Scheme of Course Structure M.Sc.(Computer Science)

Sr. No	Class	Sem	Code	Paper	Paper Title	Credit	Exam	Marks
1	M.Sc. - I	I	COMP4101	Theory	Principles of Programming Languages (C)	4	I / E	60 + 40
2	M.Sc. – I	I	COMP4102	Theory	Advanced Networking & Cryptography (C)	4	I / E	60 + 40
3	M.Sc. – I	I	COMP4103	Theory	Database Technologies (C)	4	I / E	60 + 40
4	M.Sc. – I	I	COMP4104	Theory	Design and Analysis of Algorithms (C)	4	I / E	60 + 40
5	M.Sc. – I	I	COMP4105	Theory	Programming with DOT NET (C)	4	I / E	60 + 40
6	M.Sc. – I	I	COMP4106	Pract.	Lab Course on DOT NET and PPL (C)	4	I / E	60 + 40
7	M.Sc. – I	I			Certificate Course	2		

Note: Minimum credit: 24 and Maximum Credit:24 Core subjects is compulsory and Extra credits (2) is also compulsory.

Sr. No	Class	Sem	Code	Paper	Paper Title	Credit	Exam	Marks
9	M.Sc. - I	II	COMP4201	Theory	Digital Image Processing (C)	4	I / E	60 + 40
10	M.Sc. - I	II	COMP4202	Theory	Data Mining and Data Warehousing (C)	4	I / E	60 + 40
11	M.Sc. - I	II	COMP4203	Theory	Python Programming (C)	4	I / E	60 + 40
12	M.Sc. - I	II	COMP4204	Theory	**Advanced Operating System (Elective I)**	4	I / E	60 + 40
13	M.Sc. - I	II	COMP4205	Pract.	Lab Course on Python Programming and Advance Operating System (C)	4	I / E	60 + 40
14	M.Sc. - I	II	COMP4206	Pract.	**Project (Elective II)**	4	I / E	60 + 40
15	M.Sc. - I	II	COMP4207	Theory	**Artificial Intelligence (Elective III)**	4	I / E	60 + 40
16	M.Sc. - I	II			Introduction Cyber Security – I	2		
17	M.Sc. - I	II	COMP4208	Theory	Modeling and Simulation **(Elective IV)**	4	I / E	60 + 40

Note: : Minimum credit: 28 and Maximum Credit:32 Core subjects is compulsory and Extra credits (2) is also compulsory. From elective courses 3 subjects for Minimum Credits and 4 for Maximum Credit

Sr. No	Class	Sem	Code	Paper	Paper Title	Credit	Exam	Marks
19	M.Sc. - II	III	COMP5301	Theory	Mobile Technologies (C)	4	I / E	60 + 40
20	M.Sc. - II	III	COMP5302	Theory	Soft Computing (C)	4	I / E	60 + 40
21	M.Sc. - II	III	COMP5303	Theory	Web Services (C)	4	I / E	60 + 40
22	M.Sc. - II	III	COMP5304	Theory	**Software Architecture& Design Pattern (Elective I)**	4	I / E	60 + 40
23	M.Sc. - II	III	COMP5305	Pract.	Lab Course-on Mobile Technology and Web Services (C)	4	I / E	60 + 40
24	M.Sc. - II	III	COMP5306	Pract.	**Project (Elective II)**	4	I / E	60 + 40
25	M.Sc. - II	III	COMP5307	Theory	**Recent Trends in IT (Internet of Things) (Elective III)**	4	I / E	60 + 40
26	M.Sc. - II	III			Human Rights	2		
27	M.Sc. - II	III			Certificate Course	2		
28	M.Sc. - II	III	COMP5308	Theory	Machine Learning **(Elective IV)**	4	I / E	60 + 40
29	M.Sc. - II	III	COMP5309	Theory	Network Programming **(Elective V)**	4	I / E	60 + 40

Note: : Minimum credit: 28 and Maximum Credit:36. Core subjects is compulsory and Extra credits (2+2) is also compulsory. From elective courses 3 subjects for Minimum Credits and 5 for Maximum Credit

30	M.Sc. - II	IV	COMP5401	Project	Industrial Training/ Institutional Project (**IT**) (**Core**)	16	I / E	60 + 40
31	M.Sc. - II	IV	COMP5402	Theory	Cloud Computing **(Elective I)**	4	I / E	60 + 40
32	M.Sc. - II	IV	COMP5403	Theory	Evolutionary Algorithm **(Elective II)**	4	I / E	60 + 40
33	M.Sc. - II	IV			Introduction Cyber Security – II	2		

Note: Core subject is compulsory, if students had completed 80 credits within three semesters then no need to select any elective course in Fourth Semester otherwise students should select appropriate number of elective courses to minimum complete 96 credits.

M.Sc. (Computer Science) II - Credit Structure

Subject	Semester III	Semester IV	Total
Paper – I	4 (Core)	4 (Elect) (Opt)	4
Paper – II	4 (Core)	4 (Elect) (Opt)	4
Paper – III	4 (Core)	-----	4
Paper - IV	4 (Elective)	**Industrial Training Project (IT)** 16	8
Paper – V	4 (Elective)		8
Practical / Paper VI (Sem IV)	4 (Core)		8
Practical (Project)	4 (Elective)		8
Human Rights	2	--	2
Introduction to Cyber Security – II	--	2	2
Certificate Course	2	--	2
Total ====	**32**	**18**	**50**

SYLLABUS (CBCS) FOR M. Sc. (Computer Science)-II (Semester- III)
(w.e.f from Academic Year 2020-2021

Class: M. Sc. (Computer Science) (Sem-III)   **Paper Code :** Comp5301
Title of Paper: Mobile Technologies (Core)   **Paper:** I
Credit: 4 (4 Lectures/Week)   **No. of lectures:** 48
Pre-requisites:

- Concepts of Networking, Wireless communication.
- Familiar with object oriented concepts
- Conversant with OS internals

Course Objectives:

- Students can understand basic concepts of mobile technology and mobile platforms
- Students can understand concepts related to wireless communication.
- Create native android applications using basic concepts
- Understand UI design or controls available in android.

Title and Contents	No. of Lectures
Unit 1 : Wireless Transmission 1.1 Multiplexing and Modulation 1.2 Spread Spectrum 1.3 Wireless transmission media 1.4 Migration from 2G to 4G 1.5 Wireless LAN advantages 1.6 Wireless LAN architecture 1.7 Mobility in Wireless LAN 1.8 Wireless LAN security	7
Unit 2: Introduction to Mobile Computing 2.1 Introduction and need for Mobile computing 2.2 Mobility and portability 2.3 Mobile and Wireless devices 2.4 Mobile Applications 2.5 Mobile Operating system – IOS, BlackBery, Windows ,phone, Plam OS, Symbian OS,PhoneGap 2.6 Cellular systems. 2.7 Introduction to IEEE 802.11	6

Unit 3: GSM Technology
3.1   What is GSM?
3.2   GSM Services.
3.3   GSM Network Architecture
    a.   Radio Subsystem
    b.   Network Subsystem
    c.   Radio Subsystem
3.4   Localization and Calling
3.5   Handover
3.6   GSM security
3.7   GPRS
3.8   CDMA in Mobile Communication
	8

Unit 4 : Introduction to Android Operating System & Programming 4.1 Features of Android 4.2 Android Architecture 4.3 Fundamental of Android App 4.4 Java for Android 4.5 Activities and Intents 4.6 User Interface 4.7 Services and Broadcast Receivers	8
Unit 5: Android UI Design 5.1 Basic UI Designing (Form widgets ,Text Fields , Layouts ,[dip, dp, sip, sp] versus px) 5.2 Intent (in detail) 5.3 All components (e.g Button , Slider, Image view, Toast) Event Handling 5.4 Adapters and Widgets 5.5 Menu	5
Unit 6:Android Thread and Notification 6.1 Threads running on UI thread (run On Ui Thread) 6.2 Worker thread 6.3 Handlers & Runnable 6.4 AsynTask (in detail) 6.5 Broadcast Receivers Services and notifications 6.6 Toast 6.7 Alarms	4

Unit 7: iOS Fundamentals
 7.1 **Introduction** - What is IOS ,IOS Architecture, Frameworks, Application Life
Cycle, Features
 7.2 **Swift -** Introduction to Swift ,General Concepts of Swift
 7.3 **Xcode -** Introduction to Xcode , Navigator, Editor Utility, Tools, Console,
    Document, Simulator, Instruments
 7.4 **Startup -** Application Templates, Introduction to Storyboard , Hello World
    Application, How 'Hello World' Working, Debugging Database, Plist,
    Preference, Sqlite Web Service, Restful Web Service (JSON & XML)
	6

Unit 8: Introduction to Flutter Mobile App Development 8.1 Flutter – Introduction 8.2 Features of Flutter 8.3 Basics of creating flutter application in the Android Studio. 8.4 Flutter – Architecture of Flutter Application 8.5 Flutter – Introduction to Dart Programming 8.6 Flutter – Introduction to Gestures. 8.7 Introduction to Beacon Mobile technology	04

References:

1. Schiller Jochen, (2003). Mobile Communications. **UK.: Pearson Education**
2. Wei-Meng Lee . Beginning Android Application Development. John Wiley & Sons, 2012
3. Talukder Asoke K. Hasan Ahmed,Roopa R Yavagal. Mobile Computing: Technology, Applications and Service Creation
4. Mark Murphy. Beginning Android 3. APress , ISBN 9788132203568
5. Matthijs Hollemans . IOS Apprentice

Websites :

https://www.tutorialspoint.com/
https://developer.android.com/
https://w3resource.com/
 https://abhiandroid.com/

Class: M. Sc. (Computer Science) (Sem-III)   **Paper Code :** Comp5302
Title of Paper: Soft Computing (Core)   **Paper:** II
Credit: 4 (4 Lectures/Week)   **No. of lectures:** 48

Prerequisites : Probability, First Order Mathematical Logic, Classical Logic, Linear algebra and calculus
Objective
- To understand the concepts of how an intelligent system work and its brief development process.
- Be familiar with design of various neural networks & fuzzy logic & Learn genetic programming.

Learning Outcome:
- Describe human intelligence and AI
- Explain how intelligent system works.
- Apply basics of Fuzzy logic and neural networks.
- Understand the ideas of fuzzy sets, fuzzy logic and use of human experience relate with neural networks, generalize appropriate rules for inference systems
- Understand the genetic algorithms and other random search procedures.
- Develop some familiarity with current research problems and research methods in Soft Computing Techniques.

Units & Contents	No. Of Lectures
1. **Introduction to Fuzzy Logic** • The Illusion : Ignoring Uncertainty and accuracy • Uncertainty and information • Fuzzy set and membership • Chance versus Fuzziness • Classical Sets, Fuzzy Sets • Cartesian Product • Crisp Relations • Fuzzy relations • Tolerance and equivalence Relations, Fuzzy Tolerance and equivalence Relations • Value assignments, Other Forms of the Composition Operations • Features of the membership Function • various forms, Fuzzification, Defuzzification to Crisp sets • λ-Cuts for fuzzy Relations, Defuzzification to Scalars	12
2 : **Fuzzy logic** • Fuzzy Logic • Approximate Reasoning, Others forms of implication operations • Natural Language, Linguistic Hedges • Fuzzy (Ruled-Based) system • Graphical technique of inference • Membership value assignment-Intuition, Inference.	04

3 : **Fuzzy System and Classification**
• Fuzzy System Simulation-
  ○ Fuzzy Relation
  ○ Equations
  ○ Nonlinear Simulation Using Fuzzy Systems
	10

• Fuzzy Associative Memories
• Fuzzy Classification
  ○ Classification by Equivalence Relations
  ○ Cluster Analysis
  ○ Cluster Validity
• c-Means Clustering
• Hard c-Means
• Fuzzy c-Means
• Classification Metric
• Hardening the Fuzzy c-Partition
• Similarity Relations from Clustering
• Fuzzy Arithmetic and Extension Principle
  ○ Extension Principle
  ○ Fuzzy Arithmetic
  ○ Interval Analysis in Arithmetic
  ○ Approximate Methods of Extension.	

4 : Neural Network	12

• Neural networks:
  ○ Artificial Neural Network: Definition
  ○ Advantages of Neural Networks Application
  ○ Scope of Neural Networks
• Fundamental Concept:
  ○ Artificial Neural Network
  ○ Biological Neural Network
• Brain vs. Computer-
  ○ Comparison Between Biological Neuron and Artificial Neuron (Brain vs. Computer)
• Artificial Neurons
• Neural Networks and Architectures:
  ○ Neuron Abstraction
  ○ Neuron Single Functions
  ○ Mathematical Preliminaries
  ○ Neural Networks Defined
• Architectures:
  ○ Feed forward and Feedback,
• Salient Properties of Neural Networks
• Geometry of Binary Threshold Neurons and Their Networks:
  ○ Pattern Recognition and Data Classification
  ○ Convex Sets
  ○ Convex Hulls and Linear Separability
  ○ Space of Boolean Functions
  ○ Binary Neurons are Pattern Dichotomizers
  ○ Non-linearly Separable Problems
  ○ Capacity of a Simple Threshold Logic Neuron	

• Revisiting the XOR Problem • Multilayer Networks • How Many Hidden Nodes are Enough?	

5 :Introduction to learning
• Learning and Memory:
    o An Anecodatal Introduction
    o Long Term Memory
• The Behavioural Approach to Learning
• The Molecular Problem of Memory
• Learning Algorithms
• Error Correction and Gradient Descent Rules
• Learning Objective for TLNs
• Pattern Space and Weight Space
• Linear Seperabilty
• Hebb Network
• Perceptron Network
• α- Least Mean Square Learning
• MSE Error Surface and Its Geometry
• Steepest Descent Search with Exact Gradient Information
• μ-LMS:
    o Approximate Gradient Descent
    o Application of LMS to Noise Cancelation.
	06

6 :Genetic Algorithms
• A Gentle Introduction to Genetic Algorithms:
    o What are Genetic Algorithm?
    o Robustance of Traditional Optimization and Search Methods
    o The Goals of Optimization
• How are Genetic Algorithms Different from Traditional Methods?
• A simple Genetic Algorithm
• Genetic Algorithms at Work—a Simulation by hand
• Grist for the Search Mill—Important Similarities
• Similarity Templates (Schemata)Learning the Lingo.
	04

Reference Books
1. Fuzzy Logic With Engineering Applications, 3rd Edition By Timothy Ross , Wiley Publication
2. Neural Networks By Satish Kumar, Tata McGraw Hill
3. Introduction to Soft Computing by Deepa & Shivanandan, Wiley Publication
4. Genetic Algorithms in Search, Optimization and Machine Learning By David E. Goldberg, Pearson Education.

Class: M. Sc. (Computer Science) (Sem-III)   **Paper Code :** Comp5303
Title of Paper: Web Services (Core)   **Paper:** III
Credit: 4 (4 Lectures/Week)   **No. of lectures:** 48
Pre-requisites:
  Strong knowledge about Java programming, Good Understanding of Object Oriented Programming concepts. Must be familiar with XML.
Objectives:
- To Understand Web Services and implementation model for SOA.
- To Understand the SOA, its Principles and benefits.
- To understand cloud computing as a web service.
- To understand XML concepts. To understand paradigms needed for testing Web Services.

Learning Outcomes: Student will able to :
- Understand the principles of SOA
- Efficiently use market leading environment tools to create and consume web services.
- Indentify and select the appropriate framework components in creation of web service solution.

Units & Contents	No. of Lecture
1. Web Service and SOA fundamental	**8**
1. Introduction : what are Web Services?, Concept of Saas 2. Web services Vs Web based Implementation 3.Characteristics of Web Services: Types of Web services, Functional and non functional properties , State processing , loose Coupling, Service Granularity, Service Synchronization 4.Service Interface and Implementation 5.The Services oriented Architecture (SOA): Roles of Interaction in SOA, Layer of SOA 6.Quality O f Service (QoS) 7.Web Service Interoperability(WS-I) 8.Web Services Vs Components 9.Restful Services 10.Impact and Shortcomings of Web services: Impact of web services	
2. Web Services Architecture	**8**
1. Introduction 2.Web Services Architecture and It's Characteristics: Web service characteristics , Web Service Architecture 3 Core building blocks of web services 4 Standards and technologies available for implementing web services: SOAP, WSDL,UDDI 5 Web Services Communication Models: RPC-based Communication Model, Messaging-based communication Model 6 Basic steps of implementing web services 7 Developing web services-Enabled applications: Web Services Implementation Using J2EE Environment, Developing Web Services Using J2EE:AN Example 8 Packaging and Deploying the Service: Creating web Service Clients	
3. SOAP: Simple Object Access Protocol	**8**
1 Introduction 2 Inter-application communication and wire protocols: SOAP as a wire representation, SOAP as a messaging protocol 3 Structure of a SOAP message: SOAP Envelope, SOAP Header, SOAP Body 4 SOAP communication model: RPC-Style, Document –(Message)Style Web Services 5 Building SOAP Web Services 6 developing SOAP Web Services using Java 7 Error handling in SOAP and Advantages and disadvantages of SOAP.	

4. Describing and Discovering Web Services:	**12**
1 WSDL in the world of Web Services 2 Web Services life cycle: Anatomy of WSDL definition document, Patterns Of Operations 3 WSDL bindings, WSDL Tools, limitations of WSDL 4 Service discovery: Role of service discovery in a SOA, service discovery mechanisms 5 UDDI – UDDI Registries, uses of UDDI Registry, Searching information in a UDDI Registry, Deleting information in a UDDI Registry, limitations of UDDI.	
5. Cloud Computing:	**12**
1 Introduction: What is Cloud Computing?, Essential characteristic of cloud, Cloud Deployment Model, History, Benefits, Risk 2 SOA Meets the Cloud : Comparing SOA with Cloud Computing , SOA Deployment :Cloud Computing Vs The ESB 3 Cloud Computing Technologies: Virtualization, SOA, Grid Computing, Utility Computing 4 Cloud Computing Architecture: Front End, Back End 5 Cloud Model : Public Cloud ,Private Cloud, Hybrid Cloud, Benefits, Disadvantages 6 Security and Privacy: Insecure or incomplete data and deletion, Security Planning, Understanding security of cloud ,Security Boundaries, Understanding Data Security, Isolated Access To data 7 Cloud Computing Application: Bossiness, Social, Entertainment	

Learning Resources

Text Book :

Web Services & SOA Principles and Technology

Reference Book :

1.Restful Web Services Cookbook By Subbu Allamaraju

2. Service Architecture By Thomas Eri

3.XML,Web Services, and the Data Revolution ,F.P. Coyle, Pearson Education

Class: M. Sc. (Computer Science) (Sem-III)   **Paper Code :** Comp5304
Title of Paper: Software Architecture & Design Pattern   **Paper:** IV **(Elective-I)**
Credit: 4 (4 Lectures/Week)   **No. of lectures:** 48

Objectives:
>To Understand and learn the software architecture, its styles, views and pattern for design software with minimum complexity and maintain flexibility

Prerequisites:
>System Analysis and Design, Software Engineering, OOSE, Software project Management, UML

Outcomes: Able to assist learner to utilize styles and views to state Architecture, define documentation, analyze the architectural structures and it's Influence on business and development process.

Units & Title and Contains	No of Lect.	Assignments /test
1. Introduction to Software Architecture 1.1 Introduction to concept of Software Architecture 1.2 Definition , Architectural structures 1.3 Need and Influence of software architecture in organization as business and technical aspects 1.4 Architecture Business Cycle 1.5 Introduction – Functional requirements, Technical constraints, Quality Attributes	4	**Assignment-1**
2. Quality Attribute 2.1 Introduction Quality Attribute 2.2 Documenting Quality Attributes 2.3 Six part scenarios 2.4 Case studies	4	**Assignment-2**
3. Architectural Views 3.1 Introduction ,Definitions for views 3.2 Structures and views, Representing views, available notations, 3.3 Standard views,4+1 view of RUP, Siemens 4 views SEI's perspectives and views 3.4 Case studies	6	**Assignment-3** **Test-I**
4. ARCHITECTURAL STYLES 4.1 Introduction 4.2 Data flow styles with Case study 4.3 Call-return styles with Case study 4.4 Shared Information styles with Case study 4.5 Event styles with Case study	6	**Assignment-4**
5. Common Software Architectural Patterns 5.1 Layered pattern 5.2 Client-server pattern 5.3 Master-slave pattern 5.4 Pipe-filter pattern, 5.5 Broker pattern 5.6 Peer-to-peer pattern 5.7 Event-bus pattern 5.8 Model-view-controller pattern 5.9 Blackboard pattern 5.10 Interpreter pattern	12	**Assignment-5** **Test-II**

6. Design Pattern 6.1 Important Design Patterns 6.2 Design Pattern Catalogue, Creational, Structural and behavioral patterns 6.3 Structural Decomposition 6.4 Organization of Work, Access Control.	**10**	**Assignment-6**
7. DOCUMENTING THE ARCHITECTURE 7.1 Good practices,Documenting the Views using UML, 7.2 Merits and Demerits of using visual languages 7.3 Need for formal languages 7.4 Architectural Description LanguagesACME 7.5 Case studies. Special topics: SOA and Web services,Cloud Computing, Adaptive structures	**6**	**Assignment-7**

Reference Books:

- Beyond Software architecture, Luke Hohmann, Addison wesley, 2003.
- Software architecture, David M. Dikel, David Kane and James R. Wilson, Prentice Hall PTR, 2001
- Software Design, David Budgen, second edition, Pearson education, 2003
- Head First Design patterns, Eric Freeman & Elisabeth Freeman, O'REILLY, 2007.
- Design Patterns in Java, Steven John Metsker& William C. Wake, Pearson education, 2006
- J2EE Patterns, Deepak Alur, John Crupi& Dan Malks, Pearson education, 2003.
- Design Patterns in C#, Steven John metsker, Pearson education, 2004.
- Pattern Oriented Software Architecture, F. Buschmann& others, John Wiley & Sons
- Anti-Patterns: Refactoring Software, Architectures and Projects in Crisis, William J Brown et al., John Wiley, 1998
- Object-oriented analysis, design and implementation, brahma dathan, sarnathrammath, universities press,2013
- Design patterns, erich gamma, Richard helan, Ralph johman , john vlissides ,PEARSON Publication,2013.

Class: M. Sc. (Computer Science) (Sem-III)   **Paper Code :** Comp5305

Title of Paper: Lab Course based on Comp5301 & Comp5303   **Paper:** V

Credit: 4 (4 Hours Practical/Week/batch)   **No. of Practical's:** 13

Sr.No	Practical Assignments
Mobile Technologies Practical Assignments Topics	
1	Assignment based on basic UI Design
2	Assignment based on Basic UI Design
3	Assignment based on Thread and Notification
4	Assignment based on handler on UI thread
5	Assignment based on using Intent, Toast
6	Assignment based on swift in ios
7	Assignment based on startup
8	Assignment based on flutter
Web services Practical Assignments Topics	
1	Introduction of Basic web services
2	Assignment on SOAP
3	Assignment on WSDL
4	Assignment on UDDI
5	Assignment on XML

Class: M. Sc. (Computer Science) (Sem-III) **Paper Code :** Comp5306
Title of Paper: Projects- Lab Course **Paper:** VI **(Elective-II)**
Credit: 4 (4 Hours Practical/Week/batch) **No. of Practical's:** 12

- The Project can be platform, language and technology independent.
- Project will be evaluated by the 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 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?
- You should keep all sign progress report.
- Project will not be accepted, if progress report is not submitted and all the responsibilities remain with student.

Roll No. & Name of Student:	
Title of the Project:	
Project Guide Name:	

Sr.No.	Date	Details of Project Work	Project Guide Sign (With Date)

Project Guide

Class: M. Sc. (Computer Science) (Sem-III)   **Paper Code :** Comp5307
Title of Paper: Internet of Things (IOT)   **Paper:** VII **(Elective-III)**
Credit: 4 (4 Lectures/Week)   **No. of lectures:** 48
Pre-Requisite: Basic understanding of electronics and microprocessors.
Course Objectives:
- The Internet of Things (IoT) is aimed at enabling the interconnection and integration of the physical world and the cyber space.
- To learn about SoC architectures, programming Raspberry Pi and implementation of internet of things and protocols.

Expected Learning Outcomes:
- Enable learners to understand System On Chip Architectures.
- Introduction and preparing Raspberry Pi with hardware and installation.
- Learn physical interfaces and electronics of Raspberry Pi and program them using practical's
- Learn how to design IoT based prototypes.

Unit and Contents	No. Of Lectures
Unit 1: System on Chip (SoC) and Internet of Things (IoT) Overview - **System on Chip:** What is System on chip? Structure of System on Chip. - **SoC products:** Field Programmable Gate Array (FPGA)**,** General Purpose Graphics Processing Units (GPU), Accelerated Processing Unit (APU), Compute Units. -**The IoT paradigm** giving overview of IoT supported Hardware platforms such as: Raspberry pi, SoC on ARM 8 Processors, Arduino and Intel Galileo boards. -**Network Fundamentals**: Wired Networking(Router, Switches), Wireless Networking (Access Points) **-Introduction to Raspberry Pi:** Introduction to Raspberry Pi, Raspberry Pi Hardware, Preparing your raspberry Pi. **-Raspberry Pi Boot:** Learn how this small SoC boots without BIOS. Configuring boot sequences and hardware. **-Introduction to IoT:** What is IoT? IoT examples, Simple IoT LED Program. **-IoT and Protocols** **-IoT Security:** HTTP, UPnp, CoAP, MQTT, XMPP**.** **-IoT Service as a Platform:** Clayster, Thinger.io, SenseIoT, carriots and Node RED. **-IoT Security and Interoperability:** Risks, Modes of Attacks, Tools for Security and Interoperability.	20
Unit 2: Programming Raspberry Pi **Raspberry Pi and Linux:** About Raspbian, Linux Commands, Configuring Raspberry Pi with Linux Commands **Programing interfaces:** Introduction to Node.js, Python. **Raspberry Pi Interfaces:** UART, GPIO, I2C, SPI **Useful Implementations:** Cross Compilation, Pulse Width Modulation, SPI for Camera.	10
Unit 3: Case Study & advanced IoT Applications: IoT applications in home, infrastructures, buildings, security, Industries, Home appliances, other IoT electronic equipments. Use of Big Data and Visualization in IoT, Industry 4.0 concepts. Sensors and sensor Node and interfacing using any Embedded target boards (Raspberry Pi / Intel Galileo/ARM Cortex/ Arduino)	10
Unit 4: Internet of Things Privacy, Security and Governance Introduction, Overview of Governance, Privacy and Security Issues, Contribution from FP7 Projects, Security, Privacy and Trust in IoT-Data-Platforms for Smart Cities, First Steps Towards a Secure Platform, Smartie Approach. Data Aggregation for the IoT in Smart Cities, Security	8

References:

1. The Internet of Things: From RFID to the Next-Generation Pervasive Networked Lu Yan, Yan Zhang, Laurence T. Yang, Huansheng Ning

2. Internet of Things (A Hands-on-Approach) , Vijay Madisetti , Arshdeep Bahga

3. Designing the Internet of Things , Adrian McEwen (Author), Hakim Cassimally

4. "Mobile Computing," Tata McGraw Hill, Asoke K Talukder and Roopa R Yavagal, 2010.

5. Computer Networks; By: Tanenbaum, Andrew S; Pearson Education Pte. Ltd., Delhi, 4th Edition

6. Data and Computer Communications; By: Stallings, William; Pearson Education Pte. Ltd., Delhi, 6th Edition

7. "Fundamentals of Mobile and Pervasive Computing," F. Adelstein and S.K.S. Gupta, McGraw Hill, 2009.

8. Cloud Computing Bible, Barrie Sosinsky, Wiley-India, 2010

Class: M. Sc. (Computer Science) (Sem-III)   **Paper Code :** Comp5308
Title of Paper: Machine Learning   **Paper:** VIII **(Elective-IV)**
Credit: 4 (4 Lectures/Week)   **No. of lectures:** 48

Objective :
- To discover patterns in the user data and then make predictions based on these.
- To intricate patterns for answering business questions and solving business problems.
- To helps in analyzing the data as well as identifying trends.

Prerequisites:
   Basics Of : Statistics, Linear Algebra, Calculus, Probability, Programming skills

Outcomes:
- To know discipline of designing algorithms that allow machines (eg., A Computer) to learn patterns & concepts from data without being explicitly programmed.
- It focuses on the recent real world application of machine learning algorithms.

Unit and Contents	No. Of Lectures

1.Inroduction to Machine learning
• Meaning and need for Machine Learning (ML).
• Types of Learning
   - Predictive or supervised learning
   - Classification
	03

2.Learning Frameworks • Learning framework , Domain set, Label set ,Training and test data • Learner's output, Data generation model, Empirical risk and over fitting, Hypothesis classes.	05
3. Support Vector Machine (SVM) Learning • Support Vector Machine (SVM) Learning – Definition • Maximal-margin classifier • Soft margin classifier • SVM kernels – - Linear kernel SVM, Polynomial kernel SVM, Radial kernel SVM, Data preparation for SVM, SVM regression	12
4. Artificial Neural Network (ANN) Learning • Artificial Neural Network (ANN) Learning – Perception learning, Learning • Boolean functions • XOR problem • Multilayer perception learning • Back propagation algorithm and related concepts. • Applications to classification and non-linear regression.	14
5. Classification and clustering • Dimensionality reduction • Application of rough set theory to classification and clustering. • Design and analysis of ML experiments • Cross-validation and re-sampling methods. • Introduction to open source software such as Python	14

Reference Books:

1. Understanding Machine Language: From Theory to Algorithms Shai Shalev_Shwartz and Shai Ben-David (2014)

2. Introduction to Machine Language (Second Edition) Ethem Alpaydn (2010) MIT press

3. Course on Machine Learning Andrew Ng (Open classroom.stanford.edu)

Class: M. Sc. (Computer Science) (Sem-III) **Paper Code :** Comp5309
Title of Paper: Network Programming **Paper:** VIII **(Elective-V)**
Credit: 4 (4 Lectures/Week) **No. of lectures:** 48

Prerequisites:
  ➢ Working Knowledge of C
  ➢ Basic Understanding of Networking Concepts
  ➢ User Level Knowledge of Linux

Contents:
UNIT 1: Introduction [2]
• A Simple Daytime Client, Protocol Independence, Error Handling: Wrapper Functions, A Simple Daytime Server [Book-1]

UNIT 2: Sockets Introduction [6]
• Socket Address Structures, Value-Result Arguments, Byte Ordering Functions, Byte Manipulation Functions, inet_aton, inet_addr, and inet_ntoa Functions, inet_pton and inet_ntop Functions,sock_ntop and Related Functions, readn, writen, and readline Functions, isfdtype Function [Book-1]
• What is a Socket?, Using Sockets [Book-2]

UNIT 3: Elementary TCP Sockets [4]
• Socket Function, connect Function, bind Function, listen Function, accept Function, fork and exec Functions, Concurrent Servers, close Function, getsockname and getpeername Functions [Book-1]

UNIT 4: TCP Client-Server Example [6]
• TCP Echo Server: main Function, TCP Echo Server: str_echo Function, TCP Echo Client: main Function, TCP Echo Client: str_cli Function, Normal Startup, Normal Termination, Connection Abort before accept Returns, Termination of Server Process, SIGPIPE Signal, Crashing of Server Host, Crashing and Rebooting of Server Host, Shutdown of Server Host [Book-1]

UNIT 5: I/O Multiplexing: The select and poll Functions [6]
• I/O Models, select Function, str_cli Function (Revisited), Batch Input, shutdown Function, str_cli Function (Revisited Again), TCP Echo Server (Revisited), pselect Function, poll Function, TCP Echo Server (Revisited Again) [Book-1]

UNIT 6: Socket Options [4]
• getsockopt and setsockopt Functions, Checking If an Option Is Supported and Obtaining the Default, Socket States, Generic Socket Options, IPv4 Socket Options, ICMPv6 Socket Option, IPv6 Socket Options, TCP Socket Options [Book-1]

UNIT 7: Elementary UDP Sockets [8]
• recvfrom and sendto Functions, UDP Echo Server: main Function, UDP Echo Server: dg_echo Function, UDP Echo Client: main Function, UDP Echo Client: dg_cli Function, Lost Datagrams, Verifying Received Response, Server Not Running, Summary of UDP example, connect Function with UDP, dg_cli Function (Revisited), Lack of Flow Control with UDP, Determining Outgoing Interface with UDP, TCP and UDP Echo Server Using select [Book-1]
• User Datagram Protocol, File Transfer, Error Handling [Book-2]

UNIT 8: Protocols, Sessions, State, and Implementing Custom Protocols [4]
• State vs. Stateless, Methods for Maintaining State, What Is a Protocol?, Designing a Custom Protocol, Our Chat Protocol, Protocol Registration [Book-2]

UNIT 9: Elementary Name, Address Conversions and design decisions   **[8]**

• Domain Name System, gethostbyname Function, RES_USE_INET6 Resolver Option, gethostbyname2 Function and IPv6 Support, gethostbyaddr Function, uname Function, gethostname Function, getservbyname and getservbyport Functions [Book-1]

• TCP vs. UDP, Application Protocol Choices, Client-Server Architecture, Client-Side Considerations, Server-Side Considerations [Book-2]

References:

T1: Unix Network Programming, Volume 1: The Sockets Networking API, 3/E by W. Richard Stevens, Bill Fenner, Andrew M. Rudoff, PHI

T2: The Definitive Guide to Linux Network Programming by KEIR DAVIS, JOHN W. TURNER, AND