

## M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III

### PGCS-301- Artificial Intelligence

#### Course outcomes:

At the end of this course, a student shall be able to:

- apply the basic principles, models, and algorithms of AI to recognize, model
- demonstrate knowledge of the building blocks of AI as presented in terms of intelligent agents.
- solve problems in the analysis and design of information systems
- analyze the structures and algorithms of a selection of techniques related to searching, reasoning, machine learning, and language processing.

**Total Credits: 04**

**Total lectures: 60**

#### Course content

##### **1. Artificial Intelligence: (5L)**

An overview, Intelligent Systems: Evolution of the concept.

##### **2. Intelligent Agents : (5L)**

how agent should act, Structure of intelligent agents, Environments

##### **3. Problem solving (5L)**

solving problems by searching, informed search methods, Game playing

##### **4. Knowledge and Reasoning (5L)**

A knowledge based agent, Representation, Reasoning Logic, Proportional logic, First order logic: Syntax and Semantics, Extensions and Notational variation ,Using first order logic

##### **5. Building a Knowledge Base (5L)**

Properties of good and bad Knowledge base, Knowledge Engineering, general ontology

##### **6. Interfacing first Order Logic: (5L)**

Interface rules involving quantifiers, An example proof, Forward and backward chaining, Completeness

##### **7. Acting logically (5L)**

Planning, Practical planning: Practical Planners, Hierarchical decomposition, Conditional planning

##### **8. Uncertain Knowledge and Reasoning: (5L)**

Uncertainty, Representing knowledge of uncertain domain, the semantics of belief networks, Inference in belief networks

##### **9. Learning: (5L)**

Learning from observations: General model of learning agents, Inductive learning, learning decision tree, Learning in neural and belief networks: introduction of neural networks, Perceptrons, Multilayer feed forward networks, Application of ANN, Reinforcement learning: Passive learning in a known environment, Generalization in reinforcement learning, Genetic algorithms

##### **10. Agents that Communicate: (5L)**

Communication as action, Types of communicating agents, A formal grammar for a subset of English

##### **11. Expert system: (5L)**

Introduction of expert system, Representing and using domain knowledge, Expert system shells, Explanation, Knowledge acquisition

**12.Application:****(5L)**

Natural language processing, perception, Robotics

**References Books:**

1. Stuart Russeland Peter Norvig, .Artificial intelligence: A modern Approach.
2. George F. Luger, . Artificial intelligence: Structures and Strategies for Complex Problem Solving., Person Education
3. Nils J. Nillson, .Artificial Intelligence : A New Synthesis., Harcourt Asia page 36 of 38
4. Elaine Rich and Kevin Knight,. Artificial Intelligence.,TMH
5. Patrick Winston,. Artificial Intelligence., Pearson Education
6. Ivan Brakto,.Prolog Programming for Artificial Intelligence.,Pearson Education
7. Efraim Turban Jay E.Aronson, .Decision Support System and Intelligent Systems.
8. Ed. M. Sasikumar and Others, . Artificial Intelligence :Theory and Practice.

\* \* \* \* \*

## M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III

### PGCS-302- Mobile Technologies

#### Course outcomes:

At the end of this course, a student shall be able to:

- get familiarize with the buzz words and technology of mobile communication
- understand SMS, MMS and GSM architecture
- apply mobile IP and mobile TCP
- understand the issues relating to wireless applications

**Total Credits: 04**

**Total lectures: 60**

#### Course content

##### **1. Introduction to Mobile Computing: (4L)**

Introduction and need for Mobile computing, Mobility and portability, Mobile and Wireless devices, Applications, Brief History of wireless communication

##### **2 Wireless Transmission: (6L)**

General Concepts of multiplexing and modulation, Spread Spectrum, Cellular Systems, Cellular Phone Array, Mobile Phone Technologies (1G, 2G, 2.5G, 3G)

##### **3. Medium Access Control Layer: (6L)**

Why specialized MAC?, hidden and exposed terminals, near and far terminals, General Concepts and comparison of SDMA, FDMA, TDMA, CDMA

##### **4. Global System for Mobile Communication: (10L)**

Mobile Services (Bearer, Tele-and-supplementary services), System Architecture, Radio subsystem, Network and switching subsystem, Operation subsystem, Protocols, Localization and calling, Handover, Value Added Services, SMS, Architecture, Mobile Originated and Mobile Terminated, procedures, Cell Broadcast Service, Architecture, Message Transfer Procedure, MMS, Architecture, Protocol framework, Message Transfer, Procedure, Location Services, Logical Reference Model, Control procedures, Network Architecture, determination of Location Information, Location based services, GPRS

##### **5. Mobile IP : (10L)**

Goals, assumptions and requirements, Entities and terminologies, Agent Discovery, Registration, Tunneling and encapsulation, Reverse Tunneling, IPv6, IP micro-mobility support – Cellular IP, Hawaii, Hierarchical, mobile IPv6, Mobile Routing Destination sequence distance Vector, Dynamic Source Routing, Alternative Matrix, Adhoc Routing Protocols -Flat, Hierarchical, Geographic-position-assisted

##### **6 Mobile TCP: (10L)**

Traditional TCP, Congestion Control, Slow start, Fast retransmit / Fast recovery, Implications on mobility, Classical TCP improvements, Indirect TCP, Snooping TCP, Mobile TCP, Fast retransmit / Fast recovery, Transmission / Timeout freezing, Selective Retransmission, Transaction oriented TCP, TCP over 2.5/3G wireless networks

##### **7. Wireless Application Protocol : (4L)**

Architecture, Wireless datagram protocol, Wireless transport layer security, Wireless transaction protocol, Wireless session protocol, Wireless application environment, WML, WML Scripts, Push Architecture, Push – Pull Services

**8. Java for Wireless Devices:** (2L)  
Setting up the development environment, Basic Data types, Libraries (CLDC, MIDP)

**9. UI Controls:** (2L)  
Displayable and Display Image, Events and Event Handling, List and choice, Text box, Alerts

**10. Persistent Storage:** (2L)  
Record Stores, Records, Record Enumeration

**11. Network MIDlets:** (2L)  
The Connection Framework, Connection Interface, Making a connection using HTTP, Using datagram connection

**12. Wireless Messaging:** (2L)  
Architecture for Messaging application, Messaging API, Types of applications, Pro's and con's of messaging

**Reference Books:**

1. Mobile Communication By Jochen Schiller
2. Pervasive Computing Technology And Architecture Of Mobile Internet Applications by JOCHEN BURKHARDT and HORST HENN, PEARSON INDIA
3. Mobile Computing by Rajkamal, Oxford University Press
4. Mobile Computing by TALUKDER and ASOKE K, McGraw Hill

\* \* \* \* \*

**M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III**

**PGCS-303: .Net Technologies**

**Course outcomes:**

At the end of this course, a student shall be able to:

- understand the development and deployment cycles of enterprise applications.
- develop ASP.NET Web Services, secure web services, and .NET remote applications.
- Develop web applications using a combination of client-side and server side technologies
- experiment with the Windows Form.

**Total Credits: 04**

**Total lectures: 60**

**Course content**

**1. Introduction to .Net Technology: (6L)**

Framework common Language Runtime, Common Language Specification, Intermediate Language Code, Just-In-Time Compiler, Assemblies, Manifest, Metadata ,Global assembly Cache, VB.NET, Creating Applications, Building, Running Projects, Data , Operator, Loops, Procedures, Functions.

**2. Windows Forms: (6L)**

Text Box, Buttons, Labels, Checks Boxes, radio Buttons, List Boxes, Combo Boxes, Picture Boxes, Scrollbars, Timer, Menus, Built-in Dialogs, Image List, Toolbars, Status Bar and Progress bars, Event and Delegates, Tracing, Debugging

**3. Object Oriented Programming in VB.NET: (6L)**

Class and Objects Properties, methods and events, Contractor and Destructor, Method overloading, Inheritance, Access modifiers, Public, Private, Protected, Friend, Overriding and shadowing, Interfaces, Polymorphism, Error Handling, Private and Shared Classes

**4. File Handling: (6L)**

Stream Writer, Stream Reader, MSDN, Binary Reader, Binary Writer Classes MSDN, File and Directory Classes

**5. ADO.Net Connections: (6L)**

Data adapters, and Datasets ,Data binding with controls, Navigating data source ,Data from wizard, Data reader, Connection objects, command Objects, Data Adapters, Dataset Class, Data Readers, Non-Queries, Query-Parameters, Scalar-Queries, Calling stored-Procedures, Data-Relations, Editing rows ,Filtering and sorting, XML (Reading, Writing)

**6. Web application: (8L)**

Introduction to Web form, page directive, all validation controls, Page redirection

**7. State management: (2L)**

Application state, Session State, View state, Web controls, Tracing web applications

**8. Database: (6L)**

Data grid control in web applications

**9. Web services: (4L)**

Concept of web services, MSDN, Create a small web services

**10. Deployment: (2L)**

Deploying applications using wizard MSDN

**11. Security introduction:** (2L)  
Securing a Microsoft Applications, MSDN

**12. C#:** (4L)  
Introduction to C#, Windows Forms with C#, Difference between C# and VB.Net MSDN  
Textbox, Label, Command button

**13. Object Oriented Programming in C#.Net:** (2L)  
Class and Objects, Properties, Methods and events, Contractors and Destructor, Method overloading,  
Inheritance, Interface

**Reference Books:**

1. MCAD / MCSD:Developing Windows-Based Applications With Microsoft Visual Basic.Net and Visual C# .Net By Mattjew Stoecker
2. MCAD / MCSD:Developing Web-Based Applications With Microsoft Visual Basic.Net and Visual C# .Net By Jeff Webb
3. Programming Microsoft VB.NET Microsoft Press By Belena
4. Programming Microsoft VB.NET ASP.NET By Dino Esposito

\* \* \* \* \*

## M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III

### PGCS – MIII: MINOR PROJECT-III

#### Learning outcomes:

At the end of the course, a student shall be able to:

- apply knowledge and techniques learnt in theoretical classes for developing the s/w for real problems.
- Get an insight into the working of the real organizations/companies.
- gain deeper understanding in specific functional areas.
- exploring career opportunities in their areas of interest.

**Total Credits: 04**

#### Course content

The objective of this minor project is to gear up student for preparation of final project training in Semester-IV.

Student will select individually Commercial or Technical project based on Technologies learnt in Semester I. Each student will have to prepare proper documentation consisting of SRS, Modeling Techniques, Development Strategies and Implementation and Testing Strategies. Student may use any Design Methodologies such as SSAD, OOAD and UML etc.

This is a documentation project only. The project work will be presented by student using Power Point Presentation. The Institute may appoint external expert from industry or academics if it feels so. The students will be assessed internally by such panel for this project.

- The Project can be platform, Language and technology independent.
- Project will be evaluated by project guide.
- Assessment will be done weekly in the respective batch.
- Evaluation will be on the basis of weekly progress of project work, progress report, oral, results and documentation and demonstration.
- You should fill your status of the project work on the progress report and get the Signature of project guide regularly.
- Progress report should sharply focus how much time you have spent on specific task. (The format of progress report is given as follow.)
- You should keep all signed progress report.
- Project will not be accepted if progress report is not submitted and all responsibility remains with student.
- Students should prepare design document using SE/UML techniques depends on your project.

#### About project Report: -

The report should be typed on A4 size, executive bond paper for the final submission. The report should be in the good quality Rexene bound. We suggest, using one-and-half spaced printing, Times New Roman 12 font sizes for the normal text, 14-16 font sizes for headings & page titles. Number of copies: For one project you should prepare 2 copies of the project report. One for yourself, one for college (College copy can be in CD).

**Evaluation for internal 40 marks**

<i>Description</i>	<i>Marks</i>
UML /ERD/DFD diagrams	10
Technology and design base first demo	10
Project technology based two assignments	10
Second Demo	10

**Evaluation for external 60 marks**

Description	Marks
Demo	10
Report	10
Presentation	20
Viva	20

\*\*\*\*\*



**M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III**

**CORE ELECTIVE-III**

**PGCS-304: Software Architecture**

**Course outcomes:**

At the end of this course, a student shall be able to:

- design and motivate software architecture for large-scale software systems
- implement major software architectural styles, design patterns, and frameworks
- make use of various software architectural styles to design modern systems
- apply software paradigms and software architecture

**Total Credits: 04**

**Total lectures: 60**

**Course content**

**Pre-requisites:**

Some familiarity with modern software engineering concepts and experience in designing and developing software systems.

**SYLLABUS:**

**1. Introduction :**

**(8L)**

Basic Concepts, Introduction and need of Software Architecture , Software Architecture Design Plan, Software Architecture as an abstraction , Software Architecture terminology, Different Engineering concerns addressed by different views.

**2. Designing, Describing and using Software Architecture:**

**(8L)**

System Overview, Product Features, System Interactions, The future of IS2000

**3. Global Analysis:**

**(6L)**

Overview of Global Analysis Activities, Analyze factors, Develop strategies, Analyze Organisational , Technological, Product Factors, Global Analysis Summary

**4. Conceptual Architecture view:**

**(10L)**

Design Activities for the conceptual, Architecture view, Design of Conceptual view for IS2000, Summary of Conceptual Architecture view, Traceability. Uses for the Conceptual Architecture view.

**5. Module Architecture View:**

**(8L)**

Design Activities for the Module Architecture view, Design of Module view for IS2000. Uses for the Module Architecture view.

**6. Execution Architecture View:**

**(6L)**

Design Activities for the Execution Architecture View, Design of Execution View for IS2000. Design summary for IS2000 Execution View, Summary of Execution Architecture View And Traceability. Uses for the Execution Architecture View.

**7. Code Architecture View :**

**(6L)**

Design Activities for the Code Architecture View, Design of code Architecture View for the IS2000. Summary of code Architecture, View and Traceability, Uses for the code Architecture View

**8. Software Architecture Best Practice:**

**(4L)**

Global Analysis, Conceptual Architecture View, Module Architecture View, Execution Architecture View, Code Architecture View, Software Architecture Uses

## 9. Role of software Architecture:

(4L)

Creating a Vision, The Architect as a Key Technical Consultant, The Architect makes decisions,  
The Architect Coaches, The Architect Coordinates, The Architect Implements

### Reference Books:

1. Applied software architecture, C. Hofmeister, R. Nord, D. Soni, Addison Wesley, (2000)
2. Software paradigms, S.H. Kaisler, John Wiley & sons, (2005).
3. Design Patterns: Elements of Reusable Object - Oriented software, E. Gamma, R. Helm, R. Johnson, J. Vlissides, Addison – Wesley (1995).

\* \* \* \* \*

## M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III

### CORE ELECTIVE-III PGCS-305: Software Testing

#### Course Outcomes:

At the end of this course, a student shall be able to:

- apply the principles of system and component testing.
- gain knowledge about strategies for generating system test cases.
- analyze understand the essential characteristics of tool used for test automation.
- identify differences between validation testing and defect testing.

**Total Credits: 04**

**Total lectures: 60**

#### Course content

##### Pre - requisites:

Some familiarity with modern software engineering concepts and experience in designing and developing software systems.

##### **1: Big Picture: Introduction: (6L)**

Famous case studies. What is a bug? Software failure terminologies, Why do bugs occur? Cost of bugs in software, Categories, Consequences, Testing Fundamentals. Test Axioms. Six Essentials of testing, Tester:? Competencies. Assessing Quality of the organization.

##### **2: The S/W Development Process: (6L)**

Realities of S/W testing, Life Cycle Models, Building a s/w testing process. Realities of S/W testing, Types of testing. Testing terms and definitions. Developing a test plan, Requirements verification checklist. Case study, Workbench, Do Procedures, Check Procedures.

##### **3: Testing Methods: (6L)**

Verification Testing, Validation Testing, What are test cases? Creating test cases. Test case planning overview, Reducing the number of test cases.

##### **4: Eleven Step Testing Process, Testing Process Overview: (6L)**

Clean Sheet Approach, Eleven Steps, Issues in requirements testing. Black-box, White-box, Static and Dynamic Testing.

##### **5: Examining code, Issues in Code Testing. Static methods, Case Study (10L)**

Testing Software with X-Ray glasses, Dynamic testing, Coverage Types. Validation, Validation Testing, Controlling validation Costs, Alpha & Beta Testing, Software Testing tools, Types continued, Need for Tools, Types.

##### **6: Test Documentation & Reporting (10L)**

Bug Reporting, Bug Tracking Systems, Measurement Types. Metrics, Recommendations. Types of testing: More categories, Flow Testing, Path testing, Syntax Testing, Transition Testing, Compatibility Testing, Usability Testing, Configuration Testing, Web Site Testing, Foreign Language Testing.

##### **7: Software Quality Assurance (10L)**

CMM, Career: Software Tester, Hands-On Experience. Training, Professional Organizations, Managing Test Technology.

## **8: Current Practices & Trends**

**(6L)**

What is new? Developer to Tester Ratio? Benchmarks, Test Security. Testing off-shelf s/w, Identification and precautions.

### **Reference Books:**

1. Software Testing: Ron Patton
2. Effective Methods for Software Testing: William E Perry, Wiley
3. Software Testing in the Real World: Edward Kit
4. Introducing Software Testing: Louise Tamres
5. Software testing Techniques: Boris Beizer
6. Effective Methods for Software Testing: William Perry, Wiley
7. Software Testing: Renu Rajani

\* \* \* \* \*

**M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III**  
**CORE ELECTIVE-III PGCS-306: Advanced Operating System**

**Course Outcomes:**

At the end of this course, a student shall be able to:

- manage various OS components like System calls, Schedulers, Memory management systems, Virtual Memory and Paging systems.
- and implement the policies for scheduling, deadlocks, memory management, synchronization, system calls, and file systems
- analyze the various device and resource management techniques for timesharing and distributed systems.
- analyze simple concurrent programs using transactional memory and message passing and to understand the trade-offs and implementation decisions.

**Total Credits: 04**

**Total lectures: 60**

**Course content**

- 1. Overview of Operating systems: (4L)**  
Objectives and Functions, Essentials of modern operating systems, Overview of UNIX, LINUX and Windows
- 2. Process Management and Scheduling: (6L)**  
Process and its states, Process control, Scheduling algorithms, Multi-processor and real time scheduling, UNIX Process management & scheduling, Windows Scheduling
- 3. Threads and Symmetric Multi-processing (4L)**  
Processes and Threads, SMP, Implementation of above in Popular Operating Systems
- 4. Concurrency Control (6L)**  
Mutual exclusion and synchronization, Deadlocks and Starvation, Mutual Exclusion & concurrency, Semaphores, Monitors, message Passing, Readers/Writers, Problem
- 5. Memory Management and Virtual Memory (12L)**  
Memory Management, Partitioning, Paging and Segmentation, Virtual memory and associated algorithms, Implementation of above in Windows and UNIX
- 6. I/O management and File management (12L)**  
I/O devices, I/O functions, I/O buffering, Disk Scheduling, RAID levels & Disk cache ,Virtual File Systems, UNIX and Windows support for I/O, File Organisation, directories, sharing, Storage management and Record Blocking, UNIX and Windows File management
- 7. Networking and Security: (8L)**  
Protocol architecture, TCP/IP, Sockets, Network File System, Security Threats, protection, Intruders, malicious software, trusted systems
- 8. Distributed processing, client/servers and Clusters (8L)**  
Client/server computing, message passing, RPC Clusters

**References Books:**

1. Operating Systems – Internals and Design principles by William Stallings ( Fifth Edition, Prentice Hall India – 2005 )

2. The MINIX Book: Operating System – Design and Implementation by Andrew S Tanenbaum and Albert S Woodhull (Third Edition, Prentice Hall India – 2006)
3. UNIX Systems for Modern Architectures by Curt Schimmel, Addison Wesley
4. Applied Operating Systems Concepts, Seventh Edition by Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne, John Wiley and Sons, 2004

\* \* \* \* \*

**M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III**

**PGCS - 307: Lab Course -VII**

**Course Outcomes:**

At the end of this course, a student shall be able to:

- apply software testing knowledge and engineering methods
- apply a software test process for a software testing project
- develop ability to identify the needs of software testing automation
- define and develop the test tool to support test automation

**Total Credits: 02**

**Course content**

1. Practical Title

- Problem Statement,
- Process Model

2. Requirement Analysis

- Creating a Data Flow
- Data Dictionary,
- Use Cases

3. Project Management

- Computing FP
- Effort
- Schedule, Risk Table, Timeline chart

4. Design Engineering

- Architectural Design
- Data Design, Component Level Design

5. Testing

- Basis Path Testing

6. Case study on software cost estimation

\* \* \* \* \*

## M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III

### PGCS - 308: Lab Course- VIII

#### Course Outcomes:

At the end of this course, a student shall be able to:

- develop applications using .net framework for C# programming languages.
- perform connectivity application using ADO.Net.
- perform web services.

#### Total Credits: 02

#### Course content

- Attain a detailed working knowledge of C-Sharp implicit types, object initializers, delegates, anonymous types & methods, extension methods, lambda expression, LINQ and many more.
- Acquire a working knowledge of creating and rich internet Web application using the .NET Framework 4.5 and Visual Studio 2012.
- Attain a detailed knowledge of the building blocks of Web application, including C-Sharp, ASP.NET, ADO.NET, Web Services and Ajax.
- Understand the ASP.NET programming model issues, such as state, caching and data handling management.
- Configure and deploy a Microsoft ASP.NET Web application. Our dot net training in Chennai will help you gain a strong knowledge in all Dot Net Web application concepts
- Produce ASP.NET programs for the web using solid multilayer architectures.
- Learn how to implement web applications using web forms, including programs that interact with databases.
- Work with data from multi sources like objects, XML, databases using Language Integrated Query (LINQ) and the Entity Framework (EF).
- Develop a clean, maintainable code base using the Model View Controller (MVC 4) architecture.
- Create flexible views for user interaction with view helpers.
- Learn to manage data access tasks by using LINQ.
- Learn to create a Microsoft ASP.NET AJAX application.
- Learn to consume and Windows Communication Foundation (WCF) services.
- Learn to manage data by using ASP.NET Dynamic Data.

\* \* \* \* \*



## M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III

### PGCS - 309: Lab Course - IX

#### Course Outcomes:

At the end of this course, a student shall be able to:

- perform various Linux and Unix command for managing operating system as well as user.
- to understand the basic commands of Linux operating system and can write shell scripts
- create file systems and directories and operate them.
- create shared memory segments, pipes, message queues and can exercise interprocess communication

#### Total Credits: 02

#### Course content

#### Note: Following exercises can be performed using Linux or Unix

1. Usage of following commands: ls, pwd, tty, cat, who, who am I, rm, mkdir, rmdir, touch, cd.
2. Usage of following commands: cal, cat (append), cat (concatenate), mv, cp, man, date.
3. Usage of following commands: chmod, grep, tput (clear, highlight), bc.
4. Write a shell script to check if the number entered at the command line is prime or not.
5. Write a shell script to modify “cal” command to display calendars of the specified months.
6. Write a shell script to modify “cal” command to display calendars of the specified range of months.
7. Write a shell script to accept a login name. If not a valid login name display message – “Entered login name is invalid”.
8. Write a shell script to display date in the mm/dd/yy format.
9. Write a shell script to display on the screen sorted output of “who” command along with the total number of users.
10. Write a shell script to display the multiplication table any number,
11. Write a shell script to compare two files and if found equal asks the user to delete the duplicate file.
12. Write a shell script to find the sum of digits of a given number.
13. Write a shell script to merge the contents of three files, sort the contents and then display them page by page.
14. Write a shell script to find the LCD (least common divisor) of two numbers.
15. Write a shell script to perform the tasks of basic calculator.
16. Write a shell script to find the power of a given number.

\* \* \* \* \*

**M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III**  
**PGSEC31: Android Programming**

**Course Outcomes:**

At the end of this course, a student shall be able to:

- understand Android platform and its architecture.
- learn about mobile devices types and different modern mobile operating systems.
- learn activity creation and Android User Interface designing.
- learn basics of Intent, Broadcast and Internet services.
- understand different approaches to structuring the applications
- understand application life cycle

**Total Credits: 02**

**Total lectures: 30**

**Course content**

- 1.Introduction: (4L)**  
History of Android, Introduction to Android Operating Systems, Android Development Tools, Android Architecture.
- 2.Overview of object oriented programming using Java: (6L)**  
OOPs Concepts: Inheritance, Polymorphism, Interfaces, Abstract class, Threads, Overloading and Overriding, Java Virtual Machine.
- 3.Development Tools: (8L)**  
Installing and using Eclipse with ADT plug-in, Installing Virtual machine for Android sandwich/Jelly bean (Emulator), configuring the installed tools, creating a android project – Hello Word, run on emulator, Deploy it on USB-connected Android device.
- 4.User Interface Architecture: (4L)**  
Application context, intents, Activity life cycle, multiple screen sizes.
- 5.User Interface Design: (4L)**  
Form widgets, Text Fields, Layouts, Button control, toggle buttons, Spinners (Combo boxes), Images, Menu, and Dialog.
- 6.Database: (4L)**  
Understanding of SQLite database, connecting with the database.

**Reference Books::**

1. Android application development for java programmers. By James C. Sheusi. Publisher: Cengage Learning, 2013.

**Online Reading / Supporting Material:**

1. <http://www.developer.android.com>
2. <http://developer.android.com/about/versions/index.html>
3. <http://developer.android.com/training/basics/firstapp/index.html>
4. <http://docs.oracle.com/javase/tutorial/index.htm> (Available in the form of free downloadable ebooks also).
5. <http://developer.android.com/guide/components/activities.html>
6. <http://developer.android.com/guide/components/fundamentals.html>

7. <http://developer.android.com/guide/components/intents-filters.html>.
8. <http://developer.android.com/training/multiscreen/screensizes.html>
9. <http://developer.android.com/guide/topics/ui/controls.html>
10. <http://developer.android.com/guide/topics/ui/declaring-layout.html>
11. <http://developer.android.com/training/basics/data-storage/databases.html>

**Software Lab Based on Android Programming:**

1. Create “Hello World” application. That will display “Hello World” in the middle of the screen in the emulator. Also display “Hello World” in the middle of the screen in the Android Phone.
2. Create an application with login module. (Check username and password).
3. Create spinner with strings taken from resource folder (res >> value folder) and on changing the spinner value, Image will change.
4. Create a menu with 5 options and selected option should appear in text box.
5. Create a list of all courses in your college and on selecting a particular course teacher-in-charge of that course should appear at the bottom of the screen.
6. Create an application with three option buttons, on selecting a button colour of the screen will change.
7. Create and Login application as above. On successful login, pop up the message.
8. Create an application to Create, Insert, update, Delete and retrieve operation on the database.

\* \* \* \* \*