CHOICE BASED CREDIT SYSTEM (CBCS)
CBCS Pattern
Proposed Syllabus outline of B.C.A. Second Year
Effective from Academic Year (2018-2019)

Semester	Course Code	Course Name	Credits / Marks				Total	
			Internal		External			
			Credit	Marks	Credit	Marks	Credit	Marks
Semester - III	**1. Ability Enhancement Courses (AEC) (Compulsory Course)**							
	BCA-AEC-3	English- I	--	20	--	30	2	50
	2. Core Course (CC) (Compulsory Course)							
	BCA-CC-A3	OOP's	--	20	--	30	3	50
	BCA-CC-B3	Database Management System	--	20	--	30	3	50
	BCA-CC-C3	Computer Network	--	20	--	30	3	50
	BCA-CC-D3	Multimedia Systems	--	20	--	30	3	50
	3. Skill Enhancement Course							
	BCA-SEC-1	Android O.S.		20	--	30	2	50
	4. Practical / Lab Course							
	BCA-Lab-9	Lab-Course1 (C++)	--	20	--	30	2	50
	BCA-Lab-10	Lab-Course2 (DBMS)	--	20	--	30	2	50
	BCA-Lab-11	Lab-Course3 (Computer Network)	--	20	--	30	2	50
	BCA-Lab-12	Lab-Course4 (Multimedia)	--	20	--	30	2	50
		Total (III)					**24**	**500**

Semester	Course Code	Course Name	Credits / Marks				Total	
			Internal		External			
			Credit	Marks	Credit	Marks	Credit	Marks
Semester - IV	**1. Ability Enhancement Courses (AEC) (Compulsory Course)**							
	BCA-AEC-4	English- I	--	20	--	30	2	50
	2. Core Course (CC) (Compulsory Course)							
	BCA-CC-A4	Linux OS	--	20	--	30	3	50
	BCA-CC-B4	S.E.	--	20	--	30	3	50
	BCA-CC-C4	Mathematical Foundation for Computer Science	--	20	--	20	3	50
	BCA-CC-D4	Programming with Java	--	20	--	30	3	50
	3. Skill Enhancement Course							
	BCA-SEC-2	Mobile Application Development / Hardware Networking		20	--	30	2	50
	4. Practical / Lab Course							
	BCA-Lab-13	Lab-Course5 (Linux)	--	20	--	30	2	50
	BCA-Lab-14	Lab-Course6 (S.E)	--	20	--	30	2	50
	BCA-Lab-15	Lab-Course7 (Matlab)	--	20	--	30	2	50
	BCA-Lab-16	Lab-Course8 (Java)	--	20	--	30	2	50
		Total (IV)					**24**	**500**
		Total (III + IV)					**48**	**1000**

SEMESTER – III

Course Title: Object Oriented Programming Structure
Course Code:CC-A3

Total Teaching Hours: 50 **Total Marks: 50**
Credit: 3

Learning Objective:

1.Understand object oriented programming and advanced C++ concepts

1.1 Be able to explain the difference between object oriented programming and procedural programming.

1.2 Be able to program using more advanced C++ features such as composition of objects, operator overloads, dynamic memory allocation, inheritance and polymorphism etc.

1.3 Be able to build C++ classes using appropriate encapsulation and design principles.

2. Improve your problem solving skills

2.1 Be able to apply object oriented or non-object oriented techniques to solve bigger computing problems .

Course Outcomes:

- Gain the basic knowledge on Object Oriented concepts.
- Ability to develop applications using Object Oriented Programming Concepts.

After the completion of this course, a successful student will be able to do the following:
a) Use the characteristics of an object-oriented programming language in a program.
b) Use the basic object-oriented design principles in computer problem solving.
c) Use the basic principles of software engineering in managing complex software project.
d) Program with advanced features of the C++ programming language.

Syllabus

UNIT-I : Introduction to Object Oriented Programming

1. Getting Started:

Introduction, A brief history of C++, Advantages of OOP - Usage of OOP and C++
Variable, constant, Expression, Statements, Comments and keywords of C++

2. Operator:

Arithmetic, Relational, Logical, Assignment, Increment/Decrement, Conditional,
Precedence of Operators. Data type, Type Conversion, library function.

3. Input / Output Statements:

Inputting using cin and outputting using cout statements.
Preprocessor directives. Basic program construction.

UNIT-II :Array , control and Looping Statement

4. Decision Making and Looping Statement:

If Statement, If..else statement, nesting of if statement,
Switch statement, conditional operator statement.
While loop, Do loop,
For loop, nesting of loops, break and continue statement, go to statement.

5. Arrays:

Defining an array
Array type, array elements

Accessing and averaging array elements, initializing array.
Programming of C++ with array.
String handling, array of strings.

UNIT-III : Introduction to Class and Functions

6. Functions:

What is a function?, Declaring and defining function.
Local, global variables, execution of function.
Passing argument to function, Return values.
Reference arguments. Overloading functions.
Inline function and default parameter.

7. Object OrientedProgramming:

Objects & Classes. Constructor & Destructor.
Operator overloading. Overloading unary operators.
Overloading binary operators.

UNIT IV : Inheritance and Polymarphism

8. Inheritance:

Derived class and Base Class. Derived Class Constructors. Overriding member functions.
Inheritance in the English distances class, class hierarchies. Public and Private Inheritance.
Level of inheritance.

9. Polymorphism:

Problems with single inheritance. Multiple inheritances.

REFERENCES/BOOK-

1. C++ & Graphics by Vijay Mukhi's
2. Object Oriented Programming in Turbo C++, Robert Lafore, Fourth Edition Galgotia Publications.
3. Mastering C++.
4. C++ Programming Language by Schaum' soutline series.
5. Object Oriented Programming with C++, E. Balagurusami, Fourth Edition, Tata Mc-Graw Hill

Course Code:- CC-B3
Database Management System

Total Teaching Hours: 50 **Total Marks: 50**
Credit: 3

Learning Objectives

- To learn the fundamentals of data models and to conceptualize and depict a database system using ER diagram.
- To make a study of SQL and relational database design.
- To understand the internal storage structures using different file and indexing techniques which will help in physical DB design.
- To know the fundamental concepts of transaction processing- concurrency control techniques and recovery procedure

Course Outcome

- Assemble the information that is needed to design a database management system for a business information problem.
- Create conceptual and logical database designs for a business information problem.
- Design a database management system that satisfies relational theory and provides users with business queries, business forms, and business reports.
- Analyze the core terms, concepts, and tools of relational database management systems.

SYLLABUS

UNIT I: Introduction to Database And Elements of DBMS

1. Introduction to Database
1.1. Definition of DBMS, File processing Vs DBMS
1.2. Advantages and disadvantages of DBMS
1.3. Users of DBMS
1.4. DBMS Structure

2. Elements of DBMS
2.1. DBMS Languages: DDL, DML, DCL
2.2. Terms: Entity, Entity set, attributes
2.3. Keys: Primary, secondary, foreign, composite

UNIT II: Data Models

3. Data Models And Relational Algebra and Calculus
3.1. Introduction, Object based logical model,
3.2. record based logical model (RDB, NDB, HDB)
3.3. E-R model, E-R diagram

4. Relational Algebra and Calculus
4.1. Introduction
4.2. Relation, Schemes, Domain, Tuples
4.3. Cardinality degree
4.4. Algebraic operation
4.5. Fundamental operation: Select, product, union
4.6. Set difference : Natural join, Cartesian product, rename
4.7. Relational calculus: Tuple and domain relational calculus

UNIT III: Relational Database Design And SQL

5. Relational Database Design

5.1. Normalization: INF, 2NF, 3NF, BCNF
5.2. Class diagrams and E-R tables
5.3. Functional dependency

6. SQL

6.1. Data types
6.2. Table Creation, Modify
6.3. Selecting, Deleting records
6.4. Simple queries
6.5. Oracle constraints

REFERENCE BOOKS

1. Ramez Elmasri and Shamkant B. Navathe, “Fundamental Database Systems”, Third Edition, Pearson Education, 2008.
2. Raghu Ramakrishnan, “Database Management System”, Tata McGraw-Hill Publishing Company, 2003.
3. Abraham Silberschatz, Henry F. Korth and S. Sudarshan- “Database System Concepts”, Sixth Edition, McGraw-Hill, 2010.

Course Code: CC-C3
Course Title: Computer Networks

Total Teaching Hours: 60 **Total Marks: 50**
Credits : 03

Learning Objectives: At the end of the course, the students will be able to:

- Build an understanding of the fundamental concepts of computer networking.
- Familiarize the student with the basic taxonomy and terminology of the computer networking area.
- Introduce the student to advanced networking concepts, preparing the student for entry Advanced courses in computer networking.
- Allow the student to gain expertise in some specific areas of networking such as the design and maintenance of individual networks.

Course Outcomes:

After completing this course the student must demonstrate the knowledge and ability to:

- Independently understand basic computer network technology.
- Understand and explain Data Communications System and its components.
- Identify the different types of network topologies and protocols.
- Enumerate the layers of the OSI model and TCP/IP. Explain the function(s) of each layer.
- Identify the different types of network devices and their functions within a network
- Understand and building the skills of subnetting and routing mechanisms.
- Familiarity with the basic protocols of computer networks, and how they can be used to assist in network design and implementation.

SYLLABUS

UNIT-I : Networking Fundamentals And Network Components

1) Networking Fundamentals

Terminologies Client, Server, Topology, Types of Network, Hierarchical Central Computer, Peer to Peer Network, Client Server Network, Types of Network Topologies, Types of Network Technologies, Types of Data passing Schemes

2) Network Components

Types of Cablings- Coaxial, UTP, STP, FOC, Types of Connectors- RJ-45, Terminator, T-Connector, BNC, HUB, Switch, Router

UNIT-II : Network Hardware and Components And Protocols and Services

3) Network Hardware and Components

Introduction of Network Cable like UTP, STP, Fiber Optics, Hub, Unmanageable Switch, Manageable Switch, Router, Modem, Wi-Fi, Access Point, PCI Wireless Card, USB Wireless Device, Print Server, USB Network Sharer, Backup Device, Server Hardware etc. Network Interface Card, Crimping tools and Color standards for straight crimping and Cross crimping, Repeaters, Hub, Switches, Routers, Bridges.

4) Protocols and Services

HTTP, FTP and other Different types of protocols, Media Access Method, DNS services, DHCP services, WINS services and RAS services, Web services, Proxy Services etc.

UNIT-III : Device Installation And Diagnostic Tools & PC Maintenance

5) Device Installation

Graphics Card, Sound Card, LAN Card, Wireless LAN Card, SCSI Card, External Drive, Flash Cards, Web Camera, CCTV Camera, Mobile Devices, Pen Drive, Firewire Cards, Modem, Plotter, Wireless LAN, Access Point etc.

6) Diagnostic Tools & PC Maintenance

Introduction, Virus and its types, Effect of Virus for Computer System, Scanning and Antivirus remover tools, Antivirus Utilities for Diagnostic, Safety and Preventive Maintenance Tools, Data Recovery, Concept of Fax and E-mail, PC care and Maintenance, Electrical Power Issues, Troubleshooting PC

Hardware:- O/S Troubleshooting issues in computer System.

UNIT-IV :) Network Introduction & Installation And Network Administration

7) Network Introduction & Installation Introduction About Network, Installing Network Operating System Windows 2003 Server and Windows 2008 Server, Cable Crimping, Network Sharing and user Permission, Internet Connection.

8) Network Administration
Installing and Configuring Wire & Wireless Network, Network Troubleshooting, Installing Manageable Switches, Routers, Wi-Fi Device.

Reference Books :
1) James F. Kurose & Keith W. Ross, "Computer Networking : A Top-Down Approach ", 6th edition , Pearson / Addison Wesley 2013 .
2) Networking Complete BPB Publication
3) Computer Networking Andrew S. Tanenbawan by PH

Course Title: Multimedia Systems
Course Code: CC-D3

Total Teaching Hours: 50 **Total Marks: 50**
Total Credit :-03

LEARNING OBJECTIVES:

- To learn the basic concepts of animation as an art.
- To produce exercises as well as a final project in animation using Flash.
- To understand the basic animation techniques and concepts covered in the films and projects viewed in class.

COURSE OUTCOMES:

- To develop an understanding of the tools used for creating two dimensional (2D) Graphics and animation.
- To facilitate the students to design 2D Graphics, 2D character modelling and Animation.
- To make the students create a CBT presentation.

UNIT – I

1. **Exploring The Flash Interface**
1.1. The Flash stage
1.2. Stage Settings
1.3. Creating a new Flash file
1.4. The various import formats
1.5. Timeline- Play head/Frames/Key Frames/ Blank frames
1.6. Menus, Toolbox and Properties
1.7. Color Swatches and Color Mixer
1.8. Rulers, Guides, Grids and Snappings

UNIT - II

2. **Introduction**
2.1. CDROM and Multimedia Highway
2.2. Applications of Multimedia
2.3. Stages of Multimedia Project
3. **Macintosh and Windows Productions Platforms**
3.1. Macintosh Platform
3.2. Windows Platform
3.3. Connections- SCSI and IDE
3.4. Memory and Storage devices
3.5. Input and Output Devices

UNIT - III

4. **Basic Software Tools**
4.1. Text editing and word Processing tools
4.2. Painting and drawing tools
4.3. Image Editing Tools
4.4. Sound Editing Tools
4.5. Font Editing and designing tools
4.6. Hypermedia and Hypertext
4.7. Making Still Images : BITMAPS , Vector Drawing
4.8. Colors, Image file formats

Unit - IV

5. **Animation and Video**
5.1. Principal of Animation
5.2. Making animation that work: Rolling Ball, Bouncing ball
5.3. Using Video
5.4. Broadcast Video Standards
5.5. Recording Formats

References Books:

1. Macromedia Flash MX 2004: The Complete Reference by Brian Underdahl
2. Action Script for Flash MX: The Definitive Guide, 2nd Edition By Colin Moock
3. Macromedia Flash MX 2004 Bible by Robert Reinhardt and Snow Dowd
4. Multimedia : Making it work (5th Editions) By Tay Vaughan (Tataamc)
5. Multimedia : Computing Communications and Applications By Ralf Steinmetz ,Klara Nahrstedt

Recommended Web sites:

1. http://www.webdevelopersnotes.com/tutorials/flash/
2. http://www.adobe.com/devnet/flash/
3. http://www.adobe.com/support/flash/tutorial_index.html
4. http://www.thefreecountry.com/webmaster/flash.shtml

Class :- B.C.A- S.Y [III-Sem]

Course Title :- Android Operating System **Course Code:-SEC-I**

Credit :- 02

Learning Objective:- Learn Basic of Android operating system, Learn basic of XML , and learn basic concepts of java programming like Classes, Packages, Thread, Exception Handling etc.
Course Outcome:-Student should write xml code to design android controls , students also able to write java programs with object oriented features, they should create their own packages and able to access created packages.

Syllabus

Unit –I :- Android History and Scope

Chp 1. Introduction to Android

1.1 Need of Mobile Application

1.2 Introduction to Android

1.3 Types of Mobile Applications

1.4 Android Versions

Chp 2. Android Architecture

2.1 Android Architecture

2.2 Linux Kernal

2.3 Dalvik Virtual Machine

UNIT –II:-IDE's and Java Basic

Chp 3. Android IDE's & Components

3.1 Various IDE For Android

3.2 Installtion of Android

3.3 Android Virtual Device

3.4 Android Components

Chp 4. Introduction To Java

4.1 Introduction & History of Java

4.2 Java Applications

4.3 Java Architecture

Unit – III:- Java Programming

Chp 5. Programming Basics

5.1 Variable, Constants

5.2 Hello World Program

5.3 Classes&Inheritance

5.4 Interface

Unit –IV:- Packages & UI Designing

Chp 6. Java Packages & Threads

6.1 Packages

6.2 Thread

6.3 Exception Handling

6.4 Method Overloading

Chp 7. XML &Json

7.1 Tag ,Attribute

7.2 XML

7.3 Json

Reference Books:-

1. The Complete Reference Java2 By Herbert Schildt
2. Java CookBook By Ian Darwin,PublisherO'Reilly

U-LAC-CC-A3
(OOP C++)

Practical Total Marks: 50 **CREDIT: [02]**

Learning Objective:

1.Understand object oriented programming and advanced C++ concepts

1.1 Be able to explain the difference between object oriented programming and procedural programming.

1.2 Be able to program using more advanced C++ features such as composition of objects, operator overloads, dynamic memory allocation, inheritance and polymorphism etc.

1.3 Be able to build C++ classes using appropriate encapsulation and design principles.

2. Improve your problem solving skills

2.1 Be able to apply object oriented or non-object oriented techniques to solve bigger computing problems .

Course Outcomes:

- Gain the basic knowledge on Object Oriented concepts.
- Ability to develop applications using Object Oriented Programming Concepts.

After the completion of this course, a successful student will be able to do the following:
a) Use the characteristics of an object-oriented programming language in a program.
b) Use the basic object-oriented design principles in computer problem solving.
c) Use the basic principles of software engineering in managing complex software project.
d) Program with advanced features of the C++ programming language.

Proposed Practical List:

1. Program to demonstrate encapsulation using class.
2 Program to demonstrate use of array of objects
3 Program to demonstrate use of pointers
4 Program to demonstrate use of looping statements class
5 Program to demonstrate use of function overloading
6. Program to demonstrate inline function.
7 Program to demonstrate use of friend function
8 Program to demonstrate use of different manipulators
9. Program to demonstrate use of constructor, constructor overloading & destructor
10. Program to demonstrate use of all types of Inheritance.
11. Program to demonstrate use of unary & binary operator overloading.

U-LAC-CC-B3
(DBMS)

Practical Total Marks: 50 **CREDIT: [02]**

Learning Objectives

- To implement the design of the tables in DBMS
- To write queries to get optimized outputs
- To store, retrieve and view the contents
- To generate report based on customized need

Course Outcome

- Upon successful completion of the course, students would be able to :
- Identify the information that is needed to design a database management system for a business information problem.
- Create conceptual and logical database designs for a business information problem.
- Construct a database management system that satisfies relational theory and provides users with business queries, business forms, and business reports.
- Analyze the core terms, concepts, and tools of relational database management systems.
- Demonstrate skills to work in teams in teams and utilize effective group techniques to manage a complex project.

Proposed Practical List:

1. Data Definition Language (DDL) commands in RDBMS

2. Data Manipulation Language (DML) and Data Control Language (DCL) commands in RDBMS.

3. High-level language extension with Cursors.

4. High level language extension with Triggers

5. Procedures and Functions.

6. Embedded SQL.

7. Database design using E-R model and Normalization.

8. Design and implementation of Payroll Processing System.

9. Design and implementation of Banking System.

10. Design and implementation of Library Information System.

U-LAC-CC-C3(Computer Network)

Practical Total Marks: 50 **CREDIT: [02]**

Learning Objectives: At the end of the course, the students will be able to:

- Build an understanding of the fundamental concepts of computer networking.
- Familiarize the student with the basic taxonomy and terminology of the computer networking area.
- Introduce the student to advanced networking concepts, preparing the student for entry Advanced courses in computer networking.
- Allow the student to gain expertise in some specific areas of networking such as the design and maintenance of individual networks.

Course Outcomes:

After completing this course the student must demonstrate the knowledge and ability to:

- Independently understand basic computer network technology.
- Understand and explain Data Communications System and its components.
- Identify the different types of network topologies and protocols.
- Enumerate the layers of the OSI model and TCP/IP. Explain the function(s) of each layer.
- Identify the different types of network devices and their functions within a network
- Understand and building the skills of subnetting and routing mechanisms.
- Familiarity with the basic protocols of computer networks, and how they can be used to assist in network design and implementation.

Proposed Practical List:

1) Study of different types of Network Cables.
2) Implementation of cross-wire and straight cable using crimping tool.
3) Study of Network device in detail.
4) Study of Network IP.
5) Connect computer in local area network.
6) Study of basic network commands.
7) Study of network configuration commands.
8) Setting up new network.
9) Assigning IP to a computer in network.
10) Working with DHCP
11) Installing and Configuring Server O.S. / Linux

U-LAC-CC-D3

Practical Total Marks: 50 **CREDIT: [PR-2]**

LEARNING OBJECTIVES:

- To learn the basic concepts of animation as an art.
- To produce exercises as well as a final project in animation using Flash.
- To understand the basic animation techniques and concepts covered in the films and projects viewed in class.

COURSE OUTCOMES:

- To develop an understanding of the tools used for creating two dimensional (2D) Graphics and animation.
- To facilitate the students to design 2D Graphics, 2D character modelling and Animation.
- To make the students create a CBT presentation.

Proposed Practical List:

1. Study of Flash environment.
2. Working with shape tween by designing A to Z characters animation.
3. Working with shape motion by designing A to Z characters animation.
4. Designing Bouncing Ball animation.
5. Creating a commercial Advertisement.
6. Creating a Jumping Man animation.
7. Creating a Flying Bird animation.
8. Creating a Walking Man animation.
9. Creating a Riding Bike animation.
10. Creating a Flying Honey Bee animation.
11. Creating a growing Plant animation.
12. Working with shape tween with blending options.

SEMESTER - IV

Course Title: LINUX OPERATING SYSTEMS
Course Code:CC-A4

Total Teaching Hours:50 **Total Marks:50**
Credit: 03

Learning Objective:
Teach Basics of Linux Operating System

- Teach ownership and permissions of the files and directories.
- Explain why these issues exist.
- Learn to set permissions files/directories
- Learn to manipulate files/directories – list files, create, delete, and move just about anything on the file system
- Mention vi - a standard Unix text editor
- Learn How to create users , groups and setting the passwords
- Understand Different network configuartions
- Know about Storage and backups

Course Outcome:
On completion of this course the student should be able to:
Identify and use UNIX/Linux utilities to create and manage simple file processing operations, organize directory structures with appropriate security, and develop shell scripts to perform more complex tasks.

After learning the Linux O.S student should :

- Use the Linux O.S frequently
- Expert in the all types of commands of Linux
- Make the shell programs very easily to solve the complex problem without any mistakes
- Creating ,deleting, modifying the users , groups
- Use the network file systems.
- Make the backups using backup strategies

SYLLABUS

UNIT I :Introduction to linux file system and devices

1. Introduction
1.1. History of Unix
1.2. Directory structure of Linux
1.3. History of Linux
1.4. Comparison of various operating systems
1.5. Advantages of Linux, Flavors of Linux, Installation notes, Linux Loader, Linux kernel, linuxfilesystem

2. File System and Devices
2.1. File System concept ext3, ext2.
2.2. File systems: - mount,fsconf and other related commands

UNIT II Linux commands and file permissions

3. Working with permissions
3.1. Assigning file permission
3.2. Directory Permission
3.3. Using text editors

3.4. Working with vi&emacs
3.5. System services and run levels
3.6. Controlling services at boot with administration tools (chkconfig& using GUI based services)

4. Linux commands
4.1. General Purpose Utilities – - Man, who, login, logout, shutdown, passwd.
4.2. File management commands – cat, cp, rm, mv, wc,
4.3. Filter related commands – grep ,wc, head, tail, sort
4.4. Directory management commands – pwd, cd, mkdir, rmdir, ls
4.5. Checking directories and permissions – pwd, chmod, cd, ls
4.6. Compression commands zip ,unzip, gzip, gunzip, printer related commandslpc ,lpr,lpq.lprm

UNIT III : Administration Tools

5. SystemAdministration
5.1. Performing system maintenance
5.2. Communication commands :- write, wall, talk, mesg, motd,
5.3. Pre-login Message
5.4. Managing software with RPM :- Installing, Uninstalling, Upgrading
5.5. Managing users and managing Groups and managing passwords.

UNIT IV : Network configuration and Backups

6. Backup strategies
6.1. Choosing Backup Strategies and Operations
6.2. Choosing Backup hardware and media.
6.3. Using backup software and commands

7. Network configuration for Linux
7.1. Network configuration tools
7.2. Dynamic host configuration protocol.
7.3. Network files system.
7.4. Introduction to samba
7.5. Introduction to DNS & Apache web server

References Books:
1 Bill Ball, David Pitts, “Red Hat Linux 7 Unleashed”, Techmedia SAMS Publication
2 EviNemeth, Garth Snyder, Scott Seebass, Trent R. Hein, “UNIX System Administration
3 Handbook” Person Education Asia (LPE) (IIIrd Edition)
4 Red hat Linux & fedora unlashed Authors Bill Ball & Hoyt Dust.

Course Title: SOFTWARE ENGINEERING
Course code: CC-B4

Credit:03

TOTAL TEACHING HOUR :60 **TOTAL MARKS :50**

LERANING OBJECTIVE:

The basic objective of software engineering is to develop methods and procedures for software development that can scale up for large systems and that can be used consistently to produce high-quality software at low cost and with a small cycle of time. In software engineering you develop your skills for developing new and useful software's. Main objectives are: Understanding user conceptual manual and develop better specifications. Improvement in design languages. Reusable codes. Interactive debugging. Mockup to conform specifications.

A. Be employed in industry, government, or entrepreneurial endeavors to demonstrate professional advancement through significant technical achievements and expanded leadership responsibility;

B. Demonstrate the ability to work effectively as a team member and/or leader in an ever-changing professional environment; and

C. Progress through advanced degree or certificate programs in computing, science, engineering, business, and other professionally related fields.

COURSE OUTCOME :

- An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability.
- An ability to function on multi-disciplinary teams.
- An ability to identify, formulate, and solve engineering problems.
- An understanding of professional and ethical responsibility.
- An ability to communicate effectively.

Syllabus:

UNIT I: FUNDAMENTALS

1. Software Engineering Fundamentals:

Definition of software, Software characteristics, software applications.

2. Software Process:

Software Process Models- waterfall model, prototyping model spiral model, incremental model, concurrent model.

Project Management Concepts: The management spectrum- the people, the product, the process and the project.

UNIT II: METRICS

3. Software Process and Project Metrics:

Measures, metrics and indicators, software measurement : Size-oriented Metrics , function - oriented metrics, extended function point metrics.

4. Software Project Planning:

Project planning objectives, software project estimation, and decomposition techniques- problem based estimation, process based estimation, empirical estimation models- the COCOMO Model.

UNIT III: RISK MANAGEMENT

5. Risk Analysis And Management:

Software risks, risk identification, risk projection, risk refinement, risk mitigation, monitoring and management.

6. Software Quality Assurance:

Basic concepts- quality, quality control, quality assurance, cost of quality ,software quality assurance(SQA), formal technical review.

UNIT IV: TESTINGS

7. Coding And Unit Testing:

Programming principles and guidelines, incrementally developing code, managing evolving code, unit testing ,code inspection, and metrics.

8. Testing:

Testing concepts, testing process, black-box testing, white-box testing, and metrics.

REFERENCE BOOKS:

1. R.PRESSMAN: Software Engineering- Mc Graw -Hill
2. R.K. Agrawal and Y.Sing: Software Engineering- New Age International.
3. P. Jalote : Software Project Management in practice- Pearson.

Course Code:- CC-C4
Course Title: Mathematical Foundations for Computer Science

Total Teaching Hours: 60 **Total Marks: 50**
Credits :- 3

Learning Objectives

This course aims at introducing the students into the world of Mathematical Foundations for Computer Science. It includes the topic like Sets, Relation , Functions, Mathematical Logic, Matrices and Graph theory. Also, this course emphasizes general techniques of problem solving and explores the creation of mathematical patterns.

Course Outcome

- Formulate and interpret statements presented and determine their validity by applying the rules and methods of propositional logic.
- Demonstrate a working knowledge of set notation and elementary set theory, recognize the connection between set operational and logic.
- Apply the different properties of injections, surjections , bijections and composition functions.
- Determine when a relation is reflexive, symmetric, antisymmetric or transitive, apply the properties of equivalence relations and partial ordering relation.

UNIT I: SETS, RELATIONS AND FUNCTIONS

1. SETS, RELATIONS AND FUNCTIONS

1.1. Definition and types of sets
1.2. Equal sets, subsets, universal sets, Venn diagram.
1.3. Set operations
1.4. Properties of set union and intersections
1.5. Cartesian product
1.6. Relation , types of relation
1.7. Function, domain, range, types of function
1.8. Numerical examples

UNIT II : MATHEMATICAL LOGIC

2. MATHEMATICAL LOGIC

2.1 Propositions
2.2 Truth values and truth table
2.3 Logical connectives and compound statements
2.4 Statement pattern and logical equivalence
2.5 Tautology, contradiction, contingency

UNIT III: MATRICES AND DETERMINANTS

3. MATRICES AND DETERMINANTS

3.1. Definition of Determinant
3.2. Definition and types of matrices
3.3. Equality of Matrices and transpose of matrices
3.4. Algebra of matrices : addition, subtraction of matrices, scalar
3.5. Multiplication of matrix ·
3.6. Adjoint of matrices
3.7. Inverse of matrices

UNIT IV : GRAPH THEORY AND TREE

4. GRAPH THEORY

4.1 Definition and types of graphs
4.2 Incidences and degree of vertices
4.3 Isomorphism of graphs

4.4 Connected and disconnected graphs
4.5 Walks, paths and circuits
4.6 Directed graph

5. Tree

5.1 Centre of Tree
5.2 Binary Tree
5.3 Spanning tree
5.4 Cut sets and Cut vertices – Fundamental circuits and cut sets
5.5 Edge Connectivity - Vertex connectivity

Reference books:

1) Elements of Discreet Mathematics by C.L. Liu
2) Discreet Mathematics by Olympia nicodemi
3) Mathematical Structures for Computer Science by Alon Doerr and k. Levasieur
4) A first step in graph theory by raghunathan, Nimkar & Solapurkar
5) Graphs theory with applications to computer science by Narsing Deo
6) Computer Fundamentals by P. K. Sinha
7) Basic Mathematics by Mittal and Agarwal
8) Tremblay and Manohar : "Discrete Mathematical Structures with Application to Computer Science" , McGraw Hill Book Company.

Course Code:CC-D4
Course Title:- Java Programming

Total Teaching Hours: 55 **Max. Marks: 50**
Total Credit:- 03

Learning Objectives:

- Learn the basic concepts of Object-Orientation and how they are handled in Java
- Covers techniques for better class construction
- Understand Exceptions. How and when they should be handled
- An overview of database access and details for managing information using the JDBC API
- Examines the use of Object Serialization
- Learn how to use Servlet and JSP and XML with JSP
- Be able to create and use custom JSP tags
- A presentation of Enterprise JavaBeans and how to use it

Course Outcome: After Completion of this course students are able to :

- Use the methods of the Applet and Component classes required for a basic applet
- Describe the classes in the AWT package that relate to the Applet class
- Describe the AWT graphics explain controls and how to apply them in the container
- Develop programs using Event class and Event Listener Interface
- Develop a program for steps to connect a database
- Describe the use of JDBC
- Develop program to use JDBC to query a database and modify
- Describe life cycle of servlet
- Develop program using javax.servlet package
- Explain JSP Architecture and its Life cycle

Syllabus

Unit I

1.Introduction to AWT: Working with windows, Graphics Text

1.1AWT Classes
1.2Windows Fundamentals
1.3Working with Frame window
1.4Working with Graphics
1.5Working with Colors & Fonts

2. Swing Components

2.1 Icons & Labels Button & Label, TextField & Buttons,
2.2 CheckBoxes, Radio buttons
2.3 Combo Box & Lists
2.4 Scroll panes
2.5 Trees
2.6 Tables
2.7 Menu Bars & Menus

Unit-II

3. Networking

3.1 The java.net package
3.2 Connection oriented transmission – Stream Socket Class
3.3 Creating a Socket to a remote host on a port (creating TCP client and server)
3.4 Simple Socket Program Example
3.5 Programs on chatting 1-1 .

4. JDBC

4.1 The design of JDBC

4.2 Basic JDBC Concept
4.3 Drivers
4.4 Making the Connection, Statement
4.5 Executing SQL commands
4.6 Executing queries
4.7Scrollable and updatable result sets
4.8Metadata, transactions

Unit-III

5. Servlet

5.1 Introduction
5.2 Life cycle of servlet
5.3 Handling HTTP Get Request
5.4 Handling HTTP Post Request

6. Introduction to JSP

6.1 Getting Familiar with JSP Server
6.2 First JSP
6.3 Adding Dynamic contents via expressions
6.4 Scriptlets, Mixing Scriptlets and HTML
6.5 Directives, Declaration, Tags and Session

Unit-IV

7. Introduction to Java Beans & Hibernate

7.1 What is bean
7.2 Advantages
7.3 The bean-writing process
7.4 Introduction to jar and manifest files
7.5 The java beans API
7.6 Overview Of hibernate
7.7 Hibernate Architecture

Reference Books:

1) Complete reference Java by Herbert Schildt(5th edition)
2) Java 2 programming black books, Steven Horlzner
3) Programming with Java , A primer ,Forth edition , By E. Balagurusamy
4) Java servlet Programming by Jason Hunter, O'Reilly
5) Core Java Volume-II-Fundamentals, Eighth Edition, Cay S. Horstmann, Gary Cornell, Prentice Hall, Sun Microsystems Press.

Class B.C.A. S.Y. (IV Sem)
Course Title:- Mobile Application Development Using Android
Course Code:- SEC-II

Total Credit :-02 **Marks: 50**

Learning Objective :-Learn Designing of android application, writing java code, joining xml with java, testing application on real mobile device or virtual device, database connectivity etc.
Course Outcome:- Student should design xml controls, join xml controls with java object, run app on real mobile device, create services, works with different java android classes like LocationManager, SensorManager, SQLiteOpenHelper etc.
Students can get job of Android developer or xml-android app UI designer.

SYLLABUS

Unit-I :- Android Basic

Chp1. Android Basic
- 1.1 What is Android?
- 1.2 Activity Life Cycle
- 1.3 Call Back Methods
- 1.4 Logcat usage

Chp2. Android Application Structure
- 2.1 Android Application Structure
- 2.2 AndroidManifest.xml, <Uses-permission>, < uses-sdk>
- 2.3 First Sample Android Application
- 2.4 Activity Registration
- 2.5 Activity & Intent

Unit-II :- Android Widgets

Chp3.UI Widget – I
- 3.1 LinearLayout, RelativeLayout
- 3.2 Button, EditText,TextView
- 3.3 Event Handling
- 3.4 RadioButton , CheckBox
- 3.5 ImageView

Chp 4.UI Widget – II
- 4.1 SeekBar, ProgressBar
- 4.2 Switch
- 4.3 Analog Clock , Digital Clock
- 4.4 ListView
- 4.5 WebView

Unit-III :- Android Menus &Database Connectivity

Chp 5.Menus & Notifications
- 5.1 Alert Dialog
- 5.2 Prompt Dialog
- 5.3 Android Menus
- 5.4 Toast Notification
- 5.5 Custom Toast

Chp6. Services & SQLite
- 6.1 Android Services
- 6.2 SQL Commands
- 6.3 SQLiteOpenHelper Class

6.4 SQLite Based Application

Unit- IV :- Telephony & Mini Projects

Chp7. Telephony & Sensor API

7.1 TelephonyManager
7.2 PhoneCall
7.3 Send SMS
7.4 SensorManager

Chp8. Mini Projects

8.1 Torch App
8.2 Media Player
8.3 Voice To Text Conversion

Reference Books:-

3. Learning Android , OREILLY By:- Marko Gargenta
4. Android Black Book

U-LAC-CC-A4

Practical Total Marks: 50 **CREDIT: [02]**

Learning Objective:

Teach Basics of Linux Operating System

- Teach ownership and permissions of the files and directories.
- Explain why these issues exist.
- Learn to set permissions files/directories
- Learn to manipulate files/directories – list files, create, delete, and move just about anything on the file system
- Mention vi - a standard Unix text editor
- Learn How to create users , groups and setting the passwords
- Understand Different network configuartions
- Know about Storage and backups

Course Outcome:

On completion of this course the student should be able to:
Identify and use UNIX/Linux utilities to create and manage simple file processing operations, organize directory structures with appropriate security, and develop shell scripts to perform more complex tasks.

Proposed Practical List:

1. Study of installation of linuxo.s.
2. Study of file related commands
3. Study of utility commands
4. Study of directory related commands
5. Study of compression related commands
6. Study of filter commands
7. Study of communication commands
8. Study of network related commands
9. Study of printer related commands
10. Study of vi editor.
11. Write a simple shell program

U-LAC-CC-B4

Practical Total Marks: 50 **CREDIT: [02]**

LERANING OBJECTIVE:

The basic objective of software engineering is to develop methods and procedures for software development that can scale up for large systems and that can be used consistently to produce high-quality software at low cost and with a small cycle of time. In software engineering you develop your skills for developing new and useful software's. Main objectives are: Understanding user conceptual manual and develop better specifications. Improvement in design languages. Reusable codes. Interactive debugging. Mockup to conform specifications.

A. Be employed in industry, government, or entrepreneurial endeavors to demonstrate professional advancement through significant technical achievements and expanded leadership responsibility;

B. Demonstrate the ability to work effectively as a team member and/or leader in an ever-changing professional environment; and

C. Progress through advanced degree or certificate programs in computing, science, engineering, business, and other professionally related fields.

COURSE OUTCOME :

- An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability.
- An ability to function on multi-disciplinary teams.
- An ability to identify, formulate, and solve engineering problems.
- An understanding of professional and ethical responsibility.
- An ability to communicate effectively.

Proposed Practical List:

1. Perform the practical on Requirement gathering.
2. Perform the practical on Requirement Specification.
3. Perform the practical on Requirement Analysis.
4. Practical on designing the software.
5. Creating the Login form.
6. Testing the all forms.
7. Practical on designing the Tables.
8. Testing the all the tables.
9. Perform the practical on Database connectivity.
10. Perform the practical on Testing's
11. Perform the practical on Error testing.

U-LAC-CC-C4

Practical Total Marks: 50 **CREDIT: [02]**

Learning Objectives

This course aims at introducing the students into the world of Mathematical Foundations for Computer Science. It includes the topic like Sets, Relation , Functions, Mathematical Logic, Matrices and Graph theory. Also, this course emphasizes general techniques of problem solving and explores the creation of mathematical patterns.

Course Outcome

- Formulate and interpret statements presented and determine their validity by applying the rules and methods of propositional logic.
- Demonstrate a working knowledge of set notation and elementary set theory, recognize the connection between set operational and logic.
- Apply the different properties of injections, surjections , bijections and composition functions.
- Determine when a relation is reflexive, symmetric, antisymmetric or transitive, apply the properties of equivalence relations and partial ordering relation.

Proposed Practical List:

1. Set and types of sets
2. Relation and types of relations
3. Function and types of functions.
4. Logical connectives and truth tables
5. Creating matrix by using Matlab
6. Types of matrices and algebra of matrices by using Matlab
7. Inverse and adjoint matrix by using Matlab
8. Graph theory
9. Types of graph theory
10. Tree

Java Programming
U-LAC-CC-D4

Practical Total Marks: 50 **CREDIT: [PR-2]**

Learning Objectives:

- Learn the basic concepts of Object-Orientation and how they are handled in Java
- Covers techniques for better class construction
- Understand Exceptions. How and when they should be handled
- An overview of database access and details for managing information using the JDBC API
- Examines the use of Object Serialization
- Learn how to use Servlet and JSP
- Be able to create and use custom JSP tags
- A presentation of Enterprise JavaBeans and how to use it

Course Outcome: After Completion of this course students are able to :

- Write programs based upon java concepts.
- Create animation & events based upon advanced java concepts.
- Connect an application with database.
- Develop programs using java collection API as well as java Standard Library.
- Write, debug & document well structured java application

Proposed Practical List:

1. Write a program to demonstrate life cycle of applet and display Hello world on applet UI.
2. Write a program to create a frame window.
3. Write a program to make use various methods of graphics class object.
4. Write a program to display buttons, labels, and Image icons using swing.
5. Write a program to display check boxes and radio buttons using swing.
6. Write a program to display combo box and scroll pane using swing.
7. Write a program to display a tree using swing.
8. Write a program to display a table using swing.
9. Write a program to display Menu bar and menus using swing.
10. Write a program to demonstrate socket programming. E.g. send hello world to server from client.
11. Write a program to chat between client and server.
12. Write a program to connect to db and to execute the queries.
13. Write a Servlet code to demonstrate GET & POST methods.
14. Write a program to demonstrate JSP use.