Rajarshi Shahu Mahavidyalaya, Latur
(Autonomous)
Department of Computer Sci. and Information Technology

Syllabus of
B.Sc.C.S.-II Year (Semester III and IV)
(Under CBCS)

w.e.f. 2018-2019

Syllabus outline of B.Sc.C.S. Second Year
Under CBCS Pattern
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)**							
	BScCS-AEC-3	English- I	--	20	--	30	2	50
	2. Core Course (CC) (Compulsory Course)							
	BScCS -CC-A3	OOP using C++	--	20	--	30	3	50
	BScCS -CC-B3	Core Java	--	20	--	30	3	50
	BScCS -CC-C3	Computer Network	--	20	--	30	3	50
	BScCS -CC-D3	Operating System	--	20	--	30	3	50
	3. Skill Enhancement Course							
	BScCS -SEC-1	Android O.S.	--	20	--	30	2	50
	4. Practical / Lab Course							
	BScCS -Lab-9	Lab-Course 1 (C++)	--	20	--	30	2	50
	BScCS -Lab-10	Lab-Course 2 (Java)	--	20	--	30	2	50
	BScCS -Lab-11	Lab-Course 3 (CN)	--	20	--	30	2	50
	BScCS -Lab-12	Lab-Course 4 (OS)	--	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)**							
	BScCS -AEC-4	English- II	--	20	--	30	2	50
	2. Core Course (CC) (Compulsory Course)							
	BScCS -CC-A4	Software Engineering.	--	20	--	30	3	50
	BScCS -CC-B4	Advance Java	--	20	--	30	3	50
	BScCS -CC-C4	C#.Net	--	20	--	30	3	50
	BScCS -CC-D4	Multimedia Systems	--	20	--	30	3	50
	3. Skill Enhancement Course							
	BScCS -SEC-2	1)Mobile Application Development OR 2) Hardware Networking	--	20	--	30	2	50
	4. Practical / Lab Course							
	BScCS -Lab-13	Lab-Course 5 (SE)	--	20	--	30	2	50
	BScCS -Lab-14	Lab-Course 6 (Java)	--	20	--	30	2	50
	BScCS -Lab-15	Lab Course 7 (C#.Net)	--	20	--	30	2	50
	BScCS -Lab-16	Lab-Course8 (Multimedia)	--	20	--	30	2	50
		Total (IV)					**24**	**500**
		Total (III + IV)					**48**	**1000**

Semester – III

Course Title: Object Oriented Programming using C++
Course Code: CC-A3

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

Learning Objectives

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, file I/O, exception handling, 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 .

2.2 Ultimate goal: to make you a good programmer.

Course Outcomes

- Gain the basic knowledge on Object Oriented concepts.
- Ability to develop applications using Object Oriented Programming Concepts.
- Ability to implement features of object oriented programming to solve real world problems.
- Use the characteristics of an object-oriented programming language in a program.
- Use the basic object-oriented design principles in computer problem solving.
- Use the basic principles of software engineering in managing complex software project.
- Program with advanced features of the C++ programming language.

Syllabus

UNIT- I: Introduction to Object Oriented Programming

1. Principles of Object Oriented Programming (OOP)

Evolution of C++ - Programming Paradigms - Key Concepts of OOP - Advantages of OOP - Usage of OOP and C++ .Input and Output in C++- Streams-Stream classes Unformatted console I/O operations-Member functions of istream class-manipulators-manipulators with parameters

2. Introduction to C++

Tokens, Keywords, Identifiers, Variables, Operators, Expressions and Control Structures: If,If. Else, Switch – Repetitive Statements- for, while, do..while - Pointers and arrays

UNIT II: Class, Functions and Constructors

3. Structures and Unions:

Declaration of structures, Accessing structure members, Structure Initialization, Arrays of structure, nested structures, structure with pointers, functions & structures, Unions, Structure/Union Versus Class in C++.

4. Class Declaration
Data Members, Member Functions, Private and Public Members, Data Hiding and Encapsulation, Array within a class.

5. Class Function Definition
Member Function definition inside the class and outside the class, Friend Function, Inline Function, Static Members & Functions, Scope Resolution Operator, Private and Public Member Functions, Nesting of Member Functions. Creating Objects, Accessing class data members, Accessing member functions, Arrays of Objects, Objects as function arguments: Pass by value, Pass by reference, Pointers to Objects.

6. Constructors and Destructors
Declaration and Definition, Default Constructors, Parameterized Constructors, Constructor Overloading, Copy Constructors. Destructors: Definition and use.

UNIT III: Inheritance and Overloading

7. Inheritance
Extending Classes, Concept of inheritance, Base class, Derived class, Defining derived classes, Visibility modes : Private, public, protected; Single inheritance : Privately derived, Publicly derived; Making a protected member inheritable, Access Control to private and protected members by member functions of a derived class, Multilevel inheritance, Nesting of classes.

8. Function Overloading & Operator Overloading
Binary & Unary

UNIT IV: Polymorphism and file operations

9. Polymorphism
Definition, early Binding, Polymorphism with pointers, Virtual Functions, late binding, pure virtual functions.

10. Working with files
Header file, redirection, Classes for File Stream Operations - Opening and Closing a File - End-of-File Detection - file input and output. File Pointers - Updating a File - Error Handling during File Operations - Command-line Arguments, buffers & iostreams

Reference Books
1. “Object Oriented Programming with C++”, E. Balagurusami, Fourth Edition, Tata Mc-Graw Hill
2. “Object Oriented Programming in Turbo C++”, Robert Lafore, Fourth Edition Galgotia Publications.
3. “The C++ Programming Language”, Bjarne Stroustrup, Third Edition, Addison-Wesley Publishing Company.
4. “Object Oriented Programming Using C++”, Salaria, R. S, Fourth Edition, Khanna Book Publishing
5. “Object Oriented Programming with ANSI & Turbo C++”, Ashok N. Kamthane, Pearson Education, 2006

Course Title: Core Java
Course Code: CC-B3

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

Learning Objective:
Syllabus helps to learn basic knowledge of Core Java. Develop dynamic web applications. Create final year project. Students can get job of Java developer as well as android application developer.

Course outcomes:
- Implement Object Oriented Programming Concepts.
- Use and create packages and interfaces in a Java program.
- Create final year project with database connectivity.

SYLLABUS

Unit I: Introduction of Java Programming Basics

1. An Introduction to Java:
A Short History of Java, Features of Java, Comparison of Java and C++, Java virtual machine, Java program structure, Creating and Running Java Programs, Command Line Arguments

2. Programming Construct : Decision making statement, switch statement, looping statement

Unit II: Object Oriented concepts in java programming

3. Classes and Objects :
Introduction, Defining a class, Adding variables, Adding Methods, Creating Objects, Accessing Class Members, Constructors. Method Overloading, Static Members

4. Inheritance:
Extending a class, Overriding Method, using super, Final variable and Methods, this keyword

Unit III: Arrays, Packages and Interface

5. Arrays, Strings:
Introduction, One-dimensional and Two-dimensional Arrays, String Arrays, String Method.

6. Packages and Interface:
Java API package, Using system packages, Creating Packages & Using a Package, Interface Introduction, creating and using interfaces

Unit IV: Multi Threading and applet

7. Multithreaded Programming:
Introduction, Life Cycle of a Thread, Creating Threads, Extending the Thread Class, Stopping and Blocking a Thread, Thread Priorities

8. Applets:
Life cycle of Applet, Creation and Execution of Java Applets, Applet tag, Parameter Passing to applet

Reference Books

1) "Complete Reference Java" by Herbert Schildt(5th edition)
2) Programming with Java , A primer ,Forth edition , By E. Balagurusamy

Websites links • http://tutorialpoint.com • https://www.w3schools.in/java-tutorial

Course Title: Computer Network
Course Code: CC-C3

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

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 12

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, 13 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. “Computer Networking: A Top-Down Approach”, James F. Kurose & Keith W. Ross, , 6th edition , Pearson / Addison Wesley 2013 .
2. “Computer Networking”, Andrew S. Tanenbaum by PHI
3.”Data and Computer Communications”, William Stalling

Course Title:- Operating System
Course Code: CC-D3

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

Learning Objectives:
The student will enable to
1. Understand the basic concepts of operating system, its functions and services.
2. Familiarize the various management policies adopted by O.S. as pertaining with processes, Deadlock , memory , File and I/O operations.
3. Understand the knowledge of basic concepts towards process synchronization and related issues.

Course Outcomes:
After the completion of this course student will be able to:
1. Understand functions, structures and history of operating systems
2. Understand process management concepts including scheduling, synchronization, deadlocks
3. Understand and implement multithreading concept
4. Analyze concepts of memory management including virtual memory
5. Design the protection and security mechanisms

SYLLABUS

UNIT –I

1. Introduction to Operating System

1.1 Definition of Operating System

1.2 Functions of Operating System

1.3 Types of Operating System

1.4 Operating System as resource manager

1.5 Hierarchical structure of Operating System

UNIT -II

2. Memory Management

2.1 Single contiguous allocation

2.2 Partitioned allocation

2.3 Paged memory management

2.4 Introduction to demand paged & segmented memory management

UNIT -III

3. Process Management

3.1 What is process?

3.2 Process Control Block

3.3 Process states

3.4 Job Scheduling & Process Scheduling

3.5 Process Synchronization

3.6 Race Condition

3.7 Introduction to Deadlocks

UNIT -IV

4. Device Management

4.1 Techniques of Device Management

4.2 Dedicated, Shared, Virtual Devices

4.3 Device Characteristics

4.4 Channels & Control Units

5. File Systems

5.1 A Simple file system

5.2 General Model of file system

5.3 Symbolic file system

Reference Books

1. “Operating System” by Stuart .E. Madnick & John. J. Donovon

2. “Operating System Concepts” by Abraham Silberschatz, Peter B. Galvin, Greg Gagne

3. “Operating Systems” by Achyut S. Godbole

4. “Operating System” by Milan Milenkovic (IBM CORPORATION)

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

Total Teaching Hours: 50 **Total Marks: 50**
Credit: 2

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

Chapter 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

Chapter 2. Android Architecture

2.1 Android Architecture

2.2 Linux Kernel

2.3 Dalvik Virtual Machine

UNIT –II:-IDE's and Java Basic

Chapter 3. Android IDE's & Components

3.1 Various IDE for Android

3.2 Installation of Android

3.3 Android Virtual Device

3.4 Android Components

Chapter 4. Introduction To Java

4.1 Introduction & History of Java

4.2 Java Applications

4.3 Java Architecture

Unit – III:- Java Programming

Chapter 5. Programming Basics

5.1 Variable, Constants

5.2 Hello World Program

5.3 Classes&Inheritance

5.4 Interface

Unit –IV:- Packages & UI Designing

Chapter 6. Java Packages & Threads

6.1 Packages

6.2 Thread

6.3 Exception Handling

6.4 Method Overloading

Chapter 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

Course Title: Lab-Course1 (C++)

Course code: U-LAC-CC-A3

Total Marks: 50 **Credit:2**

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, file I/O, exception handling, 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 .

2.2 Ultimate goal: to make you a good programmer.

Course Outcomes:

- Gain the basic knowledge on Object Oriented concepts.
- Ability to develop applications using Object Oriented Programming Concepts.
- Ability to implement features of object oriented programming to solve real world problems.
- Use the characteristics of an object-oriented programming language in a program.
- Use the basic principles of software engineering in managing complex software project.
- Program with advanced features of the C++ programming language.

Proposed Practical List:

1. Program to demonstrate encapsulation using of class.
2. Program to demonstrate use of array of objects
3. Program to demonstrate use of pointers
4. Program to demonstrate use of pointer to members of 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 static data members & member functions of class.
9. Program to demonstrate use of different manipulators
10. Program to demonstrate use of constructor, constructor overloading & destructor
11. Program to demonstrate use of all types of Inheritance.
12. Program to demonstrate use of unary & binary operator overloading
13. Program to demonstrate use of polymorphism (virtual function)
14. Program for reading and writing operations on text file.

Course Title: Lab-Course2 (Core Java)
Course code: U-LAC-CC-B3

Total Marks: 50 **Credit:2**

Learning Objective:
Syllabus helps to learn basic knowledge of Core Java. Develop dynamic web applications. Create final year project. Students can get job of Java developer as well as android application developer in IT industries.

Course outcomes:

- Implement Object Oriented Programming Concepts.
- Use and create packages and interfaces in a Java program.
- Create final year project with database connectivity.

Proposed practical list

1. Program to demonstrate simple java program

2. Program to demonstrate class and object

3. Program to demonstrate Static member function

4. Program to demonstrate method overloading

5. Program to demonstrate this and super keyword

6. Program to demonstrate method overriding.

7. Program to demonstrate use of String functions

8. Program to demonstrate creating and using packages

9. Program to demonstrate simple applet program

10. Program to demonstrate reading data from keyboard

11. Program to demonstrate applet

12. Program to demonstrate simple multithreading application

13. Program to demonstrate Inheritance using interface

Course Title: Lab-Course3 (Computer Network)
Course code: U-LAC-CC-C3

Total Marks: 50 **Credit:2**

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 various network devices
2. Study of basic network and network configuration commands.
3. Study of installation of LAN card and LAN cabling.
4. Test TCP/IP Setup and create user accounts and user group.
5. Study of network IP6.
6. Study about installation of network and file sharing.
7. Study about installation and configuration of printers.
8. Connect the computers in Local Area Network.
9. Interfacing with the network card (Ethernet).

Course Title: Lab-Course 4 (Operating System)
Course code: U-LAC-CC-D3

Total Marks: 50 **Credit:2**

Learning Objectives:

The student will enable to

1. Understand the basic concepts of operating system, its functions and services.
2. Study the various management policies adopted by O.S. as pertaining with processes, Deadlock , memory , File and I/O operations.
3. Understand the knowledge of basic concepts towards process synchronization and related issues.

Course Outcomes:

After the completion of this course student will be able to:

1. Understand functions, structures and history of operating systems
2. Familiarize of process management concepts including scheduling, synchronization, deadlocks
3. Understand and implement multithreading concept
4. Analyze concepts of memory management including virtual memory
5. Design the protection and security mechanisms

Proposed Practical List:

1. Comparative study of various operating systems
2. Study of DOS internal commands
3. Study of DOS external commands
4. Installation of Windows 7 O.S.
5. Working with Windows Desktop and utilities
6. Installation of Ubuntu Linux O.S.
7. Working with Linux Desktop and utilities
8. Study of Libre Office (Writer, Calc, Impress) in Linux
9. Study of Vi editor
10. Introduction to shell programming
11. Programs on process scheduling algorithms

Semester-IV

Course Title: Software Engineering
Course code: CC-A4

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

Learning 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.
- Be employed in industry, government, or entrepreneurial endeavors to demonstrate professional advancement through significant technical achievements and expanded leadership responsibility;
- Demonstrate the ability to work effectively as a team member and/or leader in an ever-changing professional environment.
- 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: ROLE OF SOFTWARE

1. INTRODUCTION TO SOFTWARE ENGINEERING:

The evolving role of software, changing nature of software, software myths. The software problem: Cost, Schedule and quality, scale and change.

2. SOFTWARE PROCESS:

Process and project, component software process.

Software development process models- waterfall model, prototyping , iterative development , relational unified process, time boxing model, Extreme programming and agile process, using process models in a project, project management process.

UNIT II: REQUIREMENT ANALYSIS

3. SOFTWARE REQUIREMENT ANALYSIS AND SPECIFICATION:

Value of good SRS, requirement process, requirement specification, functional specifications with use-cases, other approaches for analysis,validation.

4. PLANNING A SOFTWARE PROJECT:

Effort estimation, project schedule and staffing, quality planning, risk management plans, project monitoring plan, detailed scheduling.

UNIT III: ARCHITECTURE

5. SOFTWARE ARCHITECTURE:

Role of software architecture, architecture view, components and connector view, Architecture styles for C and C view , documenting architecture design, evaluating architectures.

6. DESIGN:

Design concepts, function-oriented design, object oriented design, detailed design, verification, and metrics.

UNIT IV: TESTINGS

7. CODING AND UNIT TESTING:

Programming principles and guidelines, incrementally developing code, managing evolving code, unit testing ,code inspection and metrics.
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 Title:- Advance Java

Course Code:CC-B4

Total Teaching Hours: 50 **Total Marks: 50**

Credit: 3

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.

Course Title: C#.Net
Course code: CC-C3

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

Learning Objectives:

- To understand the DOTNET framework and C# language features
- To develop object oriented programs on C#.
- To develop windows based applications on .NET framework.

Course Outcome:

After the successful completion of the course students will be able to:

- Describe basic concepts and develop programs in C# using object oriented features like delegates, events, errors and exceptions
- Explain Common language runtime (CLR) as a platform for managed code
- Describe the features of Common language runtime (CLR) and develop efficient code with C# on .NET framework
- Develop windows based applications & services on .NET framework
- Describe overview of .NET framework
- Interpret data access and develop windows applications
- Apply an understanding of the .NET technology and C#.net components to develop a windows based application which solves specified problem domain
- Use of ADO.NET technology for developing database oriented applications
- Understand the professional responsibility
- Apply an understanding of the need for high ethical standards in the practice of engineering towards people and the environment

---

SYLLABUS

UNIT I: Introduction to .net, Arrays and operators

What is .Net?,.Net Framework, CLR, Visual Studio.Net & .Net Languages, Integrated Development Environment, Project types, C#.Net History & design Goals, How c# differs from c++, Characterstics of c#.net, I/O Statement with C#.net, Boxing & Unboxing, Short Circuiting Operators, Array & ArrayList class, Jagged Array, String Class

UNIT II: Properties, Events, Delegates and C# namespaces

Properties & its type, Event, Delegate & Multicast Delegate, Creating & Starting thread, Exception handling, using keyword, creating and using namespaces, interface, Method overloading & method overriding, Partial Class

UNIT III: Windows Application

Event Driven Programming, Building windows application with visual studio, TextBox, Label & Button Control, ComboBox, ListBox ,CheckBox & GroupBoxControl, DateTimePicker , Timer control,Building Menu,. MDI Form, PictureBox , ProgressBarControl, Common Dialog boxes, Introduction to WPF

UNIT IV: Ado.Net and Database Oriented Applications

How Ado.net differs from Ado, Advantages of Ado.net, Connected & Disconnected Architecture, Dataset, DataReader & DataAdapter, Managed Data Providers, DataGridViewControl, Developing Ado.net Based Application, Insert, Update & Delete operation on table, Filling the Dataset

Reference books:

1. Programming in C# A Primer - Second Edition By - E Balagurusamy
2. C#.Net Programming Wrox Publication
3. .Net 4.0 programming black book by KOGENT LEARNINGSOLUTIONS INC.
4. C# 2010 programming black book by KOGENT LEARNINGSOLUTIONS INC.

Course Title: Multimedia Systems
Course Code: CC-D4

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

LEARNING OBJECTIVES:

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

COURSE OUTCOMES:

- Develop an understanding of the tools used for creating two dimensional (2D) Graphics and animation.
- Design 2D Graphics, 2D character modeling and animation.
- 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

Course Title:- Mobile Application Development Using Android
Course Code:- SEC-II

Total Teaching Hours: 50 **Total Marks :-50**
Credit : 2

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 will be able to 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

Course Title: Lab-Course 5 (Software Engineering)
Course code: U-LAC-CC-A4

Total Marks: 50 **Credit:2**

Learning 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.
- Be employed in industry, government, or entrepreneurial endeavors to demonstrate professional advancement through significant technical achievements and expanded leadership responsibility;
- Demonstrate the ability to work effectively as a team member and/or leader in an ever-changing professional environment.
- 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. Study of UML diagrams
4. Practical on designing the software.
5. Creating the Login form.
6. Testing of all forms.
7. Practical on designing the Tables.
8. Testing all the tables.
9. Perform the practical on Database connectivity.
10. Perform the practical on Testing's

Course Title: Lab-Course 6 (Advance Java)
Course code: U-LAC-CC-B4

Total Marks: 50 **Credit: 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 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 :

- 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.

Course Title: Lab-Course 7 (C#.Net)
Course code: U-LAC-CC-C4

Total Marks: 50 **Credit: 2**

Learning Objectives:

- To understand the DOTNET framework and C# language features
- To develop object oriented programs on C#.
- To develop windows based applications on .NET framework.

Course Outcome:

After the successful completion of the course students will be able to:

- Describe basic concepts and develop programs in C# using object oriented features, delegates, events, errors and exceptions
- Interpret data access and develop windows applications
- Explain Common language runtime (CLR) as a platform for managed code
- Describe the features of Common language runtime (CLR) and develop efficient code with C# on .NET framework
- Develop windows based applications & services on .NET framework
- Describe overview of .NET framework
- Apply an understanding of the .NET technology and C#.net components to develop a windows based application which solves specified problem domain
- Use of ADO.NET technology for developing database oriented applications
- Understand the professional responsibility
- Apply an understanding of the need for high ethical standards in the practice of engineering towards people and the environment

Proposed Practical List:

1. Program to demonstrate jagged array
2. Program to demonstrate String class and its methods
3. Program to demonstrate properties concept
4. Program to demonstrate delegate concept
5. Program to demonstrate creation of C# namespaces
6. Program to demonstrate interface concept
7. Program to demonstrate common windows controls
8. Program to demonstrate advance windows controls
9. Program to demonstrate simple database connectivity application
10. Program to perform insert, update and delete operation on database

Course Title: Lab-Course 8 (Multimedia)
Course code: U-LAC-CC-D4

Total Marks: 50 **Credit: 2**

LEARNING OBJECTIVES:

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

COURSE OUTCOMES:

- Develop an understanding of the tools used for creating two dimensional (2D) Graphics and animation.
- Design 2D Graphics, 2D character modeling and animation.
- 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.