**ST. XAVIER'S COLLEGE(AUTONOMOUS)**
**Palayamkottai - 627 002**
**Recognized as "College with Potential for Excellence" by UGC**
**Accredited by NAAC at A++ Grade with a CGPA of 3.66 out of 4 in IV cycle**

# SYLLABUS

# B.Sc. COMPUTER SCIENCE
**(w.e.f June 2021)**

# Programme Name: B.Sc. Computer Science
# Programme Code: CSC

## PROGRAMME SPECIFIC OUTCOMES

At the completion of the B.Sc. programme in Computer Science the students will be able to

- Apply fundamental principles and methods of Computer Science to a wide range of applications.
- Enable graduates to apply their programming skills to solve real world problems in society.
- Inculcate Algorithmic knowledge to solve mathematical problems.
- Formulate solution for computational problems.
- Design and implement software systems to meet the desired needs.
- Apply algorithms and mathematical concepts to design and Analysis of software.
- Developing software skills and training to tackle real world problems using latest Cloud technologies.

# B.Sc Computer Science Course Pattern

(With effect from June 2021)

| Sem | Part | Status | Sub. Code | Title of the Paper | Hrs | Cdt |
|---|---|---|---|---|---|---|
| I | I | Lang | 21UGT11 | General Tamil - I | 6 | 3 |
| | I | Lang | 21UGH11 | Hindi – I | | |
| | I | Lang | 21UGF11 | French –I | | |
| | II | Lang | 21UGE11 | General English - I | 6 | 3 |
| | III | Core-T1 | 21UCS11 | Programming in C | 4 | 4 |
| | III | Core-P1 | 21UCS12 | Practical – Programming in C | 4 | 2 |
| | III | Allied-T1 | 21UCSA11 | Digital Principles and Computer Architecture | 4 | 4 |
| | IV | NME1 | 21UNM11 | Microsoft Publisher | 2 | 2 |
| | IV | SBE1 | 21USB11 | Integrated Personality Development | 2 | 2 |
| | IV | VE | 21UVE11 | Ethics I / Religion I | 2 | 2 |
| | | | | | 30 | 22 |
| II | I | Lang | 21UGT21 | General Tamil - II | 6 | 3 |
| | I | Lang | 21UGH21 | Hindi – II | | |
| | I | Lang | 21UGF21 | French - II | | |
| | II | Lang | 21UGE21 | General English | 6 | 3 |
| | III | Core-T2 | 21UCS21 | Object Oriented Programming with C++ | 4 | 4 |
| | III | Core-P2 | 21UCS22 | Practical – Object Oriented Programming with C++ | 4 | 2 |
| | III | Allied-T2 | 21UCSA21 | Discrete Mathematics | 4 | 4 |
| | IV | NME2 | 21UNM21 | Photoshop | 2 | 2 |
| | IV | SBE2 | 21USB21 | Life Issues and Coping Skill Development | 2 | 2 |
| | IV | SBE3 | 21USB22 | Professional English for Computer Science | 2 | 2 |
| III | | | | | 30 | 22 |
| | III | Core-T3 | 21UCS31 | Programming in Java | 4 | 3 |
| | III | Core-T4 | 21UCS32 | Data Structures and Algorithms | 4 | 3 |
| | III | Core-P3 | 21UCS33 | Practical – Programming in Java | 4 | 2 |
| | III | Core-P4 | 21UCS34 | Practical - Data Structures and Algorithms | 4 | 2 |
| | III | Allied-T3 | 21UCSA31 | Programming in Python | 4 | 4 |
| | III | Allied-P1 | 21UCSA32 | Practical - Programming in Python | 4 | 2 |
| | IV | SBE4 | 21USB31 | Human Rights and Social Analysis | 2 | 2 |
| | IV | SBE5 | 21USB32 | R Programming | 2 | 2 |
| | IV | ES | 21UES31 | Environmental Studies | 2 | 2 |
| | | | | | 30 | 22 |
| IV | III | Core-T5 | 21UCS41 | Software Engineering | 4 | 3 |
| | III | Core-T6 | 21UCS42 | RDBMS Concepts and Oracle | 4 | 3 |
| | III | Core-T7 | 21UCS43 | Web Technology | 4 | 3 |
| | III | Core- P5 | 21UCS44 | Practical - Oracle | 4 | 2 |
| | III | Core- P6 | 21UCS45 | Practical – Web Technology | 2 | 1 |
| | III | Elective1 | 21UCSE41 | Operation Research/Graph Theory/ Theory of Computations/ Numerical and Statistical Methods | 4 | 3 |
| | III | Allied-T4 | 21UCSA41 | Microprocessor and Assembly Language Programming | 4 | 4 |
| | III | Allied-P2 | 21UCSA42 | Practical- Assembly Language Programming | 2 | 1 |
| | IV | SBE6 | 21USB41 | Internet Concepts and Web Design | 2 | 2 |
| | | | | | 30 | 22 |

|  |  |  |  |  |  |  |
|---|---|---|---|---|---|---|
| V | III | Core-T8 | 21UCS51 | Programming in Dot NET | 4 | 4 |
|  | III | Core-T9 | 21UCS52 | Operating System and Unix | 4 | 4 |
|  | III | Core-T10 | 21UCS53 | Mobile Application Development | 4 | 4 |
|  | III | Core-T11 | 21UCS54 | PHP and MySQL | 4 | 4 |
|  | III | Core-P7 | 21UCS55 | Practical - Programming in Dot NET | 4 | 2 |
|  | III | Core-P8 | 21UCS56 | Practical – Unix and Shell Programming | 2 | 1 |
|  | III | Core-P9 | 21UCS57 | Practical – Mobile Application Development | 2 | 1 |
|  | III | Core-P10 | 21UCS58 | Practical - PHP and MySQL | 2 | 1 |
|  | III | Elective2 | 21UCSE51 | Artificial Intelligence and Machine Learning, Internet of Things, Big Data Analytics and Deep Learning | 4 | 4 |
|  |  |  |  |  | 30 | 25 |
| VI | III | Core-T12 | 21UCS61 | Data Communications and Computer Networks | 5 | 5 |
|  | III | Core-T13 | 21UCS62 | Computer Graphics and Multimedia | 5 | 5 |
|  | III | Core-T14 | 21UCS63 | Cloud Computing | 5 | 5 |
|  | III | Core-P11 | 21UCS64 | Practical - Computer Graphics and Multimedia | 4 | 2 |
|  | III | Core-P12 | 21UCS65 | Practical – Cloud Computing | 4 | 2 |
|  | III | Project | 21UCSE61 | Project work and Viva Voce | 7 | 7 |
|  |  |  |  |  | 30 | 26 |
|  |  |  |  | STAND |  | 1 |
|  |  |  |  | TOTAL | 180 | 140 |

**Elective 1**

1. Operations Research
2. Graph Theory
3. Theory of Computation
4. Numerical and Statistical Methods

**Elective 2**

1. Artificial Intelligence and Machine Learning
2. Internet of Things
3. Big Data Analytics
4. Deep Learning

**ECC Papers**

1. Fundamentals of Computer
2. Internet concepts
3. Web design with style sheets
4. Visual Basic Dot Net
5. Wireless Technology

**Certificate Courses**

1. Desk Top Publishing
2. Mobile Apps with Android

**Add-on Courses**

1. Introduction to Robotics
2. PC Assembling and Trouble Shooting

## PROGRAMMING IN C -THEORY
## (21UCS11)

| SEMESTER-I | CORE-T1 | HOURS-4 | CREDITS-4 | TOTAL HOURS: 60 |
|---|---|---|---|---|

**Course Outcomes:**

**Upon completion of the course, the students will be able to**

- Show the syntax of the basic constructs of C language (K1)
- Differentiate different constructs of C language (K2)
- Apply arrays, structures, pointers and files in suitable situations (K3)
- Analyze and understand programs written  in C language (K4)
- Evaluate and debug programs written in C language (K5)
- Design algorithm and write program in C language for the given problem (K6)

**UNIT I** **(12 Hours)**

**Overview of C:** History of C-Importance of C-Basic Structure of C programs-Programming Style. **Constants, Variables and Data Types:** Character set-C Tokens-Keywords and Identifiers-Constants-Variables-Data Types-Declaration of Variables-Declaration of Storage Class-Assigning values to variables-Defining symbolic constants-Declaring a Variable as Constant. **Operators and Expressions:** Operators-Arithmetic Expressions-Evaluation of Expressions-Precedence of Arithmetic Operators-Type Conversions in Expressions-Operator Precedence and Associativity.

**UNIT II** **(12 Hours)**

**Managing Input and Output Operations:** Reading a character-Writing a character-Formatted Input-Formatted Output. **Decision making and Branching:** Decision making with IF –Simple IF Statement-The IF-ELSE statement-nesting of IF—ELSE Statements –The IF-ELSE Ladder-The Switch statement-The Ternary operator-The GOTO statement. **Decision Making and looping:** The WHILE statement-The DO Statement-The FOR Statement-Jumps in loops –concise test expressions.

**UNIT III** **(12 Hours)**

**Arrays:** One Dimensional Array-Two Dimensional Arrays-Multi-dimensional Arrays-Dynamic Arrays**. Character Arrays and Strings:** Declaring and initializing String Variables, Reading Strings from Terminal-Writing Strings to Screen-Arithmetic Operations on characters-Putting strings together-Comparing two strings-String handling functions-Array of Strings.

**UNIT IV** **(12 Hours)**

**User –Defined Functions:** Need, elements of User-defined functions-Definition of function-function calls-Function Declaration-Nesting of Functions-Recursion-Passing Arrays to functions-Passing String to functions-Scope, Visibility and lifetime of variables.

**Structure and Union:** Defining Structure-Declaring structure variables-Accessing Structure members-Structure Initialization-copying and comparing structure variables-Operations on individual members-Array of Structures-Array within Structures-Structure within Structure-Structure and functions-unions.

**UNIT V** **(12 Hours)**

**Pointers:** Understanding pointers-Accessing the address of a variable-Declaring Pointer variables-Initialization of pointer variables-Accessing a variable through its pointer-Chain of Pointers-Pointer expression-Pointers and arrays-Array of pointers-Pointers and functions-Pointers as function arguments-Functions returning pointers- Pointers and structures. **File Management in C:** Defining and Opening a File-Closing a File-Input/Output Operations on Files-Error Handling-I/O Operations-Random Access to Files-Command Line Arguments.

**Text Book:**

E.Balagurusamy, ”Programming in ANSI C”, Tata McGraw Hill Education India Pvt. Ltd., Seventh Edition, 2016.

**Reference Books:**

1. Yashavant Kanetkar, “Let us C”, BPB Publications; 15th Revised and Updated edition, 2016.
2. Salim Y. Amdani, “ 'C' Programming ”, Laxmi Publications, First edition, 2016.

## PRACTICAL - PROGRAMMING IN C
## (21UCS12)

| SEMESTER-I | CORE-P1 | HOURS-4 | CREDITS-2 | TOTAL HOURS: 60 |
|---|---|---|---|---|

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

1. Develop C programs for simple applications making use of Control Structures (K6)
2. Develop C programs using arrays and strings (K6)
3. Develop C programs involving functions, recursion (K6)
4. Develop C programs involving pointers (K6)
5. Develop C programs involving structures (K6)
6. Design applications using sequential and random access file processing (K6)

**List of Practical**

1. Simple C programs
2. Simple C program using functions
3. Programs based on control structures
    a) Decision Making (if, if-else, nested if-else, else if ladder)
    b) Looping (for, while, do-while)
4. Program using single dimensional Array
5. Program using two dimensional Arrays
6. Program using string handling functions
7. Program using recursive function
8. Program using Structure
9. Program using union
10. Program using pointers
11. Program to handle file
12. Program to handle file using command line argument

# DIGITAL PRINCIPLES AND COMPUTER ARCHITECTURE
# (21UCSA11)

| SEMESTER-I | ALLIED-T1 | HOURS-4 | CREDITS-4 | TOTAL HOURS: 60 |
|---|---|---|---|---|

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

- Definition of digital logics and Circuits(K1)
- Understand about the digital devices (K2)
- Understand about digital arithmetic circuits(K2)
- Acquire Knowledge on basics of Gates and its Applications(K4)
- Have the necessary understanding on Registers for Counting Applications (K4)
- Gain overall knowledge about the Computer Architecture (K3)

**UNIT I** **(12 Hours)**

**Digital Principles:** Definition of digital systems - Digital wave forms - Digital logic - Moving and storing information - Digital Operations. **Codes:** ASCII code - Excess 3 code - Gray code -Number System. **Digital Logic:** Basic Gates - Boolean algebra - Universal Gates - AND OR Invert gates.

**UNIT II** **(12 Hours)**

**Combinational Logic Circuits:** Boolean laws - Demorgan’s law-Sum of products - K map - K map simplifications - don’t care conditions - Product of sum simplification. **Data Processing Circuits:** Multiplexers – Demultiplexers - Decoders and Encoders - Ex OR gate - Parity generator and checkers - Read only Memory.

**UNIT III** **(12 Hours)**

**Arithmetic Circuits:** Binary addition – Binary subtraction - Unsigned binary numbers - 2’s Complement numbers - Arithmetic building blocks - The Adder-Subtracter. **Flip Flops:** RS Flip flop - Edge triggered RS Flip Flops - Edge triggered JK Flip Flops - Edge triggered D Flip Flops - JK Master Slave Flip Flop.

**UNIT IV** **(12 Hours)**

**Registers and Counters:** Introduction – Registers: Types of Registers – Universal Shift Register – Applications of Shift Registers –Counters: Asynchronous Counter – Decode Counter – Synchronous Counter – Decade Counter –Digital Clock.

**UNIT V** **(12 Hours)**

**Central Processing Unit:** General register organisation-Stack organisation - Instruction formats - Addressing Modes - Data transfer and manipulation. **Input-Output organization:** Peripheral devices - Input output interface - Asynchronous data transfer - Direct Memory Access - Input-Output processor.

**Text Books:**

1. Donald P Leach, Albert Paul Malvino, Goutam Saha, “Digital Principles and Applications”, McGraw-Hill Education, Eighth Edition. Eleventh Reprint 2018 (Units I,II, III & IV)
2. Morris Mano M, “Computer System Architecture”, Third Edition, Pearson Low Price Edition, reprinted 2014. (Unit V)

**Reference Books:**

1. Thomas C Bartee “Digital Computer Fundamentals”, Sixth Edition, Tata McGraw-Hill.
2. John P Hayes, “Computer Architecture and Organisation”, MGH, Third Edition.

## MICROSOFT PUBLISHER
## (21UNM11)

| SEMESTER – I | NME1 | HOURS –2 | CREDITS – 2 | TOTAL HOURS: 30 |
|---|---|---|---|---|

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

- Understand and use the terminology of typography and desktop publishing(K2)
- Describe what MS Publisher is and what its capabilities are.(K1)
- Determine aligning and formatting objects.(K3)
- Describe ways to customize publication.(K3)
- Demonstrate publishing, printing, and sharing publications(K4)
- Evaluate and redesign documents to improve appearance and functionality(K5)

**UNIT I** **( 6 Hours)**
Getting Started with Publisher - Starting New Publications - Saving and Closing Publications - Opening and Viewing Publications - Printing Publications - Working with Text Boxes - Working with WordArt.

**UNIT II** **( 6 Hours)**
Working with Graphics - Working with Shapes - Working with Pre-Designed Visual Elements - Creating Folded Cards - Creating Postcards - Using Mail Merge

**UNI III** **( 6 Hours)**
Creating Calendars - Working with the Master Page - Packaging Publications for Printing - Building Your Brand - Creating a Logo - Creating Flyers - Aligning and Stacking Objects.

**UNIT IV** **( 6 Hours)**
Creating Brochures - Flowing Text around Objects - Planning Longer Publication - Creating Newsletters.

**UNIT V** **( 6 Hours)**
Organizing Content - Working with a Table of Contents - Editing and Proofing Content - Creating a Basic E-Mail Message - Creating a Message from a Multi-Page Publication – Creating and Modifying a Web Site - Adding Text and Graphics to a Web Page.

**TEXT BOOK**
Joyce Cox, Joan Preppernau, Microsoft Office Publisher 2007 Step by Step, Microsoft Press, 2008.

**REFERENCE BOOK**
Jim McCarter and Jacqui Salerno Mabin, “Microsoft® Office Publisher 2007 For Dummies”, Wiley Publishing, Inc, 2008.

## List of Practical

1. Working with column layout
2. Working with Frames.
3. Formating and editing text.
4. Designing master Page.
5. Designing Invitation cards.
6. Creating table of contents.
7. Creating Index page.

## OBJECT ORIENTED PROGRAMMING WITH C++ -THEORY (21UCS21)

| SEMESTER-II | CORE-T2 | HOURS-4 | CREDITS-4 | TOTAL HOURS : 60 |
|---|---|---|---|---|

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

- Define complete overview of Data types, functions, control statements and pointers.(K1)
- Illustrate Console I/O Operations.(K2)
- Apply Object Oriented Programming Concepts.(K3)
- Demonstrate the use of virtual functions to implement polymorphism.(K3)
- Demonstrate Function Overloading and Operator Overloading concepts (K3)
- Illustrate Templates, Files and Exception Handling.(K4)

**UNIT I** **(12 Hours)**
**Principles of Object Oriented Programming:** Software Crisis - Software Evolution – Object Oriented Programming Paradigm – Basic Concepts of Object Oriented Programming - Benefits of OOP - Applications of OOP. **Beginning with C++ :** Applications of C++ - Simple Program – C++ Statements - Structure of C++ program – **Tokens, Expressions and Control Structures :** Tokens – Key words – Identifiers and Constants - Basic Data Types – User Defines Data Types – Storage Classes - Derived Data Types – Symbolic constants – Type Compatibility – Declaration of Variables -Reference variables – Operators in C++ - Scope resolution, Memory Dereferencing and Management operators – Manipulators – Type cast operator – Expressions – Operator Precedence – Control Structures.

**UNIT II** **(12 Hours)**
**Functions in C++ :** The Main Function – Function Prototyping - Call by Reference – Inline functions – Default and const Arguments – Function overloading – Friend and virtual functions – Math Library Functions - **Classes and objects :** Specifying a class - Defining member functions – A C++ Program with class - Arrays within a class – Memory Allocation for Objects - Static Data members and Member Functions – Arrays of objects – Objects as Function Arguments and Returning Objects - Const member function. **Constructors and Destructors :** Constructors - Parameterized Constructors – Copy Constructor – Dynamic Constructor – Destructors – **Operator Overloading and Type Conversions :** Defining Operator Overloading – Overloading Unary and Binary Operators – Rules for Operator Overloading – Type conversions.

**UNIT III** **(12 Hours)**
**Inheritance**: Defining Derived Classes – Single, Multilevel, Multiple, Hierarchical, Hybrid Inheritance - Virtual Base Classes – Abstract Classes – **Pointers, Virtual Functions and Polymorphism:** Pointers – Pointers to Objects - this pointer – Virtual Functions – Pure Virtual Functions.

**UNIT IV** **(12 Hours)**

**Managing console I/O operations :** C++ streams and Classes – Unformatted I/O operations - Formatted console I/O operations – Managing Output With Manipulators - **String Manipulation :** Creating and Manipulating String Objects – Relational Operations – String Characteristics – Accessing Characters in Strings – Comparing and Swapping.

**UNIT V** **(12 Hours)**

**Working with files :** Classes and File Stream Operations – Opening and Closing a File – Detecting End-of-File – Open() File Modes – File Pointers and Their Manipulations – Sequential Input and Output operations – Updating a file – Command Line Arguments - **Templates :** Class Templates – Function Templates – **Exception Handling :** Basics of Exception Handling – Exception Handling Mechanism – Throwing, Catching and Rethrowing Mechanisms.

**TEXT BOOK:**

E.Balagurusamy,"Object Oriented Programming with C++", Tata McGraw Hill, Seventh Edition, 2017.

**REFERENCE BOOKS:**

1. D.Ravichandran, "Programming with C++", Tata McGraw Hill, Third Edition, 2012.
2. Robert Lafore, "Object Oriented Programming with C++", Galgotia Publications Pvt Ltd., Fourth Edition, 2002.
3. Herbert Schildt, "C++ the Complete Reference ", Tata McGraw Hill Fourth Edition, 2006.

## PRACTICAL - OBJECT ORIENTED PROGRAMMING WITH C++ (21UCS22)

| SEMESTER-II | CORE-P2 | HOURS-4 | CREDITS-2 | TOTAL HOURS : 60 |
|---|---|---|---|---|

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

- Use the  control structure syntax while writing programs(K3)
- Apply Object Oriented Concept while writing programs(K3)
- Demonstrate function overloading concept (K3)
- Illustrate the use of Pointers in memory management (K4)
- Illustrate the use of function and class templates (K4)
- Design programs with exception handling concept(K6)

### List of Practical

1. Simple Programs.
2. Program using control structures
3. Program using one dimensional Array.
4. Program using two dimensional Arrays.
5. Program using Structure and Union.
6. Program using class and Objects
7. Program using Constructor and overloading constructor.
8. Program using Inheritance (Different forms)
9. Program using Function Overloading
10. Program using Operator Overloading
11. Program using Pointer Arithmetic
12. Program using Virtual Functions
13. Program using Friend Function and Inline function
14. Program using Templates
15. Program using Stream (File) Operations

## DISCRETE MATHEMATICS –THEORY
## (21UCSA21)

| SEMESTER-II | ALLIED-T2 | HOURS-4 | CREDITS-4 | TOTAL HOURS : 60 |
|---|---|---|---|---|

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

- Define concepts of set theory (K1)
- Describe algorithms for solving problems (K2)
- Use logics and inferences (K3)
- Apply graph theory for solving problems (K3)
- Illustrate the concept of counting (K4)
- Select proper functions and recursion for solving problems (K4)

**UNIT I** **( 12 Hours)**

**SET THEORY:** Introduction – sets and elements – universal set and empty set – Subsets – Venn Diagrams – set operations – Algebra of sets and duality – finite sets, counting principle – class of sets, power sets, and partitions – mathematical induction.
**RELATIONS** – Introduction – product sets – relations – pictorial representations of relations – composition of relations – types of relations – closure properties – equivalence relations – partial ordering relations – n-ary relations.

**UNIT II** **( 12 Hours)**

**FUNCTIONS AND ALGORITHMS**: Introduction – functions – One – to – one – Onto and Inevitable functions – mathematical functions, exponential and logarithmic functions – sequences, indexed classes of sets – recursively defined functions – cardinality – algorithms and functions – complexity of algorithms.

**UNIT III** **( 12 Hours)**

**LOGIC AND PROPOSITIONAL CALCULUS:** Introduction – propositions and Compound propositions – Basic logical operations – propositions and truth tables – Tautologies and contradictions – logical equivalences – algebra of propositions – conditional and biconditional statements – arguments – logical implication – prepositional functions, Quantifiers – Negation of quantified statements

**UNIT IV** **( 12 Hours)**

**COUNTING:** Introduction, Basic counting principles – factorial Notation – Binomial coefficients – permutations – combinations – the pigeonhole principle – the inclusion – exclusion principle – ordered and unordered partitions.

**UNIT V** **(12 Hours)**

**GRAPH THEORY:** Introduction, data structures – graphs and multigraphs – subgraphs, Isomorphic and homeomorphic graphs – paths, connectivity – the bridges of konigsberg, traversable multigraphs – labeled and weighted graphs – complete, regular, and bipartite graphs – tree graphs-Minimum Spanning Trees-Directed Graphs- Basic Definitions- Rooted Trees – Graph Coloring.

**Text Book:**

Seymour Lipschutz, Marc Lipson Discrete Mathematics Third Edition, Tata McGraw Hill, 2010.

**Books for Reference:**

1. B.S.Vatsa, "Discrete Mathematics", Wishwa Prakashan, Third Edition., 2013
2. K.D.Joshi, "Foundation of Discrete Mathematics", Wiley Eastern Ltd.

## PHOTOSHOP
## (21UNM21)

| SEMESTER – II | NME2 | HOURS – 2 | CREDITS – 2 | TOTAL HOURS : 30 |
|---|---|---|---|---|

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

- Basics of digital images and its properties(K1)
- Understand the essential of Photoshop.(K2)
- Develop simple application using Photoshop.(K3)
- Understand the layers in Photoshop.(K2)
- Understand the filters in Photoshop.(K2)
- Update knowledge to learn any advanced version of the software.(K4)

**UNIT I** **( 6 Hours)**

Working with Photoshop files – Color models

**UNIT II** **( 6 Hours)**

Toolbox and Palette Essentials – Color Management.

**UNIT III** **( 6 Hours)**

Working with layers – layer styles

**UNIT IV** **( 6 Hours)**

Working with text – text effects – channel and mask essentials

**UNIT V** **( 6 Hours)**

Working with filters.

**TEXT BOOK**

Robert Shufflebotham, “Photoshop CS6 in Easy Steps”, In Easy Steps, 2012.

## List of Practical

1. Designing 3D text.
2. Creating text effects.
3. Creating  ID – Cards
4. Designing Banners.
5. Creating text mask.
6. Applying Filters.
7. Converting black and white photo into color photo

## PROFESSIONAL ENGLISH FOR PHYSICAL SCIENCES
### (21USB22)

| SEMESTER-II | SBE3 | HOURS-2 | CREDITS-2 | TOTAL HOURS: 30 |
|---|---|---|---|---|

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

- Recognise their own ability to improve their own competence in using the language(K1)
- Use language for speaking with confidence in an intelligible and acceptable manner(K2)
- Understand the importance of reading for life(K2)
- Read independently unfamiliar texts with comprehension(K1)
- Understand the importance of writing in academic life(K2)
- Write simple sentences without committing error of spelling or grammar (K3)

**UNIT I COMMUNICATION** **( 6 Hours)**

**Listening**: Listening to audio text and answering questions
- Listening to Instructions

**Speaking**: Pair work and small group work.
**Reading :** Comprehension passages –Differentiate between facts and opinion
**Writing :** Developing a story with pictures**.**
**Vocabulary:** Register specific - Incorporated into the LSRW tasks

**UNIT II DESCRIPTION** **( 6 Hours)**
**Listening:** Listening to process description.-Drawing a flow chart.
**Speaking:** Role play (formal context)
**Reading :** Skimming/Scanning-
Reading passages on products, equipment and gadgets.
**Writing :** Process Description –Compare and Contrast
Paragraph-Sentence Definition and Extended definition-
Free Writing.
**Vocabulary:** Register specific -Incorporated into the LSRW tasks.

**UNIT III NEGOTIATION STRATEGIES** **( 6 Hours)**

**Listening:** Listening to interviews of specialists / Inventors in fields
(Subject specific)
**Speaking**: Brainstorming. (Mind mapping).
Small group discussions (Subject- Specific)
**Reading:** Longer Reading text.
**Writing:** Essay Writing (250 words)
**Vocabulary:** Register specific - Incorporated into the LSRW tasks

### UNIT IV PRESENTATION SKILLS ( 6 Hours)

**Listening**: Listening to lectures.
**Speaking:** Short talks.
**Reading :** Reading Comprehension passages
**Writing :** Writing Recommendations
Interpreting Visuals inputs
**Vocabulary:** Register specific - Incorporated into the LSRW tasks

### UNIT V CRITICAL THINKING SKILLS ( 6 Hours)

**Listening:** Listening comprehension- Listening for information.
**Speaking**: Making presentations (with PPT- practice).
**Reading** : Comprehension passages –Note making.
Comprehension: Motivational article on Professional Competence, Professional Ethics and Life Skills)
**Writing:** Problem and Solution essay– Creative writing –Summary writing
**Vocabulary:** Register specific - Incorporated into the LSRW tasks

## PROGRAMMING IN JAVA - THEORY
## (21UCS31)

**SEMESTER–III CORE-T3 HOURS–4 CREDITS–3 TOTAL HOURS: 60**

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

- Define Object Oriented Programming Paradigm using Java language (K1)
- Describe the given problem using the syntactical structures of JAVA language (K2)
- Use the programming skill to debug and run the programs (K3)
- Implement various object oriented concepts (K5)
- Design an algorithmic solution for a given problem in JAVA language (K6)

### UNIT I (12 Hours)

**An overview of Java:** Object oriented Programming – A first simple program – A second short program – Two control statements – using blocks of code – lexical issues – Java class libraries. Data Types, variables and **Arrays:** Java is a strongly typed language – The simple styles – Integers - Floating point types – Characters – Boolean – a close look at literal – variables Type Conversion casting – Automatic type promotion in expressions – Arrays.
**Operators:** Arithmetic operators - the bitwise operator - relational operator – Boolean logical operator the assignment operator – the ?: Operator – operator precedence – using parenthesis. **Control statements:** Java selection statements – Iteration statements – Jump statements.

### UNIT II (12 Hours)

**Introducing Classes:** Class fundamentals – Declaring objects – Assigning object reference variables – Introducing methods – Constructors- The this keyword – Garbage collection – The finalized method – A stack class. **Inheritance:** Inheritance basics - using super – creating a multilevel hierarchy – when constructors are called – method overriding – dynamic method dispatch – using abstract classes - using final with inheritance - the object class.
**Packages and Interfaces:** Packages – access protection – interface.

### UNIT III (12 Hours)

**Exception Handling:** Fundamentals - Exception types – Uncaught exceptions – Using try and catch clauses – nested try statements – throw –throws- Java's built in exceptions – creating your own exceptions subclasses. **Multithreaded Programming :** The Java Thread Model – The Main Thread – Creating Thread – Creating Multiple Threads – Using isAlive() and join() – Thread Priorities – Synchronization - Interthread Communication – Suspending , Resuming and stopping Threads – using Multithreading.

**UNIT IV** **(12 Hours)**

**The Applet Class :** Applet basics – Applet architecture – an applet skeleton – Simple applet display methods – Requesting repainting – using the status window – the HTML APPLET tag – passing parameter to applets – GainDocumentBase() and GainCodebase() – Applet Context and show Document – the AudioClip Interface – The Apple Stub Interface – Outputting to the console. **Event handling:** Two event handling mechanisms – the delegation event model – Event classes – Sources of events – Event listener interfaces – Using the delegation event model – Adapter classes – Inner classes - String handling.

## UNIT V **(12 Hours)**

**Introducing AWT:** AWT Classes - Window fundamentals – Frame windows – Working with graphics, color and fonts – managing text output using font metrics –**AWT Controls:** Labels – Buttons – CheckBox - Choice control – list – scrollbar – Textfields – TextArea - , Layout Managers – Menubars and menus – DialogBoxes – FileDialog – Handling events by extending AWT components.

**Text book:**

Herbert Schildt, “The Complete Reference Java 2”, McGraw Hill Publication(India), Eighth Edition, 2011.

**Reference Books:**

1. John Zukowski, “Mastering Java2”, BPB Publications, First Indian Edition, 2000
2. Aaron Walsh and John Fronckoviak, “Java Programming Bible”, IDG Books World wide Inc, First Edition, 2000.
3. Cay S. Horstmann and Garry Cornell, “Core Java 2”, Pearson Education Asia, First Edition, 2001.
4. Deborah S. Ray and Eric J. Ray, “Mastering HTML 4.0”, BPB Publications, First Indian Edition, 1998.
5. C Xavier, “World Wide Web Design with HTML”, Tata McGraw Hill Publication, First Edition, 2000.

## DATA STRUCTURES AND ALGORITHMS - THEORY
## (21UCS32)

**SEMESTER–III CORE-T4 HOURS–4 CREDITS–3 TOTAL HOURS : 60**

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

- Define data structure and algorithms (K1)
- Describe data structures like stack, queue, tree and graph (K2)
- Apply data structures in solving the problems (K3)
- Use algorithm techniques for solving problems. (K3)
- Analyze the time complexity of algorithms (K4)
- Assess various algorithmic techniques (K5)

**Unit I** **( 12 Hours)**
**Introduction :** History of Algorithms – Definitions – Structure and Properties of Algorithms – Development of an algorithm – Data Structures and Algorithms – Data Structure Definition and Classification.
**Analysis of Algorithms:** Efficiency of Algorithms – Apriori Analysis – Asymptotic Notations – Time complexity of an Algorithm using O notation – Polynomial versus Exponential Algorithms – Average, Best and Worst Case complexities – Analyzing recursive programs.
**Arrays:** Introduction – Array Operations – Number of elements in an array – Representation of arrays in memory – Applications.

**Unit II** **( 12 Hours)**
**Stacks:** Introduction – Stack operations – Applications
**Queues:** Introduction – operations on Queues – Circular Queues – Other Types of Queues – Applications.
**Linked Lists :** Introduction – Singly linked list – Circularly linked list – Doubly linked list – Multiply linked list – Applications.

**Unit III** **(12 Hours)**
**Trees and Binary Trees:** Introduction – Trees: Definition and basic terminology – Representation of Trees – Binary Trees: Basic Terminology and types – Representation of Binary Trees – Binary Tree Traversal – Threaded Binary Tree – Applications.
**Graphs:** Introduction – Definition and Basic Terminology – Representation of Graphs – Graph Traversals – Application.

**Unit IV** **( 12 Hours)**
**Searching:** Introduction – Linear Search – Transpose Sequential Search – Interpolation Search – Binary Search – Fibonacci Search – Other Search Techniques.

**Unit V** **( 12 Hours)**
**Sorting:** Internal Sorting - Introduction – Bubble sort – Insertion sort – Selection sort – Merge sort – Shell sort – Quick sort – Heap sort – Radix sort.

**TEXT BOOK**:

G.A.Vijayalakshmi Pai, “Data Structures and Algorithms Concepts, Techniques and Applications” , Tata Mcgraw Hill Publishing Company Limited, New Delhi, 2008.

**REFERENCE BOOKS:**

1. Ellis Horowitz and Sartaj Sahani, “Fundamentals of Data Structures”, Computer Science Press Inc, Galgotia Book Sources Publishers, New Delhi., 2010.
2. Ellis Horowitz and Sartaj Sahani, “ Fundamentals of Computer Algorithms”, Computer Science Press Inc, Galgotia Book Sources Publishers, New Delhi., 2016.

## PRACTICAL - PROGRAMMING IN JAVA
## (21UCS33)

**SEMESTER–III CORE–P3 HOURS–4 CREDITS – 2 TOTAL HOURS : 60**

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

- Describe the basics and syntax of JAVA language (K1)
- Write programs for simple problems using JAVA language (K3)
- Illustrate Multithreading using JAVA (K4)
- Update knowledge to learn any future advanced version of language (K5)
- Develop programming skill in handling advanced concepts in Java (K6)

1. Program using class and Objects
2. Program using Constructor and Method overloading.
3. Program using Inheritance
4. Program using Interfaces
5. Program using Packages
6. Program using Exception Handling
7. Program using Multithreading
8. Simple Applet Program
9. Program for Keyboard and Mouse event handling
10. Program for String Handling.

## PRACTICAL - DATA STRUCTURES AND ALGORITHMS (21UCS34)

**SEMESTER–III CORE–P4 HOURS–4 CREDITS–2 TOTAL HOURS : 60**

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

- Understand the use arrays in polynomial addition (K2)
- Write program for STACK operation (K3)
- Developing programming skill in handling Queues (K3)
- Effectively handle linked list(K2)
- Understand various sorting methods (k2)

1. Polynomial addition using arrays.
2. Implementation of Stack
3. Implementation of Queue
4. Single Linked list.
5. Doubly Linked list
6. Implementation of Search Techniques.
7. Merge sort.
8. Quick sort.

## PROGRAMMING IN PYTHON - THEORY
## (21UCSA31)

**SEMESTER–III ALLIED-T3 HOURS–4 CREDITS-4 TOTAL HOURS : 60**

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

1. Describe Object Oriented Programming Paradigm through Python Programming. (K1)
2. Describe a systematic approach to design, organize, write and debug programs (K2)
3. Analyze the various data structures available in Python programming language and apply them in solving computational problems. (K4)
4. Develop proficiency in creating applications using the Python Programming Language. (K5)
5. Update knowledge to learn any future advanced version of language (K6)

**UNIT I** **(12 Hours)**

**Introduction to Python:** Features of Python – Execution of a Python program – Flavors of Python – Python Virtual machine (PVM) – Memory Management in Python – Garbage Collection – Comparison of Python with C and Java. **Datatypes in Python:** Built in Data types: None Type - Numeric types: int, float, complex - datatype conversion - bool datatype - Sequences : string , bytes , bytearray , list , tuple , range - set datatype – mapping datatype - literals. **Operators:** Arithmetic operators – Assignment operators – Unary minus operator – Relational operators – Logical operators – Boolean operators – Bitwise operators - Membership operators – Identity operators - Operator precedence - Mathematical functions.

**UNIT II** **(12 Hours)**

**Input and Output:** print() - input() - command line arguments. **Conditionals and Loops:** if statement - if…else statement - if…elif statement - while loop - for loop - the else suite - break statement - continue statement - pass statement - assert statement - return statement . **Arrays in Python:** Creating array – Importing the array module – Indexing and slicing on arrays – Types of arrays – Working with arrays using numpy – Mathematical operations on arrays – Working with multidimensional array – Matrices in numpy.

**UNIT III** **(12 Hours)**

**Strings and characters:** Slicing the strings – String functions and methods – working with characters. **Functions:** Defining a function – Calling a function – Pass by object reference – Recursive functions – lambda functions – Function decorators – Generators. **Lists:** list operations – list slices – aliasing and cloning list - Methods to process lists – Nested list – list comprehension. **Tuples:** Creating tuples - Basic operations on tuples – Functions to process tuples.

**UNIT IV** **(12 Hours)**

**Dictionaries:** Operations on dictionary – Dictionary methods – Using loops with dictionaries – Converting lists, strings into dictionary – Passing dictionary to functions – Ordered dictionaries. **Exceptions:** Errors in Python program - Exception - Types of exceptions - except Block - assert statement - user defined exceptions - logging the exception. **Files:** text files - binary files - opening a file - closing a file - working with textfile - working with binary files - pickle in Python - seek() and tell() methods - random accessing of binary files - zipping and unzipping files - working with directories.

**UNIT V** **(12 Hours)**

**Classes and objects:** Creating a class - the self variable - constructor – instance variables - class variables - namespaces - Instance methods - class methods - static methods - passing members of one class to another class - inner class **Inheritance:** Constructors in inheritance - overriding super class constructors and methods - super() method - types of inheritance - Method Resolution Order (MRO) **Polymorphism:** Duck Typing Philosophy of Python - Operator overloading - Method overloading - Method overriding.

**Text book:**

R. Nageswara Rao, "Core Python Programming", Second Edition, Dreamtech Press, 2019.

**Reference Books:**

1. Guido van Rossum and the Python development team, "An Introduction to Python - Revised and updated for Python 3.6.2", Shroff Publishers & Distributors Pvt.Ltd , 2017.
2. Allen B.Downey, " Think Python: How to Think Like a computer Scientist" , Second Edition, Updated for Python 3, Shroff Publishers & Distributors Pvt.Ltd , 2015.
3. Charles Dierbach, "Introduction to Computer Science using Python; A Computational Problem-Solving Focus", Wiley India Edition, 2013.

## PRACTICAL - PROGRAMMING IN PYTHON
## (21UCSA32)

**SEMESTER–III ALLIED–P1 HOURS–4 CREDITS-2 TOTAL HOURS :60**

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

- Describe the basics and syntax of Python language (K1)
- Write programs for simple problems using Python language (K3)
- Developing programming skill in handling advanced concepts in Python (K4)
- Update knowledge to learn any future advanced version of language (K5)
- Develop programming skill in handling advanced concepts in Python (K6)

1. Simple Programs
2. Programs using Control Structures
3. Programs sing Sequences
4. Programs using Strings
5. Programs using Lists
6. Programs using Tuples
7. Programs using Files
8. Programs for Exception Handling
9. Programs using Functions
10. Programs to demonstrate Scope of Variables
11. Programs using Recursion
12. Programs using Generators
13. Programs using Classes
14. Programs using Inheritance
15. Programs using Delegation

## R PROGRAMMING
## (21USB32)

| SEMESTER–III | SBE-5 | HOURS–2 | CREDITS-2 | TOTAL HOURS: 30 |
|---|---|---|---|---|

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

- To teach how to use R for statistical programming, computation, graphics, and modeling
- To Write functions and use R in an efficient way
- To help the students expand their knowledge of R
- To learn how to handle R package
- Update knowledge to learn any advance version of the software

**UNIT I** **( 6 Hours)**

**Gainting started:** Introduction to Functions - Important R Data Structures - Gainting Help - vectors - Declarations - Common Vector Operations - Using all() and any() - Vectorized Operations - NA and NULL Values.

**UNIT II** **( 6 Hours)**

**Filtering:** The Selection Function which() - The ifelse() Function - Vector Element Names - Matrices and Arrays - General Matrix Operations - Filtering on Matrices - Applying Functions to Matrix Rows and Columns - Adding and Deleting Matrix Rows and Columns - More on the Vector/Matrix Distinction - Higher-Dimensional Arrays - Lists - Creating Lists - General List Operations - Accessing List Components and Values - Applying Functions to Lists - Recursive Lists.

**UNIT III** **( 6 Hours)**

**Data frames:** Creating Data Frames - Other Matrix-Like Operations - Merging Data Frames - Applying Functions to Data Frames - Factors and Tables - Factors and Levels - Common Functions Used with Factors.

**UNIT IV** **( 6 Hours)**

**Working with Tables:** Other Factor- and Table-Related Functions - R Programming Structures - Control Statements - Arithmetic and Boolean Operators and Values - Default Values for Arguments - Return Values - Functions Are Objects - Environment and Scope Issues - No Pointers in R - Writing Upstairs - Recursion - Replacement Functions - Anonymous Functions.

**UNIT V** **( 6 Hours)**

**Doing Math and Simulations in R:** Math Functions - Functions for Statistical Distributions - Sorting - Linear Algebra Operations on Vectors and Matrices - Set Operations - Simulation Programming in R.

**Text Book**

Norman Matloff, "The Art of R Programming: A Tour of Statistical Software Design", No Starch Press, 2011.

**Reference Books**

1. Michael J. Crawley, “The R Book”, John Wiley & Sons Ltd., 2007.
2. Jared P. Lander, “R for Everyone”, Pearson Education, Inc., 2014.

## Practical List:

1. Write a program that prints ‘Hello World’ to the screen.
2. Write a program that asks the user for a number n and prints the sum of the numbers 1 to n
3. Write a program that prints a multiplication table for numbers up to 12.
4. Write a function that returns the largest element in a list.
5. Write a function that computes the running total of a list.
6. Implement the following sorting algorithms: Selection sort, Insertion sort, Bubble Sort
7. Implement matrices addition, subtraction and Multiplication

# SOFTWARE ENGINEERING
# (21UCS41)

**SEMESTER–IV CORE-T5 HOURS–4 CREDITS-3 TOTAL HOURS: 60**

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

- Identify the key activities in managing a software project (K1)
- Describe concepts of requirements engineering and Analysis Modeling (K2)
- Apply systematic procedure for software design and deployment (K3)
- Classify different process models (K4)
- Compare and contrast the various testing and maintenance (K5)
- Manage project schedule, estimate project cost and effort required (K6)

## UNIT I SOFTWARE PROCESS AND AGILE DEVELOPMENT ( 12 Hours)

**Introduction to Software Engineering:** Software Process- Perspective and Specialized Process Models –Introduction to Agility-Agile process-Extreme programming-XP Process.

## UNIT II REQUIREMENTS ANALYSIS AND SPECIFICATION ( 12 Hours)

**Software Requirements:** Functional and Non-Functional, User requirements, System requirements, Software Requirements Document – **Requirement Engineering Process:** Feasibility Studies, Requirements elicitation and analysis, requirements validation, requirements management Classical analysis: Structured system Analysis, Petri Nets- Data Dictionary.

## UNIT III SOFTWARE DESIGN ( 12 Hours)

**Design Process**: Design Concepts-Design Model– Design Heuristic – Architectural Design - Architectural styles, Architectural Design, Architectural Mapping using Data Flow- User **Interface Design:** Interface analysis, Interface Design- **Component level Design:** Designing Class based components, traditional Components.

## UNIT IV TESTING AND MAINTENANCE ( 12 Hours)

**Software testing fundamentals**: Internal and external views of Testing-white box testing - basis path testing-control structure testing-black box testing- Regression Testing – Unit Testing – Integration Testing – Validation Testing – System Testing and Debugging – **Software Implementation Techniques:** Coding practices-Refactoring-Maintenance and Reengineering-BPR model-Reengineering process model-Reverse and Forward Engineering.

## UNIT V PROJECT MANAGEMENT ( 12 Hours)

**Software Project Management:** Estimation – LOC, FP Based Estimation, Make/Buy Decision COCOMO I & II Model – Project Scheduling – Scheduling, Earned Value Analysis Planning – Project Plan, Planning Process, RFP- **Risk Management**: Identification, Projection - Risk Management-Risk Identification-RMMM Plan

**Text Book:**

Roger S. Pressman, ―Software Engineering – A Practitioner's Approach, Seventh Edition, Mc Graw-Hill International Edition, 2010.

**Reference Books:**

1. Ian Sommerville, Software Engineering, 9th Edition, Pearson Education Asia, 2011.
2. Richard Fairly, "Software Engineering", Tata McGraw Hill publication., 2017.

## RDBMS CONCEPTS AND ORACLE -THEORY
## (21UCS42)

| SEMESTER–IV CORE–T6 HOURS–4 CREDITS-3 TOTAL HOURS : 60 |
|---|

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

- Explain the fundamentals of RDBMS (K2)
- Use SQL queries in a procedural language, PL/SQL (K3)
- Design database using ER diagram and normal forms (K4)
- Update knowledge to learn any future advanced version of language (K5)
- Create and manipulate relational database using Oracle (K6)

**UNIT I** **(12 Hours)**

Purpose of database systems – view of data – database languages – relational databases – database design – data storage and querying – transaction management - database architecture – database users and administrators - Structure of relational databases – database schema – keys – schema diagrams – relational query languages – relational operations.

**UNIT II** **(12 Hours)**

Overview of design process – E-R model – constraints - E-R diagrams – Reduction to relational schemas – E-R design issues - extended E-R features - Features of good relational design – atomic domains and first normal form – decomposition using functional dependencies – functional dependency theory

**UNIT III** **(12 Hours)**

Naming rules and conventions – data types – constraints – creating table – displaying information – altering existing table – dropping, renaming and truncating a table.
Adding new records – updating and deleting records – retrieving data from table – arithmetic operations – where clause – sorting – CASE.

**UNIT IV** **(12 Hours)**

Built-in functions – grouping data – join – set operators – subquery – top-N analysis – correlated subquery – views – sequences – synonyms – index – transactions – locking rows for update – controlling access.

**UNIT V** **(12 Hours)**

Fundamentals of PL/SQL – PL/SQL block structure – comments – data types – variable declaration – bind variable – control structures – SQL in PL/SQL – data manipulation in PL/SQL – cursors – exception handling – procedure – function – packages – trigger.

**Text Books:**

1. Abraham Silberschatz, Henry F. Korth and S. Sudarsan, ”Database system concepts”, Sixth edition, McGraw Hill, 2011. ( for units I & II Chapters 1,2, 7 & 8))
2. Nilesh Shah, ”Database Systems using Oracle A simplified guide to SQL and PL/SQL”, Second Edition, Prentice Hall of India, 2010. ( for units III, IV & V , Chapters 4-12 & 14)

**References:**

1. Alexis Leon and Mathews Leon, Fundamentals of Database Management Systems, Vijay Nicole Imprints, 2010.
2. Scott Urman, “Oracle 9i Pl/SQL programming”, Tata McGraw Hill, 2006.
3. Ivan Bayross, “SQL, PL/SQL, The programming language of Oracle”, BPB Publications, 2010.

## WEB TECHNOLOGY - THEORY
## (21UCS43)

| SEMESTER–IV | CORE–T7 | HOURS–4 | CREDITS-3 | TOTAL HOURS: 60 |
|---|---|---|---|---|

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

- Define the concepts of web application development (K1)
- Describe JavaScript as a dynamic webpage creating tool(K2)
- Use Table and Forms in web page designing process(K3)
- Illustrate CSS in web development (K4)
- Design innovative web applications (K6)
- Integrate various techniques to develop creative web applications (K6)

**Unit I** **( 12 Hours)**

HTML: Introduction - Tags - Attributes - Elements - Formatting tags - Anchor - Image - Table - Lists - Form - Form input - Iframes - Div Tag - Comments - HTML Responsive - HTML5: Introduction - HTML5 Tags - Html5 form inputs.

**Unit II** **( 12 Hours)**

Introduction - Syntax - Inline CSS - External CSS - Internal CSS - comments - background - border - border radius - cursor - buttons - flaot - fonts - colors - hover - line height - margin - padding - shadow - CSS3 : 2D and 3D Transform - Box - Animation - Web font.

**Unit III** **( 12 Hours)**

Bootstrap : Introduction - Basic Example - Container - Jumbotron - Button - Grid - Table - Form - Alert - Wells - Badge - Panels - Image - Glyphicon - Carousel - List Group - Dropdown - Collapse - Tabs - Navbar - Input types - Modals – Popover.

**Unit IV** **( 12 Hours)**

Javascript : Introduction - Basic Example - Internal and External Javascript - Comment - Varaible - Function - Control Statements - Loops - JS Objeccts: Js Object - Array - String - Math. JS DOM: Document Object - getElementById - getElementsByName() - getElementsByTagName() - innerHTML - innerText – AJAX.

**Unit V** **( 12 Hours)**

JSON : Introduction - Basic Example - Object - Array - Comments - Parse JSON Data - XML: Introduction - Features - Basic Example - Attributes - Comments - Validation - DTD - XML Parsers.

**Text Books**

1. Paul Deital, Harvey Deitel& Abbey Deitel,, "Internet and World Wide Web - How to Program", Pearson, Fifth Edition, 2012
2. Matt Lambert, "Learning Bootstrap 4", Packt Publishing, Second Edition, 2016.

**References**

1. Fabio Cimo, "Bootstrap Programming Cookbook", from Web Developers Resource Center.
2. Chris Bates, Web Programming – Building Intranet Applications, 3 rd Edition, Wiley Publications, 2009.
3. Jeffrey C and Jackson, “Web Technologies A Computer Science Perspective”, Pearson Education, 2011.
4. Gopalan N.P. and Akilandeswari J., “Web Technology”, Prentice Hall of India, 2011.

## PRACTICAL - ORACLE
## (21UCS44)

| SEMESTER–IV | CORE–P5 | HOURS-4 | CREDITS–2 | TOTAL HOURS : 60 |
|---|---|---|---|---|

**Course Outcomes:**

**Upon completion of the course, the students will be able to**

- Explain the basics and syntax of ORACLE (K1)
- Apply DDL, DML in ORACLE (K3)
- Handle functions, procedures, packages and reports in ORACLE (K4)
- Develop programming skill in handling advanced concepts using PL/SQL (K5)
- Update knowledge to learn any future version of language (K6)

1. Creating, modifying and dropping Tables.
2. Creating tables with referential and check constraints.
3. Inserting, modifying, deleting rows.
4. Dropping ,disabling /enabling constraints
5. Retrieving rows with operators in where Clause.
6. Retrieving rows with Character functions.
7. Retrieving rows with Number and Date functions.
8. Retrieving rows with Group functions and HAVING.
9. Joining Tables. (Inner and Outer).
10. Retrieving rows with Sub Queries.
11. Simple PL/SQL programs.
12. PL/SQL programs with control structures.
13. PL/SQL programs with Cursors.
14. PL/SQL programs with Exception Handling.
15. Creating and Calling Procedures.
16. Creating and Calling Functions.
17. Creating and Calling Packages.
18. Overloading Packages.
19. Working with Triggers.
20. SQL PLUS Reports.

# PRACTICAL – WEB TECHNOLOGY
# (21UCS45)

**SEMESTER–IV CORE–P6 HOURS–2 CREDITS–1 TOTAL HOURS:30**

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

- Basics and syntax of HTML tags(K1)
- Write programs for simple problems using HTML and Style Sheets(K3)
- Develop programming skill in table forms and frames using HTML(K3)
- Create simple web pages using Bootstrap Containers(K5)
- Understand and apply advanced features such as Javascript and JSON (K4)

1. Designing web page for list handling
2. Designing web page for Table
3. Designing web page using links and images
4. Working with Forms
5. Designing web page using Html5 form inputs
6. Working with CSS border and border radius
7. Working with CSS animation
8. Bootstrap Containers and grid
9. Bootstrap tables
10. Bootstrap alert, Badge and modals
11. Bootstrap collapse and Tabs
12. Form validation using javascript
13. Calculator using javascript and AJAX
14. Display json data using javascript
15. Display XML data using javascript

## OPERATION RESEARCH - THEORY
## (21UCSE41)

**SEMESTER–IV ELECTIVE-1 HOURS–4 CREDITS-3 TOTAL HOURS : 60**

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

- Understand the models and phases of Operation Research(K2)
- Solve LPP by graphical method, Transshipment Problems and Assignment problems(K3)
- Find the optimal solutions for Games and decision trees(K4)
- Identify the Critical Path and to determine the Project completion time(K2)
- Develop the problem solving skills in Operations Research(K4)

**UNIT I** **(12 Hours)**

**LINEAR PROGRAMMING**: formulation of LP model – Graphics solution to two variable LP problems - Simplex method – Big M method – Two phase method – degeneracy in simplex method.

**UNIT II** **(12 Hours)**

**TRANSPORTATION MODEL:** definitions–Formulation and solution of transportation models–finding basic feasible solutions–finding optimal solution–degeneracy in transportation models. **ASSIGNMENT MODEL**– Comparison with Transportation model – Mathematical model – Formulation and Solution–Variations of the Assignment problems.

**UNIT III** **(12 Hours)**

**THEORY OF GAMES:** rules for game theory – Two-person zero-sum game – algebraic methods for finding optimum strategies – mixed strategies – graphical method for 2*2 or m*2 games - mixed strategies (3*3 games).

**UNIT IV** **(12 Hours)**

**REPLACEMENT MODELS:** Replacement of Items that deteriorate – replacement of Items that fail suddenly – Group replacement policy – mortality and staffing problems.

**UNIT V** **(12 Hours)**

**PERT and CPM:** Phases of project management – Work breakdown Structure – Network Logic – Numbering in the events – Activity on node diagram – Forward pass computations – Backward pass computations – Representation in tabular form – Slack – Critical path – Difference between CPM and PERT – CPM terms – Critical path (in CPM) – Float – Negative float and negative slack.

**Text Book:**

Prem Kumar Gupta, D.S Hira, “Operations Research”, S. Chand & Company Ltd., 7th Edition, 2014.

**Reference Books:**

1. J.K .Sharma, “Operations Research”, Macmillan India,4th edition(2009).
2. HamdyA. Taha , “Operations Research” , Pearson Education India; 9th edition (2014).

## GRAPH THEORY - THEORY
## (21UCSE41)

| SEMESTER–IV | ELECTIVE-1 | HOURS–4 | CREDITS-3 | TOTAL HOURS : 60 |
|---|---|---|---|---|

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

- Apply concepts and fundamentals theorems of Graphs to model problems of real world. (K3)
- Implementation of Graphs algorithms.(K2)
- Understand the basic concepts of graphs and trees. (K2)
- Find the research directions in the field of Graphs. (K4)
- Gain overall knowledge about the subject (K4)

**UNIT I (12 Hours)**

**Graph definitions:** Definitions of terms such as graph - sub-graph – vertex – edge - directed/undirected graph - weighted/un-weighted edges – degree - cut vertex/articulation point - clique - complete graph - Finite and Infinite Graphs - bipartite graphs - Isolated Vertex - Pendant Vertex - Null Graph. **Path and Circuits** - Walks, paths and Circuits - Connected Graphs - Disconnected Graph - Components - Euler Graph - Hamiltonian Paths – Circuits – The Travelling Salesman problem.

**UNIT II (12 Hours)**

**Trees:** Definitions – some properties of trees – pendant vertices in a tree – distance and centers - rooted trees - binary trees – counting trees - spanning trees – fundamental circuits – finding all spanning trees. Cut-Set and its Properties- Different Cut Sets in a graph – fundamental circuits and cut-sets – connectivity and separability – network flows – 1-Isomorphism – 2-Isomorphism.

**UNIT III (12 Hours)**

**Combinatorial vs Geometric graphs:** planar graphs – Kuratowski's two graphs – detection of planarity – geometric dual – combinatorial dual – thickness and crossings. Matrix representation of graphs – incidence matrix – submatrices – cirtuit matrix – fundamental circuit matrix and rank – cut-set matrix – Path matrix – Adjacency matrix.

**UNIT IV (12 Hours)**

**Chromatic Number:** Chromatic Partitioning – Chromatic Polynomial – Matchings – Coverings – Four Color problem. **Directed graphs** – types of digraphs – digraphs and binary relations – directed paths and connections – Euler digraphs – trees with directed edges – fundamental circuits in digraphs - matrices of digraphs – adjacency matrix of digraphs – paired comparisons and tournaments – acyclic digraphs.

**UNIT V** **(12 Hours)**

**Graph theoretic algorithms and computer programs:** computer representation of graphs – basic algorithms – connectedness and components algorithm – spanning tree algorithms – shortest-path algorithms – depth-first search on a graph – breadth first search – isomorphism algorithm.

**TEXT BOOK:**

Narsingh Deo, "Graph Theory with Applications to Engineering and Computer Science", Prentice -Hall India (PHI), 2003.

**REFERENCE BOOKS:**

1. Clark J. and Holton D.A, "A First Look at Graph Theory", Allied Publishers, 1995.
2. R.J. Wilson, "Introduction to Graph Theory", Fourth Edition, Pearson Education, 2003.
3. R. Diestel, "Graph Theory", Springer-Verlag, 2nd edition, 2000.

## THEORY OF COMPUTATIONS- THEORY
## (21UCSE41)

| SEMESTER–IV ELECTIVE-1 HOURS–4 CREDITS–3 TOTAL HOURS:60 |
|---|

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

- Understand the concept of Automata theory
- Understand about regular expression and languages
- Learn about context free grammars
- Understand the properties of Context free grammer
- Gain overall knowledge about the subject

**UNIT I** **(12 Hours)**

**AUTOMATA:** Introduction to format proof – Additional forms of Proof – Inductive proofs – Finite Automata(FA) – Deterministic Finite Automata (DFA) – Non deterministic Finite Automata (NFA) – Finite Automata with Epsilon transitions.

**UNIT II** **(12 Hours)**

**REGULAR EXPRESSIONS AND LANGUAGES:** Regular Expression-FA and Regular Expression – Proving languages not to be regular Closure properties of regular languages- Equivalence and minimization of Automata.

**UNIT III** **(12 Hours)**

**CONTEXT FREE GRAMMARS AND LANGUAGES**: Context - Free Grammar (CFG) – Parse Trees Ambiguity In Grammars And Languages – Definition of the Pushdown automata – Languages of a Pushdown Automata - Equilavance of Pushdown automata and CFG – Deterministic Pushdown Automata.

**UNIT IV** **(12 Hours)**

**PROPERTIES OF CONTEXT:** FREE LANGUAGES – Normal forms for CFG – Punming Lemma for CFL – Closure Properties ofCFL – TuringMachines - Programming Techniques for TM.

**UNIT V** **(12 Hours)**

**UNDECIDABALITY:** A language that is not Recursively Enumerable (RE) – An undecidable problem that is RE – Undecidable problems about Turing Machine – Post's Correspondence Problem – The classes P and NP.

**TEXT BOOK:**

J.E. Hopcroft, R Motwani and J.D Ullman, “Introduction to Automata Theory, Languages and Computations”, Second Edition, Pearson Education, 2007.

**REFERENCES:**

1. HR. Lewis and C.H. Papadimitriou, “Elements of theory of Computation”, Second Edition, Pearson Education, 2003.

2. Thomas A. Sudkamp, “An Introduction to the Theory of Computer Science Languages and Machines ”, Theird Edition, Pearson Education, 2007.

## NUMERICAL AND STATISTICAL METHODS – THEORY (21UCSE41)

**SEMESTER–IV ELECTIVE-1 HOURS–4 CREDITS–3 TOTAL HOURS: 60**

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

- Understand the concept of errors and approximation
- Understand about iteration methods of solving numerical problems
- Learn about simultaneous Algebraic equations and their solutions
- Understand about probability and various hypothesis testing methods
- Gain overall knowledge about the subject

**UNIT I (12 Hours)**

**Approximations and Errors in computation:** Introduction - numbers - Errors - Error in the approximation of a function - Errors in a series approximation - order of approximation - propagation error. **Solution of Algebraic and Transcendental Equations:** Introduction - Basic properties of equations - bisection method - Regula - falsi method - Secant method - Iteration method - Newton - Raphson method.

**UNIT II (12 Hours)**

**Solution of simultaneous Algebraic Equations:** solution of linear simultaneous equations - Direct methods of solution - Gauss elimination method, Gauss - Jordan method, Crout's triangularization method - Iterative methods of solution - Jacobi, Gauss - Seidal. Numerical integration - Quadrature formulae - Trapezoidal rule, Simpson's one - third rule - Simpson’s three - eighth rule - Boole's rule - Weddle’s rule.

**UNIT III (12 Hours)**

**Curve fitting:** Introduction - Principle of least squares - Fitting a straight line - Fitting a second degree parabola.Correlation and Regression: Introduction - Correlation - Rank correlation - Regression - Correlation coefficient for a bipartite frequency distribution.
Interpolation: Introduction - Finite Differences - Newton's formula - Lagrange's formula.

**UNIT IV (12 Hours)**

**Probability:** Introduction - probability - Conditional probability some special Distributions - Introduction - Binomial distribution - Poisson distribution - Normal distribution - Some more continuous distributions.

**UNIT V** **(12 Hours)**

**Test Based on chi-square Distribution**: Introduction – chi-square Test – chi square test for test the goodness of fit - Test for independence of attributes. Test of significance (small samples): Introduction - Test of significance based on t - distribution (t-test) - Test of significance based on f-test - Test for significance of an observed sample correlation.

**Text Books:**

1. B.S. Grewal, "Numerical methods in Engineering & Science", Khanna Publishers, Fifth Edition, April 1999.
2. S.Arumugam, A.Thangapandi Isaac, "Statistics", New Gamma publishing House, 1999.

**Reference Books:**

1. Ajay Wadhwa, ' Numerical Analysis with Algorithms and Computer Programming in C++' PHI learning Pvt Ltd, 2012.
2. S.P.Gupta, "Statistical methods", Sultan chand & sons, 7 th Edition, 2012.

## MICROPROCESSOR AND ASSEMBLY LANGUAGE PROGRAMMING -THEORY (21UCSA41)

**SEMESTER–IV ALLIED–T4 HOURS–4 CREDITS–4 TOTAL HOURS: 60**

**Course Outcomes:**

**Upon completion of the course, the students will be able to**

- Define the general architecture and organization of 8085 & 8086 Microprocessor (K1)
- Classify the instruction set of 8085 microprocessors (K4)
- Analyse the use of different instructions and apply it in assembly language programming. (K4)
- Illustrate the design aspects of Microcontrollers (K4)
- Summarize the Interfacing of memory and various I/O devices with 8085 Microprocessor (K5)
- Develop programming skill in assembly language. (K6)

**UNIT I (12 Hours)**

Microprocessor Architecture – Introduction – Intel 8085 – Instruction Cycle – Timing Diagram – Instruction Set of Intel 8085 – Introduction – Instruction and Data Formats – Addressing Modes – Status Flags – Symbols and Abbreviations – Intel 8085 Instructions.

**UNIT II (12 Hours)**

Programming of Microprocessors – Assembly Language – High level Language – Area of Applications of Various Languages – Stack – Subroutines – System Software – Some Important Commands – Modular Programming – Structured Programming - Top-Down Design, Bottom-Up Design – MACRO.

**UNIT III (12 Hours)**

Intel 8086 – Classification of 8086 Instructions – Binary Address of 8086 Registers – Description of 8086 Registers – Assembler Directives – Assembler Directives for Intel 8086.

**UNIT IV (12 Hours)**

Peripheral Devices and Their Interfacing – Address Space Partitioning – Memory and I/O Interfacing – Data Transfer Schemes – Interrupts of 8085 – Interfacing Devices and I/O Devices – I/O Ports – Programmable DMA Controller – Programmable Interrupt Controller (PIC) – Programmable Communication Interface.

**UNIT V (12 Hours)**

Intel’s 32-Bit and 64-Bit Microprocessors – Intel 80386 – Intel 80486 - Intel’s Pentium Processor – Pentium MMX – Pentium II – Pentium III – Pentium 4 – IA-64 – Itanium - Multicore Processors – Other Microprocessors – AMD – Sun Microprocessors – MIPS Microprocessors – PowerPC Microprocessors – DEC’s Alpha Microprocessors – National Semiconductor Microprocessors – ARM Microprocessor – MOTOROLA Microprocessors.

**Text Book:**

B.Ram, “Fundamentals of Microprocessor and Microcomputers”, Dhanpat Raj Publications, Sixth Edition, 2005.

**Reference Books:**

1. Ramesh S.Goankar, “Microprocessor Architecture, Programming and Applications with 8085”, Fifth Edition.
2. Yu-Cheng Liu, Glenn A.Gibson, “Microcomputer Systems, the 8086/8088 Family Architecture, Programming and Design”, PHI, 2000.

# PRACTICAL - ASSEMBLY LANGUAGE PROGRAMMING
# (21UCSA42)

**SEMESTER–IV ALLIED–P2 HOURS–2 CREDITS–1 TOTAL HOURS:30**

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

- Define the fundamentals of assembly level programming of microprocessors and microcontroller (K1)
- Describe abstract problems and apply a combination of hardware and software to address the problem(K2)
- Select proper mnemonics and run their program on the training boards.(K4)
- Evaluate the expressions in different cases(K5)
- Develop testing and experimental procedures on Microprocessor and Microcontroller.(K6)
- Create ALP Programs and analyse the information.(K6)

**Practical List**

1. Addition of two 8-bit numbers
2. Subtraction of two 8-bit numbers
3. To add N 8-bit numbers
4. Multiply with repeated addition
5. Division with repeated subtraction
6. Combining and separating nibbles
7. Odd and even numbers
8. Fibonacci and factorial
9. Evaluating expression
10. Finding square and cube of the giving number
11. To find the max and min of n numbers
12. Block transfer
13. Complements of numbers
14. Ascending and descending order
15. Binary to BCD, BCD to Binary
16. BCD to Seven segment and BCD to ASCII
17. ASCII to BCD, ASCII to Binary

## INTERNET CONCEPTS AND WEB DESIGN
## (21USB41)

| SEMESTER–IV | SBE6 | HOURS–2 | CREDITS–2 | TOTAL HOURS : 30 |
|---|---|---|---|---|

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

- Explain the basic functions of HTML tags. (K1)
- Create simple web pages using HTML. (K3)
- Describe the concepts of tables, frames and forms (K2)
- Create web pages using tables, frames and forms. (K4)
- Update knowledge to learn any future advanced version of language (K6)

**UNIT I** **( 6 Hours)**

Introduction to the Internet – Networking – Internet – Email – Resource sharing – Gopher – World Wide Web

**UNIT II** **( 6 Hours)**

Internet Technologies – Browsers - Introduction to HTML – History of HTML – HTML documents

**UNIT III** **( 6 Hours)**

Head and Body Sections – Designing the body section - Ordered and Unordered Lists.

**UNIT IV** **( 6 Hours)**

Table Handling – DHTML and Style sheets

**UNIT V** **( 6 Hours)**

Frames – A Web page design project – Forms.

**Text Book:**

1. C. Xavier, “World Wide Web Design with HTML”, Tata McGraw Hill publication, First Edition , 2000.

**PRACTICAL LIST**

1. Designing a simple web pages
2. Designing a web page using formatting and image tags
3. Designing web page containing tables
4. Designing Web page with Frames
5. Designing Web page with Forms

## PROGRAMMING IN DOT NET - THEORY
## (21UCS51)

**SEMESTER-V CORE–T8 HOURS–4 CREDITS-4 TOTAL HOURS:60**

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

- Define the concepts of Dot Net programming(K1)
- Describe programming skill to debug and run the programs efficiently.(K2)
- Demonstrate solutions for a given problem using VB dot Net and Asp dot Net(K3)
- Illustrate various Data base concepts using ADO dot Net(K4)
- Develop real time Applications in VB.NET and ASP.NET(K6)
- Develop creative windows and web applications(K6)

**UNIT I (12 Hours)**

**Introudction to Visucal Basic.NET:** Exploring Visual basic .NET Discovering visual basic .NET Language changes -Window forms working with window forms Creating inheriting window forms working with controls Adding controls to window forms-arranging controls in window forms-working with dialog boxes MDI Working with MDI forms-Implementing Drag and drop operation Utilizing the clipboard -variables Controlling Program flow.

**UNIT II (12 Hours)**

**Procedures in VB.Net:** Procedure-Working with procedures Implementing Vb.Net classes Discovering Object oriented Programming Visual Basic and object oriented .NET programming Crating classes Implementing Inheritance -Handling Errors in VB.Net identifying error type Handling errors in program Tracing errors in program -Accessing a Database ADO.NET -ADO.NET components Displaying data on a form.

**UNIT III (12 Hours)**

**Introducing ASP.Net:** Gainting started with ASP.Net applications: Web forms -creating ASP.Net Webform applications Using ASP.Net Webforms for server controls: Beginning with server controls Talking a closer look at web controls Illustrating Basic web controls – Working with Validation Controls: The compare Validator – The Range Validator Regular Expression Validator – Custom validator – Validation Summery control – Multiple validation control.

**UNIT IV**

**(12 Hours)**

**Developing ASP.Net Server controls:** Developing ASP.Net server controls Creating and using Web User Control Creating ASP.Net Pages to web user control – Composite controls-using Rich controls-Web controls: Adrotator web server control Calendar web server control.

**UNIT V** **(12 Hours)**

**Debugging ASP.Net Web Applications:** Tracing ASP.Net Applications Handling Errors in ASP.Net applications – Debugging ASP.Net Application with the Visual Studio .Net debugger – Using ADO.Net with ASP.Net:ADO.Net ADO.Net Object model Creating a Data aware application. Deploying web applications: Creating a Deployment project Testing the Installation Program.

**Text Book:**

MridulaParihar, YeshSingal and NitinPandey, “Visual Studio .Net Programming”, PHI, 2002.

**Books for Reference:**

1. G. Andrew Duthie, Microsoft ASP.NET Step by step, Microsoft Press, 2003.
2. Kogent Learning Solutions Inc., ASP.NET 2.0 Black book, DreamTechPress, 2006.

# OPERATING SYSTEM AND UNIX - THEORY
## (21UCS52)

| SEMESTER–V CORE – T9 | HOURS – 4 CREDITS – 4 | TOTAL HOURS-60 |
|---|---|---|

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

- Describe the basic concepts of operating system. (K2)
- Illustrate how UNIX implements File System (K3)
- Analyze the CPU Scheduling Algorithms (K4)
- Compare the Memory Management Strategies (K5)
- Explore several tools to solve process Synchronization problems (K6)

**UNIT I** **(12 Hours)**

**Introduction:** Operating system - Computer System Organization-Operating System Structure – Operations – Process Management – Memory Management – Storage Management – Protection and Security **System structures**: Operating system services – user operating system interface – system calls – types of system calls – System programs - operating system structure.

**UNIT II** **(12 Hours)**

**Process Concept:** Process Scheduling – Operations on Processes – Inter Process Communication **Process Scheduling:** Basic concepts - Scheduling Criteria – Scheduling Algorithms **Synchronization:** Background – Critical Section Problem – Mutex locks - Semaphores – Classic problems of synchronization.

**UNIT III** **(12 Hours)**

**Deadlocks:** System Model – Deadlock Characterization - Methods of Handling Deadlocks – Deadlock prevention – Deadlock Avoidance **Memory Management Strategies**: Background – Swapping - Contiguous Memory allocation – Segmentation – Paging Virtual Memory Management: Background – Demand Paging – Page replacement.

**UNIT IV** **(12 Hours)**

**File System:** File Concept – Access Methods – Directory and Disk Structure – Protection.
**Implementing File System:** File System Structure – File system Implementation –Directory Implementation – Allocation Methods **Mass storage structure:** Overview of mass storage structure – Disk structure – disk scheduling.

**UNIT V** **(12 Hours)**

**UNIX:** The file system: the file-The HOME variable-pwd-cd-mkdir-rmdir-absolute and relative path names-ls-The Unix file system. Handling ordinary files : cat-cp-rm-mv-more-lp-file-wc-od-cmp-comm-diff-gzip-gunzip-tar-zip-unzip. Basic File attributes : ls options-File ownership-file permission-chmod-directory permission-changing file ownership. **Essential Shell programming:** Shell scripts-read-using command line arguments-exit-The logical operators-if conditional-test-case-expr-$0 calling a script by different names – while-for-set and shift- the here document- trap – set-x – sample validations and data entry scripts.

**Text Books:**

1. Abraham SilberSchatz, Peter Baer Galvin, Greg Gagne ,"Operating System Concepts", John Willy & Sons (Asia), NINTH Edition, 2014.
2. Sumitabha Das. "Unix Concepts and Applications", Tata McGraw Hill Publications, Third Edition., 2017.

**Reference Books:**

1. Gary J.Nutt,"Operating Systems", Pearson Education Asia, 2$^{nd}$ Edition., 2012.
2. H.M.Deital,"Operating Systems", Addison-Wesley Publishing Company, Second Edition.

## MOBILE APPLICATION DEVELOPMENT - THEORY (21UCS53)

**SEMESTER–V CORE–T10 HOURS–4 CREDITS–4 TOTAL HOURS:60**

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

- Identify the advantages of React Native (K1)
- Know how to create simple App (K2)
- Express the Project structure (K2)
- Implement the various components and Elements (K3)
- Examine the native modules (K3)
- Execute Platform specific components (K3)

**Unit I** **( 12 Hours)**

Introduction to React Native - Advantages of React Native - Risks and Drawbacks - Working with React Native -Rendering Life Cycle - Creating Components in React Native - Working with Views - Using JSX - Styling Native Components - Host Platform APIs.

**Unit II** **( 12 Hours)**

Building First Application- Setting up the Environment - Developer Setup: Create React Native App - Developer Setup: The Traditional Approach - Handling User Input - Displaying Data - Fetching Data from the Web - Adding a Background Image.

**Unit III** **( 12 Hours)**

Components for Mobile - Analogies Between HTML Elements and Native Components - Working with Touch and Gestures - Working with Lists - Styles: Declaring and Manipulating Styles - Organization and Inheritance -Positioning and Designing Layouts.

**Unit IV** **( 12 Hours)**

Platform APIs - Using Geolocation - Accessing the User’s Images and Camera - Storing Persistent Data with AsyncStorage - The SmarterWeather Application -Modules and Native Code - Installing JavaScript Libraries with npm - Installing Third-Party Components with Native Code - Objective-C Native Modules - Java Native Modules - Cross-Platform Native Modules.

**Unit V:** **( 12 Hours)**

Platform-Specific Code - iOS-or Android-Only Components - Components with Platform-Specific Implementations - When to Use Platform-Specific Components - Debugging and Developer Tools - JavaScript Debugging Practices, Translated - React Native Debugging Tools - Debugging Beyond JavaScript - Testing Your Code - Navigation and Structure in Larger Applications - The Flashcard Application - Project Structure - Using React-Navigation - State Management in Larger Applications - Using Redux to Manage State - Actions - Reducers - Connecting Redux - Persisting Data with AsyncStorage.

**TEXT BOOK:**

Bonnie Eisenman, "Learning React Native - Building Native Mobile Apps with JavaScript", SECOND EDITION, O'Reilly Media, Inc., 2018.

**REFERENCE BOOKS:**

1. Jonathan Lebensold,"React Native Cookbook", O'Reilly Media, Inc., 2018.
2. Dotan Nahum, "Programming React Native", 2016.

## PHP and MySQL - THEORY
## (21UCS54)

**SEMESTER–V CORE-T11 HOURS–4 CREDITS–4 TOTAL HOURS:60**

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

- Impart knowledge on dynamic web design issues(K3)
- Impart theoretical knowledge about PHP (K1)
- Develop programming skills in PHP (K3)
- Introduce database connections to MySQL through PHP(K4)
- Update knowledge to learn any future advanced version of language (K5)

**UNIT I** **(12 Hours)**
**Basics of PHP:** Basic Syntax- PHP Data type - Defining variables and constant- whitespace- Code Blocks-Opening and Closing Code Islands- Mixed Mode Processing -Comments- Automatic Type Conversion- Including Other Files- Operator and Expression- Conditional Statements -Control Structures.

**UNIT II** **(12 Hours)**
**Function:** Call by value - Call by reference, Recursive function-Default arguments- String Creating and accessing, String Searching - Replacing String, Formatting String, String Related Library function – Array- Creating index based and Associative array- Accessing array, Element Looping with Index based array, Looping with associative array using each () and foreach()- Array related Library functions

**UNIT III** **(12 Hours)**
**HTML Forms:** Form design-Handling data-Validating Input-GAIN ,POST methods - isset() – Superglobals - MySQL: What is MySQL - MySQL Connect - MySQL create DB - MySQL create Table - Insert Data Select Data - Delete Data-Update Data - Limit Data.

**UNIT IV** **(12 Hours)**
**Session and Cookie:** Introduction to Session Control, Session Functionality What is a Cookie, Setting Cookies with PHP. Using Cookies with Sessions, Deleting Cookies, Registering Session variables, Destroying the variables and Session. Working with file and Directories: Understanding file& directory, Opening and closing a file, Coping, renaming and deleting a file, working with directories, Creating and deleting folder, File Uploading & Downloading.

**UNIT V** **(12 Hours)**
**Object Oriented PHP:** Objects -Declaring a class-The new keyword and constructor-Destructor-Access method and properties using this variable -Public ,private, protected -Static properties and method-Class constant -Inheritance & code reusability –Polymorphism -Parent:: & self:: keyword -Instanceof operator - Abstract method and class –Interface –Final Exception Handling-Understanding Exception and error Try, catch, throw.

**Text Books:**

1. Paul Hudson, "PHP in a Nutshell”, O’ Reilly Publications, 2005.
2. Andy Harris, "PHP5/MYSQL Programming for the absolute beginner", Thomson Course Technology, 2015.

**Reference Books:**

1. Robin Nixon, “Learning PHP, MySQL and JavaScript”, O’reilly Publishers, 2009.
2. K.Meena, R.Sivakumar and A.B.Karthick Anand Babu, “Web Programming Using PHP and MySQL”, Himalaya Publishing House, 2012.

## PRACTICAL - PROGRAMMING IN DOT NET (21UCS55)

**SEMESTER–V CORE–P7 HOURS–4 CREDITS–2 TOTAL HOURS:60**

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

- Describe the concepts of Dot Net programming(K2)
- Apply programming skill to debug and run the programs efficiently.(K3)
- Apply solutions for a given problem using VB dot Net and Asp dot Net(K3)
- Use various Data base concepts using ADO dot Net(K3)
- Develop real time Applications in VB.NET ,ASP.NET (K6)
- Develop creative windows and web applications(K6)

### List of Practical

### VB.NET

1. Designing a simple application using VB.NET.
2. Designing application using various server controls.
3. Designing application for drag and drop operation.
4. Designing MDI applications.
5. Designing application using clipboard.
6. Designing application using databases.

### ASP.NET

1. Designing simple Web Forms Applications
2. Designing Applications with Web Controls
3. Designing Applications with Validator controls
4. Designing applications using Web user and Composite controls
5. Designing applications for working with Rich Web controls
6. Designing applications to work with databases

## PRACTICAL - UNIX AND SHELL PROGRAMMING (21UCS56)

**SEMESTER–V CORE–P8 HOURS–2 CREDIT–1 TOTAL HOURS:30**

**Course Outcomes:**

**Upon completion of the course, the students will be able to**

- Summarize the UNIX Commands (K2)
- Write simple shell programs (K3)
- Implement UNIX commands using C language (K4)
- Write shell programs using advanced commands of UNIX (K5)
- Update knowledge to learn any future advanced version of language (K6)

1. Implement WC Command in C
2. Implement Grep command in C
3. Implement More command in C
4. Implement LS command in C
5. Write the program to find biggest among 3 numbers using shell
6. Write a shell program to find factorial
7. Write a shell program to check the given string is Palindrome (or) not
8. Write a shell program to check whether the user is logged (or) not
9. Write a shell program using array
10. Write a shell program to find the biggest among numbers using positional parameters

# PRACTICAL – MOBILE APPLICATION DEVELOPMENT (21UCS57)

**SEMESTER–V CORE–P9 HOURS–2 CREDIT–1 TOTAL HOURS:30**

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

- Apply the basic elements (K3)
- Implementing the components (K3)
- Using the Persistent storage (K3)
- Playing with Aimations (K3)
- Displaying Remote Images (K3)
- Building App with API (K6)

**Practical List:**

1. Layout with Flexbox

2. Breaking down a UI into Components

3. Dealing with the Keyboard

4. Listing Data with the FlatList

5. Persistent Storage

6. Dealing with Remote Images on Slow Networks

7. Playing with Animations

8. Complex Navigation Structure

9. Build a Swiper Component

10. Making a Declarative API for an Imperative API

## PRACTICAL - PHP and MYSQL
## (21UCS58)

**SEMESTER–V CORE–P10 HOURS–2 CREDITS–1 TOTAL HOURS:30**

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

- Write simple PHP programs.(K2)
- Write programs using control structures.(K3)
- Develop application using COOKIES.(K2)
- Create database application.(K5)
- Update knowledge to learn any future advanced version of language (K2)

1. Simple PHP Program.

2. Sort an Array.

3. Program using if condition.

4. Palindrome.

5. Retrieve Form values and display.

6. Upload Files.

7. Session, Cookies and Query string

8. Create database and table using PHP.

9. Insert values in database.

10. Display table data in Grid Format.

11. Update, Delete table data.

12. Program using Files (Create, Read, Write and Modify).

13. Program using class.

## ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING - THEORY (21UCSE51)

**SEMESTER–V ELECTIVE-2 HOURS–4 CREDITS-4 TOTAL HOURS: 60**

**Course Outcomes:**

**Upon completion of the course, the students will be able to**

- Recall the basic concepts of Artificial Intelligence and machine learning (K1)
- Explain the different knowledge representations for AI problems (K2)
- Apply suitable heuristic search technique for the given problem (K3)
- Classify the types of reasoning (K4)
- Summarize the types of learning (K5)
- Design an associate rule learning model for a real life problem (K6)

**UNIT I (12 Hours)**

**Introduction**: Definitions of Artificial Intelligence (AI) – AI problems - Topics of AI – Production Systems – State Space Representation - Branches of AI – Applications of AI. **Heuristic search techniques**: Generate and test - Hill Climbing - Search Techniques - Depth First Search, Breadth First Search, Greedy Method, Best First Search Algorithm,0 A* Algorithm - Problem Reduction – AND-OR Graphs, The AO* Algorithm, Towers of Hanoi problem - Constraints Satisfaction - Means-ends Analysis.

**UNIT II (12 Hours)**

**Game Playing**: MINIMAX Procedure - Alpha-Beta Pruning - Combined Approach - **Knowledge representation**: Knowledge Management - Value of Knowledge Management - Categories of Knowledge - Types of Knowledge - Knowledge Representation - Approaches to Knowledge Representation - Issues in Knowledge Representation - Knowledge Base.

**UNIT III (12 Hours)**

**Knowledge representation structures**: First-order Logic - Frames - Conceptual Dependency - Scripts - Semantic Network. **Reasoning**: Types of Reasoning - Non-monotonic Inference Methods - Non-monotonic Reasoning - Truth Maintenance Systems - Reasoning with Fuzzy Logic - Rule-based Reasoning - Diagnosis Reasoning.

**UNIT IV (12 Hours)**

**Learning**: Types of Learning - Machine Learning - Intelligent Agents. **Association learning**: Basics of Association - Apriori Algorithm - Eclat Algorithm - FP Growth Algorithm - Tertius Algorithm - Case Studies -Customer Sequence: Apriori Algorithm - Bank Loan Status: Association Rule Formation - Comparison of Associate Rule Algorithms - SCADA Application by FP Growth Algorithm.

**UNIT V (12 Hours)**

**Reinforcement learning**: Markov Decision Problem - Q-learning - Temporal Difference Learning - Learning Automata - Case Studies - Super Mario: Reinforced Learning. **Artificial neural nets**: ANN Basics - ANN—Learning Process - Types of Networks -

Perceptron - Multilayer Perceptron - Error Back-propagation Algorithm - RBF Networks - ANN Summary.

**Text Book:**

Vinod Chandra S.S. and Anand Hareendran S., “Artificial Intelligence and Machine Learning”, PHI Learning Private Limited, 2014.

**Reference Books:**

1. Christopher Bishop, Pattern Recognition and machine learning; Springer Verlag, 2006.
2. Stuart Russell, Peter Norvig, “Artificial Intelligence: A Modern Approach”, 3rd Edition, Prentice Hall, 2010.

## INTERNET OF THINGS- THEORY
## (21UCSE51)

| SEMESTER–V ELECTIVE-2 HOURS–4 CREDITS-4 TOTAL HOURS: 60 |
|---|

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

- Interpret the impact and challenges posed by IoT networks leading to new architectural models.(K1)
- Compare and contrast the deployment of smart objects and the technologies to connect them to network.(K3)
- Appraise the role of IoT protocols for efficient network communication.(K1)
- Elaborate the need for Data Analytics and Security in IoT.(K4)
- Illustrate different sensor technologies for sensing real world entities and identify the applications of IoT in Industry.(K5)

**Unit I** **( 12 Hours)**

Introduction to Internet of Things – Physical design of IoT – Logical Design of IOT – IoT enabling technologies – IoT levels and deployment templates - Domain Specific IoTs - Home Automation – Cities – Environment – Energy – Retail – Logistics – Agriculture – Industry – Health and Lifestyle.

**Unit II** **( 12 Hours)**

IoT and M2M – Introduction – M2M – Difference between IoT and M2M – SDN and NFV for IoT – IoT System management with NETCONF-YANG – Need for IoT systems Management – Simple Network Management Protocol – Network Operator Requirements – NETCONF – YANG.

**Unit III** **( 12 Hours)**

IoT Platforms design methodology – IoT Design Methodology – IoT systems – Logical Design using Python – Python Data Types and Data Structures – Control Flow – Functions – Modules – Packages – File Handling – Classes – Python Packages for IoT.

**Unit IV** **( 12 Hours)**

IoT Physical Devices and Endpoints – Raspberry Pi – Raspberry Pi Interfaces – Programming Raspberry Pi with Python – Other IoT Devices - IoT Physical Servers and Cloud Offerings – WAMP AutoBrain for IoT – Xively Cloud for IoT – Python Web Application Framework – Django.

**Unit V** **( 12 Hours)**

Case Studies Illustrating IoT Design – Home Automation – Cities – Environment – Agriculture – Productivity Applications.

**Text Book**

Vijay Madisetti and Arshdeep Bahga, “Internet of Things (A Hands-on-Approach)”, First Edition, VPT, 2014.

**Reference Books**

Raj Kamal, “Internet of Things: Architecture and Design Principles”, 1st Edition, McGraw Hill Education, 2017.

# BIG DATA ANALYTICS - THEORY
# (21UCSE51)

**SEMESTER–V ELECTIVE-2 HOURS–4 CREDITS-4 TOTAL HOURS: 60**

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

- Describe Big Data analytics (K1)
- Explain the basic concept of Map Reduce (K2)
- Use Analytics process in real time applications (K3)
- Apply big data Storage techniques (K3)
- Analyze various analytics algorithms. (K4)
- Illustrate visualization techniques. (K4)

**UNIT I** **(12 Hours)**

**Introduction to Big Data:** Analytics-Characteristics of Big Data-Domain Specific examples of Big Data-Analytics flow for Big Data-Big Data Stack-Mapping Analytics flow to Big Data stack-Setting up Big Data Stack-Hortonworks Data Platform(HDP)-Cloudera CDH Stack-Amazon Elastic MapReduce-Azure HDInsight.

**UNIT II** **(12 Hours)**

**Big Data Patterns:** Analytics Architecture Components and Design Styles-Map Reduce Patterns-NoSQL:Key-Value Databases-Document Databases-Column Family Databases.Data Acquisition-Data Acquisition Considerations-Publish –Subscribe Mesaging Frameworks-Big Data Collection Systems-Messaging Queues-Custom Connectors.

**UNIT III** **(12 Hours)**

**Big Data Storage:** HDFS –Batch Analysis-Hadoop and MapReduce-Hadoop-MapReduce Examples-Pig-Case Study:Batch Analysis of News Articles-Apache Oozie-Apache Spark-Search.

**UNIT IV** **(12 Hours)**

**Real-time Analysis:** Stream Processing-Storm Case Studies-In Memory Processing-Spark Case Studies-Interactive Querying: Spark SQL-Hive –Amazon Redshift-Google BigQuery-Serving Databases and web Frameworks: Relational(SQL) Databases-Non Relational(SQL) Databases-Python Web application Framework-Django.

**UNIT V** **(12 Hours)**

**Analytics Algorithms:** Frameworks-Clustering-Classification and regression-Case Study-Classifying Handwritten Digits-Data Visualization: Frameworks and Libraries-Vizualization Examples: Line Chart-Scatter Plot-Bar chart-Box Plot-Pie Chart-Dot Chart-Gauge Chart-Radar Chart-Regression Plot-Clustered Heatmap-Pair Grid.

## TEXT BOOK

Arshdeep Bahga,Vijay Madisetti, ”Big Data Science and Analysis-A Hands-on Approach, Arshdeep Bahga & Vijay Madisetti Publishers, 2019.

## REFERENCE BOOKS

1. Boris lublinsky, Kevin t. Smith, Alexey Yakubovich, “Professional Hadoop Solutions”, Wiley, ISBN: 2015.
2. Chris Eaton, Dirk derooset al. , “Understanding Big data ”, McGraw Hill, 2012.

## DEEP LEARNING- THEORY
## (21UCSE51)

| **SEMESTER–V ELECTIVE-2 HOURS–4 CREDITS-4 TOTAL HOURS: 60** |
|---|

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

- Define complete overview of Deep Learning (K1)
- Illustrate basic operations of Deep Learning(K2)
- Apply Deep Learning algorithms in appropriate real time applications.(K3)
- Demonstrate the use Deep Learning(K3)
- Demonstrate the Deep Learning Computation Techniques (K3)
- Illustrate the methodologies for training and optimizing Deep Models(K4)

**UNIT I** **(12 Hours)**
Introduction:- Key Components- The Road to Deep Learning- Characteristics- Preliminaries: Data Manipulation- Operations- Indexing and Slicing- Data Preprocessing- Linear Algebra-Scalars- Vectors -Calculus- Derivatives and Differentiation- Automatic Differentiation-Probability- Basic Probability Theory- Dealing with Multiple Random Variables.

**UNIT II** **(12 Hours)**
Linear Neural Networks: Linear Regression-1 Basic Elements of Linear Regression- From Linear Regression to Deep Networks- Linear Regression Implementation from Scratch-Multilayer Perceptrons- Hidden Layers- Activation Functions- Implementation of Multilayer Perceptrons from Scratch- Model Selection, Underfitting, and Overfitting- Forward Propagation, Backward Propagation, and Computational Graphs- Forward Propagation-Computational Graph of Forward Propagation- Back propagation- Training Neural Networks.

**UNIT III** **(12 Hours)**
Deep Learning Computation: Layers and Blocks -A Custom Block- The Sequential Block-Executing Code in the Forward Propagation Function- Efficiency- Parameter Management-Parameter Access- Parameter Initialization- Tied Parameters- Custom Layers- Layers without Parameters- Layers with Parameters- Modern Convolutional Neural Network: Deep Convolutional Neural Networks (AlexNet) - Learning Representations AlexNet - Reading the Dataset –Training.

**UNIT IV** **(12 Hours)**
Deep Feed forward Networks-Example: Learning XOR-Gradient-Based Learning-Hidden Units- Architecture Design-Back-Propagation and Other Differentiation Algorithms-Regularization for Deep Learning-Parameter Norm Penalties-Norm Penalties as Constrained Optimization-Regularization and Under-Constrained Problems-Dataset Augmentation-Semi-Supervised Learning-Multi-Task Learning-Adversarial Training-Tangent Distance, Tangent Prop, and Manifold Tangent Classifier.

**UNIT V** **(12 Hours)**

Optimization for Training Deep Models-Challenges in Neural Network Optimization-Basic Algorithms-Parameter Initialization Strategies-Algorithms with Adaptive Learning Rates-Approximate Second-Order Methods-Optimization Strategies and Meta-Algorithms Applications: Large Scale Deep Learning-Computer Vision-Speech Recognition-Natural Language Processing-Other Applications.

**TEXT BOOKS:**

1. Aston Zhang, Zachary C. Lipton, Mu Li, and Alexander J. Smola,” Dive into Deep Learning”, 2021
2. Goodfellow, I., Bengio,Y., and Courville, A., Deep Learning, MIT Press, 2016.

**REFERENCE BOOKS**

1. Bengio, Yoshua, Ian J. Goodfellow, and Aaron Courville. "Deep learning." An MIT Press book in preparation, 2015.
2. Abadi, Martın, et al. "Tensorflow: Large-scale machine learning on heterogeneous distributed systems." arXiv preprint arXiv:1603.04467, 2016.

## DATA COMMUNICATIONS AND COMPUTER NETWORKS- THEORY (21UCS61)

**SEMESTER–VI CORE–T12 HOURS–5 CREDITS-5 TOTAL HOURS:75**

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

- Describe the functions of the layers of OSI model (K2)
- Trace the flow of information from one node to another node in the network (K3)
- Illustrate the protocols used in each layer (K4)
- Evaluate the protocols in network layer with noisy and noiseless channels (K5)
- Identify the functions of key management in Cryptography (K6)

**UNIT I (15 Hours)**

**Introduction:** Data Communications – Networks – Networks types – Internet history – Standards and Administration – **Network Models:** Protocol layering – TCP/IP Protocol Suite - OSI Model – **Transmission media:** Guided Media - Unguided media.

**UNIT II (15 Hours)**

**Digital Transmission:** Digital-To-Digital conversion – Analog-To-Digital conversion – Transmission modes. **Analog Transmission:** Digital-To-Analog conversion – Analog-To-Analog conversion. **Bandwidth Utilization:** Multiplexing – Spread spectrum.

**UNIT III (15 Hours)**

**Switching:** Circuit Switched Networks – Packet switching - Structure of Switch – **Error Detection and Correction:** Introduction - Block Coding – Cyclic Codes - Checksum. **Data Link control:** DLC services – Data link layer Protocols – POINT-TO-POINT Protocol – **Media Access Control:** Random Access – Controlled Access – Channelization.

**UNIT IV (15 Hours)**

**WirelessLAN**: Introduction - Bluetooth - **Connecting devices and Virtual LANS** – **Introduction to Network Layer:** Network layer services – Packet Switching – IPV4 Addresses – Forwarding IP packets – **Unicast Routing :** Introduction – Routing Algorithms – **Multicast Routing:** Introduction – Multicasting basics – **Introduction to Transport Layer:** Introduction – Transport layer protocols.

**UNIT V** **(15 Hours)**

**Introduction to Application layer:** Introduction – Client server Programming – **Standard Client Server Protocols:** World Wide Web and HTTP – FTP – Electronic Mail – Domain Name System – **Cryptography and Network Security:** Introduction – Symmetric key Ciphers – Asymmetric key Ciphers.

**Text Book:**

Behrouz A Forouzan, "Data Communications and Networking", Fifth Edition, McGrawHill Education (India), 2013.

**Reference Books:**

1. Andrew S Tanenbaum, "Computer Networks", Pearson Publications, Fourth Edition., 2016.
2. Achyut and Godbole, "Data Communications and Computer Networks", Tata McGraw Hill Edition, 2006.

## COMPUTER GRAPHICS AND MULTIMEDIA-THEORY
## (21UCS62)

| SEMESTER–VI | CORE-T13 | HOURS–5 | CREDITS-5 | TOTAL HOURS:75 |
|---|---|---|---|---|

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

- Defining display devices and its behavior(K1)
- Understand about the basic terminology of Graphics algorithm.(K2)
- Learn about attributes of output primitives (K4)
- Apply filling Algorithms on different objects(K3)
- To visualize 2D Transformations. (K5)
- Learn about concepts of Computer Animation and Multimedia(K2)

**UNIT I** **(15 Hours)**

**Raster scan systems**: Random scan systems - **Output primitives:** Points and lines - Line Drawing algorithms – DDA algorithm – Bresenham's Line algorithm – Parallel line algorithm – Line Function – Circle Generating algorithms – Ellipse Generating algorithms – Other Curves – Parallel Curve algorithm – Curve Functions – Boundary Fill Algorithm – Flood Fill Algorithm.

**UNIT II** **(15 Hours)**

**Attributes of Output primitives:** Line and Curve attributes – Color and Gray Scale Levels – Area fill Attributes – Character attributes – Bundled attributes – Inquiry functions - Antialiasing.
**Two Dimensional geometric transformations:** Basic transformations – Translation – Matrix representations and Homogeneous Coordinates – Composite Transformations – Other Transformations – Transformation between Coordinate Systems – Affine Transformations – Transformation Functions – Raster Methods for Transformations.

**UNIT III** **(15 Hours)**

**Two Dimensional Viewing:** The Viewing Pipeline – Viewing co-ordinate reference Frame – Window-to-Viewport- Coordinate Transformation – Two Dimensional Viewing Functions – Clipping Operations – Point Clipping – Line Clipping – Curve Clipping – Text Clipping – Exterior Clipping. **Three Dimensional Concepts:** Three Dimensional Display Methods – Three Dimensional Graphics Package.

**UNIT IV** **(15 Hours)**

**Flash Workspace:** Panels – Layers – Concept of Frame – Drawing Model – Vector and Bitmap Graphics – Drawing Models – Graphics Creating – Transforming and Aligning Graphics – Text Attributes – Modifying Text Attributes – Transforming Text.

**UNIT V** **(15 Hours)**

**Creating Symbols**: Creating Button – Editing Symbols – Instance of Symbol – Libraries – Frame by Frame Animation – Sound Mixing – Motion Tweening – Shape Tweenning – Onion Skin Features – Guided Motion Tweening – Action Script.

**Text Books:**

1. Donald Hearn & M. Pauline Baker, Computer Graphics C Version, Second Edition, Pearson Low Price Edition, 2019. (for units I ,II & III).
2. Stalini Gupta, Adity Gupta, "Flash 8 in Simple Steps, Dreamtech Press, 2012. (for units IV & V).

**Reference Books:**

1. Computer Graphics principles & Practice by Foley Van Dam , Feiner Hughes, 2nd Edition., 2015.
2. Steven Harrington," Computer Graphics",Second edition., 2016.
3. Robert Reinhardt and John Warren Lentz, "Flash 8 Bible" IDG Books India (P) Ltd**.,** 2013.
4. Ze-Nian Li and Mark S Drew, Fundamentals of Multimedia, Pearson Edn. International, Third Edition, 2005.

# CLOUD COMPUTING - THEORY
# (21UCS63)

| SEMESTER–VI CORE–T14 HOURS–5 CREDITS-5 TOTAL HOURS:75 |
|---|

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

- Perceptive of cloud architecture and model(K1)
- Understand the concept of Virtualization(K2)
- Understand the features of cloud simulator(K2)
- Understand the design of cloud Services.(K3)
- Learn to design the trusted cloud Computing system(K5)

## UNIT I (15 Hours)

**Introduction:** Cloud computing at a glance - Historical developments - Building cloud computing environments - Principles of Parallel and Distributed Computing - Eras of computing - Parallel vs. distributed computing - Elements of parallel computing - Elements of distributed computing - Components of a distributed system - Architectural styles for distributed computing - Technologies for distributed computing - Service-oriented computing.

## UNIT II (15 Hours)

**Virtualization:** Characteristics of virtualized environments - Increased security - Taxonomy of virtualization techniques - Virtualization and cloud computing - Pros and cons of virtualization - Xen: paravirtualization - VMware: full virtualization - Microsoft Hyper-V - Cloud Computing Architecture - The cloud reference model - Infrastructure- and hardware-as-a-service - Platform as a service - Software as a service - Types of clouds - Economics of the cloud - Open challenges.

## UNIT III (15 Hours)

**Cloud application programming:** Aneka - Framework overview - Anatomy of the Aneka container - Building Aneka clouds - Private cloud deployment mode - Public cloud deployment mode - Hybrid cloud deployment mode - Cloud programming and management - Aneka SDK - Management tools - Data-Intensive Computing - Characterizing data-intensive computations - Technologies for data-intensive computing - Aneka MapReduce programming - Introducing the MapReduce programming model - Example application.

## UNIT IV (15 Hours)

**Cloud Platforms in Industry:** Amazon web services - Compute services - Storage services - Google AppEngine - Architecture and core concepts - Microsoft Azure - Azure core concepts - SQL Azure - Windows Azure platform appliance - Cloud Applications - Scientific applications - Business and consumer applications - Social networking - Media applications - Multiplayer online gaming.

**UNIT V** **(15 Hours)**

**Security in the Cloud:** Cloud Security Challenges - Software-as-a-Service Security - Security Management - Security Governance - Risk Management - Risk Assessment - Security Portfolio Management - Security Awareness - Education and Training Policies, Standards, and Guidelines - Secure Software Development Life Cycle - Security Architecture Design - Vulnerability Assessment - Data Privacy - Data Governance - Data Security - Identity Access Management (IAM) -- Data Privacy - Change Management - Business Continuity and Disaster Recovery.

**Text books**:

1. Rajkumar Buyya, Christian Vecchiola, S.Tamarai Selvi, 'Mastering Cloud Computing", Tata McGraw Hill Education Private Limited,2013. (Units I-IV) Chapters: 1, 2, 3, 4, 5, 8, 9, 10.
2. John W.Rittinghouse and James F.Ransome, "Cloud Computing: Implementation, Management, and Security", CRC Press, 2010. (Unit V), Chapter: 6

**Reference books:**

1. Kai Hwang, Geoffrey C Fox, Jack G Dongarra, "Distributed and Cloud Computing, From Parallel Processing to the Internet of Things", Morgan Kaufmann Publishers, 2012.
2. Toby Velte, Anthony Velte, Robert Elsenpeter, "Cloud Computing, A Practical Approach", TMH, 2009.
3. Kumar Saurabh, "Cloud Computing – insights into New-Era Infrastructure", Wiley India, 2011.
4. Gautam Shroff, Enterprise Cloud Computing, Cambridge University Press, 2011.

## PRACTICAL - COMPUTER GRAPHICS AND MULTIMEDIA (21UCS64)

| SEMESTER–VI | CORE–P11 | HOURS–4 | CREDITS–2 | TOTAL HOURS:60 |
|---|---|---|---|---|

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

- Draw line, circle, Ellipse using Bresenham's Algorithms(K1)
- Learn about how to translate, rotate and scale objects(K2)
- Algorithmic method apply for clipping objects(K3)
- Draw images using Graphics software (K4)
- Learn Animation using coordinate movement(K2)

### COMPUTER GRAPHICS

1. Line Drawing using DDA method
2. Line Drawing using Bresenham's Algorithm
3. Circle Drawing using Bresenham's Algorithm
4. Area filling  using Flood fill Algorithm
5. Translate an object
6. Scaling an object
7. Rotating an object
8. Line Clipping

### MULTIMEDIA

1. Frame by frame animation of an object.
2. Animate an object using Multiple layer.
3. Perform a simple animation using button control.
4. Create a Banner text using a mask.
5. Animating an image using different Tweenings(Tint, Shape, Guided and Motion).
6. Digital Clock Displaying
7. Design a calculator using Action Script.

## PRACTICAL - CLOUD COMPUTING
## (21UCS65)

| SEMESTER–VI | CORE-P12 | HOURS–4 | CREDITS–2 | TOTAL HOURS : 60 |
|---|---|---|---|---|

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

- Work with Google Drive(K1)
- Explore Google cloud.(K2)
- Learn to install Google App engine (K3)
- Create project in cloud9 (K5)
- Work with Codenvy (K3)

1. Working with Google Drive to make spreadsheet and notes.
2. Installation and Configuration of Just cloud.
3. Creating a Warehouse Application in SalesForce.com.
4. Exploring Google cloud for the following a) Storage b) Sharing of data c) manage your calendar, to-do lists, d) a document editing tool
5. Working and installation of Google App Engine
6. Working and installation of Microsoft Azure
7. Creating a project in Cloud9
8. Working in Codenvy to demonstrate Provisioning and Scaling of a website

# Extra Credit Course Papers

## ECC-1 FUNDAMENTALS OF COMPUTER-THEORY (21UEC11)

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

- Introduce components of a computer.(K1)
- Understand about operating system.(K2)
- Understand about programming principles.(K2)
- Understand about different types of software.(K3)
- Analysis the relationship between software and hardware.(k4)

**UNIT I**

**INTRODUCTION:** Components of computer (block diagrams) – characteristics of computers – generation of computers – classification of computer – application of computers – number systems (decimal, binary, octal, hexadecimal) – memory hierarchy.

**UNIT II**

**SECONDARY STORAGE:** Introduction – Classification of storage devices – Input devices – Key board – Pointing devices – Classification of output – softcopy output devices – Monitors – Projectors.

**UNIT III**

**OPERATING SYSTEMS:** Function of operating systems – Managing files with mycomputer and windows explorer - general dos commands – Internal commands – External commands – Creating, Copying, Renaming, Deleting Moving files and folders – searching files.

**UNIT IV**

**COMPUTER PROGRAM:** Computer program – Developing a program – Algorithm – Flow chart – Characteristics of good program –Computer languages – Evaluation of Programming languages – Classification of programming languages.

**UNIT V**

**COMPUTER SOFTWARE:** Definition – Relationship between hardware and Software – Software categories – System software – Application software – Software terminology.

**Text book:**

Introduction to Computer Sciene, ITL Education Solution Ltd., Pearson Education, 2014.

**Reference Book:**

Introduction to Computers and BASIC Programming by Dr. C. Xavier.

# ECC 2 - INTERNET CONCEPTS-THEORY
## (21UEC21)

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

- Basic concepts of internet .(K1)
- Learn about charting.(K3)
- Understand about WWW concepts.(K2)
- Familiar with email operations.(K4)
- Gain overall knowledge about the subject(K4)

### UNIT I

**Connecting to the Internet:** Internet Connection Concepts – Configuring the Internet Connection – Connecting LAN to Internet.

### UNIT II

**Exchanging E-mail:** E-mail concepts – Configuring E-mail program – Sending and Receiving Files by E-mail – Fighting Spam, Sorting Mail, and Avoiding

### UNIT III

**Chatting and Conferencing on the Internet:** Online chatting – Messaging, and Conferencing Concepts – Usenet Newsgroups Concepts – Internet Relay Chat (IRC) – Instant Messaging – Voice and Video Conferencing.

### UNIT IV

**World Wide Web:** World Wide Web Concepts – Streamlining the Browsing – Web Security, Privacy, and Site Blocking.

### UNIT V

**File Transfer and Downloading**: File Transfer by Web, FTP, and Peer – to – Peer – Downloading and Installing software.

### Text Book:

Margaret Levin Young, "**The Complete Reference Internet**", Second Edition, Tata McGraw-Hill, 2017.

## ECC 3 - WEB DESIGN WITH STYLE SHEETS-THEORY (21UEC31)

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

- Learn the features of XML.(K1)
- Knowledge about style sheets.(K3)
- Understand how to customize mouse (K2)
- Work with simple web pages with CSS.(K4)
- Gain over all knowledge about the subject(K5)

**UNIT I**

**Introducing cascading style sheets**-the bits that make up a style sheet- selectors

**UNIT II**

**The cascade and inheritance**-applying font faces- Manipulating the display of text

**UNIT III**

**Background colors and images**-the box model: Controlling margins

**UNIT IV**

**Borders**, padding, width, and height-styling tables

**UNIT V**

**Customizing the mouse cursor**-controlling opacity and visibility

**Text Book:**

Ian Pouncey, Richard York, "Beginning Css-Cascading Style Sheets For Web Design", Third Edition, Wiley Publications, 2011.

## ECC 4 - VISUAL BASIC DOT NET - THEORY
## (21UEC41)

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

- Understand the basic concepts of visual programming(K1)
- Design simple applications(K4)
- Work with GUI applications(K3)
- Understand database applications(K2)
- Gain overall knowledge about the subject(K5)

### UNIT I

**VB.NET 2005 Training:** The .NET Framework Architecture Part 2-Introducing Windows Forms- Implementing Class Library Object in VB.NET 2005 -Introduction and Implementing Inheritance in VB.NET 2005- Visual Studio.NET Namespaces.

### UNIT II

**Windows Designing a Form using Forms** Designer Window-Exploring the Forms Designer generated code-Using Application Class and Message Class-Setting and Adding Properties to a Windows Form - Event Handling In Visual Basic .NET 2005.

### UNIT III

**Building graphical interface elements**-Adding Controls -Common Controls and Handling Control Events-Dialog Boxes in Visual Basic .NET 2005 -Common Windows Forms Controls Section-DomainUpDown and NumericUpDown -Creating Menu and Menu Items-Creating Multiple-Document Interface (MDI) Applications  Validation-Exceptions.

### UNIT IV

**Creating and Managing Components** Section-Creating and Managing .NET Assemblies-Simple Data Binding-Complex Data Binding-Using the Data Form Wizard-Access and Manipulate Data - The ADO .NET Object Model-Access and Manipulate Data - Using DataSets-Using XML Data.

### UNIT V

**Finding and Sorting Data in DataSets**-Editing Data With ADO .NET-Web Services - SOAP, WSDL, Disco and UDDI-Instantiating - Invoking Web Services, Creating Proxy Classes with WSDL-Creating Web Service Project.

**Text Book:**

Programming Visual Basic .NET, Dave Grundgeiger, 2008.

**Reference Books:**

1. Beginning VB.Net, Richard Blaire, Jonathan Crossland, Mathew Renolds, 2nd Edition, 2008.
2. Programming VB.Net, Garry Cornell, Jonathan Morrison, APress Publications, 2007.

## ECC 5 - WIRELESS TECHNOLOGY - THEORY
## (21UEC51)

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

- About wireless technology(K1)
- Understand the layers of network(K2)
- Understand about the layers of network planning(K3)
- Knowledge about the applications of wireless technology(K4)
- Gain overall knowledge about the subject(K5)

### UNIT I

**OVERVIEW OF WIRELESS NETWORKS:** Introduction, Different generations of wireless networks. **CHARACTERISTICS OF THE WIRELESS MEDIUM**: Introduction, radio propagation mechanisms, path-loss modeling and signal coverage, effects of multi path and Doppler, channel measurement and modeling techniques.

### UNIT II

**PHYSICAL LAYER ALTERNATIVES FOR WIRELESS NETWORKS:** Introduction, applied wireless transmission techniques, short distance base band transmission, UWB pulse transmission, Carrier Modulated transmission, Broadband modems for higher speeds, Spread Spectrum transmissions, High-speed Modems for Spread spectrum technology, Diversity and Smart Receiving Techniques, Comparison of modulation schemes, Coding techniques for wireless communications - **WIRELESS MEDIUM ACCESS ALTERNATIVES:** Introduction, fixed-assignment access for Voice-Oriented networks, Random access for Data-Oriented Networks, Integration of Voice and Data Traffic.

### UNIT III

**NETWORK PLANNING:** Introduction, wireless network topologies, Cellular Topology, Cell Fundamentals, Signal-to-interference ratio calculation, capacity Expansion Techniques, network planning for CDMA systems. **WIRELESS NETWORK OPERATION:** Introduction, mobility management, radio resources and power management, security in wireless networks.

### UNIT IV

**WIRELESS APPLICATION PROTOCOL:** Design and Principles of Operation, WAP Architecture & Components, WAE Overview, WAE Model, WTA Architecture, WTA Framework Components, WSP Specification, WTP Specification, WTLS Specification, WDP Specification.

## UNIT V

**BLUETOOTH:** Design and Principles of Operation, Transmitter Characteristics, Bluetooth Security, Link Manager Protocol, Logical Link Control and Adaptation Layer Protocol, Alternatives to Bluetooth. **WIRELESS LANs:** Benefits of WLANs, Design and principles of Operation, WLAN Configurations, Microcells and Roaming, Types of WLANS, IEEE802.11, IEEE802.11a, IEEE802.11b.

### Text books:

1. Kaveh Pahlavan and Prashant Krishnamurthy," Principles of Wireless Networks-a Unified approach", Pearson , 2004.
2. Gary S.Rogers et al, "An Introduction to Wireless Technology", Pearson, 2007.

### Reference books:

1. William Stallings,"Wireless communications and Networks", Pearson education, 2005, ISBN 81- 7808-560-7
2. Jim Geier, "Wireless Networks first-step", Pearson, 2005.
3. Sumit Kasera et al, "2.5G Mobile Networks: GPRS and EDGE", TMH, 2008.
4. Matthew S.Gast, "802.11 Wireless Networks", O'Reilly, Second Edition, 2006.
5. Theodore s. Rappaport,"Wireless Communications –principles and practice", second edition, PHI, 2002
6. C.S.R.Prabhu et al, "Bluetooth Technology and its Applications with Java and J2ME", PHI, 2007.

# Certificate Course Papers

## DESKTOP PUBLISHING - THEORY
## (21UCC01)

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

- Basics of PageMaker.(K1)
- Learn how to create documents using PageMaker.(K2)
- Work with Photoshop Software.(K3)
- Understand the knowledge of Colors and Brushes.(K2)
- Knowledge of Restoring and Aligning(K4)

### UNIT I

**PageMaker:** PageMaker – the PageMaker environment – the basics of Creating a new document (The basics of creating a new document) the basics of using PageMaker text – the PageMaker text- the basics of using PageMaker Graphics – Applying Color to simple graphics.

### UNIT II

**DOCUMENTS IN PAGEMAKER**: Meaning of document layout – Managing documents – Managing PageMaker text – Creating document consistency using – Creating tables of contents and indexes –Creating a book in PageMaker.

### UNIT III

**PHOTOSHOP:** Introduction to Photoshop – When to use Photoshop - When to use a drawing program – Inside Photoshops: the Photoshop Desktop – Navigate in Photoshop – Customizing the Interface. Image management: How to open, Duplicate and Save Images. Adding file information- and annotations - Resumption and Cropping.

### UNIT IV

**COLORS AND BRUSHES:** Defining Colors: Selecting And Editing colors – Working in Different Color Modes – Using Photoshop's other color selecting methods. Printing and brushes: Meet the Paint and Edit Tools –Basic Techniquers – Brush size and shape - Brush Modes. Filling and Stroking: Filling selection with color or patterns : Applying gradient files= Applying Strokes and Arrow heads.

### UNIT V

**RESTORING AND ALINGNING**: Retouching and restoring: Cloaning and Healing: Retouching Photographs – Corrective Filtering: Filter Basics – Hightening forms and contrast. Working with Layers – Selecting the content of Layers – Moving, Linking and Aligning.

**Text Books**:

1. Carolyn M. Connolly, ”PageMaker 7 – The Ultimate Reference”, Dreametch Press, 2005.
2. Deke McClelland, ”Photoshop CS Bible”, Wiley Publishing Inc.

**Reference Books**:

1.Vikas Gupta, “Comdex DTP Course Kit ”, Dreametch Press,2002.
2. Roger C. Parker,”Desktop Publishing & Design For Dummies”,Wiley 1995.

# MOBILE APPS WITH ANDROID
# (21UCC02)

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

- Understand the basics Android technology(K1)
- Learn how to create user interface using Android.(K1)
- Work with file systems.(K3)
- Understand about Adaptors and content providers(K2)
- Understand about NDK ( Native Development Kit)(K4)

## UNIT I

**Android Overview** – The Stack – Quick Start

## UNIT II

**Main Building Blocks** – Android User Interface

## UNIT III

**Preferences** – The File System – The Options Menu, and Intents – The Database

## UNIT IV

**Lists and Adapters** – Content Providers

## UNIT V

**The Android Interface Definition Language** – The Native Development Kit(NDK)

**Text Book:**

Marko Gargenda, ”Learning ANDROID”, First Edition 2011, O’REILLY Publication. Chapters-1,2,3,4,67,9,10,12,14,15

**Reference Book:**

Android Community Experts, “Android Cookbook”, First Edition 2011, O’REILLY Publications.

## PRACTICAL – MOBILE APPs WITH ANDROID

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

- Create simple applications using Android (K3)
- Apply Android Components to create applications (K3)
- Implement various features to create innovative applications (K4)
- Create application using various UI components (K3)
- Update knowledge to learn any future advanced version of language (K4)

1. Simple Program
2. Create sample application with login module. (Check username and password)
3. Open browser and any URL.
4. Understand resource folders
5. Understand Menu option
6. Create an application that will display toast(Message) on specific interval of time.
7. Create a background application that will open activity on specific time.
8. Understanding of UI:
9. Understanding content providers and permissions
10. Read messages from the mobile and display it on the screen.
11. Create an application that will create database with table of User credential.
12. Create an application that read file from asset folder and copy it in memory card.
13. Create an application that will play a media file from the memory card.
14. Create an application to make Insert, update, Delete and retrieve operation on the database.
15. Create an application to read file from the sdcard and display that file content to the screen.
16. Create an application to draw line on the screen as user drag his finger.
17. Create an application to send message between two emulators.
18. Create an application to take picture using native application.

# ADD-ON COURSES

## AOC- 1 INTRODUCTION TO ROBOTICS
### (21UAC01)

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

- Define the concept of Embedded Systems(K1)
- Develop program to work with Embedded Systems and Robotics(K2)
- Design models of Robots for simple problems(K3)
- Ensure sustained use of robotics (K4)
- Implement projects in Embedded Systems and Robotics(K4)
- Evaluate the various models of Robots used in households and industries(K5)

**UNIT I**

Installation guide to AtmelStudio – Setting up project in AtmelStudio – Installing and Using AVR Bootloader.

**UNIT II**

Introduction to Firebird V - I/O Interfacing – LCD interfacing – Motor Interfacing.

**UNIT III**

Interrupts - Interrupts - Timer Overflow Interrupt - Position and Velocity Interrupts - Serial Interrupt.

**UNIT IV**

USB to Serial Communication using Zigbee - ADC Interfacing - Seven Segment Display Interfacing - Stepper Motor Interfacing - Keypad Interfacing - Keypad Interfacing .

**UNIT V**

Temperature Sensor Interfacing - 8255 (Programmable Peripheral Interface) Interfacing - 8255 (Programmable Peripheral Interface) Interfacing - Motion Control using Pulse Width Modulation.

**REFERENCES**

1. e-yantra Lab Materials.

2. e-yantra 8051 Lesson Plan.

## AOC- 2 PC ASSEMBLING AND TROUBLESHOOTING-THEORY (21UAC02)

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

- Recognize basics of hardware components and its characteristics(K1)
- Understand about different processors(K2)
- Learn about installation, configuration and upgrading software(K3)
- Learn to trouble shoot in the microcomputer(K4)
- Gain overall knowledge about the subject(K5)

### UNIT I

**Assemble and setup and upgrade personal computer systems :** Identify modules that make up a computer system and its operation - Understand that a computer requires both hardware and software to work - Describe the different hardware components inside of and connected to a computer.

### UNIT II

Identify each type of computer bus structure - Learn about the many different processors – processor history – processors used for personal computers and notebook computers.

### UNIT III

**Perform installation, configuration, and upgrading of microcomputer hardware and Software :** Assemble/setup microcomputer systems, accessory boards - Learn about the different types of motherboards and how to select one - Install or replace a motherboard - Troubleshoot problems with memory.

### UNIT IV

**Install/connect associated peripherals :** Learn how printers and scanners work- Install printers and scanners and how to share a printer over a local area network - Troubleshoot printer and scanner problems - Solve hard drive problems –

### UNIT V

**Diagnose and troubleshoot microcomputer systems hardware and software, and other peripheral equipment:** Understand how to approach and solve a PC problem -
Troubleshoot a failed boot before the OS is loaded - Describe the general approaches you need to take when installing and supporting I/O devices - diagnose and isolate faulty components.

**Text Book:**

A+ Guide to Hardware: Managing, Maintaining, and Troubleshooting, “Jean Andrews” Fourth Edition, 2016.