# ANNA UNIVERSITY
# CENTRE FOR DISTANCE EDUCATION
# MASTER OF SCIENCE IN COMPUTER SCIENCE
# REGULATIONS - 2018 CURRICULUM

## SEMESTER – I

<table>
<thead>
<tr>
<th>CODE NO.</th>
<th>COURSE TITLE</th>
<th>CREDITS*</th>
<th>MARKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCS5101</td>
<td>Computer Organization</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5102</td>
<td>Problem Solving and Programming</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5103</td>
<td>Database Management System</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5104</td>
<td>Software Engineering</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5105</td>
<td>Mathematical Foundations of Computer Science</td>
<td>4</td>
<td>100</td>
</tr>
<tr>
<td>DCS5111</td>
<td>Programming Lab</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5112</td>
<td>Database Management System Lab</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td><strong>16</strong></td>
<td><strong>700</strong></td>
</tr>
</tbody>
</table>

## SEMESTER – II

<table>
<thead>
<tr>
<th>CODE NO.</th>
<th>COURSE TITLE</th>
<th>CREDITS*</th>
<th>MARKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCS5201</td>
<td>Computer Networks</td>
<td>4</td>
<td>100</td>
</tr>
<tr>
<td>DCS5202</td>
<td>Object Oriented Programming</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5203</td>
<td>Data Structures and Algorithms</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5204</td>
<td>Operating System</td>
<td>4</td>
<td>100</td>
</tr>
<tr>
<td>DCS5205</td>
<td>Software Project Management</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5211</td>
<td>Object Oriented Programming Lab</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5212</td>
<td>Data Structures and Algorithms Lab</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td><strong>18</strong></td>
<td><strong>700</strong></td>
</tr>
</tbody>
</table>

## SEMESTER – III

<table>
<thead>
<tr>
<th>CODE NO.</th>
<th>COURSE TITLE</th>
<th>CREDITS*</th>
<th>MARKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCS5301</td>
<td>Information Security</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5302</td>
<td>Data Warehousing and Mining</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5303</td>
<td>Web Programming</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5304</td>
<td>Object Oriented Analysis and Design</td>
<td>4</td>
<td>100</td>
</tr>
<tr>
<td>E1</td>
<td>Elective I</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5311</td>
<td>Web Programming Lab</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5312</td>
<td>Software Development Lab</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td><strong>16</strong></td>
<td><strong>700</strong></td>
</tr>
</tbody>
</table>

## SEMESTER – IV

<table>
<thead>
<tr>
<th>CODE NO.</th>
<th>COURSE TITLE</th>
<th>CREDITS*</th>
<th>MARKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>E2</td>
<td>Elective II</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>E3</td>
<td>Elective III</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5411</td>
<td>Project Work</td>
<td>12</td>
<td>200</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td><strong>16</strong></td>
<td><strong>400</strong></td>
</tr>
</tbody>
</table>

**Total No. of Credits** 66  2500

*Each credit is equivalent to 30 hours of student study comprising of all learning activities.*
## ELECTIVES

### SEMESTER – III

#### ELECTIVE – I

<table>
<thead>
<tr>
<th>CODE NO.</th>
<th>COURSE TITLE</th>
<th>CREDITS*</th>
<th>MARKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCS5001</td>
<td>Mobile Computing</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5002</td>
<td>XML and Web Services</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5003</td>
<td>Theory of Computation</td>
<td>2</td>
<td>100</td>
</tr>
</tbody>
</table>

### SEMESTER – IV

#### ELECTIVE - II

<table>
<thead>
<tr>
<th>CODE NO.</th>
<th>COURSE TITLE</th>
<th>CREDITS*</th>
<th>MARKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCS5004</td>
<td>Computer Graphics and Multimedia Systems</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5005</td>
<td>Ethical Hacking and Cyber Forensics</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5006</td>
<td>Software Testing</td>
<td>2</td>
<td>100</td>
</tr>
</tbody>
</table>

#### ELECTIVE – III

<table>
<thead>
<tr>
<th>CODE NO.</th>
<th>COURSE TITLE</th>
<th>CREDITS*</th>
<th>MARKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCS5007</td>
<td>Cloud Computing</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5008</td>
<td>Visual Programming</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>DCS5009</td>
<td>E Commerce</td>
<td>2</td>
<td>100</td>
</tr>
</tbody>
</table>
COURSE OBJECTIVES

Upon Completion of the course, the students should be able to:

- Master the binary and hexadecimal number systems including computer arithmetic.
- Design and implement digital systems with basic gates and other components using combinational and sequential circuits.
- Be familiar with the Von Neumann architecture.
- Be familiar with the functional units of the processor and addressing modes, instruction sets.
- Be familiar with the memories and cache subsystem.
- Be familiar with different ways of communicating with I/O devices and standard I/O interfaces.

COURSE OUTCOMES

- Understand the fundamentals of Boolean logic and functions.
- To have a thorough understanding of the basic structure and operation of a digital computer.
- Design and realize digital systems with basic gates and other components using combinational and sequential circuits.
- Discuss in detail the operation of the arithmetic and logic unit.
- To study the instruction sets and operation of a processor.
- To study the different ways of communicating with I/O devices and standard I/O Interfaces.
- To study the hierarchical memory system including cache memories and virtual memory.

UNIT I
Digital systems, binary numbers, octal, hexadecimal conversions, signed binary numbers, complements, logic gates, Boolean algebra, K-maps, standard forms, NAND-NOR implementation.

UNIT II
Combinational circuits, adder, subtractor, ALU design, decoder, encoder, multiplexers, Sequential circuits: latches, flip-flops, registers, memories, up-down counters.

UNIT III
Von-Neumann architecture, processor: definition, structure, category, technology, ALU concept, stored programs, fetch execute cycle, instruction formats, clock rate instruction rate, pipeline, current processors, multi core processors.

UNIT IV
Physical memory, addressing, virtual memory, address translation, paging, cache, L1, L2, L3 cache memories, cache mapping, LRU replacement.

UNIT V
Data transfer, Serial and Parallel data transfer, Full duplex- half duplex interaction, Bus interface, Programmed I/O, Polling, Interrupt driven I/O, Hardware interrupt mechanism, Interrupt vectors, Multi level of interrupts, DMA, buffer chaining, operation chaining.

REFERENCE BOOKS:

   Douglas E.Comer Pearson sixth edition 2012
COURSE OBJECTIVES

- Understand the various problem solving techniques.
- To be aware of the top down design technique.
- To learn the syntax of C.
- To be exposed to the file processing techniques of C.
- To be familiarized with the preprocessor directives.

COURSE OUTCOMES

Upon completion of the course, the students should be able to

- Design and implement C programs for any given problem.
- Work with existing programs and modify it as per the requirements.
- Identify the errors in a C program.
- Identify the output of a C program without actually executing it.


REFERENCE BOOKS:

COURSE OBJECTIVES

- Learn the fundamentals of data models and to conceptualize and depict a database system using ER diagram.
- To make a study of SQL and relational database design.
- Understand the internal storage structures using different file and indexing techniques which will help in physical DB design.
- Know the fundamental concepts of transaction processing- concurrency control techniques and recovery procedure.
- Gain a fundamental knowledge about the Storage and Query processing Techniques.

COURSE OUTCOMES

Upon completion of the course, the students should be able to

- Design and create tables in database and query them.
- Know how transaction processing is done.
- Analyze and appraise different types of databases.

UNIT I


UNIT II


UNIT III


UNIT IV


UNIT V

ADVANCED TOPICS - Distributed Databases-Architecture-Transaction Processing-Data Warehousing and Mining-Classification-Association rules-Clustering-Information Retrieval-Relevance ranking-Crawling and Indexing the Web- Object Oriented Databases-XML Databases.

REFERENCE BOOKS:

COURSE OBJECTIVES

- To provide information about wider engineering issues that form the background to develop complex, evolving (software-intensive) systems.
- To plan a software engineering process to account for quality issues and non-functional requirements.
- To employ a selection of concepts and techniques to complete a small-scale analysis and design in mini projects.
- To impart knowledge to translate requirement specifications into a design, and then realize that design practically, all using an appropriate software engineering methodology.
- To provide basic knowledge about software project management.

COURSE OUTCOMES

Upon completion of the course, the students should be able to

- Familiar with basic concepts of Software design and implementation.
- Perform software testing on various applications.
- Understand and apply various software metrics on software quality products.

UNIT I


UNIT II


UNIT III


UNIT IV


UNIT V


REFERENCE BOOKS:

COURSE OBJECTIVES

To introduce mathematical logic, combinatorial and counting techniques, Algebraic structures, Finite state system and grammar as Mathematical Foundation of computer Science so as to understand algorithms, computability and other theoretical aspects of Computer science.

COURSE OUTCOMES

Upon completion of the course, the students should be able to

- Understand mathematical logic and to develop analytical solutions for logical problems and they will be equipped with counting techniques to solve combinatorial problems.
- Comprehend the algebraic structure and formal languages with their applications to handle abstract generalizations and computability.

UNIT I


UNIT II

COMBINATORICS - Permutations and Combinations - Mathematical Induction - Pigeonhole principle - Principle of Inclusion and Exclusion - Recurrence relations - Solution by generating functions and characteristics equations.

UNIT III

ALGEBRAIC STRUCTURES - Groups - Cyclic group - Permutation group (Sn and Dn) - Substructures - Homomorphism - Cosets and Lagrange’s Theorem - Normal Subgroups - Rings and Fields (definition and examples).

UNIT IV

LATTICES - Partial order relation – Posets - Hasse diagram - Lattices - Special Lattices - Boolean Algebra.

UNIT V

FINITE STATE AUTOMATA AND GRAMMARS - Finite state automata - Deterministic and non-deterministic model - languages accepted by Finite State Automata - Regular expressions - Context-free grammars - Derivation trees.

REFERENCE BOOKS:

COURSE OBJECTIVES

- To practice the syntax of C.
- To be exposed to the file processing techniques of C.
- To be familiarized with control structures, functions, arrays and files.

COURSE OUTCOMES

Upon Completion of the course, the students should be able to:

- Design and implement C programs for any given problem.
- Understand an existing program and modify it as per the requirements.
- Identify the errors in a C program.
- Produce the output of a C program by actually executing it.

EXPERIMENTS IN THE FOLLOWING TOPICS:

- Non-iterative control structures.
- Iterative control structures and arrays.
- Functions with parameters.
- Functions with arrays, structures as arguments.
- Character and String handling Libraries.
- Files – Sequential access and random access.
- Preprocessor directives for other features like macros, conditional compilation.
COURSE OBJECTIVES

- Understand the concepts of DBMS practically.
- To familiarize with SQL queries.
- To write stored procedures in DBMS.
- Learn front end tools and to integrate them with databases.

COURSE OUTCOMES

Upon completion of the course, the students should be able to:

- Design and implement databases practically.
- Formulate complex queries using SQL and execute them.
- Design and implement applications that have GUI and access databases for backend connectivity.

EXPERIMENTS IN THE FOLLOWING TOPICS:

- Data Definition, Manipulation of Tables and Views
- Database Querying – Simple queries, Nested queries, Sub queries and Joins
- Triggers
- Transaction Control
- Embedded SQL
- Database Connectivity with Front End Tools
- Front End Tools / Programming Languages
- High level language extensions - PL/SQL Basics
- Procedures and Functions
- Database Design and Implementation (Case Study)
SEMESTER – II

DCS5201 COMPUTER NETWORKS

COURSE OBJECTIVES

- Understand data communication techniques.
- To know network Fundamentals.
- Understand Network layers and its functionalities.

COURSE OUTCOMES

Upon Completion of the course, the students should be able to:

- Trace the flow of information from one node to another node in the network.
- Identify the component required to build different types of networks.
- Understand the division of network functionalities into layers.
- Identify solution for each functionality at each layer.
- Choose the required functionality at each layer for given application.


REFERENCES

COURSE OBJECTIVES

- Understand the OO paradigm.
- To be aware of the OO design technique.
- To learn the syntax of C++.
- To be exposed to the file processing and exception handling techniques of C++.
- To be familiarized with the Standard Template Library.

COURSE OUTCOMES

Upon Completion of the course, the students should be able to:

- Design and implement C++ programs for any given problem.
- Understand an existing program and modify it as per the requirements.
- Identify the errors in a C++ program.
- Identify the output of a C++ program without actually executing it.
- Write generic programs using STL.


UNIT II IMPLEMENTING ADTS AND ENCAPSULATION - Aggregate Type struct – Structure Pointer Operators – Unions – Bit Fields – Data Handling and Member Functions – Classes – Constructors and Destructors – Static Member – this Pointer – reference semantics – implementation of simple ADTs.


REFERENCE BOOKS:

COURSE OBJECTIVES

- Gain comprehensive introduction of common data structures, and algorithm design and analysis.
- To master the design of tree, sets and graph structures and its applications.
- Learn about sorting techniques and understand how common computational problems can be solved efficiently on a computer.

COURSE OUTCOMES

Upon Completion of the course, the students should be able to:

- Describe, explain, and use abstract data types including stacks, queues and lists.
- Design and Implement Tree data structures and Sets.
- Design algorithms using graph structure to solve real-life problems.
- Implement a variety of algorithms for sorting, including insertion sort, selection sort, merge sort, quick sort, and heap sort.
- Describe the asymptotic performance and algorithm design techniques studied in this course and understand the practical implications of that information.

UNIT I     BASIC DATA STRUCTURES - From Problems to programs - Abstract Data Types - Data Types, Data Structures, and Abstract Data Types - The Running Time of a program - Calculating the Running Time of a program - Good Programming Practice; Basic Data Types: The Data Type “List” - Implementation of Lists – Stacks – Queues – Mappings - Stacks and Recursive Procedures.


REFERENCE BOOKS:

COURSE OBJECTIVES

- Learn the Operating System basics.
- Study the process management of Operating system.
- Gain knowledge in the storage management and I/O systems of Operating system.

COURSE OUTCOMES

Upon Completion of the course, the students should be able to:

- Ability to discuss on the basics of OS.
- In depth knowledge in process management, memory management and I/O Management of various operating systems.
- To explore the case studies with various operating systems.


REFERENCE BOOKS:

COURSE OBJECTIVES

- Understand the cost evaluation techniques.
- Learn the concepts of project planning and monitoring.
- Understand the concepts of organizing teams for software projects.

COURSE OUTCOMES

- To perform planning and scheduling activities.
- Ability to draw activity network.
- Ability to manage people and project.

UNIT I

INTRODUCTION TO SOFTWARE PROJECT MANAGEMENT - Project Definition – Contract Management – Activities Covered By Software Project Management – Overview of Project Planning – Stepwise Project Planning.

UNIT II


UNIT III


UNIT IV


UNIT V


REFERENCE BOOKS:

COURSE OBJECTIVES

- To implement the OO design technique.
- To learn the syntax of C++ and features of C++.
- To be exposed to the file processing and exception handling techniques of C++.
- To be familiarized with the Standard Template Library.

COURSE OUTCOMES

- Ability to write C++ programs
- Implement the Stack & queue structures using C++
- Ability to create Class Template files

EXPERIMENTS IN THE FOLLOWING TOPICS:

- Write a C++ Program to illustrate Enumeration and Function Overloading
- Write a C++ Program to illustrate Scope and Storage class
- Implementation of ADT such as Stack and Queues
- Write a C++ Program to illustrate the use of Constructors and Destructors and Constructor Overloading
- Write a Program to illustrate Static member and methods
- Write a Program to illustrate Bit fields
- Write a Program to overload as binary operator, friend and member function
- Write a Program to overload unary operator in Postfix and Prefix form as member and friend function
- Write a Program to illustrate Iterators and Containers
- Write a C++ Program to illustrate function templates
- Write a C++ Program to illustrate template class
- Write C++ Programs and incorporating various forms of Inheritance
- Write a C++ Program to illustrate Virtual functions
- Exception Handling
COURSE OBJECTIVES

- To develop skills in design and implementation of data structures and their applications.
- To learn and implement linear, non linear and tree data structures
- To learn Set ADT and Graph data structures and its applications
- To study, implement and analyze of different sorting techniques.

COURSE OUTCOMES

Upon Completion of the course, the students should be able to:

- Work with basic data structures that are suitable for problems to be solved efficiently.
- Implementation of linear, tree, and graph structures and its applications.
- Implementation of various sorting techniques its algorithm design and analysis.

EXPERIMENTS IN THE FOLLOWING TOPICS:

- Abstract Data type Implementation of List, Stack and Queues.
- Tree ADT
- Tries Implementation
- Set ADT- Bit Vector Implementation
- Graph Representations
- Graph Traversals
- Shortest Path Implementation
- Spanning Tree Implementation
- Sorting Algorithms
- Implementation of Algorithms using Dynamic Programming, Backtracking
SEMESTER – III

DCS5301 INFORMATION SECURITY

COURSE OBJECTIVES

- Understand the security analysis principles
- Able to become familiar with Security Design.

COURSE OUTCOMES

Able to Develop

- Security Models using SDLC
- Implement Logical and Physical Security
- Perform Recovery using Disaster Recovery Techniques


UNIT III SECURITY ANALYSIS - Risk Management: Identifying, Assessing, Mitigating, Transferring and Accepting Risk, Issues of Due Care and Due.


UNIT VI BUSINESS CONTINUITY & DISASTER RECOVERY - The need for BC and DR Process, The BIA process, Understanding RTO and RPO as a business driven metrics, Selecting the appropriate BC and DR strategies, Testing and Maintenance of BC and DR Plans.

REFERENCE BOOKS:

COURSE OBJECTIVES

- Understand Data mining principles and techniques and Introduce DM as a cutting edge business intelligence.
- Expose the students to the concepts of Datawarehousing Architecture and Implementation.
- Study the overview of developing areas – Web mining, Text mining and ethical aspects of Data mining.
- Identify Business applications and Trends of Data mining.

COURSE OUTCOMES

Upon Completion of the course, the students will be able to

- Evolve Multidimensional Intelligent model from typical system.
- Discover the knowledge imbibed in the high dimensional system.
- Evaluate various mining techniques on complex data objects.


UNIT II  DATA MINING & DATA PREPROCESSING - Introduction to KDD process – Knowledge Discovery from Databases - Need for Data Preprocessing – Data Cleaning – Data Integration and Transformation – Data Reduction – Data Discretization and Concept Hierarchy Generation.

UNIT III ASSOCIATION RULE MINING - Introduction - Data Mining Functionalities - Association Rule Mining - Mining Frequent Itemsets with and without Candidate Generation - Mining Various Kinds of Association Rules - Constraint-Based Association Mining.

UNIT IV CLASSIFICATION & PREDICTION - Classification vs Prediction – Data preparation for Classification and Prediction – Classification by Decision Tree Introduction – Bayesian Classification – Rule Based Classification – Classification by Back propagation – Support Vector Machines – Associative Classification – Lazy Learners – Other Classification Methods – Prediction – Accuracy and Error Measures – Evaluating the Accuracy of a Classifier or Predictor – Ensemble Methods – Model Section.


REFERENCE BOOKS:

COURSE OBJECTIVES

- To understand the basics of HTML.
- To learn the concepts of XML related technologies.
- To learn the fundamentals of Java.
- To understand the importance of server side programming and web development.

COURSE OUTCOMES

Upon Completion of the course, the students should be able to:

- Design and implementation of web forms and client side validation.
- XML authoring, Parsing, and related technologies.
- Object oriented concept programming using Java.
- Design and development of GUI based applications using Swing components.
- Design and development of servlet and JSP application with database connectivity.


UNIT IV  JAVA GUI AND DATABASE CONNECTIVITY - Generic classes – Generic methods – Applets – Applet life cycle methods – Applets based GUI – GUI components – Basic of Swings – Accessing database with JDBC basics.


REFERENCE BOOKS:

COURSE OBJECTIVES

- Understand the basics of object oriented analysis and design.
- Learn UML models and tools.
- To apply design patterns to various applications.

COURSE OUTCOMES

Upon Completion of the course, the students should be able to:

- Familiarize with the topics of object oriented System designs.
- Design patterns using UML.
- Apply design patterns to various applications.


REFERENCE BOOKS:

COURSE OBJECTIVES

- To learn web page creation.
- To understand the real time requirements of web page such as validation, use of DOM, role of XML.
- To understand OOP concepts and basics of Java language.
- To learn and use client server architecture based applications.
- To explore server side functionalities of an application.

COURSE OUTCOMES

Upon Completion of the course, the students should be able to:

- Make Web site creation and validation.
- Work with XML based technologies.
- Develop simple console application using Java.
- Develop GUI application using Swing and Applet.
- Build web based applications using JDBC, Servlet / JSP.

EXPERIMENTS IN THE FOLLOWING TOPICS:

- Creation of web pages having dynamic contents and validation using java script
- Creation of XML file and validation using XML schema and generation of XML using tools
- Simple xml based applications using DOM, SAX and XSL
- Basic Java programming covering objects, inheritance, polymorphism, interfaces, packages and exception handling
- String handling programs and regular expression programs
- Creation of applet based GUI's
- Application involving applet based GUI, JDBC, Servlet, JSP, cookies and session tracking.
COURSE OBJECTIVES

- Understand project planning
- Able to Analysis and Design Application
- Able to Implement and Test Software

COURSE OUTCOMES

- Able to Draw DFD and UML Diagrams
- Able to Write programs for Design
- Generate and Execute Test cases.

EXPERIMENTS IN THE FOLLOWING TOPICS:

Apply the following to typical application problems:

1. Project Planning
2. Software Requirement Analysis
3. Software Estimation
4. Software Design
5. Data Modelling & Implementation
6. Software Testing
7. Software Debugging

A possible set of applications may be the following:

a. Library System
b. Student Marks Analyzing System
c. Text Editor.
d. Create a dictionary.
e. Telephone dictionary.
f. Simulator Software for Parallel Processing Operation.
g. Inventory System.
LIST OF ELECTIVES

DCS5001 MOBILE COMPUTING

COURSE OBJECTIVES

Able

- To understand the principles of wireless communication
- To understand wireless LAN protocols
- To become familiar with mobile Routing protocols

COURSE OUTCOMES

Able to Develop Mobile Application

- Implement MAC protocols
- Able to perform Optimal Routing


UNIT IV MOBILE COMPUTING PLATFORM - PDA - Device characteristics and Software components - Smart Phone - Convergence of Mobile devices - J2ME - Modes, Data store, GUI support - HTTP Connection Interface Push Registry - Application development using Android APIs - Palm OS Architecture and Program Development - Overview of other mobile Operating Systems.


REFERENCE BOOKS:

COURSE OBJECTIVES

- Able to understand XML Data Representation
- Create Web Services

COURSE OUTCOMES

- Able to Design Web Databases
- To compose and Deploy Web Services
- Implement Security principles


REFERENCE BOOKS

COURSE OBJECTIVES

Able to

- Understand the equivalence of NFA, DFA and Regular Expression
- Write Grammar for a Language
- Develop a PDA

COURSE OUTCOMES

- Able to write programs for Logical Analysis
- Write programs to Syntax Analysis
- Implement stack operations for PDA

UNIT I  REGULAR EXPRESSIONS AND LANGUAGES - Introduction to Formal Proof – Additional Forms of proof – Inductive proofs – Regular Expressions – Regular and Non Regular Languages - Closure Properties of Regular Languages - Proving Languages Not to Be Regular - Decision Properties of Regular Languages.


UNIT V UNDECIDABILITY - Not Recursively Enumerable Language – Recursively Enumerable Undecidable problem– Undecidable Problems about Turing Machines – Post’s Correspondence Problem - The classes P and NP - NP-complete problems.

REFERENCE BOOKS:

COURSE OBJECTIVES

- To understand the basic concepts of graphics designs.
- To familiarize the student with the transformation and projection techniques.
- To expose the student to various color models.

COURSE OUTCOMES

Upon completion of the course, the students should be able to:

- Implement basic graphics transformation and projection techniques.
- Design an application that incorporates different concepts of various color models.
- Apply and explore new techniques in the areas of compression techniques.
- To appreciate the use of multimedia authoring tools and multimedia compression techniques.

UNIT I


UNIT II

2D TRANSFORMATIONS - Two dimensional transformations – Scaling and Rotations - Interactive Input methods - Polygons - Splines – Bezier Curves - Window view port mapping transformation.

UNIT III


UNIT IV


UNIT V


REFERENCE BOOKS:

COURSE OBJECTIVES

- To understand the hacking techniques of computer forensics.
- To learn about data recovery methods.
- To identify the threats in computer forensics

COURSE OUTCOMES

- Able to distinguish between hackers and normal users.
- To apply the principles of computer forensics for security.
- To implement the data recovery methods.
- To manage threats and the tactics.

UNIT I  ETHICAL HACKING - Foundation for Ethical Hacking-Ethical Hacking in Motion-Hacking Network Hosts-Hacking Operating Systems-Hacking Applications.


REFERENCE BOOKS:

COURSE OBJECTIVES

- To introduce the basics and necessity of Software testing.
- To introduce various testing techniques along with software production.
- To introduce the concepts of Software bugs and its impact.

COURSE OUTCOMES

Upon Completion of the course, the students should be able to:

- Perform automated testing using test tools.
- Document the testing procedures.

UNIT I  INTRODUCTION - Software Testing background – software bugs- cost of bugs-software testing realities- Testing Axioms – Precision and Accuracy-verification and validation- quality and reliability-testing and quality assurance.


UNIT III SOFTWARE TESTING TECHNIQUES - Configuration testing – Compatibility testing – foreign language testing – usability testing – testing the documentation - testing for software security – website testing.


REFERENCE BOOKS:

COURSE OBJECTIVES

Understand

- Cloud Services and Application
- Become familiar with MAP REDUCE Techniques
- Know about Cloud Tools and Programming

COURSE OUTCOMES

Able to

- Implement Virtualization Techniques
- Store and Retrieve Data in Cloud
- Provide Security through programs


UNIT IV  PROGRAMMING MODEL - Parallel and Distributed programming Paradigms – MapReduce , Twister and Iterative MapReduce – Hadoop Library from Apache – Mapping Applications - Programming Support - Google App Engine, Amazon AWS - Cloud Software Environments - Eucalyptus, Open nebula, OpenStack.


REFERENCE BOOKS:

3. Cloud Computing, A Practical Approach by Toby Velte, Anthony Velte, Robert Elsenpeter: TMH, 2018
4. Cloud Application Architectures: Building Applications and Infrastructure in the Cloud: Transactional Systems for EC2 and Beyond (Theory in Practice (O'Reilly)) by George Reese: O'Reilly
5. James E. Smith, Ravi Nair, Virtual Machines: Versatile Platforms for Systems and Processes, Elsevier/Morgan Kaufmann, 2005
COURSE OBJECTIVES

Able to

- Understand windows Environment
- Write programs in Visual Basic
- Understand the concepts of VC++

COURSE OUTCOMES

Able to

- Develop windows Applications
- User Interface Design using VB
- Application Development using VC++


REFERENCE BOOKS:

6. Herbert Sheildt, “MFC from the Ground Up”
COURSE OBJECTIVES

Able to Understand

- the TCP/IP networks
- Electronic payment systems
- E-Security principles

COURSE OUTCOMES

Able to create

- XML and Web Databases
- Implement Electronic payment systems
- Implement E-Security systems

UNIT I

UNIT II
Core Technology: Electronic Commerce Models - Shopping Cart Technology - Data Mining - Intelligent Agents – Internet Marketing - XML and E-Commerce.

UNIT III

UNIT IV

UNIT V

REFERENCE BOOKS: