programming.

TEXT BOOKS

1. M. Morris Mano, "Digital Design", 6th Edition, Prentice Hall of India Pvt. Ltd.2018,
2. Samir Palnitkar," Verilog HDL: A Guide to Digital Design and Synthesis" Prentice Hall, Second Edition, 2009.
3. Bhurchandi and Ray, "Advanced Microprocessors and Peripherals", Third Edition McGraw Hill, 2006.

REFERENCES

1. A. Nagoor Kani, Microprocessors and Microcontrollers, Second Edition, Tata McGraw Hill,2012
2. Thomas L. Floyd & R P Jain, "Digital Fundamentals", PHI, 10th Edition, 2009.

****

19-381-0201 Data Structures

Course Objective

1. To introduce techniques for designing and implementing common data structures on modern computers
2. To Introduce linear and non-linear data structures and best practices to choose appropriate data structure for a given application
3. To explain various sorting and searching techniques
4. Overview of data structures like Linked Lists, Graphs, Trees
5. Introduce complexity classes and ways of classifying problem.

Learning Outcome

1. Students will be able to solve problems using appropriate data structures to implement solutions.

UNIT I Elementary Data Structures - Stacks, Stack ADT and Implementation, Applications, Queues Queue ADT and Implementation, Applications, List, Notion of position in lists, List ADT, Array and linked list implementations.

UNIT II Non-Linear Data Structures- Trees, Terms and Definition, Tree ADT, Applications, Binary Trees, Properties, Representations (Vector Based and Linked), Binary Tree traversal (In Order, Pre Order, Post Order), Applications. Binary Search Tree - Properties of BST, Searching an element in BST, Insertion and Removal of Elements.

UNIT III Heaps - Definition and Properties, Representations (Vector Based and Linked) Insertion and deletion of elements, Heap implementation of priority queue, Heap sort.

UNIT IV Graphs - Properties, Representations (Edge List, Adjacency list, Adjacency Matrix), Graph Traversals (Depth First and Breadth First Search), Directed Graph and Reachability. Spanning trees.

UNIT V Dictionaries, Unordered Dictionary , ADT Specification, Hash Tables, Notion of Hashing and Collision with a simple vector based hash table, Hash Functions, Applications, Properties, Methods for Collision Handling - Separate Chaining, Open Addressing - Linear & Quadratic Probing, Double Hash.

TEXT BOOK

1. Ellis Horowitz. SartajSahni and Anderson Freed, 'Fundamentals of Data Structures in C", 2nd Ed, Universities Press, 2008.

REFERENCES

1. Aaron M.Tanenbaum, Moshe J.Augenstein, “Data Structures using C”, Prentice Hall International Inc., Englewood Cliffs, NJ, 1986.
2. Aaron M Tanenbaum, YedidyahLangsam, Moshe J Augenstein, “Data Structures using C”, Prentice Hall International, Inc, 2009.

****

19-381-0202 Object Oriented Programming with C++

Course Objective

1. To understand the basic concepts of object oriented programming
2. To learn C++ programming.

Learning Outcome

1. To enable students to design software that is modular and extensible.
2. Understand Fundamental concepts of the object model; classes, objects, methods and messages, encapsulation and inheritance, interface and implementation, reuse and extension of classes, inheritance and polymorphism

UNIT I Introduction to Object oriented paradigm, Basic concepts of Object oriented programming, Applications of OOP. Introduction to C++ - I/O Streams, Datatypes and declarations, Operators, Arrays, Strings, Control flow, Storage classes and linking, File streams, Pointers, Reference variables, Functions, Inline functions, Default arguments, Function Overloading.

UNIT II Classes and objects, Static members and functions, Const objects and Const member Functions, Friend functions, Object initialization and cleanup Constructors, Different types of constructors, Destructors, Container classes.

UNIT III Dynamic Object creation-new and delete Operators, this pointer, Operator overloading. Inheritance - Different types of inheritance, Abstract classes, Inheritance versus Composition.

UNIT IV Polymorphism and virtual functions, Pure virtual functions, Abstract classes, Dynamic binding, Casting, Object slicing.

UNIT V Templates- Function Templates, Class templates, Overloading of templates, Exception handling, Namespace.

TEXT BOOK

1. K R Venugopal, Rajkumar Buyya, “Mastering C++”, Tata Mc Graw Hill, 2013.

REFERENCES

1. Bjarne Stroustrup , “The C++ programming language” , Pearson , 2000

2. Herbert Scheldt, The Complete Reference C++ “,Tata Mc Graw Hill, 2003.

3. Object Oriented Programming in C++, Robert Lafore, 2000 Edition, GALGOTIA

****

19-381-0203 Operating Systems

Course Objective

1. To learn the fundamentals of Operating Systems.
2. To learn the mechanisms of OS to handle processes and threads and their communication
3. To learn the mechanisms involved in memory management in OS
4. To know the components and management aspects of concurrency management
5. To learn programmatically to implement simple OS mechanisms

Learning Outcome

1. Appreciate the role of operating system as System software.
2. Compare the various algorithms and comment about performance of various algorithms used for management of memory, CPU scheduling, File handling and I/O operations.
3. Apply various concept related with Deadlock to solve problems related with Resources allocation, after checking system in Safe state or not.
4. To appreciate role of Process synchronization towards increasing throughput of system

UNIT I Operating System Basics: Computer System Structures: Computer System Operations, I/O Structure, Storage Structure, Storage hierarchy, Hardware Protection, Network Structure, Operating System Structures: System Components, OS Services , System Calls, System Programs, System Structure, Process: Process concepts, Process scheduling, Operation on processes, IPC. Threads: Overview, Multithreading models, Threading issues.

UNIT II CPU Scheduling: Basic concepts, Scheduling criteria, scheduling algorithms, Multiple processor scheduling, Real time scheduling. Process Synchronization: Critical section problems, Synchronization hardware, Semaphore, Classic problems of synchronization, Critical regions, Monitors.

UNIT III Deadlocks: system model, Deadlock characterization, Methods of handling deadlocks, Deadlock prevention, deadlock avoidance, Deadlock detection, Recovery from deadlock. Memory Management: Background, swapping, contiguous memory allocation, Paging, Segmentation, Segmentation with paging. Virtual Memory: Background, Demand Paging, Process Creation, Page replacement, Allocation of frames, Thrashing.

UNIT IV File System Interface and Implementation: File Concept, Access Methods, Directory Structure, File Protection. I/O Systems: Overview, I/O Hardware, Application I/O Interface, Kernel I/O Subsystem, Transforming I/ O to Hardware Operations, Streams, Performance.

UNIT V Mass Storage Structure: Disk structure, Disk Scheduling, Disk Management, Swap-space Management. Protection: Goals of Protection, Domain of Protection, Access Matrix, Implementation of Access Matrix, Revocation of Access rights

TEXT BOOK

Silberschatz, Galvin, Gagne, "Operating System Concepts", 9th Ed, Wiley-India, 2012.

REFERENCES

1. Andrew Tanenbaum "Introduction to Operating System",3rdedn . Pearson, 2006
2. Williams Stallings "Operating Systems: Internals and Design Principles" 5thedn. Pearson,2006

****

19-381-0204 Data Communication and Networks

Course Objectives

1. The course gives an overview into network models
2. Design issues of layers of network model
3. Common Protocols used in each layer and
4. An introduction to mobile computing

Learning Outcome

1. Get an outlook on to how to design different layers of a network layer
2. Basic concepts of the working of mobile networking

UNIT I Need for networking, Data Representation in n/w. Categories of Network, Protocols & Standards, Layered models, Reference Models – ISO/OSI Models, TCP/IP models, Connecting Devices used in Networks. Physical Layer: Analog and Digital Signals, Transmission Impairments, Data Rate, Baud Rate, Digital & Analog Transmission, Multiplexing, Guided and Unguided Transmission Media, Wireless Transmission, Communication Satellites. Connection Oriented & Connectionless Communication.

UNIT II Data Link Layer – Function of DLL, Types of Errors in transmission, Detection vs Correction of Errors, Retransmission, Framing, Protocols for Noisy & Noiseless Channels. Protocols for Multiple Access – ALOHA, CSMA, CSMA/CD, CSMA/CA, Reservation, Polling, Token Passing.Channel Allocation Techniques – FDMA, TDMA, CDMA.Wired LAN – Ethernet & Wireless LAN – Bluetooth, Backbone Networks, Virtual LANs

UNIT III Network Layer – Need for network layer, Routing – Optimality Principle, Routing Algorithms – Unicast & Multicast. Addressing- Physical, Logical & Port addressing.IPv4 Addressing, Classful & Classless addressing, NAT, Subnetting.IPv6 addressing. Packet format – IPv4 & IPv6.Internet Protocol – Datagram. Fragmentation, Tunneling. Address Mapping – ARP, RARP, BOOTP, DHCP, ICMP, IGMP.

UNIT IV Transport Layer – Functions of Transport Layer, Process to Process Delivery, UDP, TCP, Congestion Control Techniques. Quality of Service – Need & Techniques. Integrated and Differentiated Services.

UNIT V Application Layer – DNS – Resolver, Remote Login, E-Mail, Telnet, FTP, SMTP, SNMP, POP3, IMAP. WWW – Architecture, URL, HTTP, HTTPS.Mobile Networking – Structure, Cell Signal Encoding, Handing over, Generations of Wireless Communication, EDGE, LTE

TEXT BOOK

1. Data Communications and Networking, Behrouz A. Forouzan, 5th Ed McGraw-Hill, 2013,
2. Fundamentals of mobile computing, Prasant kumar Pattnaik, Rajib Mall, Second Edition, PHI Learning Pvt. Ltd., 2015

REFERENCES

1. Computer Networks, Andrew S. Tanenbaum, David J. Wetherall, 5th Edition, Pearson Education, 2012
2. Peterson, Larry L, Computer networks,Elsevier 2003
3. White, Curt M, Data communications and networking, Cengage Learning India, 2010
4. Stojmenovic, Ivan, Handbook of wireless networks and mobile computing, Wiley, 2002

****

19-381-0205 Software Engineering

Course Objective

1. To provide an overview of software development lifecycle phases including requirements elicitation, analysis, specification, design, development, testing, and deployment.
2. To teach about software process concepts and definition, their application and assessment.
3. To provide the knowledge and skills necessary to participate in/or lead a software project team, understand the relationship of software development to overall product engineering, estimate time, resources and costs, and appreciate uncertainties in the software development.

Learning Outcome

1. Acquire the basic knowledge of software development life cycle processes.
2. Apply sound software engineering practices to design, develop, and test software systems.

UNIT I Introduction- The Software Engineering discipline-its evolution and impact, Software Development Projects, Software Life Cycle Models – Classical Waterfall Model, Iterative Waterfall Model, Prototyping Model, Evolutionary Model, Spiral Model, Agile methodology, Comparison of different Life Cycle Model for a project. Agile Software Engineering Fundamentals, Agile Planning, Agile Execution.

UNIT II Requirements Analysis and Specification – Requirements gathering and Analysis, Software Requirements Specification, Formal System Specification, Axiomatic Specification, algebraic Specification.

UNIT III Software Design – Outcome of a Design Process, Characteristics of a good Software Design, Cohesion and Coupling, Layered arrangement of modules, Approaches to Software Design.

UNIT IV Object Modelling using UML – Unified Modelling Language (UML), UML Diagrams, Use Case Model, Class Diagrams, Interaction Diagrams, Activity Diagrams, State Chart Diagram, Postscript.

UNIT V Coding and Testing – Coding, Code Review, Software Documentation. Testing – Testing in the large Vs. Testing in the small, Unit Testing, Black Box Testing, White Box testing, Debugging, Integration Testing, System Testing, some general issues associated with Testing.

TEXT BOOK

1. Software Engineering: A Practitioner’s approach 8th Edition by Roger S Pressman, McGraw-Hill Education 2014

REFERENCES

1. Fundamentals of Software Engineering fourth edition by Rajib Mall, Prentice-Hall of India Pvt.Ltd; 2014
2. Software Engineering 10th Edition by Ian Sommerville, Pearson, 2016

****

19-381-0301 Design and Analysis of Algorithms

Course Objective

1. To introduce the concepts of Algorithm Analysis, Time Complexity, Space Complexity.
2. To discuss various Algorithm Design Strategies with proper illustrative examples.
3. To introduce Complexity Theory.
4. To engage in analysis and design of complex algorithms for real-world problems in current application domains.

Learning outcome

1. The students will be able to analyze a given algorithm and express its time and space complexities in asymptotic notations.
2. Able to solve recurrence equations and also able to perform an amortized cost analysis.
3. Understanding Divide and Conquer Strategy and able to make comparison between Dynamic Programming and Divide and Conquer Strategies.
4. Understanding the optimization problems and able to solve using Greedy strategy.
5. Design efficient algorithms using back tracking and branch bound Techniques for solving problems.
6. Understanding about the computational problems into P, NP, NP-Hard and NP-Complete

UNIT I Introduction - Definition of Algorithm – time and space complexity-Asymptotic notations. Probabilistic Analysis and Randomized Algorithms-The hiring problem, Indicator random variables, Randomized algorithms. Recurrence Equations-substitution method -recursion tree method-masters theorem(proof not required).

UNIT II Divide-and-conquer-maximum subarray problem, strassens's algorithm for matrix multiplication.AVL Trees – rotations, Red-Black Trees insertion and deletion. B-Trees – insertion and deletion operations. Heap sort -heaps, maintaining the heap property, building a heap, The heapsort algorithm, Analysis of heap sort.

UNIT III Dynamic Programming-elements of dynamic programming, 0/1 Knapsack, Rod cutting, Longest common subsequence. Greedy Algorithms-The Control Abstraction- the Fractional Knapsack Problem, huffman codes. Amortized Analysis-aggregate method, the accounting method, the potential method.

UNIT IV Back Tracking: -8-queens - Sum of subsets - Graph Coloring – Hamiltonian cycles Branch and Bound: Traveling Salesman Problem. Single source shortest paths- the bellman-ford algorithm. All pairs shortest path-The Floyd Warshall Algorithm.

UNIT V Introduction to Complexity Theory :-Tractable and Intractable Problems-The P and NP Classes- Polynomial Time Reductions - The NP- Hard and NP-Complete Classes

TEXT BOOKS

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Introduction to Algorithms,3rd edition MIT Press, 2009.
2. A.V. Aho, J.E. Hopcroft, J.D. Ullmann, 1974, The design and analysis of Computer Algorithms, Addison Wesley, Boston.

REFERENCES

1. S.E.Goodman and S.T.Hedetniemi, 1977, Introduction to the Design and Analysis of algorithms, Tata McGraw Hill Int. Edn, New Delhi.
2. Anany Levitin, Introduction to the Design and Analysis of Algorithms, Pearson, 3rd Edition, 2011.
3. E. Horowitz, S. Sahni and S. Rajasekaran, 1999, Computer Algorithms, Galgotia, New Delhi.

****

19-381-0302 Applied Probability and Statistics

Course Objective

1. Design hypothesis tests for a given set of data and select the appropriate thresholds for the tests.
2. Use linear regression analysis to develop an empirical model of experimental data.

Learning outcome

1. Apply the basic rules and theorems in probability including Bayes's theorem
2. Apply the method of least squares to estimate the parameters in a regression model.

UNIT I Basic Statistics: Collection, tabulation and presentation of data, measure of central tendency, dispersion, correlation, association and grouping of data.

UNIT II Probability: Sample space and events, Axioms of Probability, Conditional Probability and Baye's theorem, Random experiments, Discrete and continuous random variables, Distribution function, Mean, Variance and moment generating function. Probability Distributions- Binomial, Poisson and Normal distributions.

UNIT III Sampling Distributions: Population and Samples, Simple random sampling with and without replacement. Sampling distribution of sample means when variance is known and unknown, Chi-Square, Student's-t- and F distributions. Estimation: Properties of estimates, Methods of estimation – method of maximum likelihood, method of moments and method of least squares.

UNIT IV Interval estimation: Confidence interval for the mean of normal distribution when the variance is known and unknown, Testing of Hypothesis: Simple and composite hypotheses, Type I and Type II errors, power of a test, Tests of hypotheses on single sample, two-sample, Chi-square test.

UNIT V Regression Analysis:Linear Regression Models and least squares, Gauss-Markov Theorem, Multiple Regression from simple Univariate Regression, Best Subset Selection ,Forward and Backward- Stepwise Selection, Forward Stage wise Regression, Ridge Regression , Lasso, Regression analysis of variance.

TEXT BOOKS

1. Hines, W.W, Montgomery, D.C, Goldman, D. M. and Borror, C.M, 'Probability and Statistics in Engineering'. 4/e. 2003, John Wiley & Sons.
2. Trevor Hastie, Robert Tibishirani Jerome Friedman 'The Element of statistical learning' , Second Edition.
3. Walpole, R. E., Myers, R. H., Myers S L & Keying Ye, 'Probability and Statistics for Engineers and Scientists'. 8/e, 2007, Pearson Education

REFERENCES

1. Gupta, S C and Kapur, V K, 'Fundamentals of Mathematical Statistics', Sultan Chand and Co.
2. Erwin Miller and John E.Freund, 'Probability and statistics for engineers' Prentice-Hall of India / Pearson , 7th Ed.

****

19-381-0303 JAVA Programming

Course Objective

1. To improve the programming skills of students by imparting the object oriented concepts
2. To give insights to complete software development including GUI programming, thread management and exception handling
3. To extend programming to find solutions to real world problems/applications which include applets, databases and graphical user interfaces

Learning Outcome

1. Will be able to write classes and create objects
2. Will learn to connect front end and back end
3. Will be able to crate and synchronize threads
4. Will be able to write robust programs incorporating exception handling routines

UNIT I Review of OOPs and Java Basics: Java programming environment, Fundamental programming structures in Java: Comments, Data types, Variables, Operators, Strings, Arrays, Input and Output, Control flow – Branching and Looping. Object Oriented Programming: Classes and Objects, Access specifiers (Public, Private & Protected), Packages, Inheritance, Constructors, Method Overriding.

UNIT II Abstract classes and Interfaces, Object cloning. Inner classes, Local, Anonymous and Static Inner classes. Final fields in classes. Exception handling basics – Try, throw & catch, User defined exceptions. Threads: Thread states, Thread creation – Inheriting thread class and Implementing runnable interface, Thread properties, Synchronization.

UNIT III Streams and Files: Streams, Text Input and Output, Random Access Files, Object Streams and serialization, File Management. Collections: collection interfaces, concrete collections: LinkedList, ArrayList, HashSet, TreeSet, Priority Queue, Maps. The collection frame works, Algorithms

UNIT IV Swing programming: the model-view-controller design pattern, Introduction to layout management, Text Input, Choice components, Menus and Dialog Boxes. Basic event handling - Action listeners, Applets.

UNIT V Database Programming: The design of JDBC, JDBC configuration, executing SQL statements, Scrollable and updatable result sets, Row sets, Transactions

TEXTBOOK

1. Java: The Complete Reference by Herbert Schildt, 9th Ed, 2017

REFERENCES

1. Horstmann and Coronell ,"Core Java -, Volume 1 and 2" , 10 th Ed, Pearson, 2016
2. The Java Programming Language, Ken Arnold, David Holmes, James Gosling, Prakash Goteti, 3rd Edition, Pearson

****

19-381-0401 Technical Communications

Course Objective

1. Introduce the dynamics of Communication in the Business world.
2. Help to familiarize and practice the different kinds of communication tools.
3. Give practice in the nuances of spoken communication.
4. Expose to the different forms of Business communication.

Learning Outcome

1. To help the transition from a student to entrepreneur/employee.
2. To be friend students to the corporate skills needed for employment in the industry/MNCs.
3. To analyse the SWOT in the candidate and improvement techniques.

UNIT I Importance of communication/role of communication in corporate success - Concepts & goals - Vocabulary enhancement (Word Power) - LSRW & English Language - Antonyms/Synonyms - confusing word usage - analogies, grammatical correct usage - phonetics & accent correction

UNIT II Improve listening and speaking, different impediments in listening - Training is listening – language for presentations – Group Discussions – Interview – Telephone - Versant Test – Tips for effective listening / Tips to improve listening – How to improve speaking

UNIT III Reading practices : Loud and silent reading benefits – dictionary use – reading comprehension – answering questions – critical reasoning – analysing a given text – finding solutions – how reading will compliment speaking.

UNIT IV Writing practice – use of prepositions – subject verb agreement – tenses – ordering of words/sentences – Email writing – letters (formal, informal) – Report writing – message writing.

UNIT V Interviews – grooming – body language – attitude – time management – stress management – Managing anxiety – stress during communication – Instrumental values & terminal values – Goal Setting

TEXT BOOKS / REFERENCES

1. Simon Sweeney, “English for Communication”, 2nd Edition, CUP, 2003
2. Leo Jones and Richard Alexander, “New International Business English”, CUP, 2000

****

19-381-0402 Data Mining and Machine Learning

Course Objective

1. To provide the fundamentals on information retrieval and data mining techniques.

Learning Outcome

1. Students will be able to apply various data mining techniques to real time problems.

UNIT I Introduction to Data Mining and Machine Learning, Knowledge Discovery Process, Technologies for Data Mining, Application areas of Data Mining; Major issues in Data Mining; Various type of attributes and data values. Data Preprocessing: Data Cleaning – Missing Values, Noisy Data, Data Reduction – Overview , Wavelet Transforms, Principal Component Analysis, Parametric Data Reduction; Data Transformation and Discretization – Data Transformation Strategies Overview, Data transformation by Normalization, various methods of Discretization, Concept Hierarchy Generation for nominal data.

UNIT II FOUNDATIONS OF LEARNING - Components of learning – learning versus design – characteristics of machine learning – learning models – types of learning – training versus testing – Features – error measures, Descriptive Statistics-Probability Distributions, Inferential Statistics-Inferential Statistics through hypothesis tests.

UNIT III Mining Frequent Patterns, Associations, and correlations: Basic Concepts – Market Basket Analysis, Frequent Itemsets and Association Rules; Frequent Itemset. Mining Methods - The Apriori Algorithm, Generating Association Rules from Frequent Itemsets, Finding Frequent Itemsets without Candidate Generation, FP-Growth, FP-Tree.

UNIT IV Classification - Basic Concepts, Decision tree induction, Bayes Classification, Rule Based Classification, Model evaluation and selection, Advanced Classification methods – Bayesian Belief Networks, Introduction to classification by Back propagation and Support Vector Machines.

UNIT V Classification - Basic Concepts, Decision tree induction, Bayes Classification, Rule Based Classification, Model evaluation and selection, Advanced Classification methods – Bayesian Belief Networks, Introduction to classification by Back propagation and Support Vector Machines.

Cluster Analysis: Introduction to Cluster Analysis, Overview of Clustering Methods, Distance Measures, Partitioning methods - k-Means, k-Medoids; Hierarchical methods - Agglomerative versus Divisive Clustering, BIRCH, Chameleon, Density based methods - DBSCAN, Grid based methods – STING; Evaluation of Clustering.

TEXT BOOKS

1. Jiawei Han, Micheline Kamber, Jian Pei, “Data Mining - Concepts and Techniques” - Morgan Kaufmann Publishers, Third Edition, 2012.
2. T. M. Mitchell, “Machine Learning”, McGraw Hill, 2017.

REFERENCES

1. Ian H. Witten, Eibe Frank, “Data Mining - Practical Machine Learning Tools and Techniques”, Morgan Kaufmann Publishers, Third Edition, 2011.
2. Soman, Divakar and Ajay, “Data Mining – Theory and Practice”, PHI, 2006.
3. Pang-Ning Tan, Michael Steinbach, Vipin Kumar, “Introduction to Data Mining”, *Pearson* Addison Wesley, 2006.
4. Arun K Pujari, “Data Mining Techniques”, Universities Press, 2001.
5. Margaret H Dunham, “Data Mining: Introductory and Advanced Topics”, Pearson Education India, 2006.

****

19-381-0403 Web Technology and Internet Programming

Course Objective

1. This course introduces students to Internet programming, among the many existing web technologies,
2. The course also focuses on Java servlets and JSP technologies.

Learning Outcome

1. Create a basic website using HTML and Cascading Style Sheets.
2. Design and implement dynamic web page with validation using JavaScript objects and by applying different event handling mechanisms.
3. Design and implement server side programs using Servlets and JSP.

UNIT I An overview of Java – Basic JAVA programming, Object Oriented Concepts, Exception handling and Multithreading , Streams and I/O, Applets.Web 2.0: Basics, Understanding Internet - Internet technologies Overview , HTML , CSS , XHTML, Web site creation using HTML and CSS.

UNIT II Java Script: An introduction to JavaScript–JavaScript DOM Model-Date and Objects,-Regular Expressions- Exception Handling-Validation-Built-in objects-Event Handling- DHTML with JavaScript.

UNIT III Servlets: Java Servlet Architecture- Servlet Life Cycle- Form GET and POST actions- Session Handling- Understanding Cookies- Installing and Configuring Apache Tomcat Web Server;- DATABASE CONNECTIVITY: JDBC perspectives, JDBC program example – JSP: Understanding Java Server Pages-JSP Standard Tag Library(JSTL)-Creating HTML forms by embedding JSP code.

UNIT IV An introduction to PHP: Creating simple web pages using PHP, Connecting to Database – Using Cookies. XML: Basic XML- Document Type Definition- XML Schema DOM and Presenting XML, XML Parsers and Validation, XSL and XSLT Transformation, News Feed (RSS and ATOM).

UNIT V AJAX: Ajax Client Server Architecture-XML Http Request Object-Call Back Methods; Web Services: Introduction- Java web services Basics – Creating, Publishing ,Testing and Describing a Web services (WSDL)-Consuming a web service, Database Driven web service from an application – SOAP.

TEXTBOOKS

1. Deitel and Deitel and Nieto, "Internet and World Wide Web – How to Program", Prentice Hall, 5th Edition, 2011.
2. Herbert Schildt, "Java-The Complete Reference", Eighth Edition, Mc Graw Hill Professional, 2011.

REFERENCES

1. Stephen Wynkoop and John Burke "Running a Perfect Website", QUE, 2nd Edition,1999.
2. Chris Bates, Web Programming – Building Intranet Applications, 3rd Edition, Wiley Publications, 2009.
3. Jeffrey C and Jackson, "Web Technologies A Computer Science Perspective", Pearson Education, 2011.
4. Gopalan N.P. and Akilandeswari J., "Web Technology", Prentice Hall of India, 2011.
5. Paul Dietel and Harvey Deitel, "Java How to Program", , 8th Edition Prentice Hall of India.
6. Mahesh P. Matha, "Core Java A Comprehensive Study", Prentice Hall of India, 2011.
7. Uttam K.Roy, "Web Technologies", Oxford University Press, 2011.

****

19-381-0404 Python Programming

Course Objective

1. To improve the programming skills of students by imparting both procedural and object oriented concepts
2. To give insights to complete software development including GUI proogramming, thread management and exception handling
3. To extend programming to find solutions to real world problems/applications which include databases and graphical user interfaces.

Learning Outcome

1. Will be able to write programs for solving a given problem
2. Will be able to write classes and create objects
3. Will learn to connect front end and back end
4. Will be able to crate and synchronize threads
5. Will be able to write robust programs incorporating exception handling routines

UNIT I Introduction to computer programming: Brief discussion on algorithms, programs, programming languages and Python as a programming language.Introduction to Python: Python Data Types, Expressions, Variables, and Assignments, Strings, Lists.

UNIT II Imperative Programming: Python Programs, Execution Control Structures, Looping and Branching. User-Defined Functions, Parameter Passing (thrusting mutable and immutable parameters). Recursion, Memory Management During Recursive Function Calls. Global versus Local Namespaces. Containers: Strings and Lists, String Methods, Formatted Output, Two Dimensional Lists, Iterating through Two Dimensional Lists, Dictionaries, Tuples and Sets, Properties Operators and Methods of Containers. Object Oriented Programming

UNIT III Objects and Classes, Defining a Class in Python, Constructors. Classes as Namespaces Inheritance: Multiple and Multilevel Inheritance, Modifying Built in Classes Using Inheritance, Operator Overloading (Integer Class Operators only) Using Inheritance. Errors and Exceptions: Exception Types, Exception Handling using Try & Except. User Defined Exceptions.

UNIT IV Graphical User Interfaces: Tkinter Widgets – Label, Text, Entry, Button, Canvas & Frames, Event-Based tkinter Widgets, Designing GUIs, OOP for GUIs.Turtle Graphics: Familiarization of various Turle Graphics Methods, Moving and Repositioning Pointer, Drawing Geometric Shapes, Colouring of Drawings. Basic programming using Open GL, Simple animations using Open CV.

UNIT V NumPy: Creating Arrays (array() and arange), reshape(), sum(), min() and max() methods, Item wise arithmetic operations. Pattern Matching Using Regular Expressions: Python Standard Library Module RE.Database Programming in Python with sqlite3: Creating Tables, Querying (Inserting Tuples, Selecting Rows and Updating Tuples) Using Cursor to Iterate over Selected Tuples.Files: Opening and Closing a File, Opening Modes, Various Read and Write Methods.

TEXT BOOKS/REFERENCES

1. Ljubomir Perkovic, “Introduction to Computing Using Python: An Application Development Focus”, Wiley, 2012.
2. Charles Dierbach, “Introduction to Computer Science Using Python: A Computational Problem-Solving Focus”, Wiley, 2013.

****

19-381-0405 Information Security

Learning Objective

1. Provide a basic understanding of the various issues related to information systems security .
2. Learn basic concepts of Information security and help students to understand the three important security goals.
3. Understand the requirements and techniques for security management.
4. The course provides the understanding of security needs in Program, OS, Database and Network level.

Learning outcome

1. Students are able to analyses OS and network for security flaws and fool proofing.
2. The course will present an overview of the risks encountered in information systems security and the tools used for resolving these risks.

UNIT I Introduction and Basic concepts: threats, vulnerabilities, controls; risk; confidentiality, integrity, availability; security policies, security mechanisms; assurance; prevention, detection, deterrence. Basic cryptography - Basic cryptographic terms, Historical background, Symmetric crypto Systems - Conventional systems ,Asymmetric crypto primitives –RSA.

UNIT II Program security - Malicious code: viruses, Trojan horses, worms , Program flaws: buffer overflows, time-of-check to time-of-use flaws, incomplete mediation of Defenses , Software development controls, Testing techniques.

UNIT III Security in conventional operating systems - Memory, time, file, object protection requirements and techniques Identification and authentication . Trusted operating systems.

UNIT IV Database management systems security - Database integrity , Database secrecy , Inference control , Multilevel databases.

UNIT V Network security - Network threats: eavesdropping, spoofing, modification, denial of service attacks, Introduction to network security techniques: firewalls, intrusion detection systems. Cyber crimes and control measures.

TEXT BOOK

1. Charles P. Pfleeger, Shari Lawrence Pfleeger, Jonathan Margulies, Security in Computing” , 5th Ed, Prentice hall, 2015.

REFERENCES

1. Michael E. Whitman, ‘Information Security: incident response and disaster recovery’, Cengage Learning, 2009
2. WM. Arthur Conklin, Gregory B. White, Chuck Cotheren, Dwayne Williams, Roger Lavis, “Principles of Computer Security”, 4th Ed,Mc Graw Hill 2016.

****

19-381-0501 Applied Artificial Intelligence

Course Objective

1. To understand the concepts of intelligence, modeling, simulation, knowledge Representation, reasoning, issues.
2. Basic understanding of Natural Language Processing, Image Processing and Robotics.

Learning Outcome

1. The students would be able to understand Artificial Intelligence, techniques and applications of Artificial Intelligence.
2. Also able to understand the application of AI in Real Time location based systems.

UNIT I Introduction to artificial intelligence - Artificial Intelligence- Definitions, Programming Methodologies, Techniques, Intelligent Systems, Propositional calculus, Predicate Calculus, Rule-Based Knowledge Representation.

UNIT II Intelligent Agents – Agents and Environments, Good Behavior: The Concept of Rationality, The Nature of Environments, The structure of Agents. The Present and Future of AI- Agent Components, Agent Architectures.

UNIT III Heuristic search and state space search **-** Techniques for Heuristic Search, State Space Search-Strategies for State Space Search -Applications of Search Techniques in Game Playing- Minimax strategy and Alpha Beta Pruning, and Planning.

UNIT IV Perception –Image Formation, Early Image-Processing Operations, Object Recognition by Appearance, Object Recognition from Structural Information. Introduction to Recommender Systems -Case study: Real time application.

UNIT V Natural Language Processing- Language Models, Text Classification, Information Retrieval, Information Extraction. Robotics- Robot Hardware, Robotic Perception, Robotic Software Architecture.

TEXT BOOKS

1. Stuart Russell, Peter Norvig: "Artificial Intelligence: A Modern Approach ", 3rd Ed, Pearson, 2016.
2. Elaine Rich, Kevin Knight, B.Nair: "ARTIFICIAL INTELLIGENCE ", 3rd Ed, Mc Graw Hill, 2017.

REFERENCES

1. Charu C. Aggarwal. "*Recommender Systems*. The *Textbook",* Springer, 2016.
2. N.P.Padhy: Artificial Intelligence and Intelligent Systems, Oxford University Press, 2009.

****

ELECTIVES

19-381-0311 COMPUTER GRAPHICS

Course Objective

1. To provide an insight into the components techniques of graphic systems that enables the students to design a graphic system.

Learning Outcome

1. Learn the principles behind display devices, to generate 2D scenes using the algorithms designed.
2. Analyze and Implement algorithms for clipping in 2D & 3D. Learn visible surface detection methods as well as illumination techniques for rendering 3D images.
3. Learn animation techniques and introduction to image processing techniques.

UNIT I Display devices. Raster Scan/ Random Scan Displays/Systems. Output Primitives & their attributes– Point, Line, Curve, Character, Fill area. Line Drawing Algorithms – DDA Algorithm, Bresenham's Line Drawing algorithm. Circle – Bresenham's Mid-point circle Drawing Algorithm. Mid-point ellipse Drawing Algorithm, Polygon filling – Inside-Outside Tests, Splitting Polygons, Scan Line Polygon filling Algorithm, Character display methods. Anti-aliasing Techniques.

UNIT II Two Dimensional View Pipeline, Homogenous Co-ordinate System, 2D Transformations, 2D clipping – Point Clipping, Line clipping – Cohen Sutherland Line Clipping, Liang-Barsky Line Clipping, Nichol-Lee-Nichol Line Clipping, Curve Clipping, Text Clipping, Exterior Clipping.

UNIT III Three Dimensional Concepts, 3D Transformations, 3D view pipeline, Projections – Parallel and Perspective. 3D object representation – Spline, Bezier curves & surfaces. Logical Classification of Input Devices, Interactive Picture Construction Techniques. Fractals – Classification, Dimension, Generating fractals

UNIT IV Visible Surface Detection Methods, Illumination Models and Surface Rendering Methods, Computer animation Techniques.

UNIT V Image Processing – Digital Image Representation, Edge detection – Robert, Sobel, Canny Edge Detectors. Image Segmentation – Using Clustering.

TEXT BOOKS

1. Donald Hearn and M. Pauline Baker, Computer Graphics, PHI, 2e, 2016
2. Gonzalez, Rafael C, Digital image processing. 3rd Ed, PHI 2008.

REFERENCES

1. Guha Sumanta, Computer graphics through openGL from theory to experiments, 2nd ed,CRC Press, 2015.
2. Newman, William M,Principles of interactive computer graphics 2nd ed, McGraw-Hill, 2014
3. Malay K Pakhira, Computer Graphics Multimedia and Animation, PHI, 2e, 2010
4. Apurva A Desai, Computer Graphics, PHI, 2008
5. Foley, James D, Computer Graphics, Addison-Wesley Longman, 1996

****

19-381-0312 THEORY OF COMPUTATION

Course Objective

1. Introduce the concept of formal languages
2. Familiarize finite automata, push down automata and turing machine

Learning outcome

1. Understand solve DFA and NFA problems
2. Use DFA and NFA for lexical analysis for compilation

UNIT I Introduction to Automata Theory and its significance. Formal representation of languages –Chomsky Classification. Finite state automata – Properties of transition functions, Designing finite automata, NFA, Finite Automata with Epsilon Transitions, Equivalence of NFA and DFA, Conversion of NFA to DFA.

UNIT II Equivalence and Conversion of NFA with and without Epsilon Transitions. Myhill-Nerode Theorem, Finite State Machines with Output- Mealy and Moore machine, Regular Grammar, Regular Expressions, Converting Regular Expressions to NFA , converting DFA to Regular Expressions. Pumping Lemma for Regular Languages.

UNIT III Context Free Grammar – Simplification of CFG-Normal forms-Chomsky Normal form and Greibach Normal form-pumping lemma for Context free languages - Pushdown Automata –Formal definition –Language acceptability by PDA through empty stack and final state –Pumping Lemma for CFLs

UNIT IV Turing Machine (TM) – Basics and formal definition, TMs as language acceptors, TMs as Transducers, Designing Turing Machines. Variants of TMs -Universal Turing Machine, Multi- tape TMs, Non Deterministic TMs.

UNIT V Recursively Enumerable Languages, Recursive languages, Properties of Recursively Enumerable Languages and Recursive Languages, Decidability and Halting Problem.

TEXT BOOKS

1. J E Hopcroft and J D Ullman Introduction to Automata Theory and Languages and Computation, Addison Wesley ,1974
2. Michael Sipser, Introduction to the Theory of Computation, Cengage Learning,2013

REFERENCES

1. H R Lewis Papadimitrou, "Elements of Theory of Computation", 2nd Ed, Pearson 2015
2. Peter Linz, "An Introduction to Formal Languages and Automata", Narosa Publucation 6th Ed, 2017

****

19-381-0313 MOBILE COMPUTING

Course Objective

1. To build strong understanding on the concepts of mobile agents used in mobile communication
2. A detailed study of the routing and transport layer protocols in Mobile and Ad Hoc Networks
3. To impart basic understanding of the wireless communication systems
4. To expose students to various aspects of mobile and ad-hoc networks

Learning Outcome

1. Student will be able to understand the characteristics and limitations of networks and transport layer protocols.
2. Develop a new set of improved protocols in mobile and Ad Hoc networks

UNIT I **Introduction to mobile computing**, Middleware and Gateways, Application and services, Internet-Ubiquitous networks, Architecture and three-tier architecture for Mobile Computing, Design consideration for Mobile Computing

UNIT II **Spread spectrum –** Direct sequence, Frequency hoping. Medium Access Control - SDMA, FDMA, TDMA, CDMA, Cellular concepts- channel assignment strategy-hand off strategy interface and system capacity- improving coverage and capacity in cellular system, Satellite Systems-GEO, LEO, MEO. Wireless Communication Systems- Telecommunication Systems- GSM- GSM services & features, architecture -DECT features & characteristics, architecture

UNIT III **Wireless LANS**: Wireless LAN Standards – IEEE 802 Protocol Architecture, IEEE 802.11 System Architecture, Services, Cellular Networks: Channel allocation, multiple access, location management, Handoffs.MAC Layer & Management, Routing - Classification of **Routing. Algorithms**-Algorithms such as DSR, AODV, DSDV, Mobile Agents, Service Discovery

UNIT IV **Mobile internet**-mobile network layer-mobile IP-dynamic host configuration protocol-, mobile transport layer-implications of TCP on mobility-indirect TCP-snooping TCP- mobile TCP transmission-selective retransmission, Transaction oriented TCP- Support for mobility-file systems-WAP.

UNIT V **Mobile Transport Layer** - Conventional TCP/IP Protocols, Indirect TCP, Snooping TCP, Mobile TCP, Other Transport Layer Protocols for Mobile Networks. Protocols and Platforms for Mobile Computing - WAP, Bluetooth, XML, J2ME, JavaCard, PalmOS, -Next Generation Networks- LTE – Architecture & Interface – LTE radio planning and tools, 5G architecture, MIMO, Super core concept, Features and Application.

TEXT BOOKS

1. Asoke K. Talukder, Hasan Ahmad, Mobile Computing Technology- Application and Service Creation, 2nd Edition, McGraw Hill Education.
2. Jochen Schiller, Mobile Communications, Pearson Education Asia, 2008.
3. Jonathan Rodriguez , Fundamentals of 5G Mobile Networks, ,Wiley Publishers, 2015
4. Theodore S. Rappaport, Wireless Communications Principles and Practice, 2/e, PHI, New Delhi, 2004.us aspects of mobile and ad-hoc networks

REFERENCE

1. Andrew S. Tanenbaum, Computer Networks, PHI, Third edition, 2003.

****

19-381-0314 EMBEDDED SYSTEMS

Course objective

1. The goal is to understand the basic structure and design of an Embedded System
2. To learn different ways of communicating with I/O devices and standard I/O interfaces
3. To introduce the programming concepts of Embedded Systems.

Learning Outcome

1. Students will be able to develop an embedded system.

UNIT I Introduction to Embedded Systems– Components of embedded system hardware–Software embedded into the system – Classification of embedded system, Embedded Processors - CPU architecture of ARM processor (ARM9) – CPU Bus Organization and Protocol. Design and Development life cycle model - Embedded system design process – Challenges in Embedded system design.

UNIT II 8051 Micro controller hardware- I/O pins, ports and circuits- External memory- Counters and Timers- Serial Data I/O- Interrupts. 8051 instruction set- Addressing modes- Assembly language programming- I/O port programming- Timer and counter programming- Serial communication- Interrupt programming- 8051 interfacing to LCD, Sensors and Keyboard.

UNIT III Memory - memory technologies – DRAM, SRAM, EPROM, EEPROM – Memory Organizations- I/O Devices – Timer / Counter, Real time clock, ADC and DAC, Keyboards and Displays .DMA – DMA Controllers .Interrupts and Exceptions– Interrupt Controller, ISR ,Device drivers for handling ISR , Memory Device Drivers ,Device Drivers for on-board bus. Serial Communication Standards and Devices , Parallel Port Devices ,Serial Bus ,Parallel bus device protocols.

UNIT IV Inter Process Communication and Synchronization -Process, tasks and threads, Shared data, Multiple process in an application, Multiple threads in an application, Tasks, Task state Inter process communication ,Signals ,Semaphore , Message Queues ,Mailboxes ,Pipes ,Sockets – Remote Procedure Calls (RPCs).Real time operating systems - Services- Goals – Structures - Kernel - Process Management – Memory Management – Device Management – File System Organization. Micro C/OS-II RTOS - System Level Functions – Task Service Functions – Memory Allocation Related Functions – Semaphore Related Functions.

UNIT V Programming concepts of Embedded programming – Features of Embedded C++ and Embedded Java (basics only). –Components for Embedded programs – Assembling, Linking and Loading – Compilation Techniques –Program Optimization, Testing and Validation, Emulation Design Examples: Software Modem- Development using Raspberry Pi.

TEXTBOOKS

1. Rajkamal, Embedded Systems Architecture, Programming and Design, TMH, 2003
2. Wayne Wolf, Computers as Components: Principles of Embedded Computing System Design, Morgan Kaufman Publishers - Elsevier 3ed, 2008
3. Muhammed Ali Mazidi and Janice Gillispie Mazidi, The 8051 Microcontroller and Embedded Systems, 2/e, Pearson Education 2007.

REFERENCES

1. Embedded System Design by PETER MARWEDEL University of Dortmund, Germany, 2005.

****

19-381-0315 REAL TIME SYSTEMS

Course Objective

1.To define constraints in Real Time Operating Systems design.
2. Exploring how sensors could capture external data and interface the same on real time operating system.
3. Using RTOS to facilitate IOT

Learning Outcome

1.To mathematically model a real time system.
2.Exploring how sensors could capture external data and interface the same on real time operating system.

UNIT I Real Time Systems – Introduction, PID control concept, basic components - sensors, embedded processors, actuators, open and closed loop system, Example of embedded applications. Hard and Soft real time systems

UNIT II Real Time Scheduler – Components, optimality and feasibility constraints for hard and soft real time systems. Task, Periodic, Aperiodic, Sporadic tasks. Task graph, DAG's and schematics to model tasks using graphs, commonly used approaches for Real Time Scheduling.

UNIT III Clock driven scheduling, Frame size constraints, cyclic executives, Slack Stealing approach. Priority driven Scheduling – Static and Dynamic – Laxity Based Approach. Deferrable server, Bandwidth Preserving server, Acceptance test.

UNIT IV Resources and Resources access control, Resource control using task graphs, Priority ceiling and priority inheritance protocols.

UNIT V Introduction to IoT – Physical and logical design of IoT, IoT enabled technologies – examples and Raspberry Pi.

TEXT BOOKS

1. Liu, Jane W.S., Real Time Systems, Pearson Education, 2000
2. Laplante, Phillip A., Real-Time Systems Design and Analysis, Wiley, 3rd Ed., 2006.

REFERENCES

1. Arshdeep Bahga, Vijay Madisetti, Internet Of Things A hands on Approach, 2015
2. Jared Hendrix, Raspberry Pi: Essential guide on starting your own raspberry pi3 projects with ingenious tips and tricks, 2016.

****

19-381-0316 MULTICORE PROCESSING

Course Objective

1. To learn the concepts of parallel processing as it pertains to high performance computing
2. To learn the issues of parallel programming
3. To learn the concepts of Multi-core processor

Learning Outcome

1. To Understand pipelining, instruction set architectures, memory addressing.
2. To understand the various techniques to enhance a processors ability to exploit Instruction-level parallelism (ILP), and its challenges.
3. To Understand multithreading by using ILP and supporting thread-level parallelism (TLP).
4. To understand the performance and efficiency in advanced multiple-issue processors.
5. To understand multiprocessor cache coherence using the directory based and snooping class of protocols.
6. To understand the performance of multicore processors using SPEC benchmarks.
7. To understand the several advanced optimizations to achieve cache performance.

UNIT I Fundamentals Of Quantitative Design And Analysis - Classes of Computers – Trends in Technology, Power, Energy and Cost – Dependability –Measuring, Reporting and Summarizing Performance – Quantitative Principles of Computer Design – Classes of Parallelism - ILP, DLP, TLP and RLP - Multithreading - SMT and CMP Architectures – Limitations of Single Core Processors - The Multi core era – Case Studies of Multi core Architectures.

UNIT II DLP in vector, SIMD and GPU architectures - Vector Architecture - SIMD Instruction Set Extensions for Multimedia – Graphics Processing Units - Detecting and Enhancing Loop Level Parallelism - Case Studies.

UNIT III TLP and MULTIPROCESSORS- Symmetric and Distributed Shared Memory Architectures – Cache Coherence Issues -Performance Issues – Synchronization Issues – Models of Memory Consistency -Interconnection Networks – Buses, Crossbar and Multi-stage Interconnection Networks.

UNIT IV RLP and DLP in warehouse-scale architectures - Programming Models and Workloads for Warehouse-Scale Computers – Architectures for Warehouse-Scale Computing – Physical Infrastructure and Costs – Cloud Computing – Case Studies.

UNIT V Architectures For Embedded Systems - Features and Requirements of Embedded Systems – Signal Processing and Embedded Applications – The Digital Signal Processor – Embedded Multiprocessors - Case Studies.

TEXT BOOK

John L. Hennessey and David A. Patterson, “Computer Architecture – A Quantitative Approach”, Morgan Kaufmann / Elsevier, 5th. Edition, 2012.

REFERENCES

1. Kai Hwang, “Advanced Computer Architecture: Parallelism, Scalability and Programmability” , McGraw-Hill, 1993.
2. Richard Y. Kain, “Advanced Computer Architecture: A System Design Approach”, PHI, 1999.
3. Rohit Chandra, Ramesh Menon, Leo Dagum, and David Kohr, “Parallel Programming in OpenMP”, Morgan Kaufmann, 2000.
4. Joseph JaJa, ” Introduction to Parallel Algorithms”, Addison-Wesley, 1992.

****

19-381-0317 INTRODUCTION TO CRYPTOGRAPHY

Course Objective:

1. Learn about basic cryptographic algorithms for resolving security risks.

Learning Outcome:

1. Provide theoretically sound foundation in cryptography.
2. Will be able to resolve threats and attacks.

UNIT I Classical cryptography: Shift cipher, Substitution cipher, Ane cipher, Vigenere cipher, Hill cipher, Permutation cipher, Stream ciphers, LFSR, Cryptanalysis of Vigenere cipher and LFSR.

UNIT II Block ciphers: Feistel cipher, Data Encryption Standard, Multiple encryption, 3-DES, Substitution Permutation networks, Advanced Encryption Standard and Modes of operation.

UNIT III Hash functions: Hash functions and data integrity, Secure Hash Algorithm, Message Authentication Code, Nested MAC, HMAC, CBC-MAC.

UNIT IV Public Key Cryptosystems: Integer factorization problem, Discrete logarithm problem, RSA cryptosystem, Attacks on RSA, Diffie-Hellman Key agreement scheme, ElGamal cryptosystem, Elliptic curve cryptography.

UNIT V Signature schemes:RSA signature, ElGamal signature, Digital Signature Algorithm, ECDSA. Certificates - X.509 certification standard.

TEXTBOOKS

1. Behrouz A Forouzan, Cryptography and Network Security, Tata Mc Graw Hill, 2005.
2. Cryptography: Theory and Practice, (Third Edition), Douglas R. Stinson.
3. William Stalings, Cryptography and Network Security, Principles and Practices. 6th Ed., Pearson Education, 2014.

References:

1. Handbook of Applied Cryptography, (Second Edition), Alfred J. Menezes, Paul C. van Oorschot, Scott A. Vanstone.
2. Introduction to Modern Cryptography, (Second Edition), Jonathan Katz, Yehuda Lindell.
3. Understanding Cryptography: A Textbook for Students and Practitioners, (2010 Edition),Christof Paar, Jan Pelzl.
4. Introduction to Cryptography with Coding Theory, (Second Edition),Wade Trappe, Lawrence C. Washington.
5. Network Security and Cryptography, Bernard Menezes.

****

19-381-0318 Object Oriented Analysis And Design

Course Objective

1. Understand Object orientation concepts, theories and principles; To enable students to design software that is easy to understand, easy to maintain, and is modular and extensible
2. Understand Fundamental concepts of the object model; classes, objects, methods and messages, encapsulation and inheritance, interface and implementation, reuse and extension of classes, inheritance and polymorphism;
3. Understand the issues involved in implementing an object-oriented design, Analyze requirements and produce an initial design. Develop the design to the point where it is ready for implementation.

Learning Outcome

1. Knowledge of Unified Process; Students will know how to gather requirements, analyse requirements and design software for them.
2. The student should be able to apply the principles of object-oriented concepts using the Unified Process and Unified Modeling Language (UML) to any software development effort; Students will know the characteristics of good design.
3. Ability to recognize Design Patterns in a Software Product; Students will know the best practices used in industry to design software

UNIT I **Introduction-**Two views of software Developments: SSAD and OOAD. **The Object Paradigm-**Object and classes**,** Abstraction and encapsulation, Methods and Message**,** Interfaces, Inheritance and Polymorphism.Access Control. **Introduction to UML & Modeling**- Review of the object Oriented Methodologies by Booch, Rumbaugh, Cood Yourdon, Ivar Jacobson**. Unified Approach -** Diagramming and Notational Techniques using the UML,UML Diagrams and software Development Phases.

UNIT II **Object-Oriented Systems Development Process-** Rational Unified Process**,** Four Major phases:- Inception , Elaboration, Construction, Transition. Requirements Engineering**.** Problem analysis - Understanding Stockholders need,Type of requirements. Road Map For OOA & OOAD : Analysis & Design Road Map. Steps in UML Based Process.

UNIT III **Structural Modeling-**Common Structural Modeling Techniques – Approaches to find classes**.** Modeling Structural Elements -Classes, Relationships, Interfaces, Packages**.** Class Diagrams**,** Difference between ERD & Class Diagram,Object Diagram.

UNIT IV **Behavioral Modeling-**Common Behavioral Modeling Techniques**,** Interactions**.** Use Cases and Use Case Diagrams. Interaction Diagrams : Sequence Diagrams, Collaboration Diagrams , Activity diagrams and State chart Diagram. Forward & Reverse Engineering **Architectural Modeling-** Common Architectural Modeling Techniques**,** Modeling Architecture of the system**,** Components & Component Diagrams**,** Deployment & Deployment Diagrams**,** Collaborations.

UNIT V **Persistent Object and Database Issues-**The Cood Data Management Domain**.,** Object persistence, Object-oriented Database Management System**,** Object-Oriented verses Relational Database, Mapping object to Relational Data structure**Testing of Object** oriented applications- Introduction to Testing Strategies, Impact of Object Orientation on Testing. Testing Business Process. **CASE Tools-Any Tool to draw UML diagrams**

TEXT BOOKS

1. Object Oriented Analysis and Design with Applications by Grady Booch., Benjamin /Cummings , 1994., Pearson Pub.
2. Object – Oriented Modeling and Design with UML by J Rumbaugh, M Blaha, W. Premerlani,PHI Pub, 2004
3. Magnifying Object Oriented Analysis and Design by Arpita Gopal and Netra Patil : PHI Publication, 2010

REFERENCES

1. Principles of Object- Oriented Software Development - Anton Eliens , Addison Wesley, 2000.
2. Object Oriented System Development - Ali Bahrami McGRAW-HILL International Edition Object-Oriented Software Engineering - Ivar Jacobson Pearson Education INC, 2016

****

19-381-0511 Web Enabled Java Programming

Course Objective

1. Understand the JAVA concepts needed to design and build Web based application

Learning Outcome

1. Student will be able to apply java server programming to build various real time application

UNIT I Core Java Overview: Object oriented concepts, Exception Handling, Multi Threading Introduction to JDBC : Overview of JDBC API, The Java.sql package, JDBC Drivers, Executing SQL commands using JDBC Drivers, static and dynamic Execution of SQL statements, Execution of Stored Procedures using JDBC. Introduction to Transactions and Transaction Methods. Introduction to JNDI, Introduction to Data Source and Connection pooling, Introduction to Web Applications, Web Servers Overview of J2EE Technologies

UNIT II Introduction to Java Servlets, Static and Dynamic contents, Servlet life Cycle and Life cycle methods, Servlet Request and Response Model, Deploying a Servlet, Servlet State Transitions, Servlet Config and Servlet Context, Servlet Redirection and Request Dispatch, Servlet Synchronization and Thread Model. Maintaining Client State: Cookies, URL rewriting, Hidden form fields, Session Tracking.

UNIT III Introduction to JSP : JSP & Servlet as Web Components,Servlets vs. JSP, JSP Lifecycle, JSP Page Lifecycle Phases, General Rules of Syntax, JSP syntactic elements, JSP element syntax, Templatecontent. JSP elements-directives, declarations, expressions, scriptlets, actions. JSP Standard Actions: jsp:useBean, jsp:getPreoperty, jsp:setProperty, jsp:include, jsp:forward, jsp:plugin,jsp:param,java Server Pages Standard Tag Library(JSTL).

UNIT IV Introduction to JSF Frameworks: Getting started: A Simple Example, Sample Application Analysis, Development Environments for JSF. Managed Beans: A Sample Application, Bean Scopes Configuring Beans, Navigation, Static Navigation ,Dynamic Navigation, Standard JSF tags, Datatables, conversion and validation Overview of the Conversion and Validation Process ,Using Standard Converters. Event Handling: Life Cycle Events, Value Change Events, Action Events , Event Listener Tags, Immediate Components, Passing Data from the UI to the Server ,.Custom Components, Converters and Validators: Classes for Implementing Custom Components, Tags and Components, The Custom Component Developer's Toolbox, Encoding: Generating Markup, Decoding: Processing Request Values ,Using Converters, Implementing Custom Component Tags, The TLD File, The Tag Handler Class, Defining Tag Handlers in JSF 1.1 .

UNIT V AJAX :Ajax Fundamentals ,JavaScript Libraries, The Prototype Library ,The Fade Anything Technique Library ,Form Completion. Realtime Validation, Propagating Client-Side View State Direct Web Remoting, Ajax Components, Hybrid Components, Keeping JavaScript Out of Renderers, Transmitting JSP Tag Attributes to JavaScript Code,Ajax4jsf,Implementing Form Completion with Ajax4jsf,Implementing Realtime Validation with Ajax4jsf.Introduction to Java Web Services, Future Trends in Web Technology WEB 2.0 and Beyond- Flex.

TEXT BOOKS

1. Professional Java Server Programming- J2EE 1.3 Edition- SubrahmanyamAllamaraju and Cedric Buest- Apress publication, 2007.

2. Core JavaServer Faces-Second Edition-David Geary, CayHorstmann-Prentice Hall-2007

REFERENCES

1. Inside Servlets-A Server Side Programming for the Java platform- Dustin R Callaway-Pearson Education Asia-2005
2. Beginning JavaServer Pages- VivekChopra,JonEaves,Rubertjones,SingLi,JohnT.Bell-wrox publications-2005
3. Beginning J2EE 1.4 With foreword by Ivor Horton-Kevin Mukhar and James L.Weaver- Apress publication2004
4. Pro JSP 2-Fourth Edition-Simon Brown, Sam Dalton, Daniel Jepp,David Johnson, Sing Li, and Matt RaibleEdited by Kevin Mukhar-Apress Publication-2005
5. JavaServer Faces in Action-KITO D. MANN- MANNING publications Co- 2005
6. Pro JSF and Ajax ,Building Rich Internet,Components- Jonas Jacobi and John R. Fallows Apresss Publications-2006
7. Flex 3 In Action-Manning Publications Company-2009.

****

19-381-0512 Visual Programming Vb.Net

Course Objective

1. Understand the Visual Programming environment
2. To design and develop both Windows and Web based application.

Learning Outcome

1. Student will be able to apply Visual programming techniques.

UNIT I Introduction to the Program , Visual Basics Keywords and Syntax, Enhancing a sample application. Object Oriented Terminology - Problem Solving - Class & Objects, Properties, Constructor, Inheritance, Polymorphism. Class Collaboration and relationships, Collections, Abstract class, Interface.

UNIT II CLR, Memory management, Namespaces – Creating your own Namespaces,
User Interface – Master page, Controls, Validations using Validation Controls. Debugging using breakpoint.

UNIT III SQL Server 2012 - Table Creation & Constraints, DDL, DML, DCL, Sub query, Joins, triggers, Aggregate Function, Stored Procedures, Indexes, Views.

UNIT IV SQL Server 2012 - Ado .net architecture, Connected Model, Disconnected Model.
Layering, State Management, Implementation of CRUD using layering.

UNIT V Working with ASP.NET – Building ASP.Net application using webforms, Data driven Applications, Windows Services, Using IIS application services, Interacting with windows services, creating windows services, Security in Windows framework.

TEXTBOOK

1. Professional Visual Basic 2010 and .NET4; Bill Sheldon, Kent Sharkey, Jonathan Merbutt, Rob Windsor, Gatson C Hiller, Wiley publishing 2010

REFERENCES

1. Professional VB 2005 with .NET 3.0 – Bill Evjen, Billy Hollis, Rockford Lhotka, Tim Mc Carthy, Wiley Publishimg 2007
2. Steven Holzner, 'Visual Basic .NET Programming Black Book'.
3. Carneron Wakefield, Henk-Evert Sonder, Wei Meng Lee, 'VB.NET Programming Developer's Guide'.
4. Professional VB.NET 2003-– Bill Evjen, Billy Hollis, Rockford Lhotka, Tim Mc Carthy, Jonathan Pinnovk, Rama Ramachandran, Bill Sheldon. 2004

19-381-0513 Android Application Development

Course Objective

1. Understand the Android studio environment
2. To design and develop mobile applications.

Learning Outcome

1. Will be able to develop mobile applications for real time environment.

UNIT I **Introduction to Android**

Introduction to Android Architecture: Introduction, History, Features and Android Architecture. Application Environment and Tools, Android Studio, Android SDK, AVD. Application Components- Activity, Content providers, Broadcast receivers, Services. Intents- Explicit and Implicit Intents, Intent Filter, Manifest File. Debugging android application.

UNIT II **User Interface Design**

User Interface Design: Views &View Groups, Views : Button, Text Field, Radio Button, Toggle Button, Checkbox, Spinner, Image View, Image switcher, Event Handling, Listeners, Layouts : Linear, Relative, List View, Grid View, Table View, Web View, Adapters. Creating the user interface programmatically, Managing changes to screen orientation, Displaying notifications- Setting up notifications, Notification manager. Designing for Tablets – Working with tablets: Developing for different android platforms, Fragments, combining fragments into a multilane UI, Specialized Fragments

UNIT III **Mobile Data Management**

Shared Preferences – Saving and Loading User Preferences, Persisting Data to Files, Creating and using Databases, SQLite Databases. Content Providers - Using a Content Provider, Built in Content Provider - Browser, CallLog, Contacts, Media Store and Settings. Creating Your Own Content Providers -Uri, CRUD access.

UNIT IV **Native Capabilities, Messaging, Location based services**

Camera, Audio, Sensors and Bluetooth: Android Media API: Playing audio/video, Media recording. Sensors - how sensors work, listening to sensor readings. Bluetooth , Messaging – SMS Messaging, Broadcast Receiver, Sending Email . Maps & Location: Maps : Map - Based Activities, How to load maps, To finding map API key, GPS, Working with Location Manager, Working with Google Maps extensions, Location based Services. Location Updates, location-based services (LBS), Location Providers, Selecting a Location Provider, Finding Your Location.

UNIT V **Threading, Services, Web services**

Tasks & Processes: Tasks, Switching between Task, Process, Process lifecycle. Threads, Thread Life cycle, Worker Threads, Thread Handlers, Threads & Loopers and IPC.Services: Services and Notifications – bound/unbound services, Starting and stopping services, Android Interface Definition Language, Handler and Messenger, Passing objects over IPC, Scheduling of services. Web Services – Android Server Communication: communication protocols, interacting with server-side applications, develop clients for web services, Exchanging Data over the Internet data parsing using json and xml parsing. Integrating with 3rd party Apps using Web Services.

TEXTBOOK

1. Beginning Android Programming with Android Studio, 4ed, by J. F. DiMarzio, 2016

REFERENCES

1. Android Application Development Cookbook, by Wei-Meng Lee, John Wiley and Sons, 2013
2. Professional Android 4 Development by Reto Meier, John Wiley and Sons, 2012
3. Android in Action, Third Edition, by W. Frank Ableson, RobiSen, Chris King, C. Enrique Ortiz, 2012

****

19-381-0514 WEB Application Design using PHP

Course Objective

1. To gain ability to develop responsive web applications
2. To explore different web extensions and web services standards.
3. To be familiarized with open source Frameworks for web development.

Learning Outcome

1. Students are able to develop web applications using PHP

UNIT I Introduction to PHP, Language Features. PHP Basics, PHP's Supported Data Types, Identifiers, Variables, Constants, Expressions, String Interpolation, Control Structures, Functions, Arrays, Strings and Regular Expressions, Working with the File and Operating System.

UNIT II Object-Oriented PHP, Advantages of OOP, OOP Concepts, Constructors and Destructors, Static Class Members, The instance of Keyword. Advanced OOP Features - Object Cloning, Inheritance, Interfaces, Abstract Classes, Namespaces.

UNIT III PEAR - Using the PEAR Package Manager, Introducing Pyrus. Date and Time - PHP's Date and Time Library, Date Fu. Error and Exception Handling - Error Logging, Exception Handling.

UNIT IV Handling File Uploads - Uploading Files via HTTP, PHP. PHP and LDAP - Using LDAP from PHP. Session Handlers - Configuration Directives, Working with Sessions, Creating Custom Session Handlers. Working with HTML Forms - PHP and Web Forms, Validating Form Data.

UNIT V Authenticating Your Users - HTTP Authentication Concepts, Authenticating Users with PHP. Web Services – XML – Loading and Parsing XML. Security - Hiding Configuration and Sensitive Data, Data Encryption. Introducing the Zend Framework - Introducing MVC, PHP and Zend Framework. Using PHP with MySQL - Interacting with the Database.

TEXT BOOK

1. Beginning PHP and MySQL,W. Jason Gilmore, Apress, 2010, Fourth Edition

REFERENCES

1. PHP6 and MySQL, Steve Suehring, Tim Converse and Joyce Park, Wiley India 2010, Second Edition
2. HTML4 Complete ,E.Stephen Mask, Janan Platt BPB Publications, First Edition 1998.

****

19-381-0515 Linux and Shell Programming.

Course Objective

1. To provide insight into LINUX commands and Shell scripting.
2. To Explore understanding of the files, file systems and System calls and its usage

Learning Outcome

1. Students will be able to explain how the Unix shell processes commands
2. Students will be able to Explain how the Unix file system stores information

UNIT I **Directories and Files in UNIX / Linux system.**-Basic UNIX commands-Getting Introduction to UNIX / Linux-Logging in to Linux system-UNIX files and directories-Basic UNIX commands**.**

UNIT II **Internals of Unix file system**- I/O in UNIX-Basics of files, UNIX file system, inode structure in UFS, Links in UNIX, Directories in UNIX, Conversion of a path name to an inode , open files and descriptor management, File system layout, Superblock, I/O in UNIX – Kernel I/O structure, I/O devices – Block and Character devices, Device drivers, I/O queuing and interrupt handling.

UNIT III **VI editor, File and directory commands, I/O indirection and Filters**-Working with VI editor: Basic operating modes-Commands for copying & pasting text and pattern searching & substitution of text in command mode, Recovering files in vi editor,File and directory commands – cat, cp, mv, rm, chmod, cmp, diff and top,I/O indirection, Filters – wc, sort, head, tail, grep, pipe and tee .

UNIT IV **Shell Scripting**-Basic concepts, Pattern matching and Shell Meta characters, Shell Scripting: simple programs, interactive shell scripting, arithmetic in shell, variables in shell , Decision making constructs , Loop constructs , Files using Shell scripting with sample programs

UNIT V **System calls**-Need for protection, System call – User mode and Kernel mode communication, execution of a system call, difference between system call and function call, system call examples-System calls for low level file I/Os – open, creat, close, read, write, lseek, mkdir, rmdir, rename, link, symlink, unlink-System calls related to process – fork, wait, waitpid, exec, signal, kill and raise.

TEXT BOOKS

1. Brian. W. Kernighan and Rob Pike, “The UNIX Programming Environment”, Prentice – Hall of India, Pvt. Ltd, 1984
2. Operating System - Linux, NUT Press, PHI Publisher,2006 Edition
3. Red Hat Linux Bible, Cristopher Negus, Wiley Dreamtech India
4. Linux Administration Handbook, EviNemeth,Garth Snyder, Trent KHein -Pearson Education

REFERENCES

1. Muster J.C., Introduction to UNIX and LINUX, McGraw Hill, 2002.

2. Quigley E., UNIX Shells by Example, Prentice Hall, 2001.
3. O'Reilly, Learning the vi Editor, 6th Edition, 1998, Arnold Robbins, Linda Lamb

4.O'Reilly, Learning the bash Shell, 2nd Edition, By Cameron Newham, Publisher: O'Reilly Media, Released: March 2005, 3rd edition

5. W. Richard Stevens, "Advanced Programming in the UNIX environment" 2nd edition, Addition-Wesley, 2005
6. Beginning Linux Programming by Neil Mathew & Richard Stones, Wiley Dreamtech India

****

19-381-0521 Software Testing

Course Objective

1. The course aims at providing a sound conceptual foundation in the area of Software Testing with emphasis on concepts and techniques for testing and analysis of software
2. Be familiar with different types of testing tools, their uses and the issues and challenges in test automation

Learning Outcome

1. Will be able to apply different Software Testing methods and design test consolidation.

UNIT I **A Perspective on Testing–** Software Testing Life Cycle - (STLC),Quality Attributes, Types and Levels of Testing—Acceptance Testing, Overview of Testing Techniques- black box testing-boundary value analysis- Robust bva ,Worst-Case bva, robust worst case bva- Equivalent class partitioning-Decision table based testing -Cause effect graph. White-Box Testing-code based testing -Statement Coverage—Branch Coverage—Condition Coverage-Data Flow Coverage, path testing-metric-Manual versus Automated Testing, Static versus Dynamic Testing

UNIT II **Transforming Requirements to Testable Test Cases**- Introduction, Software Requirements as the Basis of Testing, Requirement Quality Factors, Process for Creating Test Cases from Good Requirements, Transforming Use Cases to Test Cases, Ad Hoc Testing, Exploratory Testing

UNIT III **Emerging specialized areas in testing:** Nonfunctional Testing - Performance Testing, Stress Testing, Performance Monitoring, Knowledge Acquisition Process, Test Development, Performance Deliverables, Security Testing, Types of Security Testing, Usability Testing, Goals of Usability Testing, Compliance Testing

UNIT IV **Test Process and Automation Assessment**: Test Automation Framework Basic Features of an Automation Framework, Standard Automation Frame works- Data driven Framework, Modular Framework- Keyword Driven Framework -Hybrid framework.

UNIT V **Automated Tools for Testing**: Tool Selection and Implementation- Test Data/Test Case Generation - vendor testing tools – open source testing tools -Introduction of Object Oriented Testing –Life Cycle Based Testing Overview and Perspective, Introduction of Automation Tools- selenium- QTP.

TEXT BOOKS

1. Software Testing – A Craftsman's Approach, Fourth Edition, Paul C Jorgenson, CRC Press.
2. Software Testing and Continuous Quality Improvement Third Edition, William E Lewis, Auerbach Publications, 2009.

REFERENCES

1. Foundations of Software Testing, Second Edition, Aditya P Mathur, Pearson.
2. Glenford J.Myers," The Art of Software Testing" John Wesley & Sons, 2 edn. 2004.
3. Boris Beizer, "Software Testing Technologies" 1st edition Dreamtech 2000.

19-381-0522 Distributed and Cloud Computing

Course Objective

1. To learn the various concept of Distributed and Cloud computing.
2. To study the Architecture and service models in Cloud computing.

Learning Outcome

1. Students will be able to Acquire Knowledge on the features and development of Cloud Computing.
2. Students will be able to use various performance criteria to evaluate the quality of the cloud architecture.
3. Identify the Service-Oriented Architecture for Distributed Computing workflow.

UNIT I Distributed Systems Models and Enabling Technologies: Scalable Computing – Technologies for Network-Based Systems – System Models for Distributed and Cloud Computing – Clusters of Cooperative Computers, Peer-to-Peer Network Families, Cloud Computing over the Internet. Software Environments for Distributed and Clouds – Trends toward Distributed Operating Systems, Parallel and Distributed Programming Models.

UNIT II Virtualization: Basics of Virtualization, Types of Virtualization, Implementation Levels of Virtualization- VMM Design Requirements and Providers, Virtualization Support at the OS Level, Middleware Support for Virtualization-Virtualization Structures, Tools and Mechanisms- Hypervisor and Xen Architecture, Binary Translation with Full Virtualization, Para-Virtualization with Compiler Support- Virtual Clusters and Resource management-Virtualization for Data-Center Automation- Cloud OS for Virtualized Data Centers, Trust Management in Virtualized Data Centers

UNIT III Cloud Platform Architecture over Virtualized Data Centers: Cloud Computing and Service Models- Architectural Design of Compute and Storage Clouds- Inter-cloud Resource Management- Resource Provisioning and Platform Deployment, Virtual Machine Creation and Management, Cloud security and resource management- Cloud Security Defense Strategies. Distributed Intrusion/Anomaly Detection, Data and Software Protection Techniques. Public Cloud Platforms: GAE, AWS, and Azure

UNIT IV Service-Oriented Architectures for Distributed Computing: Services and Service-Oriented Architecture- Message-Oriented Middleware-Workflow in Service-Oriented Architectures - Workflow Architecture and Specification, Workflow Execution Engine

UNIT V Cloud Programming and Software Environments :Features of Cloud and Grid Platforms – Grid Service Protocol Stack, Open Grid Services Architecture ,Grid Data Access Models, Resource Brokering with Grid bus- Parallel and Distributed Paradigms – MapReduce, Iterative MapReduce,Hadoop Library from Apache, Support of Google App Engine – Amazon AWS and Microsoft Azure

TEXT BOOKS

1. Distributed and Cloud Computing: From Parallel Processing to the Internet of Things – Kai Hwang, Geoffrey C. Fox and Jack J. Dongarra – Morgan Kauffmann, 2011.
2. Mastering Cloud Computing – Rajkumar Buyya, Christian Vecchiola and S.Thamarai Selvi – Tata McGraw Hill Education, 2013
3. George Coulouris, Jean Dollimore and Tim Kindberg , Distributed Systems: Concepts and Design, Fifth Edition , Pearson Education, 2011

REFERENCES

1. Enterprise Cloud Computing : Technology, Architecture, Applications - Shroff, Gautam - Cambridge University Press, 2013.
2. Cloud Computing – A Sreenivasan and J. Suresh – Pearson, Chennai,2014.
3. Cloud Computing: A Practical Approach – Anthony Velte, Toby J Velte and Robert Elsenpeter - McGraw Hill Education (India) Private Limited, 2009.

****

19-381-0523 Software Project Management

Course Objective

1. To give basic understanding of software project engineering, management, monitoring and control.

Learning Outcome

1. *Student will be able to manage real time software projects.*

UNIT I Introduction to Management – Management: Science, Theory and Practice – Definition of Management: It's nature and purpose, The aim of all managers Science or Art. The Systems approach to Operational Management, The functions of Managers. Software Engineering Project Management: Major issues of Software Engineering, Functions and activities of Management, planning, organizing, staffing, directing and controlling a software Engineering Project.

UNIT II Project Evaluation: Strategic Assessment, technical Assessment, Cost-benefit analysis, cash flow forecasting, cost-benefit evaluation techniques, Risk evaluation. Selection of an appropriate project approach: Choosing Technologies, technical plan contents list, choice of process models, structure versus speed of delivery, The Waterfall model, The V-process model, the spiral model, Software prototyping, other ways of categorizing software prototypes. Controlling changes during prototyping, incremental delivery, dynamic systems development method, Extreme programming Managing iterative processes, selecting the most appropriate process model.

UNIT III Software Effort Estimation: Problems with over and under estimates, The basis for Software estimating, Software effort estimation techniques, expert judgment, estimating by analogy, Albrecht function point analysis, function points Mark II, Object points, a procedural code –oriented approach, COCOMO: A parametric model.

UNIT IV Activity planning: The objectives of activity planning, When to plan, Project Schedules, Projects and activities, Sequencing and scheduling activities, Network planning models, Formulating a network model, Adding the time dimension, The forward pass, The backward pass, Identifying the critical path, Activity float, Shortening the project duration, Identifying critical activities, Activity-on-arrow networks. Risk Management: The nature of risk, types of risks, Managing Risk, Hazard Identification, Hazard Analysis, Risk planning and control, Evaluating risks to the schedule.

UNIT V Monitoring and control: Creating the framework, Collecting the data, Visualizing progress, Cost monitoring, Earned value, prioritizing monitoring, Getting the project back to target, change control Managing Contracts: Types of contracts, Stages in contract placement, Typical terms of a contract, Contract Management, acceptance. Managing people and organizing teams: Understanding behavior, Organizational behavior: a background, Selecting the right person for the job, instruction in the best methods, Motivation, The Oldham-Hackman job characteristics Model, Working in groups, Becoming a team, Decision making, Leadership, Organizational structures, Stress, Health and Safety.

TEXTBOOKS

1. Software Project Management by Bob Hughes and Mike Cotterell, Tata McGraw-Hill Edition 2004.
2. Software Engineering Project Management Edited by Richard H Thayer, Wiley-IEEE, Computer Society Press, 2004.

REFERENCES

1. Software Project Management- A unified framework by Walker Royce, Pearson Education, 2003.
2. Software Engineering-a Practitioner's approach by Roger S Pressman, Sixth Edition, Tata McGrawHill 2004.
3. Software Management By Donald J Reifer, Sixth Edition, Wiley-IEEE Computer Society Press, 2002.

****

19-381-0524 Business Analytics

Course Objective

1. To understand various data analytics applied in business domain

Learning Outcome

1. *Will be able to assess and apply appropriate machine learning techniques for data analytics in the business domain.*

UNIT I Introduction – Ubiquity of Data Opportunities, Data Science and Data Driven decision Making, Data Processing and Big Data, Data Analytic Thinking, Data Science and Data Mining. Business Problems and Data Science Solutions - Fundamental concepts, From Business Problems to Data Mining Tasks, Supervised Versus Unsupervised Methods, The Data Mining Process, Other Analytics Techniques and Technologies. Introduction to Predictive Modeling: Fundamental concepts, Models, Induction, and Prediction, Supervised Segmentation, Visualizing Segmentations, Trees as Sets of Rules, Probability Estimation. Fitting a Model to Data - Fundamental concepts, Classification via Mathematical Functions, Regression via Mathematical Functions, Class Probability Estimation and Logistic "Regression", Nonlinear Functions, Support Vector Machines, and Neural Networks.

UNIT II Overfitting - Fundamental concepts: Generalization, Overfitting, Overfitting Examined, From Holdout Evaluation to Cross-Validation, Learning Curves, Overfitting Avoidance and Complexity Control. Similarity, Neighbors, and Clusters - Fundamental concepts, Similarity and Distance, Nearest-Neighbor Reasoning, Important Technical Details Relating to Similarities and Neighbors, Clustering - Hierarchical Clustering, Clustering Around Centroids, Understanding the Results of Clustering. Solving a Business Problem versus Data Exploration

UNIT III Decision Analytic Thinking I: Fundamental concepts, Evaluating Classifiers, Generalizing Beyond Classification, A Key Analytical Framework: Expected Value, Evaluation, Baseline Performance, and Implications for Investments in Data, Visualizing Model Performance - Fundamental concepts, Ranking Instead of Classifying, Profit Curves, ROC Graphs and Curves, The Area Under the ROC Curve (AUC), Cumulative Response and Lift Curves. Evidence and Probabilities - Fundamental concepts, Combining Evidence, Applying Bayes' Rule to Data Science, A Model of Evidence ``Lift''.

UNIT IV Representing and Mining Text - Fundamental concepts, Representation – Bag of Words, Term Frequency, Measuring Sparseness: Inverse Document Frequency, Combining Them: TFIDF, The Relationship of IDF to Entropy; Beyond Bag of Words – N-gram Sequences, Named Entity Extraction, Topic Models. Decision Analytic Thinking II: Fundamental concept, Targeting the Best Prospects for a Charity Mailing – The Expected Value Framework: Decomposing the Business Problem and Recomposing the Solution Pieces, A Brief Digression on Selection Bias.

UNIT V Other Data Science Tasks and Techniques - Fundamental concepts, Co-occurrences and Associations, Profiling, Link Prediction and Social Recommendation, Data Reduction, Latent Information, and Movie Recommendation, Bias, Variance, and Ensemble Methods, Data-Driven Causal Explanation.

TEXT BOOK

Foster Provost & Tom Fawcett, “Data Science for Business”, SPD, 2013 .

REFERENCE

Jared P Lander, “R for Everyone: Advanced Analytics and Graphics”, Addison-Wesley Data & Analytics Series, 2013.

****

19-381-0525 Software Quality

Course Objective

1. Impart students the knowledge and learning about software quality and software testing.
2. Develop a broad understanding of SQA processes from planning until execution.

Learning Outcome

1. Student will learn in detail about various quality assurance models
2. Understand the audit and assessment procedures to achieve quality.
3. Perform various testing techniques to improve software quality
4. Conduct formal inspections, record and evaluate results of inspections

UNIT I Software Quality, Software Quality Factors, Quality Frameworks and ISO-9126, Quality Assurance, QA Activities in Software Processes, Components of SQA system, Verification and Validation Perspectives of QA, SQA Activities, Software Reviews – Cost impact of Software Defects, Defect Amplification and removal. Formal Technical reviews- The review meeting, Review reporting and Record keeping, Review guidelines, Sample-driven reviews. Formal Approaches to SQA, Software Reliability – Measures of Reliability and Availability, Software Safety. The SQA plan.

UNIT II Need of Testing, Fundamentals of Testing Process, Principles of Testing, V&V Techniques, ISO/IEC/IEEE Software Testing Standards ,Testing Techniques, Functional Testing, Boundary Value Analysis, Decision Table Based Testing, Structural Testing, Path Testing, Dataflow Testing, Mutation Testing, Slice based Testing, Domain Testing, Object Oriented Testing, Testing Web Applications, Agile Testing, Scrum Testing, Mobile Application Testing, Static and Dynamic Testing Tools, Test Metrics and test reports.

UNIT III Defect Metrics, Metrics for Software Maintenance, Classification of Software Metrics, Requirements related metrics, Measurements and Process Improvement, Measurement principles, Identifying appropriate Measures and Metrics for Projects, Metrics implementation in projects, Earned Value Analysis, Issues in Software Measurements and Metrics program implementation, Object- Oriented Metrics: An Overview.

UNIT IV Quality Engineering Activities and Process, Quality Planning Goal Setting and Strategy Formation, Quality Assessment and Improvement, Quality Assurance beyond testing, Defect Prevention and Process Improvement, Fault Tolerance, Failure Containment, Comparing Quality Assurance Techniques and Activities.

UNIT V Feedback Loop and Activities for Quantifiable Quality Improvement, Quality Models and Measurements, Risk Identification for Quantifiable Quality Improvement, Software Reliability Engineering. Software CMM and other process improvement models

TEXT BOOKS

1. Jeff Tian, Software Quality Engineering: Testing, Quality Assurance, and Quantifiable Improvement, Wiley-IEEE Computer Society Press Edition First Edition, 2005.
2. Nina S Godbole, ‘Software Quality Assurance: principles and practice’, Narosa Publishing House, 2004 Edition.

REFERENCES

1 Roger S. Pressman, ‘Software Engineering: a practitioner’s approach 6th Ed. (International Edition, 2005) Tata McGraw-Hill.

2 Alka Jarvis and Vern Crandall, ‘Inroads to software quality: how to guide and toolkit’, Prentice-Hall PTR, 1997

3 Pankaj Jalota, ‘Software Engineering principles’, Narosa Publishing House, 2000.

4 Richard Fairley, ‘Software Engineering concepts’ , Tata McGraw-Hill , 2001.

5 Software Project Management – collection of white papers – foreword by Richard H Thayer – Wiley Student Edition, 2000.

****

19-381-0526 Design Patterns

Course Objective

1. Understand the importance of Object- Oriented Approaches.
2. To understand description of patterns at the micro- architectural level.

Learning Outcome

1. Students will be able to identify and implement various design patterns suitable for the software development.

UNIT I Describing Design Patterns - MVC – How DPs solve Design Problems – How to Select a DP – How to use a DP, Design patterns – UML Diagrams, Software patterns – Overview of UML. Case study – Software life cycle.

UNIT II Creational (Abstract, Builder, Factory, Prototype, Singleton)– Structural (Adapter, Bridge, Composite, Decorator, Façade, Flyweight, Proxy)

UNIT III Behavioral (Chain of Responsibility, Command, Interpreter, Iterator, Mediator, Memento, observer, State, Strategy, Template, Visitor)

UNIT IV The JFC, or Swing – Writing a simple JFC Program – Radio Buttons and Toolbars – Menus and Actions – The JList class – The JTable Class – The JTree Class.

UNIT V Expert-Creator-Low Coupling-High Cohesion-Contrller- Polymorphism-Pure Fabrication – Indirection – Don't talk to Strangers.

TEXT BOOKS

1. Elements of Resuable Object-Oriented Software(Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, 2003.

REFERENCES

1. Java Design Patterns – A Tutorial (James W. Cooper), 2000.

19-381-0531 System Software and Compiler Design

Course Objective

1. To get an idea of different system software.
2. Provide a thorough understanding of the internals of Compiler Design.

Learning Outcome

1. Students are able to understand phases of compilation.
2. Use and compare different compilers.

UNIT I Introduction to System software- Assemblers-Two pass assembler: algorithm and data structures- Loaders-absolute loader, bootstrap loader, relocative loaders-Linkers- Interpreters- Macro processors

UNIT II Introduction to compilers – Analysis of the source program, Phases of a compiler, Grouping of phases-Lexical Analysis: The role of Lexical Analyzer, Input Buffering, Specification of Tokens using Regular Expressions, Review of Finite Automata, Recognition of Tokens.

UNIT III Syntax Analysis: Review of Context-Free Grammars – Derivation trees and Parse Trees, Ambiguity. Top-Down Parsing: Recursive Descent parsing, Predictive parsing, LL(1) Grammars.

UNIT IV Bottom-Up Parsing: Shift Reduce parsing LR parsing – Constructing SLR parsing tables, Constructing, Canonical LR parsing tables and Constructing LALR parsing tables. Type Checking: Type systems, Specification of a simple type checker.

UNIT V Intermediate code generation: Intermediate languages – Graphical representations, Three-Address code, Quadruples, Triples. Assignment statements, Boolean expressions. Code optimization : loop optimization and global optimization, sources of sample code generation.

TEXT BOOKS

1. Leland L Beck, D.Manjula: “System software:An introduction to system programming” 3rd edition, 7th impression, Pearson, 2010
2. Aho A. Ravi Sethi and D Ullman. “Compilers – Principles Techniques and Tools”, Pearson Education,2014.

REFERENCES

1. Modern Complier Implementation In C, Andrew.W.Appel, First Edition, Cambridge University Press,2004
2. Keith D Cooper, Linda Torezon, “Engineering a Compiler”, 2nd Edition,Elsevier,2012

****

19-381-0532 Network Security and Wireless Security

Course Objective

1. Understand different security implementations at network layer and transport layer.
2. Learn the protocols for implementing wireless security.

Learning Outcome

1. Understand wireless networking.
2. Impact of wireless network on security.
3. Download and install an e-mail and file security software, PGP.

UNIT I Model for network Security : Electronic Mail Security- PEM, Pretty Good Privacy, S/MIME IP and Web Security, Secure Sockets and Transport Layer,

UNIT II Buffer Overflow, TCP Session Hijacking, ARP Attacks, Route Table Modification, UDP Hijacking, Man in the Middle attack.

UNIT III Wireless Application Protocol, How WAP works, Security status of WAP – Viruses, Authorization, Non Repudiation, Authentication, Secure Sessions. WAP Security Architecture, Marginal Security – Wireless access to the internet, Wireless middleware.

UNIT IV Block Ciphers and Stream Ciphers, RC4 Encryption. WEP Security, Cracking WEP, WPA.

UNIT V RFID – Application types, RFID System components – Tag, Reader, Middleware, Attaching RFID tags, Tracking the movements of items. Architecture Guide lines. Wireless Hacking Techniques, Wireless attacks, mobile Security, Securing Wi- Fi.

TEXT BOOKS

1. Tyler Wrightson, “Hacking exposed Wireless Network Security A Beginner's Guide “, 1st Edition, Tata Mc – Graw Hill, 2012.
2. Bill Glover, Himanshu Bhatt, “RFID Essentials”, 2nd Edition, Oreilly, 2007.

REFERENCES

1. William Stallings ,”Cryptography and Network Security Principles and Practice “, Sixth Edition , Pearson, 2014.
2. Shiva Sukula, RFID essentials , innovations and beyond, ESS, 2012.

****

19-381-0533 Wireless Sensor Networks

Course objective

1. To introduce concepts of Wireless Sensor Networks
2. To provide the fundamentals of the communication in Wireless Sensor Networks.
3. To learn about fundamentals of security in WSN.

Learning Outcome

1. Will be able to understand fundamentals of the communication systems.

UNIT I Fundamentals of WSN: Overview And Introduction, Applications, Unique Constraints and challenges, Mobile ad hoc networks and wireless sensor networks, Enabling technologies for wireless sensor networks, Sensor node hardware and network architecture-Single and multi node architecture, Hardware components, Operating systems and execution environments, TinyOS and nesC. Network architecture- Optimization goals and figures of merit, Design principles for WSNs, Service interfaces of WSNs, Gateway concepts.

UNIT II Communication Protocols: Physical layer and transceiver design considerations in WSNs, MAC protocols –fundamentals of wireless MAC protocols, low duty cycle protocols and wakeup concepts, contention-based protocols, Schedule-based protocols, The IEEE 802.15.4 MAC protocol. Link-layer protocols-Error control, Framing, Link management.

UNIT III Time synchronization in WSN: Introduction, Protocols based on sender/receiver synchronization, Protocols based on receiver/receiver synchronization. Localization and positioning: Properties of localization and positioning procedures, Single-hop localization, Positioning in multihop environments.

UNIT IV Routing protocol: Gossiping and agent-based unicast forwarding, Energy-efficient unicast, Broadcast and multicast, Geographic routing, mobile nodes. Data-centric and content-based networking: Introduction, Data-centric routing, Data aggregation, Data-centric storage.

UNIT V Security In WSN: Fundamentals, Challenges of Privacy and Security in Wireless Sensor Networks, Security Attacks in Sensor Networks, Protocols and Mechanisms for Security.

TEXT BOOKS/REFERENCES:

1. Holger Karl and Andreas Willing, "Protocols and Architectures for Wireless Sensor Networks", John Wiley and Sons, 2005. 10

2. Waltenegus Dargie and Christian Poellabauer, "Fundamentals of Wireless Sensor Networks:
Theory and Practise", First Edition, John Wiley and Sons, 2010

****

19-381-0534 Software Defined Networks

Course Objectives

1. To introduce concepts of Software Defined Networks.
2. To learn about fundamentals of security in SDN.

Learning Outcome

1. Students will be able to apply various Software Defined Networks.

UNIT I Introduction – History of Software Defined Networking (SDN), Modern Data Center, Traditional Switch Architecture, Evolution of SDN ,How SDN Works – Centralized and Distributed Control and Date Planes

UNIT II **Open flow & SDN controllers -**Open Flow Specification, Drawbacks of Open SDN, SDN via APIs, SDN via Hypervisor-Based Overlays – SDN via Opening up the Device – SDN controllers and application models

UNIT III **Data centers-** Data Center Demands, Tunneling Technologies for the Data Center, Path Technologies in the Data Center, Ethernet Fabrics in the Data Center, SDN Use Cases in the Data Center, Comparison of Open SDN, Overlays, and APIs

UNIT IV **Network function virtualization** (NFV)-Definition, standards , OPNFV,SDN v/s NFV ,In-Line Network Functions. **SDN security**:-FRESCO, *FortNOX*

UNIT V **SDN applications**- Application Types , Controller Considerations ,Network Device Considerations, Creating Network Virtualization Tunnels, Offloading Flows in the Data Center, Access Control for the Campus, Traffic Engineering for Service Providers. Programming SDN networks- Northbound Application Programming Interface, Current Languages and Tools

TEXTBOOKS/ REFERENCES

1. Software Defined Networks: Paul Goransson, Chuck Black, Timothy Culver 2nd Edition, 2014.
2. Thomas D. Nadeau, Ken Gray, ―SDN: Software Defined Networks, O'Reilly Media, 2013.
3. Network Innovation through OpenFlow and SDN: Principles and Design, Edited by Fei Hu, CRC Press, 2014.

****

19-381-0535 Security Threats and Vulnerabilities

Course Objective

1. Analyze existing authentication and key agreement protocols, identify the weaknesses of these protocols.
2. Perform analysis on products to identify vulnerabilities.

Learning Outcome

1. Identify computer and network security threats, classify the threats and develop a security model to prevent, detect and recover from the attacks.

UNIT I OSI Security architecture – Security Attacks, Services and Mechanisms. Model for network Security – Open and Closed Security models. Security Goals, Identifying potential risks to Network security Simplified DES, Simplified AES.

UNIT II Threat, Types of Threat, Vulnerabilities, Controls and Counter measures, . Attacks examples – Reconnaissance attack, Access attacks, Masquerading, IP Spoofing, Denial of Service attack, Distributed Denial of Service, Salami Techniques, Botnets, Onion Routing.

UNIT III Entity authentication – Password, Challenge/ Response, Zero Knowledge Protocol. Biometrics, Introduction to identity based Public Key Cryptography, Oblivious transfers. Legal and Ethical issues, Cybercrime and computer crime, Intellectual Property Management – Copyright, Patent and Trade Secret, Digital Right Management.

UNIT IV Intruders, Hackers, Intrusion Techniques – Password Guessing, Password Capture. Intrusion Detection Systems- Types, Honeypots. Malicious Softwares - Virus and Types of Virus, Countermeasures, Worms, Trojan Horse , Trap door, Logic Bomb. Firewalls – Characteristics, Types – Packet filters, Application level gateways, Circuit level gateways, Baston host, Personal firewalls.

UNIT V Email attacks – SPAM , Electronic Mail Security - PGP, S/MIME, Smart cards, SET, Clipper, ISO Authentication framework.

TEXT BOOKS

1. William Stallings ,"Cryptography and Network Security Principles and Practice ", Sixth Edition , Pearson, 2014.
2. Behrouz A Forouzan, "Cryptography & Network Security , Second Edition, McGraw-Hill, 2007.

REFERENCE

Charles P Fleeger, Shari Lawerence P Fleeger, Jonathan Margulies, "Security in Computing" Edition, Prentice Hall, 2015.

****

19-381-0536 Blockchain Technology

Course Objective

1. To understand the concepts of a public digital ledger to share information in a trustworthy and secure way.
2. To discuss and cover both the conceptual as well as application aspects of Blockchain.

Learning outcome

1. The course will be providing a foundation of Blockchain with related technologies.
2. Student will be able to understand the applications and implement in other domains like business process management, smart contracts, IoT etc.

UNIT I Introduction to Blockchain, Blockchain Data structure, Hash chain, distributed database, Index structure, Transactions, Asymmetric-Key Cryptography, Addresses and Address Derivation, Private Key Storage, Ledgers, Blocks, Chaining Blocks.

UNIT II Consensus and multiparty agreements: protocols, Proof of Work, Proof of Stake, Delegated Proof of Stake, Proof of Elapsed Time, Deposit based consensus, Proof of importance.
Federated consensus or federated Byzantine consensus, Reputation-based mechanisms, Practical Byzantine Fault Tolerance.

UNIT III Blockchain implementation. Forking-Soft Fork, Hard Forks. Smart contract programing. Blockchain Platforms – Cryptocurrencies - Bitcoin, Litecoin, Ethereum, Ripple.

UNIT IV Hyperledger, Ethereum. Decomposing the consensus process , Hyperledger fabric components , Chaincode Design and Implementation.

UNIT V Blockchain-Outside of Currencies. IPFS protocol and Blockchain. Blockchain Concurrency and scalability. Network models and timing assumptions.

TEXT BOOK

1. Andreas M Antonopoulos , "Mastering Bitcoin: Unlocking digital crypto currencies", ORELLY,2015.
2. Melanie, "Blockchain: Blue print for new economy", ORELLY,2015.
3. https://www.ibm.com/blockchain/in-en/hyperledger.html.

REFERENCES

1. Don Tapscott, "Block chain and Crypto currency", 2016. Draft NISTIR 8202, Blockchain Technology Overview - NIST CSRC, 2018.
2. Imran Bashir, Mastering Blockchain, 2017.
3. Andreas M. Antonopoulos, Mastering Bitcoin - Programming the Open Blockchain, O'Reilly Media, Inc., 2017

4. Ethereum Programming, Alex Leverington, Packt Publishing Limited, 2017

5. Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction. A. Narayanan, J. Bonneau, E. Felten, A. Miller, S. Goldfeder, Princeton University Press, 2016

6. The Science of the Blockchain, Roger Wattenhofer, CreateSpace Independent Publishing Platform, 2016

7. Melanie Swan, Blockchain - Blueprint for a new economy, O'Reilly Media, Inc., 2015.

8. Abhijit Das and VeniMadhavan C. E., Public-Key Cryptography: Theory and Practice: Theory and Practice, Pearson Education India, 2009.

****

19-381-0541 Big Data Analytics

Course Objective

1. Understand the fundamental concepts of Big Data Analytics and identify its Use cases
2. Understand the Apache Hadoop platform and learn its application in different domains
3. Acquire the requisite skills in Computer Science and Data Engineering needed by the industry for the development of Big Data Platforms and Applications

Learning Outcome

1. Select and implement appropriate data structures to solve big data problems and also write Map and Reduce codes for distributed processing of data.
2. Perform batch processing operations on Big Data on their own computer
3. They will also be able to retrieve and store data in HDFS & HBase
4. Ingest structured and unstructured data into big data processing systems and perform data transformations using Hive.
5. Build real time data pipelines using Apache Spark and also process the data stream for subsequent analysis
6. Perform analytics on the big data using Spark MLLib and build dashboards for visualizing the results.

UNIT I Introduction to Big Data: Big Data – Introduction, Structuring Big Data, Elements of Big data, Big data analytics, Big data applications. Big Data in business context, Technologies for handling big data – Distributed and Parallel computing for Big Data, Data Models, Computing Models, Introducing Hadoop – HDFS and MapReduce.

UNIT II Hadoop Ecosystem: Hadoop EcoSystem, Hadoop Distributed file system, HDFS architecture, MapReduce framework, Techniques to Optimize MapReduce, Uses of MapReduce, Processing Data with MapReduce, MapReduce execution and Implemention, YARN Architecture, Advantages of YARN, Working of YARN, YARN Schedulers, Configurations, Commands, Containers.

UNIT III Apache Spark: Spark Architecture, Resilient Distributed Datasets (RDD) – Persistance and Cashing , In-memory distributed processing using Apache Spark, Components of the Spark unified stack, Spark SQL, Spark Streaming, Spark GraphX, Spark MLLib

UNIT IV NoSQL Databases: Types NoSQL databases, Introducing HBase, Hive and Pig, Role of HBase in Big data processing, HBase Architecture, Hive Architecture and Data Model, HiveQL, SQL vs HQL, Pig Architecture and Data Model, Pig Latin

UNIT V Analytics: Machine Learning on Large Datasets using Spark MLLib, Building visualizations using Big Data, Case Studies on applications of Big Data Analytics

TEXT BOOKS

1. Authored by DT Editorial Services , "Big Data, Black Book: Covers Hadoop 2, MapReduce, Hive, YARN, Pig, R and Data Visualization WileyIndia, 2016.
2. Mike Frampton, "Mastering Apache Spark", Packt Publishing, 2015

REFERENCES

1. Borislublinsky, Kevin t. Smith, AlexeyYakubovich, "Professional Hadoop Solutions", Wiley, 2015.
2. Chris Eaton,Dirkderooset al. , "Understanding Big data ", McGraw Hill, 2012.
3. Sima Acharya, Subhashini Chellappan, BIG Data and Analytics , Willey
4. Tom White, "HADOOP: The definitive Guide", O Reilly 2012.
5. Vignesh Prajapati, "Big Data Analytics with R and Hadoop", Packet Publishing 2013.
6. Kulkarni, Parag, Joshi, Sarang, Brown, Meta S. Big Data Analytics, PHI 2016
7. Andy Konwinski, Holden Karau, Matei Zaharia, and Patrick Wendell, "Learning Spark: Lightning-Fast Big Data Analysis", O Reilly, 2015.

****

19-381-0542 Natural Language Processing

Course Objective

1. Introduce basic mathematical models and methods used in NLP applications to formulate computational solutions.
2. Introduce the fundamental concepts and techniques of natural language processing (NLP).

Learning Outcome

1. Students will get familiarity with the current Natural Language Processing tasks and softwares like NLTK, ScikitLearn etc.
2. Use key concepts from NLP are to describe and analyse language

UNIT I Overview of Natural Language Processing,Natural Language Processing and Python: Under standing Natural language Processing and applications, NLTK, Corpus and Dataset, Understanding structure of sentence -Defining Context free grammar, Morphological Analysis

UNIT II Syntactic Analysis,Semantic analysis, Ambiguity resolution, Discourse integration Preprocessing - tokenization , stemming , lemmatization,Word tokenization and lemmatization

UNIT III Feature engineering and and NLP algorithms- parsers, context-free grammars,different types of parsers, POS tagging and different types of POS parsers.

UNIT IV Basic statistical features of NLP: TF-IDF, Vectoirization Encoders and Decoders, Normalization, Advanced feature engineering and NLP algorithms- Basics of Word2Vec

UNIT V Rule - Based system for NLP, Machine Learning for NLP problems, Applications of NLP - Text Summarization, Sentiment Analysis

TEXT BOOK

Thanaki J. Python Natural Language Processing. Packt Publishing Ltd; 2017.

REFERENCE

Jurafsky D. Speech & language processing. Pearson Education India; 2000.

****

19-381-0543 Digital Image Processing

Course objective

1. To have a thorough understanding of the basic image processing techniques.

Learning Outcome

1. Students will be able to apply various image processing techniques for real time applications.

UNIT I Introduction-Digital Image Fundamentals: elements of visual perception, light and electromagnetic spectrum, image sensing and acquisition, image sampling and quantization, some basic relationship between pixels. Intensity Transformations: Basics of intensity transformations, some basic intensity transformation functions, histogram processing.

UNIT II Filtering-Spatial Filtering: fundamentals of spatial filtering, smoothing and sharpening filters. Frequency domain Filtering: Background, preliminary concepts, sampling, Fourier transforms and DFT, 2-D DFT and properties, frequency domain filtering, low pass filters, high pass filters, implementation.

UNIT III Image restoration and Reconstruction- Noise models, restoration in the presence of noise, linear-positive invariant degradations, inverse filtering, Wiener filtering, constrained least square filtering, geometric mean filter.

UNIT IV Image Compression-fundamentals, basic compression methods. Morphological Image Processing: preliminaries, erosion and dilation, opening and closing, basic morphological algorithms.

UNIT V Image Segmentation-fundamentals, point, line and edge detection, thresholding, region based segmentation, use of motion in segmentation.

TEXT BOOK

1. Digital Image Processing, by Rafael C. Gonzalez & Richard E. Woods, 3rd edition, PHI 2008

REFERENCE

1. Fundamentals of Digital Image Processing, by Anil K. Jain, Prentice Hall, 1995. 2. Digital Image Processing, by William K. Pratt, John Wiley & Sons Inc., 3rd edition, 2001.

****

19-381-0544 Deep Learning

Course objective

1. To learn the concepts of deep learning and their applications

Learning Outcome

1. Student will be able to design and implement suitable deep learning models for various problems.

UNIT I Basics-Biological Neuron, Idea of computational units, McCulloch–Pitts unit and Threshold logic, Linear Perceptron, Multilayer Perceptron, Perceptron Learning Algorithm, Linear separability. Convergence theorem for Perceptron Learning Algorithm. Deep Neural Networks: Difficulty of training deep neural networks, Greedy layer wise training. Better Training of Neural Networks: Newer optimization methods for neural networks (Adagrad, adadelta, rmsprop, adam, NAG), second order methods for training, Saddle point problem in neural networks.

UNIT II Deep Feed forward Networks- Gradient Descent, hidden units, Back propagation, Regularization for deep learning, Optimization for training deep models-Empirical Risk Minimization.

UNIT III Convolutional Neural Networks:- LeNet, AlexNet. Recurrent Neural Networks: Back propagation through time, Bidirectional RNNs, Long Short Term Memory, Gated Recurrent Units, Bidirectional LSTMs

UNIT IV Autoencoder- Generative models: Restrictive Boltzmann Machines (RBMs), Introduction to MCMC and Gibbs Sampling, gradient computations in RBMs, Deep Boltzmann Machines.

UNIT V Recent trends: Variational Auto encoders, Generative Adversarial Networks, Multi-task Deep Learning, Multi-view Deep Learning Applications: Computer Vision, NLP, Speech.

TEXT BOOK

1. Deep Learning, Ian Goodfellow and Yoshua Bengio and Aaron Courville, MIT Press, 2016.

REFERENCES

1. Neural Networks: A Systematic Introduction, Raúl Rojas, 1996

2. Pattern Recognition and Machine Learning, Christopher Bishop, 2007

****

19-381-0545 Bioinformatics

Course Objective

1. Familiarize with fundamental concept of molecular biology,
2. Understand models of DNA and DNA mapping,
3. computational problems in molecular biology and algorithms to solve them

Learning Outcome

1. An introduction to computational aspects of computational biology and related algorithms.
2. An introduction to tools used in different aspects of molecular biology.

UNIT I Bioinformatics and Computational Biology – Scope. Central Dogma of Molecular Biology. Nucleic Acid Structure and Function. Genetic Code. Gene Expression – Transcription, Translation, Microsatellites, Minisatellites, Tandem repeats, Inverted repeats, SNPs. Biological Databases – Primary DBs, Secondary DBs, Nucleic Acid DBs, Protein DBs. Basic file formats.

UNIT II Basic Gene Statistics – Base counts, word (n-mer), frequency. Sequence Alignment – Local/Global Alignment, Pairwise Sequence Alignment, Needleman-Wunsch Algorithm. Multiple Sequence Alignment. PAM Matrix, BLOSUM. Tools: CLUSTAL W, Muscle, Tcoffee, Dotplot.

UNIT III Molecular Phylogenetic –Need & applications, DNA Barcoding, Genetic Finger Printing. Dendrogram, Cladogram; Rooted/Unrooted tree; Distance Based tree construction – UPGMA, NJ algorithm. Character Based Methods – Maximum Parsimony. Validating – Jack Knifing, Bootstrapping. Tree calibration. Tools: Phylip, NJ Plot, CLUSTAL X

UNIT IV Protein Structure- Primary, Secondary – alpha helics, beta sheets & turns, Tertiary and Quaternary structures. Protein Folding, Ramachandran Plot. Homology Modelling. CADD – Introduction to Molecular Docking. HMM-Introduction, Forward algorithm, Viterbi algorithm, Applications in bioinformatics Protein Databases, Tools: Swiss PDB, Pymol

UNIT V Systems Biology – Introduction, Protein Interaction networks, Metabolic and Signaling networks, SBML. E-Cell, Synthetic Biology- Introduction & Applications. Machine Learning in Bioinformatics

TEXT BOOK

1. Lesk, Arthur, Introduction to genomics, Oxford University Press, 2017

REFERENCES

1. Bergeron, Bryan P, Bioinformatics Computing, Prentice Hall Professional, 2003
2. Neil James, Pavel A Pevnezer, An Introduction to Bioinformatics Algorithms, MIT Press, 1st ed, 2004
3. Gibas , Cynthia, Developing bioinformatics computer skills, O'reilly 2003

****

19-381-0546 Internet of Things

Course Objective

1. To learn emerging technology
2. Familiarize tiny sensors supporting IoT

Learning Outcome

1. Design an IoT based system
2. Understand how an object evolved to a smart object.

UNIT I Introduction to IoT-Physical Design of IoT, Logical Design of IoT, IoT Levels, Deployment templates, IoT enabling technologies. Domain Specific IoTs - NETCONF-YANG- IoT Platforms Design Methodology.

UNIT II Networking IoT and Communication protocols – Link Layer, Network Layer, Transport layer, Application Layer.Sensor Networks and Machine to Machine communication – Differences and Similarities between M2M and IoT, Software defined networking , Network function virtualization.

UNIT III M2M and WSN Protocols – SCADA and RFID Protocols – Unified Data Standards – Protocols – IEEE 802.15.4 – BACNet Protocol –Modbus– Zigbee Architecture – Network layer –LowPAN - CoAP – Security.

UNIT IV Building IoT - RASPERRY PI- IoT Systems - Logical Design using Python – IoT Physical Devices & Endpoints - IoT Device -Building blocks -Raspberry Pi-Board - Linux on Raspberry Pi- Raspberry Pi Interfaces -Programming Raspberry Pi with Python - Other IoT Platforms - Arduino. Introduction to Arduino Programming, Integration of Sensors and Actuators with Arduino, Implementation of IoT.

UNIT V Big data platform for the internet of things, - Improving Data and Service Interoperability with Structure, Compliance, Conformance and Context Awareness: interoperability problem in the IoT context- Big Data Management Systems for the Exploitation of Pervasive Environments - Big Data challenges and requirements coming from different IoT based applications. Fog Computing: A Platform for Internet of Things and Analytics

TEXT BOOK/REFERENCES

1. Adrian McEwen and Hakim Cassimally, “Designing the Internet of Things”, John Wiley &Sons, 2013.
2. Cuno Pfister, “Getting Started with the Internet of Things: Connecting Sensors and Microcontrollers to the Cloud”, Maker Media, 2011.
3. Pethuru Raj and Anupama C. Raman , “The Internet of Things: Enabling Technologies, Platforms, and Use Cases", (CRC Press).
4. Arshdeep Bahga and Vijay Madisetti, "Internet of Things: A Hands-on Approach", by (Universities Press)2015.

****

19-381-0547 Data Science and Analytics

Course Objective

1. Provide the knowledge and expertise to become a proficient data scientist.
2. Demonstrate and understanding of statistics and machine learning concepts that are vital for data science.
3. Produce Python code to statistically analyse a dataset.
4. Critically evaluate data visualisations based on their design and use for communicating stories from data.

Learning outcome

1. Explain how data is collected, managed and stored for data science;
2. Understand the key concepts in data science, including their real-world applications and the toolkit used by data scientists;
3. To mathematically analyse various machine learning approaches and paradigms.
4. Students are able to analyse data by Machine learning algorithms.
5. Visualize data and enable communication with data

UNIT I Introduction to core concepts and technologies: Introduction, Terminology, data science process, data science toolkit, Types of data, Example applications.

Data collection and management: Introduction, Sources of data, Data collection and APIs, Exploring and fixing data, Data storage and management, Using multiple data sources

UNIT II Data analysis: Introduction, Terminology and concepts, Introduction to statistics, Central tendencies and distributions, Variance, Distribution properties and arithmetic, Samples/CLT, Basic machine learning algorithms, Linear regression, SVM, Naive Bayes.

UNIT III Data visualisation: Introduction, Types of data visualisation, Data for visualisation: Data types, Data encodings, Retinal variables, Mapping variables to encodings, Visual encodings.

UNIT IV Applications of Data Science, Technologies for visualisation, Bokeh (Python)

Recent trends in various data collection and analysis techniques, various visualization techniques, application development methods of used in data science.

UNIT V Unsupervised Learning: Clustering: K-means/Kernel K-means, Dimensionality Reduction: PCA and kernel PCA. Sparse Modeling and Estimation, Modeling Sequence/Time-Series data, Deep Learning.

<u>REFERENCES</u>

1. Cathy O'Neil and Rachel Schutt. Doing Data Science, Straight Talk From The Frontline. O'Reilly, 2013
2. Jure Leskovek, Anand Rajaraman and Jeffrey Ullman. Mining of Massive Datasets. v2.1, Cambridge University Press, 2016.
3. Kevin Murphy, Machine Learning: A Probabilistic Perspective, MIT Press, 2012
4. Trevor Hastie, Robert Tibshirani, Jerome Friedman, The Elements of Statistical Learning, Springer 2009 (freely available online)
5. Christopher Bishop, Pattern Recognition and Machine Learning, Springer, 2007.*************