Gujarat University

**Course Name: Advanced Networking & Network Administration**

**Course Code: MCA 311**

**Objectives:**

- Develop strong analysis, design, implementation, testing and troubleshooting skills in students regarding TCP/IP based networks and services as relevant to the computer networking needs of the IT industry
- Establish a strong conceptual foundation of the TCP/IP protocol stack, services and related tools/technologies so as to facilitate the development of the above mentioned skills
- Give in-depth understanding of all the commonly used protocols used in the TCP/IP protocol stack
- Design and implement customized TCP/IP based application layer services
- To familiarize with security and performance issues in TCP/IP networks
- Give practical exposure through rigorous laboratory exercises involving programming, network administration assignments like server installation/configuration/troubleshooting, etc. as well as exploratory and research assignments to fulfill the above objectives in a realistic and a relevant way especially from industry and research point of view
- Create a strong conceptual foundation for TCP/IP which can be leveraged for studying advanced topics in TCP/IP, dissertation and System Development Project and further studies
- Create a strong conceptual foundation and offer maximum possible development of required theoretical and practical skills for students aspiring to make a career in Computer Networking like Network Designer, Network administrator, etc.

**Prerequisites:**

- Digital Data communication concepts
- An overview of the layered architecture as per OSI and TCP/IP model
- Overview of functionality of all layers in the OSI and TCP/IP model
- Differences between OSI and TCP/IP model
- Concepts of LAN, WAN, Internet, Ethernet
- General concepts in routing and basic routing algorithms like Djkistra's shortest path, distance vector routing, link state routing, etc.
- Overview of popular application layer services like DNS, HTTP, SMTP, etc.
- Knowledge of core Java (for practical)

*(All above pre-requisites are covered in the subject MCA SEM IV M-223 Fundamentals of Networking and MCA SEM II M-212 Java Programming)*

## Contents:

1. **Classful Internet Addresses, Mapping Internet Addresses to Physical Addresses (ARP), Internet Protocol: Connectionless Datagram Delivery (IPv4)   [20%]**
   Introduction, Universal identifiers, IP addresses and network connections, original classful addressing scheme, special purpose IP addresses like directed broadcast and network broadcast, Limited broadcast, Loopback address, Subnet and Classless extensions, Weaknesses in Internet addressing, Dotted decimal notation, Network byte order, Special address conventions.
   Concept of physical address, Address resolution problem, Relationship between network address and physical address, Two types of physical addresses, Resolution through direct mapping, Resolution through dynamic binding, ARP cache Timeout, ARP refinements, Relationship with other protocols, ARP implementation, ARP Encapsulation and identification, ARP protocol format, Automatic ARP cache revalidation
   Concept and reason for connectionless and best effort delivery system at network layer, purpose of Internet protocol, IPv4 datagram format , interpretation and significance of each header fields, IP options.

2. **Internet Protocol: Forwarding IP Datagrams, Error And Control Messages (ICMP), Classless And Subnet Address Extensions (CIDR), User Datagram Protocol (UDP)   [20%]**
   Introduction, Forwarding in the Internet, Indirect and Direct delivery, Table driven IP forwarding, Next hop forwarding, Default routes, Host specific routes, IP forwarding algorithm, Forwarding with IP addresses, Handling incoming datagrams, Establishing routing tables.
   Introduction of ICMP, Need for a controlling protocol, Error Reporting versus Error Correction, ICMP message delivery, ICMP message format, Ping, Formats of different type of messages like Echo(Request and Reply), Congestion , Unreachable Destinations, Source Quench, Router advertisement and Solicitation, etc..
   Introduction of CIDR, Minimizing Network Numbers, Proxy ARP, Subnet Addressing, Flexibility in subnetting, Variable length subnets, Subnet masks, Unified Forwarding Algorithm, Broadcasting to Subnets, Anonymous point to point networks, Classless Addressing and Supernetting, CIDR address blocks and Bit masks, Data Structures and Algorithms for classless lookup, Searching by mask length, Binary trie structures, Longest match and mixture of route types, PATRICIA and level compressed tries.
   Introduction of UDP, Need for UDP, UDP message format, UDP Pseudo header, UDP encapsulation and protocol layering, Layering and UDP Checksum computation, UDP multiplexing, demultiplexing, and role of ports in multiplexing and demultiplexing, UDP applications, port numbers of well known UDP based applications

3. **Reliable Stream Transport Service (TCP), Private Network Interconnection (NAT, VPN), Bootstrap And Auto configuration (DHCP)   [20%]**
   Introduction, Need for stream delivery, Properties of reliable delivery service, Providing reliability, Concept of sliding windows, Ports, connections and endpoints, Active and Passive opens, Segments, Streams and sequence numbers, Variable

window size and flow control, TCP segment format, Out of band data, TCP options, Acknowledgment, Retransmission and timeouts, Accurate measurement of RTT, Karn's algorithm and timer backoff, Explicit feedback mechanism, Congestion control, TCP connection establishment and termination, Dealing with silly window syndrome.

Introduction to VPN, Private and hybrid networks, VPN addressing and routing, Extending VPN technology to individual hosts, VPN with private addresses,

Introduction to NAT, NAT translation table creation, multi-address NAT, port mapped NAT, Interaction between NAT and ICMP, Interaction between NAT and Applications, NAT in presence of fragmentation, Conceptual address domains, Introduction to slirp and IPtables.

Introduction to DHCP, History of bootstrapping, Using IP to find IP Address, DHCP Retransmission Policy, DHCP Message format, Need for dynamic configuration, DHCP Lease concept, Multiple addresses and Relays, Lease renewal States, Address acquisition states, Early lease termination, DHCP options and message type, Options overload, DHCP and DNS

4. **The Domain Name System (DNS), Remote Login And Desktop (TELNET, SSH), File Transfer And Access (FTP, TFTP, NFS), Electronic Mail (SMTP, POP, IMAP, MIME) [20%]**
Need for DNS, Flat versus hierarchical namespace, Centralized versus distributed Names database, Delegation of authority for names, Subset authority, Internet domain Names, Top-level domains, Mapping domain names to addresses, Domain Name Resolution, Efficient translation, Caching, DNS message format, Compression, Inverse mappings, Pointer queries, DNS resource records, Dynamic DNS, DNSSec.
Introduction, Remote interactive computing, Telnet protocol, Accommodating Heterogeneity, Client side and server side control commands, Telnet options and Options negotiation, SSH.
Different ways of sharing a file, Features, Process model, TCP Port numbers, Data connection and control connection, User's view of FTP, Anonymous FTP, Secure FTP, TFTP, NFS, RPC, XDR.
Introduction to E-mail protocols., Mailboxes, Names and Aliases, Alias expansion and mail forwarding, SMTP, POP, IMAP, MIME Extensions for non ASCII data, MIME Multipart messages.

5. **World Wide Web (HTTP), Internet Security And Firewall Design (IPsec, SSL), A Next Generation IP (IPv6) [20%]**
Importance of Web, Architectural components, URL, HTTP, HTTP methods, HTTP error messages, Connection types, Significance of different HTTP header fields, Negotiation, Conditional requests, Proxy servers, Caching, HTTP security and E- Commerce.
Introduction to IPsec and SSL, Need for Security, IPSec, AH, SA, ESP, Authentication and mutable header fields, Tunneling, Required security algorithms, SSL and TLS, Firewalls, Firewall implementation issues, Packet filtering, Stateful firewalls, proxy servers, Monitoring and logging,
Introduction to IPv6, Need for new IP protocol, IPv6 features, IPv6 base header format.

## Main Reference Book(s):

1) Douglas E. Comer, "Internetworking with TCP/IP - (Vol. 1.) Principles, Protocols, and Architecture", 5th Edition, Prentice Hall of India (PHI).

## Suggested Additional Reading Book(s):

1) Behrouz A. Forouzan, "TCP/IP protocol suite", Tata McGraw-Hill (TMH).
2) W. Richard Stevens, "TCP/IP- Illustrated, Vol. 1 (The Protocols)", Pearson Education Asia Publishers.
3) Karanjit S. Siyan, "Inside TCP/IP", Techmedia Publishers.
4) Elliote Rusty Harold, "Java Network Programming", Oreilly Publishers.
5) Scott Oaks & Henry Wong, "Java Threads", Oreilly Publishers.

## Chapter wise coverage from the main reference Books:

| Book | Chapters |
|------|----------|
| 1 | 4- 9, 11,12, 19, 22- 27, 30, 31 (31.1 – 31.9) |

## Accomplishment of the student after completing the course:

At the end of the work student will be able to

- Have thorough understanding of TCP/IP based systems, services and related tools and technologies
- Be fluent in design and developing Java based TCP/IP socket based networking solutions
- Effectively use available OS commands/utilities as well as popular third party tools for TCP/IP networking depending upon the needs
- Be geared to adapt to more sophisticate networking related packages in Java and hence develop relatively complex applications more reliably and faster.

## Practical/Term-work

### 1. OS commands and utilities for TCP/IP

TCP/IP commands and utilities available in any of the following OS (Microsoft Windows 2000 server/ Red Hat Linux/Fedora) A suggestive list for Windows is given below:

- ipconfig
- ping
- arp
- nslookup
- netstat
- tracert
- route print
- telnet
- ftp

Some of the above commands may vary depending upon the OS.

**Students may contact different Internet services like FTP, HTTP, SMTP. Daytime, echo, quote, etc. through telnet giving appropriate host name and well known port no. and observe the interaction on command prompt**

### 2. Server Administration & Client Configuration

- Installation, configuration and  operation of :
  1) Web Server (Preferably free and open source like Apache)
  2) FTP Server ( Preferably free and open source like FileZilla)
- Configuring a new TCP/IP client (preferably windows or Linux)
  Configuring IP address/s, default gateway, DNS server addresses, DNS suffix, subnet mask, proxy information, etc.
- Installation, configuration and operation of non web based email clients like Outlook Express.

### 3. Socket Programming in Java

Developing TCP and UDP based servers (both iterative and
Concurrent using multi-threading). Common Services like echo, daytime, chat, quote of the day, HTTP, DNS, ARP, EMAIL etc. may be implemented trivially. For ARP assume as if it is an application layer service. Students are expected to develop above applications with appropriate GUI using AWT or Swings. Some applications should also be developed using applet as network client.

**It is compulsory to include the work done in all the above three units as Term-work. However, any additional assignments /exploratory work given to students may also be included in the term-work.**

***** ***** *****

## Course Name: Web Technologies

## Course Code: MCA 312

## Objectives:
The objective of the course is make the students learn the concepts of web technologies and apply it in real life applications.

## Prerequisites:
HTML, DHTML and Object-Oriented programming

## Contents:

**1. Java Script [15%]**
Overview of Java Script, Primitives, Operations and Expressions, Screen Output and Keyboard Input, Control Statement, Object Creation and modification, Arrays, Functions, Constructors, Pattern Matching using regular expressions

**2. AJAX Framework [15%]**
Introducing Prototype and Scriptaculous, Introducing Quick Gallery, Simplifying Ajax with Prototype, Using Prototype's advanced Ajax features

**3. JSP [30%]**
Overview of JSP Technology, Invoking Java Code with JSP Scripting elements, Controlling the structure of generated servlets: The JSP Page directive, Including files and Applets in JSP Pages, Using JavaBeans Components in JSP Documents, Integrating Servlets and JSP: The MVC Architecture, Simplifying access to Java Code: The JSP 2.0 Expression Language

**4. Using JSTL [10%]**
Tag Libraries: The Basics, JSP Standard Tag Library (JSTL)

**5. Struts [30%]**
The Struts Framework Basics: Understanding struts, Setting up struts, Struts flow of control, Processing requests with action objects, Handling request parameters with form beans, prepopulating and redisplaying input forms, Struts framework – i18n and layout: Using properties files, internationalizing applications, Laying out pages with tiles, Using tiles Definitions, Struts framework – validating input: validating in the Action class, validating in the Form Beans, Using the automatic validation framework

## Main Reference Book(s):

1) Ivan Bayross, "Web Enabled Commercial Application Development Using HTML,DHTML, PERL, Java Script", BPB publications, Revised Edition
2) Dave Crane , Bear Bibeault and Tom Locke, "Prototype and Scriptaculous in

Action", Manning  Publication
3) Marty Hall and Larry Brown, "Core Servlets and JavaServer Pages Volume - 1", Pearson Education, 2nd Edition
4) Marty Hall and Larry Brown, "Core Servlets and JavaServer Pages Volume - 2", Pearson Education, 2nd Edition

## Suggested Additional Reading:

1) "Web Technologies  Black Book", Dreamtech press
2) Chuck Cavaness , "Programming Jakarta Struts",  O'Reilly Publication
3) Sue Spielman and Meeraj Kunnumpurath , "Pro J2EE 1.4 From Professional to Expert" , APress  Publication
4) Robert W. Sebsta, "Programming the World Wide Web", Pearson Education

## Chapter wise Coverage from Main Reference Book(s):

| Book | Chapters |
|------|----------|
| 1 | 8,9,10 |
| 2 | 1,2,3,4 |
| 3 | 10,11,12,13,14,15,16 |
| 4 | 7,9,10,11,12 |

## Accomplishments of the student after completing the Course:
After completion of this course students will be able to:
- Understand the concepts of web technologies.
- Apply web technologies concepts to develop web applications.

***** ***** *****

## Course Name:  Software Engineering

## Course Code:  MCA 313

## Objectives:

A large percentage of Software projects are termed as failures because they overshoot their planned budget and / or schedule or are of poor quality. Such failed projects then have far reaching economic and other consequences. The main reasons for this failure are poor engineering and poor project management. Hence, there is an urgent need for a systematic, disciplined and quantifiable approach to software development and maintenance. The field of Software engineering which has evolved considerably over the last few decades deals with this approach. Thus, it becomes mandatory for software developers to understand and apply the techniques of software engineering.

Hence, the objective of this course is
- To understand the basic concept and importance of Software Engineering
- To understand the process of  Software Engineering
- To understand all the activities required to develop and maintain software.

## Prerequisites:

Basic knowledge of Programming, Systems Analysis and Design.

## Contents:

1. **Software Engineering and Process Models   [15%]**
   Nature of Software, Unique nature of WebApps, Software Engineering, Software Engineering Practice and Principles.
   Generic Process Model, Process Assessment and Improvement, Prescriptive and Specialized Process Models, Unified Process, Personal and Team Process Model, Process Technology, Product and Process.
   Agility, Agility and cost of change, Agile Process.

2. **Guiding Principles, Requirements understanding and modeling   [10%]**
   Software Engineering Knowledge, Core Principles, Framework Activity Guiding Principles.
   Requirements Engineering, Establishing the Groundwork, Eliciting Requirements, Developing Use Cases, Building the Requirements Model, Negotiating Requirements, Validating Requirements.
   Requirement modeling Strategies, Flow-Oriented Modeling, Behavioral Model, Patterns of Requirements Modeling, Requirement  Modeling of WebApps.

3. **Software Design –Concepts, Architecture, Component, User Interface, Web [30%]**

Design within the Context of Software Engineering, The Design Process, Design Concepts, The Design Model.

Software Architecture, Architectural Genres, Architectural Styles, Architectural Design.

Introduction of Component, Designing class-based Components, Conducting Component level Design, Component level design for WebApps, Designing Traditional Components, Component based Development.

Golden Rules, User Interface Analysis and Design, Interface Analysis, Interface Design Steps, WebApp Interface Design, Design Evolution.

WebApp Design Quality, Design Goals, Design Pyramid for WebApps, WebApp Interface Design, Aesthetic Design, Content Design, Architecture Design, Navigation Design, Component level Design, Object oriented Hypermedia Design Method.

4. **Quality Concepts, Review Techniques and Software Quality Assurance [10%]**

Introduction of Quality, Software Quality, Software Quality Dilemma, Achieving Software Quality.

Cost Impact of Software Defects, Defect Amplification and Removal, Review Metrics and its Use, Reviews: A Formality Spectrum, Informal Review, Formal Technical Reviews.

Background Issues of SQA, Elements of Software Quality Assurance, SQA Tasks, Goals and Metrics, Formal Approaches to SQA, Statistical Software Quality Assurance, Software Reliability, ISO - 9000 Quality Standards, SQA Plan.

5. **Testing Strategy, Testing of Conventional, Object Oriented and Web Applications [20%]**

A Strategic Approach to Software Testing, Strategic Issues, Test Strategies for Conventional Software, Test   for Object Oriented Software, Test Strategies for WebApps, Validation Testing, System Testing, Art of Debugging.

Software Testing Fundamentals, Internal and External Views of Testing, White-Box Testing, Basis Path Testing, Control Structure Testing, Black-Box Testing, Model Based Testing, Testing for Specialized Environments, Architecture, and Applications, Patterns of Software Testing.

Broadening the View of Testing, Testing OOA and OOD Models, Object oriented Testing Strategies, Object oriented Testing Methods, Testing Methods Applicable at Class Level, Interclass Test-Case Design.

Testing Concepts for WebApps, Overview of Testing Process, Content Testing, User Interface Testing, Component-level Testing, Navigation Testing, Configuration Testing, Security Testing, Performance Testing.

6. **Software Configuration Management and Software Metrics [15%]**

Software Configuration Management, SCM Repository, SCM Process, Configuration Management for WebApps.

Framework for Product Metrics, Metrics for Requirements Model, Metrics for the Design Model, Design Metrics for WebApps, Metrics for Source Code, Metrics for Testing, Metrics for Maintenance.

## Main Reference Book(s):

1) Roger Pressman, "Software Engineering – A Practitioner's Approach", 7th Edition, McGraw Hill Higher Education.

## Suggested Additional Reading:

1) Sommerville, "Software Engineering", Pearson Education.
2) W S Jawadekar, "Software Engineering – Principles and Practices", TMH Publication.
3) S A Kelar, "Software Engineering – A Concise Study", PHI Publication.

## Chapter wise Coverage from the Main Reference Book (s):

| Book | Chapters |
|------|----------|
| 1 | 1, 2 (Except 2.1.3), 3-3.3,4, 5, 7(7.1 – 7.5), 8(Except 8.3.3), 9 (9.1 – 9.4), 10, 11, 13, 14, 15, 16, 17, 18(Except 18.6.4, 18.7, 18.9), 19, 20, 22, 23 |

## Accomplishments of the student after completing the Course:

- Understand and appreciate the importance of Software Engineering in today's world.
- Understand and perform the various activities required to develop good quality software within time and cost budget.

***** ***** *****

## Course Name:   Soft Computing

## Course Code: MCA 314(1)

## Objectives:
- To emphasize on learning the design, implementation and application of soft computing methodologies.
- To give a detailed understanding of how to obtain the solution of cross-disciplinary problems quickly, accurately and acceptably.
- To discuss and understand the capability of neural networks, fuzzy systems and genetic algorithms to acquire and apply knowledge in an intelligent manner.

## Prerequisites:
Knowledge of Set Theory and Artificial Intelligence is desirable.

## Contents:

**1. Introduction  [5%]**
   Neural Networks,  Application  Scope of Neural Network, Fuzzy Logic, Genetic Algorithm, Hybrid Systems, Soft Computing.

**2. Artificial Neural Network: An Introduction  [10%]**
   Fundamental Concept, Evolution of Neural Networks, Basic Models of Artificial Neural Network, Important Terminologies of ANNs, McCulloch-Pitts Neuron, Linear Separability, Hebb Network.

**3.  Supervised Learning Network   [8%]**
   Introduction, Perception Networks,  Back-Propagation Network, Radial Basis Function Network, Time Delay Neural Network.

**4. Associative Memory Networks   [10%]**
   Introduction, Training Algorithm of Pattern Association, Autoassociative Memory Network, Heteroassociative Memory Network, Bidirectional Associative Memory, Hopfield Networks (Discrete Only).

**5. Unsupervised Learning Networks  [8%]**
   Introduction, Fixed Weight Competitive Nets, Kohonen Self-Organizing Motor Maps, Adaptive Resonance Theory Network.

**6. Special Networks  [5%]**
   Introduction,  Simulated Annealing Network, Boltzmann Machine.

**7. Fuzzy Set Theory [5%]**
   Fuzzy versus Crisp, Crisp Sets, Fuzzy Sets, Crisp Relations, Fuzzy Relations.

**8. Fuzzy Systems  [10%]**
Crisp Logic, Predicate Logic, Fuzzy Logic, Fuzzy Rule Based System, Defuzzification Methods, Applications.

**9. Fuzzy Logic Control  Systems  [ 5%]**
Introduction, Control System Design, Architecture and Operation of FLC   System, FLC system models, Applications of FLC systems.

**10. Fundamentals of Genetic Algorithms  [12%]**
Genetic Algorithms: History, Basic Concepts, Creation of Offsprings, Working Principle, Encoding, Fitness Function, Reproduction.

**11. Genetic Modeling  [12%]**
Inheritance Operators, Cross Over, Inversion and Deletion, Mutation Operator, Bit-wise Operators.

**12. Integration of Neural Networks, Fuzzy Logic, And Genetic Algorithms    [10%]**
Hybrid Systems, Neural Networks and Fuzzy Logic and Genetic Algorithms Hybrids, Preview of Hybrid Systems.

# Main Reference  Book(s):

1)   S.N. Sivanandam and S.N. Deepa, "Principles of  Soft Computing", Wiley India (P) Ltd.
2)  S. Rajasekaran and G.A. Vijayalakshmi Pai, "Neural Networks, Fuzzy Logic, and Genetic Algorithms (Synthesis and Applications)", PHI Education.

# Suggested Additional Reading Book(s):

1)  S.N. Sivanandam and M. Paulraj, "Introduction to Artificial Neural Networks", Vikas Publishing House.
2)  Fakhreddine O. Karray and Clarence De Silva, "Soft Computing and Intelligent Systems Design (Theory, Tools and Applications)", Pearson education.
3)  D.K. Pratihar, "Soft Computing", Narosa Publishing House.
4)  Sudarshan K. Valluru and T. Nageswara Rao, "Introduction To Neural Networks, Fuzzy Logic and Genetic Algorithms, JAICO Publishing House.
5)  Pinaki Mazumder and Elizabeth M. Rudnick, "Genetic Algorithms for VLSI Design, Layout & Test Automation", Addison Wesley Ltd.

## Chapter wise Coverage from the main reference book(s):

| Book | Chapters |
|------|----------|
| 1 | 1, 2, 3(3.1, 3.2, 3.5–3.7), 4(4.1–4.6 (4.6.1)), 5(5.1–5.3,5.6), 6(6.1–6.3), 14 |
| 2 | 6, 7, 8, 9(9.1 – 9.5), 10 |

## Accomplishments of the student after completing the course:

At the end of the work student will be able to

- Get the knowledge of Soft Computing as Networks, Fuzzy Logic and Genetic Algorithms and its Hybrid Applications.
- Apply hybrid nature of Soft Computing techniques to real world problems by extending the capabilities of existing technologies in more effective and efficient manner.

***** ***** *****

## Course Name: Embedded Systems

## Course Code: MCA 314(2)

## Objectives:

To give basic idea of an embedded system and to give in-depth knowledge on designing the embedded systems to help students become embedded software engineers.

## Prerequisites:

- Good Knowledge of C/C++
- Good Understanding of Computer Organization

## Contents:

**1. Introduction to embedded systems  [10%]**
Embedded Systems, Processor Embedded into a system, Embedded Hardware Units and Devices in a system, Embedded Software in a System, Examples of Embedded System, Embedded Systems on a Chip (SoC) and  use of VLSI circuits designed technology, Complex System Design  and Processors, Design Process in Embedded System, Formalization of System Design, Design Process and Design Examples, Classification of Embedded Systems, Skills Required for an Embedded System Designer.

**2.  8051 and advanced processor architecture, memory organization and real – world interfacing  [15%]**
8051 architecture, real world interfacing, introduction to advanced architectures, processor and memory organization, Instruction-level parallelism, Performance Metrics, Memory-Types, Memory Maps and Addresses, Processor Selection, Memory Selection

**3.  Devices and communication buses for device network  [15%]**
I/O Types and Examples, Serial Communication Devices, Parallel Devices Ports, Sophisticated Interfacing Features in Device Ports, Wireless Devices, Timer and Counting Devices, Watchdog Timer, Real Time Clock, Networked Embedded Systems, Serial Bus communication protocols, Parallel Bus Device Protocols – parallel communication network using ISA, PCI, PCI – X and advanced Buses, Internet Enabled Systems – Network Protocols, Wireless and Mobile System Protocols

4. **Programming concepts and embedded programming in c, c++ ,java [15%]**
Programming in assembly language (ALP) and High Level Language  C,  C Program Elements: Header and source Files  and Preprocessor Directives,  Macros and functions, Data Types, Data Structures, Modifiers, Statements, Loops and Pointers, Object Oriented Programming, Embedded Programming in C++, Embedded Programming in Java

5. **Interprocess communication and synchronization of process, threads and tasks [15%]**
Multiple Processes in an application, Multiple threads in an application, Tasks , Task States  Task and Data ,  Clear-cut Distinction between Functions, ISRS and Tasks by their Characteristics ,  Concept of  Semaphores ,  Shared Data ,  Interprocess Communication ,  Signal Function ,  Semaphore Functions ,  Message Queue Functions ,  Mailbox Functions ,  Pipe Functions ,  Socket Functions ,  RPC Functions

6. **Real time operating systems [15 %]**
OS Services , Process Management , Timer Functions , Event Functions ,  Memory Management , Device, File and IO Subsystems Management , Interrupt Routines in RTOS Environment and Handling of Interrupt Source Calls ,  Real-time Operating Systems ,  Basic Design Using an RTOS ,  Rtos Task Scheduling Models, Interrupt Latency and Response of the Tasks as Performance Metrics ,  OS Security Issues .

7. **Design examples and case studies of program modeling and programming [15%]**
Basic functions and Types of RTOSES, RTOS  mCOS-II, Case Study of Embedded System Design and Coding for an Automatic , Chocolate Vending Machine (ACVM) Using MUCOS RTOS

## Main Reference  Book(s):

1) Rajkamal, Embedded Systems Architecture, Programming and Design, TATA McGraw-Hill, Second Edition.

## Suggested Additional Reading Book(s):

1) Dr. K.V.K.K. Prasad, Embedded/Real –Time Systems: Concepts, Design and Programming Black Book, Dreamtech press.
2) David E.Simon, An Embedded Software Primer, Pearson Education Asia.
3) Frank Vahid and Tony Givargis, Embedded Systems Design – A unified Hardware /Software Introduction, Wiley India.
4) Shibu K.V., Introduction to Embedded Systems, Tata McGraw-Hill.
5) Deniel W. Lewis,  Fundamentals of Embedded Software (where C and assembly meet), PHI.
6) Muhammad Ali Mazidi, Janice Gillispie Mazidi and Rolin D. McKinly, The 8051 Microcontroller and Embedded Systems Using Assembly and C, Pearson Education.
7) James K. Peckol, Embedded Systems A Contemporary Design Tool, Wiley.

## Chapter wise Coverage from the main reference book(s):

| Book | Chapters |
|------|----------|
| 1 | 1, 2, 3, 5, 7, 8, 9.1, 9.2, 11.1 |

## Accomplishments of the student after completing the course :

After completion of the course students will have understanding of

- Embedded system architecture
- Basic hardware and software elements of embedded systems
- Programming models and software techniques to embed the code into the system

***** ***** *****

**Course Name:   Wireless Communication and Mobile Technology**

**Course Code:   MCA 314(3)**

**Objectives:**

This course is introduction to wireless communication with focus on digital mobile communication system  and digital data transfer from computer science point of view. It shows integration of services and applications from fixed networks into networks supporting mobility of end user and wireless access. It emphasizes both on technology and standards of mobile communication and shows merging of classical data transmission technologies and extension of today's Internet applications onto mobile and wireless devices.

**Prerequisites:**

Knowledge of signals, transmission media, protocols and standards of  Internet and networking.

**Contents:**

1.  **Introduction  [10%]**
    Definition, types, applications and history of wireless communication systems. Wireless Transmission:  frequencies for radio transmission, signals, antennas, signal propagation, multiplexing, modulation, spread spectrum, introduction to cellular systems.

2.  **Medium Access Control  [10%]**
    Motivation for specialized MAC: hidden and exposed terminals, near and far terminals., DMA, FDMA, TDMA: Fixed TDM, classical Aloha, slotted Aloha, CSMA, DAMA, PRMA, Reservation TDMA, multiple access with collision avoidance, Polling ISMA, CDMA, comparison of  S/F/T/CDMA.

3.  **Telecommunication systems  [20%]**
    GSM**:**Mobile   services, **s**ystem   architecture,   radio   interface,   protocols, localization  and  calling,  handover.  Localization  management,  addressing, mobile  QOS,  security,  new  data  services,  other  digital  cellular  networks (listing).

4.  **Wireless networks and Wireless LANs  [20%]**
    Satellite communication, wireless system operations and standards (cordless, WLL, WImax, IEEE 802.16 broadband) wireless access standards. Introduction to mobile computing. Infrared, spread spectrum, microwave LANs. Wireless LAN standards: Wi-fi, IEEE 802.11, Bluetooth.

5. **Mobile network and transport layer  [20%]**
   Mobile IP, Dynamic host configuration protocol, mobile adhoc networks, traditional TCP networks, classical TCP improvements, TCP over 2.5/3G wireless networks.

6. **Wireless Application Protocol  [20%]**
   Architecture, WAP client, WAP gateway, WAP Application server, WAP internal structure, WTA and PUSH features, wireless datagram protocol, WML and WML script. WAP 2.0

## Main Reference  Book(s):

1) Jochen Schiller, "Mobile Communications", Pearson Education.
2) Theodore S. Rappaport, "Wireless communications, principles and practices" Pearson   Education.

## Suggested Additional Reading Book(s):

1) William C.Y.Lee, "Mobile cellular Telecommunication Analog and digital system", Mcgraw Hill
2) Rajpandya , "Mobile and personal Communication system and services" by
3) Dr. Kamilo Feher  , "Wireless communication" (WIELY INDIA  EDITION)
4) Asoke K Talukder, Roopa R Yavgal, "Mobile Computing" , TMH

## Chapter wise Coverage from the main reference book(s):

| Book | Chapters |
|------|----------|
| 1 | 1,2,3,4(4.1),7(7.1,7.2,7.3,7.5),8,9,10(10.4,10.6) |

## Accomplishments of the student after completing the course:
At the end of the work student will be able to understand terminologies of wireless communication systems, technical background of wireless transmission media, functioning of mobile phone, technologies in Wireless Networks and Wireless LANs, Mobile Computing, influence of mobility on applications, security or IP networks, higher layers of communication like WAP and current trends in wireless communications and cellular technology

***** ***** *****

## Course Name: Distributed Operating Systems

## Course Code: MCA 314(4)

## Objectives:
- Present the principles underlying the functioning of distributed operating systems
- Create an awareness of the major technical challenges in distributed systems design and implementation
- Expose students to modern and classic technology used in distributed systems and their software
- Provide experience in the implementation of typical algorithms used in distributed systems.

## Prerequisites:
C programming, Computer Networking, Operating Systems

## Contents:

1. **Fundamentals  [5%]**
   Introduction, Distributed Computing Models, Software Concepts, Issues in Designing Distributed Systems,  Case Studies

2. **Interprocess Communication  [10%]**
   Message Passing, Group Communication, Case Studies

3. **Remote Procedure Calls [15%]**
   Introduction, RPC Basics, RPC Implementation, RPC Communication, Issues, Overview of RMI, Case Studies

4. **Synchronization  [15%]**
   Introduction, Clock Synchronization, Logical Clocks, Global State, Mutual Exclusion, Election Algorithms, Deadlocks in Distributed Systems, Case Studies

5. **Distributed System Management  [15%]**
   Introduction, Resource Management, Task Assignment Approach, Load Balancing Approach, Load Sharing Approach, Process Management in a Distributed Environment, Process Migration, Threads, Fault Tolerance.

6. **Distributed Shared Memory  [10%]**
   Introduction, Basic Concepts of DSM, Hardware DSM, Design Issues in DSM Systems, Issues in Implementing DSM Systems,  Heterogeneous and Other DSM, Case Studies.
7. **Distributed File System  [10%]**
   Introduction, File Models, Distributed File System Design, Semantics of File Sharing, DFS Implementation, File Caching, Replication, Case Studies
8. **Real Time Distributed Operating Systems  [15%]**

Introduction, Design Issues in Real-Time Distributed Systems, Real-Time Communication, Real-Time Scheduling, Case Studies.

9. **Case Studies  [5%]**
   Unix IPC Mechanism, RPC, DCE Distributed File Service, Amoeba System and Chorus System

## Main Reference Book(s):

1) Mahajan S., Shah S., "Distributed Computing", Oxford University Press, India

## Suggested Additional Reading:

1) Sinha P.K, "Distributed Operating Systems", Prentice Hall India.
2) Tanenbaum A.S., "Distributed Operating Systems", Pearson Education
3) Singhal Mukesh, Shivaratri N.G., "Advanced Concepts in Operating Systems", Tata McGrawHill Publishing Company Limited
4) George Coulouris, Jean Dollimore and Tim Kindberg", Distributed Systems Concepts and Design", Pearson Education
5) Steen, Maarten Van, Tanenbaum A.S., "Distributed Systems – Principles and Paradigms", Prentice Hall India
6) Crichlow, Joel M., "Distributed Systems – Computing Over Network", Prentice Hall India
7) Garg Vijay, "Concurrent and Distributed Computing in Java", IEEE Press, A John Wiley and Sons Inc. Publication
8) Hagit Attiya, Jennifer Welch, "Distributed Computing – Fundamentals, Simulations and Advanced topics", Wiley India.

## Chapter wise Coverage from Main Reference Book(s):

| Book | Chapters |
|------|----------|
| 1 | 1, 3, 4, 5, 6, 7, 8, 11 |

## Accomplishments of the student after completing the Course:

- Appreciate the principles underlying the functioning of distributed systems, describe the problems and challenges associated with these principles, and evaluate the effectiveness and shortcomings of their solutions
- Recognize how the principles are applied in contemporary distributed systems, explain how they affect the software design, and be able to identify features and design decisions that may cause problems
- Design a distributed system that fulfills requirements with regards to key distributed systems properties (such as scalability, transparency, etc.)
- Build distributed system software using basic OS mechanisms, as well as higher-level middleware and languages.

***** ***** *****

## Course Name: Multimedia

## Course Code: MCA 314(5)

## Objectives:
The course aims at providing the students with the knowledge and skills needed to develop multimedia applications. The focus is on designing, organizing, and producing multimedia projects of all kinds.

## Prerequisites:
Windows navigation skills, experience at developing applications in both graphical or non-graphical environment and knowledge of HTML and Internet.

## Contents:

1. **Introduction to Multimedia and Multimedia Projects  [5%]**
   Introduction, Usage of Multimedia, Stages of a Multimedia Project, the Multimedia Team


2. **Text in Multimedia  [10%]**
   Power of Meaning, Fonts and Faces, Using Text in Multimedia, Computers and Text, Font Editing, Hypermedia, Hypertext

3. **Sound in Multimedia  [10%]**
   Power of Sound, Digital Audio, Audio File Formats, MIDI versus Digital Audio, Adding Sound to Your Multimedia Project

4. **Images in Multimedia  [10%]**
   Organizing Tools, Bitmap Images, Vector Drawings, 3-D Drawing and Rendering, Color, Image File Formats

5. **Animation  [10%]**
   Power of Motion, Principles of Animation, Animation by Computer, Making Animations

6. **Video in Multimedia  [10%]**
   Using Video, Analog Display Standards, Digital Display Standards, Digital Video, Video Recording, Shooting and Editing Video, Optimizing Video Files

7. **Multimedia Hardware  [5%]**
   Macintosh vs Windows, Connections, Memory and Storage Devices, Input Devices, Output Hardware, Communication Devices

8. **Multimedia Software  [5%]**
   Text Editing, OCR Software, Painting and Drawing Tools, 3-D Modeling and Animation Tools, Image Editing Tools, Sound Editing Tools, Animation Tools

9. **Designing for the World Wide Web  [5%]**
   Working on the Web, Text for the Web, Images for the Web, Sound for the Web, Animation for the Web

10. **Development of Animation using – Adobe Flash [30%]**
   Introduction to Action Script 3.0, Controlling Actions with Events, Organizing Objects with the Display List, Controlling the Timeline and Animation, Components for Interactivity, Choosing Using and Animating Text, Drawing with Action Script

## Main Reference Book(s):

1) Tay Vaughan, "Multimedia : Making it Work", 7th Edition, Tata McGraw Hill Publication
2) Chris Grover, "Flash CS5 – The Missing Manual", First Edition, SPD O'Reilly Publication

## Suggested Additional Reading:

1) Ranjan Parekh, "Principles of Multimedia", Tata McGrawHill Publication
2) Judith Jeffcoate, "Multimedia in Practice – Technology and Applications", Pearson Education
3) John F. Koegel Buford, "Multimedia Systems",  Pearson Education
4) James E. Shuman, "Multimedia in *Action"*, Cengage Learning
5) John Villamil, Louis Molina, "Multimedia : An Introduction", PHI Publication
6) Stethen McGloughin, "Multimedia : On the Web", PHI Publication
7) John Villamil, Leony Fernandex-Elias, "Multimedia : Graphics", PHI Publication
8) Jose Lozano, "Multimedia : Sound & Video", PHI Publication
9) Ralf Steinmetz, Klara Nahrstedt, "Multimedia : Computing, Communications & Applications", Pearson Education
10) S Gokul, "Multimedia Magic", BPB Publication
11) "Adobe®Photoshop 7.0 : Classroom in a Book", Adobe Creative Team
12) Emily A. Vander, "Macromedia's Flash 8", O'Reilly
13) "Adobe®Flash CS3 Professional : Classroom in a Book", Adobe Creative Team
14) Philip Kerman, "Adobe Flash CS3 Professional – in 24 Hours", Pearson Education
15) Robert Reinhardt, Snow Dowd, "Adobe Flash CS3 Professional – Bible", Wiley India

## Chapter wise Coverage from the Main Reference Book:

| Book | Chapters |
|------|----------|
| 1    | 1 – 10, 14 |
| 2    | 12 – 18 |

## Practicals :

1. In addition to practical term-work, develop Multimedia Project (like advertisement of product, social message, etc.) using :
   - Adobe Flash – for animation using motion tween, button, publishing content, programming using ActionScript, animating text, etc.
   - Adobe Photoshop – for development of digital images
   - SoundForge – for editing digital audio stream
2. The project documentation must be attached with term-work.

## Accomplishments of the student after completing the course:

- Design and develop multimedia projects
- Use Adobe Photoshop, Adobe Flash and SoundForge to develop multimedia content for various usage
- Make use of multimedia to develop good Web Sites and Web Portals

**\*\*\*\*\* \*\*\*\*\* \*\*\*\*\***

## Course Name: Windows Programming

## Course Code: MCA 314(6)

## Objectives:
- To get an introduction about Win32 environment.
- To be familiar with event driven programming.
- To introduce the concepts of windows programming
- To introduce GUI programming using Microsoft Foundation Classes

## Prerequisites:
Fundamentals of C++ programming such as constructors, function overloading, inheritance and virtual functions

## Contents:

1. **MFC and Windows  [5%]**
   Introduction of MFC, MFC and C++, 16 Bits or 32 Bits OS, Version of MFC, The Windows Programming Environment, Windows and  Program Interaction, Windows Programming Fundamentals, Win Main (),The window Procedure, The Message Loop, Window Classes, Windows Data Types and Naming Conventions, The Components of a Window

2. **MFC Fundamentals  [5%]**
   The MFC Class Hierarchy, MFC Member Functions, MFC Global Functions, AFXWIN.H., An MFC Application , Skeleton, Creating Frame Windows, Using CWinAPP, ANSI vs. Unicode.

3. **Processing Messages  [10%]**
   Introduction to Messages, Responding to Messages – MFC Style, Adding Message Macros to Message Map, Adding Message Handlers to Window Class, Implementing Message Handlers, Using BEGIN_MESSAGE_MAP() Macro, Responding to a keypress , Device Contexts ,Processing the WM_PAINT Message, Responding to Mouse Messages, Generating a WM_PAINT Message, Responding to WM_DESTROY Messages, Exploring  Windows Messages, Generating Timer Messages.

4. **Message Boxes and Menus  [10%]**
   Introducing Menus, Using Resources, Compiling RC Files, Creating a Simple Menu, Including a Menu in Program, Handling WM_COMMAND Messages, Responding to Menu Selections, Adding Menu Accelerator Keys, Loading the Accelerator Table, Creating a Hot Key, Introducing Dialog Boxes, Dialog Boxes Interaction with the User, The Control Classes, Modal vs. Modeless Dialog Boxes, The Dialog Box Resource, The CDialog Class, Processing Dialog Box Messages, Activating a Dialog Box, The Dialog Box Resource File, Manually Closing a Dialog box, Initializing a Dialog box,

Adding a list box, Adding an Edit Box, Exploring Edit Boxes. Using a Modeless Dialog box, Creating a Modeless Dialog Box

5. **More Controls [5%]**
   Using Check Boxes, Managing Check Boxes, Adding Radio Buttons

6. **Working with Icons, Cursors, and Bitmaps [10%]**
   Defining an Icon and a Cursor, Changing the Icon and the Cursor, Loading the Icon and Cursor, Obtaining a Background Brush, Using the Built-in Cursors and Icons, Using a Bitmap, Creating a Bitmap, Displaying a Bitmap, Using Multiple Bitmaps.

7. **Managing Text and Solving the Repaint Problem [10%]**
   Window Coordinates, Setting the Text and Background Color, Setting the Background Display Mode, Obtaining the Text Metrics, Computing the Length of a string, Obtaining the System Metrics, Solving the Repaint Problem, Virtual Window Theory, Using Additional API Functions., Creating and Using a Virtual Window, Changing Fonts, Using Built-in Fonts.

8. **Working with Graphics [8%]**
   The Graphics Coordinate System, Pens and Brushes, Graphics ,Functions, Setting a Pixel, Drawing a Line, Setting the Current Location, Drawing an Arc, Displaying Rectangles, Drawing Ellipses and Pie Slices, Working with Pens, Creating Custom Brushes, Deleting Custom Pens and Brushes.

9. **Introducing Common Controls [7%]**
   Including and Initializing the Common Controls, Using a Toolbar, Adding the Toolbar Buttons, Adding the Toolbar Bitmaps, Resizing the Toolbar, Using Additional CToolBarCtrl Member Functions, Adding Tooltips.

10. **More Common Controls [20%]**
    Creating Up-Down Controls, Creating a Spin Control, Creating a Slider Control, Creating a Progress Bar, Creating Smooth and Vertical Progress Bars, Creating Status Bars, Creating Tab Controls, Creating a Month Calendar, Creating a Tree View Control.

11. **Adding Help [10%]**
    Types of Help, Invoking Help, The Help File, Creating a Help File, The General Form of a help File, Using RTF Commands, Help Macros, Using Help Context IDs, Using the WinHelp() API Function, Enabling Help, Handling the Help Messages, Implementing the Help Menu, Including the **?** Button.

## Main Reference Book(s):

1) MFC Programming from the ground up, by Herbert Schield – TMH

## Suggested Additional Reading:

1) Programming Visual C++, by David J. Kruglinski, George Shepherd and Scot Wingo – Microsoft Press.
2) Visual C++ 6 Programming, by Steve Holtzner - Wiley Dreamtech India Pvt. Ltd.
3) Programming Windows With MFC, by Jeff Prosise – Microsoft Press, 2000.

## Chapter wise coverage from main reference Book(s):

| Book | Chapters |
|------|----------|
| 1 | 1 – 12, 16 |

## Software Requirement:
Visual Studio

## Accomplishments of the Student after completing the course:
- To develop programs and simple applications using Visual C++
- Creating Windows and Child windows control.
- To create powerful, full featured windows application

***** ***** *****

Gujarat University

## Course Name: Bioinformatics

## Course Code:  MCA 315(1)

## Objectives:
- Applications of Computer Science in Biology.
- Biological database creation and management.
- Helps to store, retrieve, analyze and predict the huge biological data.
- Identification, Prediction, Visualization and Pattern Matching from sequences.
- Understanding of Algorithms and its analysis.
- Software development for implementing algorithms.
- Developing applications and tools for bioinformatics.

## Prerequisites:
RDBMS, Programming & Scripting Languages, .Net Technology, Concept of Biology, Statistics.

## Contents:

1. **Bioinformatics: Objectives and Applications, Biology for Bioinformatics [10%]**
   Bioinformatics:  Applications and Research, Present Bioinformatics Scenario in India.

2. **Bioinformatics Database: Sequence Databases, Structure Databases & Other Databases [25%]**
   Characteristics of Bioinformatics Databases, Categories of Bioinformatics Databases, Navigating Databases, Information Retrieval Systems, Sequence Databases, Structure Databases, Other Databases

3. **Tools for Sequence submission and Sequence Analysis, Sequence Alignment: Compare Sequences, Two sequences & multiple sequences alignment [25%]**
   Need for tools, Knowledge Discovery, Industry Trends, Data-mining Tools, Data Submission tools, Data Analysis Tools

4. **Algorithms for data analysis [20%]**
   Classification of Algorithms, Implementing Algorithms, Biological Algorithms, Bioinformatics Tasks and Corresponding Algorithms, Algorithms and Bioinformatics Software, Data Analysis Algorithm

5. **Basics and language elements of Python, Applications programs using Python [20%]**
   Programming Basics and Strings, Numbers and Operators, Variables—Names for Values, Making Decisions, Functions, Classes and Objects, Organizing Programs, Files and Directories, Other Features of the Language, Building a Module, Text Processing, Testing, Writing a GUI with Python, Accessing Databases.

## Main Reference Book(s) :

1) Orpita Bosu , Simminder Kaur Thukral,"BIOINFORMATICS Databases, Tools and Algorithms", Oxford.
2) Peter C. Norton, Alex Samuel, Dave Aitel, Eric Foster-Johnson, Leonard Richardson, Jason Diamond, Aleatha Parker, Michael Roberts, "Beginning Python", Wrox.

## Suggested Additional Reading:

1) Cynthia Gibas, Perl Jambeck, "Developing Bioinformatics Computer Skills**,** O'Reilly .
2) Jean-Michel Claverie, Cedric Notredame,"Bioinformatics – A Beginner's Guide**,** Wiley Publications.
3) Zhumar Ghosh & Bibekanand Mallick,"BIOINFORMATICS Principles and Applications**,** Oxford Press.
4) Kenneth Baclawski and Tianhua Niu , "Bioinformatics**,** Jaico Books
5) Vittal R. Srinivas, "Bioinformatics A Modern Approach", PHI
6) Puneet Mehrotra, Dr. Kumud Sarin, Swapna K. Srivastava, " The New Handbook of Bioinformatics", Vikas Publications.
7) Mark Summerfield, "Programming in Python 3: A Complete Introduction to the Python Language ", Pearson Education
8) Timothy Budd, "Exploring Python", TMH

## Chapter wise coverage from the textbook(s):

| Book | Chapters |
|------|----------|
| 1 | PART A, Databases, Tools, Ch 1- 5,8, All Appendix |
| 2 | 1-14 |

## Accomplishments of the student after completing the course:

- Biological database searching for similar sequences.
- Prediction of protein sequences and structures.
- For identification of patterns from sequences.
- Software skills used for developing user interface, static/dynamic web pages and graphical representations of data for prediction and visualization.
- For developing next generation operating system, networking, and software for data transfer and execution of tools.

***** ***** *****

Gujarat University

## Course Name:  Network Security

## Course Code:  MCA 315(2)

## Objectives:
After completion of this course student will be able to appreciate
- What security threats and attacks are and what are the counter measures
- Symmetric and asymmetric encryption methods
- Authentication applications, Web, IP and Email security
- Intruders and Firewalls

## Prerequisites:
- Fundamentals of Networking
- Number theory
- Basic Mathematics

## Contents:

1. **Network Security and Symmetric Encryption   [20%]**
   Security Trends, The OSI Security Architecture, Security Attacks, Security Services, Security Mechanism, A Model for Internetwork Security, Internet Standards the Internet Society.
   Symmetric Encryption Principles, Symmetric Block Encryption Algorithms, Stream Ciphers and RC4, Cipher Block Modes of Operation, Location of Encryption Devices, Key Distribution.

2. **Public Key Cryptography and Authentication   [20%]**
   Approaches to Message Authentication, Secure Hash Functions and HMAC, Public Key Cryptography Principles,  Public  Key Cryptography Algorithms, Digital Signatures, Key Management.
   Kerberos, X.509 Directory Authentication Service, Public Key Infrastructure.

3. **Email and IP Security   [20%]**
   Pretty Good Privacy (PGP),  S/MIME.
   Overview of IP Security,  IP Security Architecture, Authentication Header, Encapsulating Security Payload, Combining Security Associations, Key Management.

4. **Web Security and Intrusion   [20%]**
   Web Security Requirements, Secure Sockets Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET).
   Intruders,  Intrusion Detection.

5. **Passwords and Firewalls   [20%]**
   Password  Management.
   Firewall Design Principles, Trusted Systems, Common Criteria for Information Technology Security Evaluation.

## Main Reference Book(s):

1) William Stallings, "Network Security Essentials:- Applications and Standards", 3$^{rd}$ Edition, Pearson Education.

## Suggested Additional Reading:

1) Behrouz Forouzan, "Cryptography and Network Security", TMH Publication.
2) Nina Godbole, "Information Systems Security", Wiley Publication.
3) William Stallings, "Cryptography and Network Security", Pearson Education.

## Chapter wise Coverage from the Main Reference Book (s):

| Book | Chapters |
|------|----------|
| 1    | 1-7, 9, 11 |

## Accomplishments of the student after completing the Course:

- Understand and appreciate the importance of Network Security in today's world.
- Understand and use good Network Security applications and standards in various applications.

***** ***** *****

Gujarat University

## Course Name: Distributed & Cluster Computing

## Course Code: MCA 315(3)

## Objectives:
- Present the principles of distributed computing.
- Create an awareness of the major technical challenges in the area of distributed and cluster computing.
- Expose students to modern and classic technology used in distributed computing and related software.
- Provide experience in building algorithms and implementing them on clusters and distributed systems.

## Prerequisites:
Computer Networking, Operating Systems, Parallel Processing

## Contents:

1. **Overview of Cluster Computing  [20%]**
   Cluster Computing at a Glance, Cluster Setup and its Administration, Constructing Scalable Services, Dependable Clustered Computing, Deploying a High Throughput Computing Cluster, Metacomputing: Harnessing Informal Supercomputers

2. **Networking, Protocols and I/O  [20%]**
   High Speed Networks, Lightweight Messaging Systems, Load Balancing over Networks, Distributed Shared Memory, Parallel I/O for Clusters, Software RAID and Parallel File Systems

3. **Process Scheduling, Load Sharing and Balancing  [15%]**
   Job and Resource Management Systems, Scheduling Parallel Jobs on Clusters, Load Sharing and Fault Tolerance Manager, Mapping and Scheduling on Heterogeneous Systems

4. **Message Passing Programming  [20%]**
   Message Passing Paradigm, Intro to MPI, Environment management routings, point to point communication, collective communication, derived data types, group and communicator management, virtual topologies, Overview of OpenMP Programming Model, Advances in MPI

5. **Advances in Distributed Computing  [10%]**
   Grid Computing, Service Oriented Architecture, Cloud Computing

6. **Case Studies  [15%]**
   Clusters: Beowulf, OpenSSI, MOSIX, OpenMosix, OSCAR, ROCKS
   Grid Middleware: Globus Toolkit,  UNICORE,
   Cloud Computing: Aneka Software from Manjrasoft, Eucalyptus, Nimbus

## Main Reference Book(s):

1) Buyya R. (ed), "High Performance Cluster Computing – Systems and Architecture", Volume I, Pearson Education India.
2) Mahajan S., Shah S., "Distributed Computing", Oxford University Press – India
3) MPI Tutorials - Web reference: https://computing.llnl.gov/tutorials/

## Suggested Additional Reading:

1) Buyya R. (ed), "High Performance Cluster Computing – Programming and Applications", Volume II, Prentice Hall
2) Prabhu C.S.R., "Grid and Cluster Computing", Prentice Hall India
3) Quinn M., "Parallel Programming in C with MPI and OpenMP", Tata McgrawHill
4) George Coulouris, Jean Dollimore and Tim Kindberg", Distributed Systems Concepts and Design", Pearson Education
5) Steen, Maarten Van, Tanenbaum A.S., "Distributed Systems – Principles and Paradigms", Prentice Hall India
6) Crichlow, Joel M., "Distributed Systems – Computing Over Network", Prentice Hall India
7) Jani N. N. et al., "Parallel Processing and High Performance Computing", Everest Publishing House
8) Joseph J., Fellenstein C., "Grid Computing", Pearson Education
9) Jankiram D., "Grid Computing – A research monograph", Tata Mcgraw Hill
10) Velte A., "Cloud Computing : A practical Approach", Tata Mcgraw Hill
11) Hagit Attiya, Jennifer Welch, "Distributed Computing – Fundamentals, Simulations and Advanced topics", Wiley India.

## Chapter wise Coverage from Main Reference Book(s):

| Book | Chapters |
|------|----------|
| 1 | 1, 2, 3, 4, 5, 7, 9 (Upto 9.4), 10, 14, 17, 20, 21, 22, 25 |
| 2 | 1, 13 |

## Accomplishments of the student after completing the Course:
- Appreciate the principles underlying the functioning of distributed computing, describe the problems and challenges associated with these principles, and evaluate the effectiveness and shortcomings of their solutions
- Recognize how the principles are applied in contemporary distributed systems, explain how they affect the software design, and be able to identify features and design decisions that may cause problems
- Implementation of Clusters, Grids and Cloud computing environments
- Designing of applications suitable to be executed on Distributed Computing Environment and Clusters

***** ***** *****

Gujarat University

## Course Name:  Image Processing

## Course Code:  MCA 315(4)

## Objectives:
The objective of this course is to
- Provide an introduction to basic concepts and methodologies for digital image processing, and to develop a foundation that can be used as the basis for further study and research in this field.
- Provide understanding of the different types of image representations, enhancing image characteristics, image filtering, and reducing the effects of noise and blurring in an image.

## Prerequisites:
Knowledge of Computer Graphics is desirable.

## Contents:

### 1. Introduction [10%]
Image Processing, The origins of Digital Image Processing, Examples of Fields that use Digital Image Processing, Fundamentals Steps in Digital Image Processing,  Components of an Image Processing System.

### 2. Digital Image Fundamentals   [15%]
Elements of Visual Perception, Light and the Electromagnetic Spectrum, Image Sensing and Acquisition, Image Sampling and Quantization, Some Basic Relationship between Pixels, An Introduction to the Mathematical Tools Used in Digital Image Processing

### 3. Intensity Transformation and Spatial Filtering   [15%]
Background, Some Basic Intensity Transformation Functions, Histogram Processing, Fundamentals of Spatial Filtering, Smoothing Spatial Filters, Sharpening Spatial Filters

### 4. Filtering in the Frequency Domain   [25%]
Background, Preliminary Concepts, Sampling and the Fourier Transform of Sampled Functions, The Discrete Fourier Transform (DFT) of One Variable, Extension to Functions of Two Variables,, Some Properties of the 2-D Discrete Fourier Transform, The Basics of Filtering in the Frequency Domain, Image Smoothing Using Frequency Domain Filters, Image Sharpening Using Frequency Domain Filters.

### 5.  Color Image Processing   [10%]
Color Fundamentals, Color Models, Pseudocolor Image Processing, Basics of Full-Color Image Processing, Color Transformations, Smoothing and Sharpening.

**6. Image Compression   [15%]**
   Fundamentals, Some Basic Compression Methods (Huffman Coding, Arithmetic Coding, LZW Coding, Run-Length Coding, Block Transform Coding).

**7. Image Segmentation   [10%]**
   Fundamentals, Thresholding, Point, Line and Edge Detection.

## Main Reference Book(s) :

1) Rafael C. Gonzalez and Richard E. Woods, "Digital Image Processing", 3$^{rd}$ Edition, Pearson Education.
2) Rafael C. Gonzalez, Richard E. Woods and Steven L. Eddins, "Digital Image Processing Using MATLAB", 2$^{nd}$ Edition, Tata McGraw Hill Education

## Suggested Additional Reading Book(s):

1) B. Chanda and D. Dutta Majumder, "Digital Image Processing and Analysis", PHI Publication.
2) Madhuri A. Joshi, "Digital Image Processing – An Algorithmic Approach, PHI Publication.

**Chapter wise coverage from the main reference Books:**

| Book | Chapters |
|------|----------|
| 1 | 1(1.1 – 1.5), 2(2.1 – 2.6), 3(3.1 – 3.6), 4(4.1 – 4.9), 6(6.1 – 6.6), 8(8.1,8.2(8.2.1,8.2.3 – 8.2.5, 8.2.8)), 10(10.1 – 10.3) |

## Accomplishment of the student after completing the course:
At the end of the work student will be able to
- Use histogram processing techniques; introduce fuzzy set theory and its application to Image Processing.
- Deal with frequency domain and data compressing.

**\*\*\*\*\*   \*\*\*\*\*   \*\*\*\*\***

Gujarat University

## Course Name:  ERP (Enterprise Resource Planning)

## Course Code:  MCA 315(5)

## Objectives:
Every type of organization – large / medium / small, manufacturing / service / government / NGO – belonging to any sector / industry needs to have complete integration of its operations in order to survive and thrive in the extremely competitive world of 21$^{st}$ century. The integration of operations requires an integrated and centralized information system so that right information is available to the right person at the right time for efficient use of resources. ERP systems fulfill this requirement perfectly if planned and implemented successfully. The IT professional is required to play a very significant role in the planning and implementation of ERP system and hence needs to understand different modules in a typical ERP package. The basics of ERP have been covered in an earlier course in detail.

Hence, the objective of this course is
- To make the student aware about all the key issues involved in successful planning and implementation of ERP package.
- To make the student aware about the main components (modules) of ERP packages in general.

## Prerequisites:
Basics of ERP ( covered in MCA 115 –ERFM)

## Contents:

**1. ERP Implementation (Equal weight age to all 18 topics)   [60%]**
  To go for ERP or not, Implementation Challenges, ERP Implementation (Transition) Strategies, ERP Implementation Life Cycle, Pre-implementation Tasks – Getting Ready, Requirements Definition, Implementation Methodologies, Package Selection, ERP Project Teams , Process Definition, Vendors and Consultants,  Dealing with Employee Resistance, Contracts with Vendors / Consultants/Employees, Training and Education, Data Migration, Project Management and Monitoring,  Post Implementation Activities, Success and Failure Factors of an ERP Implementation.

 **2. The Business Modules of ERP (Equal weight age to all 7 modules)   [40%]**
   Manufacturing, Human Resources, Plant Maintenance, Materials Management, Quality Management,  Marketing,  Sales Distribution and Service.

## Main Reference Book(s) :

1) Alexis Leon, "ERP Demystified", 2$^{nd}$ Edition, Tata McGraw Hill Education.

## Suggested Additional Reading Book(s):

1) Mary Sumner, "Enterprise Resource Planning", Pearson Education.
2) E Monk and B Wagner, "Concepts in ERP", Thomson Press.

## Chapter wise coverage from the main reference Books:

| Book | Chapters |
|------|-----------|
| 1 | 18-35, 42-48 |

## Accomplishment of the student after completing the course:

At the end of the work student will be able to

- Understand and appreciate the importance of key issues involved in planning and implementing an ERP package in an organization.
- Perform the various activities required to plan and implement an ERP package in an organization.
- Understand the basics of major modules of ERP packages.

**\*\*\*\*\*    \*\*\*\*\*    \*\*\*\*\***

## Course Name: Linux Programming

## Course Code: MCA 315(6)

## Objectives:
- To get a good understanding of Linux internals.
- To develop proficiency in creating applications on Linux platform.
- To create a base for advanced Computer Science Courses

## Prerequisites:
Knowledge of Operating Systems, C programming language

## Contents:

1. **Linux Environment [20%]**
   - o Introduction to Linux, Environment variables, Working with files
   - o Advanced Shell programming
   - o Standards for Linux

2. **X-windows [10%]**
   - o Terminals, termios structure, Detecting keystrokes
   - o Managing text-based screens with curses

3. **Data Management [10%]**
   - o Managing memory, File locking, Databases
   - o Accessing MySQL data from C

4. **Linux Tools [10%]**
   - o Development tools like make, RPM, GNU gcc, RCS, CVS etc.
   - o Debugging tools like gdb, Lint, ElectricFence, Valgrind etc..

5. **Process and Memory Management [25%]**
   - o Linux processes and signals
   - o POSIX threads : creation, synchronization, attributes, canceling
   - o Semaphores, shared memory and Message queues
   - o Inter-process communication using Pipes
   - o Socket programming

6. **GUI Programming [25%]**
   - o Programming GNOME using GTK+
   - o Programming KDE using Qt

## Main Reference Book(s):

1) Neil Matthew, Richard Stones, "Beginning Linux Programming", Wrox Publication (Wiley India), 4$^{th}$ edition.

## Suggested Additional Reading:

1) Arnold Robbins, "Linux Programming by example – The Fundamentals", Pearson Education.
2) Richard Stevens, "Advanced Unix Programming", Pearson Education.
3) Jon Masters, Richard Blum, "Professional Linux Programming", Wrox Publication (Wiley India).
4) Robert Love, "Linux Kernel Development", Pearson Education.
5) Beck Michael, "Linux Kernel Programming", Pearson Education.
6) Richard Peterson, "Linux Programming – A beginner's Guide", Dreamtech Publication.
7) Alan, Paul, "Introduction to design patters in C++ with QT4", Pearson Education.
8) Jasmin Blanchette, Mark Summerfield, "C++ GUI programming with QT4", Pearson Education.
9) N.B. Venkateshwarlu, "Linux Programming Tools", B. S. Publication – Hyderabad.
10) Eric S. Raymond, "The Art of Unix Programing", Pearson Education.
11) John K. Ousterhout, "Tcl and Tk Toolkit", Addison-Wesley.
12) Linux man pages especially Section 1, 2 and 3.

## Chapter wise Coverage from Main Reference Book(s):

| Book | Chapters |
|------|----------|
| 1    | 1-18     |

Programs are to be implemented in any Linux system either in C or C++ language.

## Accomplishments of the student after completing the Course:

- Ability to develop console applications on Linux
- Ability to develop GUI applications using GTK or Qt on Linux
- Proficiency in using Linux API's
- Creating and Using Shared Libraries and Tweaking Kernel

***** ***** *****

Gujarat University

## Course Name: Dissertation

## Course Code: MCA 316

## Objectives:
Dissertation aims to provide a real opportunity to the students to explore a subject of their interest from within or outside the course curriculum. As a part of dissertation, the student can
- Make an initial attempt to do some serious "research"
- Review of ongoing research work published in reputed journals or proceedings of conferences of repute
- Present results of some innovative experiments
- Study and analysis of Advanced Computer Science Topics

**Prerequisites:** Technical Writing Skills, Knowledge of at least one Word-processing software, Knowledge of Internet.

## Philosophy:
A group of students (2-3) shall either choose themselves or shall be assigned experiment/topic of their interest by the instructor / mentor, which will form the basis of their dissertation. The group in consultation with their instructor / mentor shall discuss and formulate a description of the dissertation. This should include :

- Title : A clear indication of the content of the Dissertation.
- Aims & Objectives : An overall statement of the nature of the work and what is intended to be done.
- Methodology : The Identification of the ways by which the above stated objectives are to be achieved.
- Timescale : Listing of the tasks involved in the Dissertation and estimation of the timescale involved for each task so as to ascertain the milestones by which the instructor / mentor can assess the progress.

## Guidelines for Report Preparation:

- The report should be in the range of 50-100 pages.
- The contents of the report shall include :
  - ✓ Title Page
  - ✓ Acknowledgements
  - ✓ Table of Contents
  - ✓ Main Text
  - ✓ References
  - ✓ Appendices
- The Main Text shall consist of :
  - ✓ Introduction
  - ✓ Literature Survey

- ✓ Methodology
- ✓ Results
- ✓ Discussions
- ✓ Conclusion

- Paper Size : International standard paper size A4 (297 x 210 mm).
- Typing : On one side only.
- Margins : Top 1.0", Bottom 1.0", Left 1.5", Right 1.0".
- Line Spacing : 1.5.
- Character Spacing : Normal.
- Font : Times New Roman or any other non-fancy font.
- Font Size : 12 for Main Text; 10 for References.
- Page number : Pages should be numbered clearly and consecutively; Numbering should be done uniformly throughout the work.
- Diagrams / Charts : Should be arranged so as to open to the right.
- Larger font size may be used for chapter headings and sub headings; the chapter headings, sub headings and sub sub headings should be clearly distinguishable from each other; to distinguish chapter headings from sub headings and sub sub headings any consistent scheme may be adopted.
- References :
    - ✓ Should contain all the work that is consulted.
    - ✓ Should be arranged alphabetically and numbered consecutively.
    - ✓ Font : Same as the main text.
    - ✓ Line Spacing – single.
    - ✓ Documentation Style – IEEE referencing standards to be followed.

## Accomplishments of the student after completing the course:

- Writing the dissertation will enable the student to be innovative and develop research bent of mind and be aware of new advancements in the field of Computer Science and Information Technology.
- Thinking analytically, synthesizing complicated information, writing well, and organizing the time will all serve well regardless of the career the student begins.
- It will prepare the students for serious research work in future.

***** ***** *****