GURU KASHI UNIVERSITY

Post Graduate Diploma in Computer Application
Session: 2024-2025

Department of Computer Applications

GRADUATE OUTCOME OF THE PROGRAMME

This program focuses on the design and implementation of programming, database management, software development, networking, security and web development. It also emphasizes the ability to adapt to new technologies and emerging trends in the field.

PROGRAM LEARNING OUTCOMES

After completion, the program the Learner will be able to:

- Apply mathematical foundation, computing knowledge for the conceptualization of computing models from defined problems.
- Identify, critically analyze and formulate complex computing problems using fundamentals of computer science and application domains.
- Use the modern programming languages, tools, techniques, and skills necessary for designing, developing, and deploying software-based applications.
- Apply ethical principles and come to professional ethics, responsibilities, and norms of the computer practice.
- Adopt a research culture and implement policies to address pressing local and global concerns.

Program Structure of the Post Graduate Diploma in Computer Application

SEMESTER 1st						
Course Code	**Course Title**	**Type of course**	**L**	**T**	**P**	**Credits**
GCA122	Introduction to programming languages	Core	4	0	0	4
GCA123	Relational Database Management Systems	Core	4	0	0	4
GCA124	Computer System Architecture	Multidisciplinary	3	0	0	3
GCA125	Fundamentals of Information Technology Lab	Compulsory Foundation	0	0	4	2
GCA126	Introduction to programming languages Lab	Technical Skill	0	0	2	1
GCA127	Relational Database Management Systems Lab	Technical Skill	0	0	2	1
GCA128	Ethical Hacking	VAC	2	0	0	2
Discipline Elective I (Any one of the following)						
GCA111	Internet Concepts and Web Designing	Discipline Elective I	3	0	0	3
GCA116	Data Warehousing and Data Mining					
GCA117	IoT and Its Applications					
Discipline Elective II (Any one of the following)						
GCA129	Deep Learning	Discipline Elective II	3	0	0	3
GCA119	Data Network & Security					
GCA120	Software Project Management					
Total			**19**	**0**	**8**	**23**

SEMESTER 2nd						
Course Code	**Course Title**	**Type of course**				
			L	**T**	**P**	**Credits**
GCA201	Data Structures	Core	4	0	0	4
GCA202	Digital Electronics	Core	4	0	0	4
GCA219	Programming Using Python	Core	4	0	0	4
GCA220	Data Structures Lab using Python	Technical Skill	0	0	2	1
GCA299	XXXX	MOOC	0	0	0	3
Discipline Elective III (Any one of the following)						
GCA210	Machine Learning	Discipline Elective III	3	0	0	3
GCA221	Data Visualization					
GCA222	Natural Language Processing					
Discipline Elective IV (Any one of the following)						
GCA216	Software Engineering & Testing	Discipline Elective IV	3	0	0	3
GCA223	Data Visualization Lab					
GCA224	Natural Language Processing Lab					
Total			**18**	**0**	**2**	**22**

Evaluation Criteria for Theory Courses

A. Continuous Assessment: [25 Marks]
   CA1- Surprise Test (Two best out of three) (10 Marks)
   CA2- Assignment(s)(10 Marks)
   CA3-Term Paper/Quiz/Presentations (05 Marks)

B. Attendance (5 Marks)
C. Mid SEMESTER Test: [30 Marks]

D. End-SEMESTER Exam: [40 Marks]

Evaluation Criteria for Practical Subjects

Total 20 Marks (Each Practical)

A. Performance of each practical (10 Marks)
B. Report (05 Marks)
C. Practical Viva (05 Marks)

Evaluation Criteria for Training/Internship/Survey Camp etc.

Total 25 Marks

A. Each Report (25 Marks) – Weekly/Monthly (25 Marks)

Evaluation Criteria for other courses has been given separately with the respective courses.

SEMESTER-I

Course Title: Introduction to programming languages
Course Code: GCA122

L	T	P	Credits
4	0	0	4

Total Hours: 60

Learning Outcomes:

After the Completion of this course the learner will able to

1. Describe all the basic concepts of C++ and its features such as composition of objects, Operator overloading.
2. Implement the various access modifiers in C++ programs.
3. Analyze inheritance with the understanding of early binding and late binding.
4. Analyze and explore various Stream classes, I/O operations and exception handling.

Course Content

UNIT I **17 Hours**

Programming Basics: Introduction to Programming, Programming Paradigms, Programming Languages and Types. Basic Program Structure, Execution flow charts of Program, Directives, Basic Input /Output, Advantages, Applications, Data Types, Control Structures, Operators and Expressions.

Introduction Structure, Execution flow, Classes and Objects, Access modifiers, Data Members, Member Functions, Inline Functions, Passing parameters to a Function (pass by Value, Pass by Address, Pass by Reference), Function with default arguments, Function Overloading, Object as a Parameter, Returning Object Static data members and functions, Constant Data members and functions Constructors- Default, Parameterized, Copy, Constructor Overloading, Destructors Arrays, Array as a Class Member, Array of Objects, Strings String Class.

UNIT II **14 Hours**

Operator Overloading and Pointers: Operator Functions-Member and Non Member Functions, Friend Functions Overloading Unary operators Overloading binary operators(Arithmetic, Relational, Arithmetic Assignment, equality), Overloading Subscript operator Type Conversion Operators- primitive to Object, Object to primitive, Object to Object Disadvantages of operator Overloading, Explicit and Mutable Pointers, Pointer and Address of Operator, Pointer to an Array and Array of Pointers, Pointer arithmetic, Pointer to a Constant and Constant Pointer, Pointer Initialization, Types of Pointers(void, null and dangling), Dynamic Memory Allocation, Advantages and Applications of pointers .

UNIT III **13 Hours**

Inheritance and Polymorphism: Inheritance Concept, protected modifier, Derivation of Inheritance- Public, Private and Protected,

Types of Inheritance-Simple, Multilevel, Hierarchical, Multiple, Hybrid, Constructors and Inheritance, Function Overriding and Member hiding Multiple Inheritance, Multipath inheritance – Ambiguities and solutions Polymorphism, Static and Dynamic Binding, Virtual Functions, Pure Virtual Functions, Virtual destructors, Abstract Classes, Interfaces

UNIT IV **16Hours**

Streams and Exceptions: Files, Text and Binary Files, Stream Classes, File IO using Stream classes, File pointers, Error Streams, Random File Access, Manipulators, Overloading Insertion and extraction operators Error handling, Exceptions, Throwing and catching exceptions, Custom Exceptions, Built in exceptions , Casting- Static casts, Const Casts, Dynamic Casts, and Reinterpret Casts. Creating Libraries and header files. Namespaces Generic Programming, Templates, Class Templates, Function Templates, Template arguments.

Transactional modes

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning**.**

Suggested Readings

- *Kamthane, A. (2012). Programming in C++, 2/e. Pearson Education India.*
- *Salaria, R. S. (2016). Mastering Object-Oriented Programming with C++. KHANNA PUBLISHING HOUSE.*
- *Balagurusamy, E. (2001). Object-Oriented Programming with C++, 7e. McGraw-Hill Education.*

Web Sources

- *https://www.tutorialspoint.com/basic-concepts-of-object-oriented-programming-using-cplusplus*
- *https://www.geeksforgeeks.org/operator-overloading-cpp/*
- *https://www.simplilearn.com/tutorials/cpp-tutorial/types-of-inheritance-in-cpp*

Course Title: Relational Database Management Systems
Course Code: GCA123

L	T	P	Credits
4	0	0	4

Total Hours: 60

Learning Outcomes:
After the Completion of this course the learner will able to
1. Develops an Entity-Relationship model based on user requirements.
2. Implements the role of the database administrator and his responsibilities.
3. Apply Normalization techniques to normalize a database.
4. Declares and enforces integrity constraints on a database

Course Content

UNIT I **14 Hours**

Traditional file processing system: Characteristics, limitations, Database: Definition, composition. Database Management System: Definition, Characteristics, advantages over traditional file processing system, User of database, DBA and its responsibilities, Database schema, instance.

UNIT II **16 Hours**

DBMS architecture, data independence, mapping between different levels. Database languages: DDL, DML, DCL. Database utilities, Data Models, Keys: Super, candidate, primary, foreign.

UNIT III **15 Hours**

Entity relationship model: concepts, mapping cardinalities, entity relationship diagram, weak entity sets, strong entity set, aggregation, generalization, Overview of Network and Hierarchical model. Relational Data Model: concepts, constraints. Relational algebra: Basic operations, additional operations.

UNIT IV **15 Hours**

Database Design: Functional dependency, decomposition, problems arising out of bad database design, Normalization- Normal forms based on primary keys (1 NF, 2 NF, 3 NF, & BCNF), multi-valued dependency, Database design process, database protection, database integrity.
Database concurrency: Definition and problems arising out of concurrency.

Transactional modes

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning**.**

Suggested Readings

- *Ramakrishnan, R., Gehrke, J., &Gehrke, J. (2003). Database management systems (Vol. 3). New York: McGraw-Hill.KorthF. Henry. Database System Concepts, McGraw Hill.*

- *Dittrich, K. R., Gatziu, S., &Geppert, A. (1995, September). The active database management system manifesto: A rulebase of ADBMS features. In International Workshop on Rules in Database Systems (pp. 1-17). Springer, Berlin, Heidelberg.*

Web Sources

- *https://www.tutorialspoint.com/dbms/dbms_architecture.htm*
- *https://www.geeksforgeeks.org/introduction-of-er-model/*
- *https://www.javatpoint.com/dbms-tutorial*
- *https://www.w3schools.in/dbms*
- *https://www.youtube.com/watch?v=T7AxM7Vqvaw*
- *https://www.youtube.com/watch?v=c5HAwKX-suM*
- *https://www.youtube.com/watch?v=DxoRUmW44JE*

Course Title: Computer System Architecture
Course Code: GCA124

L	T	P	Credits
4	0	0	4

Total Hours: 60

Learning Outcomes

After the Completion of the course the learner will be able to

1. Determine the designing process of combinational and sequential circuits.
2. Understanding of instruction pipelining and RISC architecture.
3. Simplify Boolean expressions.
4. Design basic Gates, Sequential & Combinational circuits.

Course Content

UNIT I **14 Hours**

Boolean Algebra: Boolean operations, Truth Tables, Boolean Laws, K-maps 2,3 and 4 variable maps, don't care about conditions).Basic Gates, Combinational logic design: half-adder, full adder, parallel adder.

UNIT II **16 Hours**

Sequential circuits: concept, flip-flops (D, RS, JK, T), counters (Ripple, Asynchronous, Synchronous). Instruction codes, Instruction formats, Instruction cycle, addressing modes.

UNIT III **15 Hours**

Register Transfer Language, Arithmetic, Logic and Shift micro-operations, Arithmetic Logic Shift Unit Control Memory: Design of control unit, Micro programmed and hardwired control unit (overview only), Features of RISC and CISC.

UNIT IV **15 Hours**

Memory Organization: memory hierarchy, Memory types: cache, associative and other types. I/O organization: I/O interface, Modes of data transfer: Programmed I/O, Interrupt initiated I/O, DMA, Block diagram depicting architecture of 8085 machine.

Transaction Modes

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning**.**

Suggested Readings

- *M.M. Mano**.**Computer System Architecture**. Third Edition, Prentice-Hall of India, 2002.*
- *A.S.Tanenbaum. (1999).Structured Computer Organisation**. Prentice-Hall of India,*

- *William Stallings.(2002)Computer Organisation and Architecture. 6th Edition, Pearson Education.*

Web Sources

- *https://www.javatpoint.com/computer-organization-and-architecture-tutorial*
- *https://www.geeksforgeeks.org/computer-organization-and-architecture-tutorials/*
- *https://www.learncomputerscienceonline.com/computer-organization-and-architecture/*
- *https://www.gatevidyalay.com/computer-organization-architecture/*

Course Title: Fundamentals of Information Technology Lab
Course Code: GCA125

L	T	P	Credits
0	0	4	2

Total Hours: 30

Learning Outcomes:

After the Completion of the course the learner will be able to

1. Compose, format and edit a word document.
2. Discover, Navigate and search through the internet.
3. Use Open Office (Word processing, Spreadsheets and Presentation).
4. Utilize the MS PowerPoint.

Course Content

1. Creating, opening, closing, saving and editing a word Document.
2. Insert header and footer in the document.
3. Create a link between two files using Hyperlink.
4. Create a mail-merge and add data of 5 recipients.
5. Protect a document.
6. Implement macro.
7. Create duplicate slides in PowerPoint. Give an example.
8. Make a master slide.
9. Design a chart of population.
10. Insert Animation.
11. Insert a background in PowerPoint.
12. How you can filter your data.
13. Sort data in ascending and descending order.
14. To show the use of goal seek
15. To show the use of scenarios.
16. Perform any 5 Date and Time functions.
17. Perform any 5 Math & Trig functions.

Course Title: Introduction to Programming Languages Lab

Course Code: GCA126

L	T	P	Credits
0	0	4	2

Total Hours: 30

Learning Outcomes: After the Completion of the course the learner will be able to

1. Design an algorithmic solution for a given problem.
2. Debug a given Program.
3. Identify solutions to a problem and apply control structures and use defined functions for solving the problem.
4. Implement Programs with pointers and arrays, perform pointer arithmetic, and use the pre-processor.

Course Content

1. Program to display Names, Roll No., and grades of 3 learner who have appeared in the examination. Declare the class of name, Roll No. and grade. Create an array of class objects. Read and display the contents of the array.
2. Program to swap two Characters of different data types using function overloading.
3. Program to demonstrate the use of inline, friend functions and this keyword.
4. Program to implement static data members and member functions.
5. Program to implement Constructor and Destructor.
6. Program to demonstrate Constructor Overloading.
7. Program to calculate factorial using Copy Constructor.
8. Program to allocate & deallocate memory using new [ ] and delete [ ].
9. Program to demonstrate the use of function overloading.
10. Program to overload comparison operator operator== and operator !=.
11. Program to create an array of pointers.
12. Create a base class containing the data member roll number and name. Also create a member function to read and display the data using the concept of single level inheritance. Create a derived class that contains marks of two subjects and total marks as the data members.
13. Program to create multilevel inheritance. (Hint: Classes A1, A2, A3)
14. Program to demonstrate the concept of function overriding.
15. Program to demonstrate the use of virtual functions and polymorphism.
16. Program to demonstrate the use of pure virtual functions.
17. Program to demonstrate the concepts of abstract class.
18. Program to perform exception handling.
19. Program to copy the contents of one file to another file.
20. Program to create Generic Functions using Template.

Course Title: Relational Database Management Systems Lab
Course Code: GCA127

L	T	P	Credits
0	0	4	2

Total Hours: 60

Learning Outcomes:

After the Completion of the course the learner will be able to:

1. Populate and query a database using SQL DML/DDL commands.
2. Designs SQL queries to create database tables and make structural modifications.
3. Design the concept of inbuilt functions.
4. Implement the concept of join, views and indexes.

Course Content

1. Data Definition, Table Creation, Constraints,
2. Insert, Select Commands, Update and Delete Commands.
3. Nested Queries and Join Queries
4. Views
5. High level programming language extensions (Control structures, Procedures and Functions).
6. Front end Tools
7. Forms
8. Triggers
9. Menu Design
10. Reports
11. Database Design and implementation (Mini Project).

Course Title: Ethical Hacking

Course Code: GCA128

L	T	P	Credits
2	0	0	2

Total Hours: 30

Learning Outcomes: After completion of this course, the learner will be able to:

1. Evaluate new Hacking Methodology.
2. Install hacking software on a closed network environment.
3. Identify tools and techniques to carry out penetration testing.
4. Exemplify security techniques used to protect system and user data.

Course Content

UNIT I **8 Hours**

Introduction to Ethical Hacking: Hacking Methodology, Process of Malicious Hacking, Footprinting and Scanning: Foot printing, Scanning. Enumeration: Enumeration. System Hacking and Trojans: System Hacking, Trojans and Black Box Vs White Box Techniques.

UNIT II **9 Hours**

Hacking Methodology: Denial of Service, Sniffers, Session Hijacking and Hacking Web Servers: Session Hijacking, Hacking Web Servers. Web Application Vulnerabilities and Web Techniques Based Password Cracking: Web Application Vulnerabilities, Web Based Password Cracking Techniques

UNIT III **7 Hours**

Web and Network Hacking: SQL Injection, Hacking Wireless Networking, Viruses, Worms and Physical Security: Viruses and Worms, Physical Security. Linux Hacking: Linux Hacking. Evading IDS and Firewalls: Evading IDS and Firewalls

UNIT IV **6 Hours**

Report writing & Mitigation: Introduction to Report Writing & Mitigation, requirements for low level reporting & high-level reporting of Penetration testing results, Demonstration of vulnerabilities and Mitigation of issues identified including tracking

Transactional Modes

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning**.**

Suggested Readings

- *Karake-Shalhoub, Z., & Al Qasimi, L. (2010). Cyber law and cyber security in developing and emerging economies. Edward Elgar Publishing.*
- *Palmer, C. C. (2001). Ethical hacking. IBM Systems Journal, 40(3), 769-780.*
- *Farsole, A. A., Kashikar, A. G., &Zunzunwala, A. (2010). Ethical hacking. International Journal of Computer Applications, 1(10), 14-20.*

Web Sources

- *https://www.javatpoint.com/ethicalhacking#:~:text=Ethical%20hacking%20involves%20an%20authorized%20attempt%20to%20gain,hackers%20improve%20the%20security%20posture%20of%20an%20organization.*
- *https://www.bing.com/ck/a?!&&p=075f840600d8da28JmltdHM9MTY4MzUwNDAwMCZpZ3VpZD0yYzYwNzgyMS05YmI2LTY0ZDItMDJmNi02OGJmOWEyZDY1NjkmaW5zaWQ9NTIwMw&ptn=3&hsh=3&fclid=2c607821-9bb6-64d2-02f6*

+

Course Title: Internet Concepts and Web Designing
Course Code: GCA111

L	T	P	Credits
3	0	0	3

Total Hours: 45

Learning Outcomes:

After the Completion of the course the learner will be able to

1. Recognize the basic HTML Tags, List, Types of lists, Adding graphics to HTML documents.
2. Design forms with various attributes, Buttons, Text Area and Radio Button.
3. Develop a web site with the help of HTML tags and CSS.
4. Apply the fundamentals of PHP to develop a dynamic website.

Course Content

UNIT I **10 Hours**

Introduction The World Wide Web (WWW) , History, Hypertext and Hypertext Markup Language, Microsoft Front Page, HTML Documents, various Tags.

Elements of an HTML Document: Text Elements, Tag Elements, Special Character Elements Structural elements of HTML documents: Header tags, Body tags, Paragraphs, Titles, Numbered list, Non-Numbered lists, and Definition lists.

Formatting HTML Documents: Logical styles (source code, text enhancements, variables), Physical Styles (Bold, Italic, underlined, crossed).

UNIT II **10 Hours**

Managing images in Html: Image format (quality, size, type), Importing images (scanners), Tags used to insert images, Frames.

Tables in HTML documents Hypertext and Link in HTML Documents, URL/FTP/HTTP

Types of links:Internal Links, External Links, Link Tags, Links with images and buttons, Links that send email messages

UNIT III **12 Hours**

Special effects in HTML documents: Text fonts, Sensitive Images, Tip tables, Page background (Variable, Fixed), Rotating messages (Marquee)

Managing forms: Interactive forms, creating data entry forms

Cascading Style Sheets: ways of inserting a style sheet: External style sheet, Internal style sheet, Inline style.CSS Id and Class, Inheritance in CSS

UNIT IV **13 Hours**

Scripting and websites: Java scripting

PHP: This course is an introduction to the PHP programming language. Topics include installation and configuration with the Apache http server, variables and data types, language syntax, control structures, functions, strategies and tools for handling input and generating output, error handling, sending email, manipulating dates and times, string manipulation and regular expressions, SQL and MySQL database access, object oriented programming (OOP). Though primarily focused on PHP 5.0. We will emphasize security and sound coding practices throughout.

Transactional modes

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning**.**

Suggested Readings

- *Duckett, J. (2014). Web design with HTML, CSS, JavaScript and jQuery set (Vol. 1). IN: Wiley.*
- *Raggett, D., Lam, J., Alexander, I., &Kmiec, M. (1998). Raggett on HTML 4. Addison-Wesley Longman Publishing Co., Inc*

Web Sources

- *https://www.tutorialspoint.com/internet_technologies/website_designing.htm*
- *https://tutorial.techaltum.com/webdesigning.html*
- *https://www.w3schools.com/css/css_intro.asp*
- *https://www.w3schools.com/js/js_operators.asp*
  https://www.codecademy.com/catalog/subject/web-design
- *https://www.entheosweb.com/website_design/responsive_web_design.asp*

Course Title: Data warehousing and Data Mining
Course Code: GCA116

L	T	P	Credits
3	0	0	3

Total Hours: 45

Learning Outcomes:

After the Completion of the course the learner will be able to

1. Understand the functionality of various Data mining techniques.
2. Familiarize yourself with the process of data analysis, identifying the problems, and choosing the relevant models and algorithms to apply.
3. Identify the Classifications & Prediction Data Mining Techniques
4. Compare the classification Techniques.

Course Content

UNIT I **10 Hours**

Data Warehousing: Definition, Characteristics of a Data Warehouse, Data warehouse Usage, DBMS vs. Data warehouse.

Developing Data Warehouse: Data warehousing components, Steps and Crucial decisions for the design and construction of Data Warehouses, Three-tier Data warehouse architecture, Data Warehouse Implementation, Design, performance and technological considerations, Metadata.

UNIT II **10 Hours**

Developing Data Mart based Data warehouse: Types of data marts, Metadata for a data mart, Data model for a data mart, Maintenance of a data mart, Software components for a data mart, Performance issues, Security in data mart.

OLAP Systems: Types of OLAP, Relational vs. Multidimensional OLAP, Data modeling: Star schema, Snowflake schema, OLAP tools.

UNIT III **12 Hours**

Data Mining: Introduction to data mining, Data mining process, Major issues and Application of Data mining, Data preprocessing: Data cleaning, Data integration and transformation and Data reduction; Tools for data mining.

Data Mining Techniques: Association rules: Introduction, Market basket analysis, Frequent Pattern Mining algorithms: Apriori algorithm, Partition algorithm.

UNIT IV **13 Hours**

Classification and Prediction: Definition, Issues regarding Classification and Prediction, Classification by Decision Tree Induction, Support Vector Machines, k-Nearest-Neighbour Prediction: Linear and Non-Linear Regression.

Clustering: Definition, Types of data in cluster analysis, Clustering paradigms: K-Means and K-Medoids, Mining Sequence patterns: Generalized Sequential Patterns(GSP) mining algorithm, Hidden Markov Model, Social Network Analysis.

Transaction Mode

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning**.**

Suggested Readings

- *Inmon, W. H., 2002: Building the Data Warehouse, John Wiley.*
- *Prabhu, C.S.R., 2010 : Data Warehousing, PHI.*
- *Jiawei Han, MichelineKamber, 2000: Data Mining: Concepts and Techniques, Morgan KoffmanElsvier.*
- *Pujari, Arun K, 2013 : Data Mining Techniques, Universities Press*

Web Sources

- *https://www.javatpoint.com/data-mining-cluster-vs-data-warehousing*
- *https://www.ibm.com/topics/data-warehouse*
- *https://www.geeksforgeeks.org/difference-between-data-warehousing-and-data-mining/*
- *https://www.investopedia.com/terms/d/data-warehousing.asp*

Course Title: IOT & Its Applications
Course Code: GCA117

L	T	P	Credits
3	0	0	3

Total Hours: 45

Learning Outcomes:

After the Completion of the course the learner will be able to

1. Identify the different types of sensors and devices used in IoT.
2. Understand the security and privacy challenges associated with IoT.
3. Compare and contrast different IoT platforms and architectures
4. Develop IoT prototypes using hardware and software components.

Course Content

UNIT I **10 Hours**

FUNDAMENTALS OF IoT- Evolution of Internet of Things, Enabling Technologies, M2M Communication, IoT World Forum (IoTWF) standardized architecture, Simplified IoT Architecture, Core IoT Functional Stack, Fog, Edge and Cloud in IoT, Functional blocks of an IoT ecosystem, Sensors, Actuators, Smart Objects and Connecting Smart Objects.

UNIT II **10 Hours**

IoT PROTOCOLS- IoT Access Technologies: Physical and MAC layers, topology and Security of IEEE 802.15.4, 802.11ah and Lora WAN, Network Layer: IP versions, Constrained Nodes and Constrained Networks,6LoWPAN, Application Transport Methods: SCADA, Application Layer Protocols: CoAP and MQTT

UNIT III **12 Hours**

DESIGN AND DEVELOPMENT- Design Methodology, Embedded computing logic, Microcontroller, System on Chips, IoT system building blocks IoT Platform overview: Overview of IoT supported Hardware platforms such as: Raspberry pi, Arduino Board details

UNIT IV **13 Hours**

Data Analytics: Introduction, Structured Versus Unstructured Data, Data in Motion versus Data at Rest, IoT Data Analytics Challenges, Data Acquiring, Organizing in IoT/M2M

Supporting Services: Computing Using a Cloud Platform for IoT/M2M Applications/Services, Everything as a service and Cloud Service Models.

CASE STUDIES/INDUSTRIAL APPLICATIONS: IoT applications in home, infrastructures, buildings, security, Industries, Home appliances, other IoT electronic equipment, and Industry 4.0 concepts.

Transaction Mode

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning**.**

Suggested Readings

- *David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Rob Barton and Jerome Henry,Cisco (2017),IoT Fundamentals: Networking Technologies, Protocols and Use Cases for Internet of Things, Press.*
- *ArshdeepBahga, Vijay Madisetti (2015),Internet of Things – A hands-on approach, Universities Press.*
- *Rajkamal, Internet of Things: Architecture, Design Principles and Applications, McGraw Hill Higher Education.*

Web Sources

- *https://www.javatpoint.com/iot-internet-of-things*
- *https://www.simplilearn.com/tutorials/data-analytics-tutorial/what-is-data-analytics*
- *https://www.tutorialspoint.com/iot-network-protocols*

Course Title: Deep Learning

Course Code: GCA129

L	T	P	Credits
3	0	0	3

Total Hours: 45

Learning Outcomes

After the Completion of the course the learner will be able to

1. Understand the mathematics behind functioning of artificial neural networks.
2. Analyze the given dataset for designing a neural network-based solution.
3. Design and implementation of deep learning models for signal/image processing applications.
4. Able to design and deploy simple Tensor Flow-based deep learning solutions to classification problems

Course Content

UNITI **15 Hours**

Artificial Neural Networks- The Neuron-Expressing Linear Perceptron's as Neurons-Feed-Forward Neural Networks- Linear Neurons and Their Limitations –Sigmoid – Tanh – and ReLU Neurons -Softmax Output Layers – Training Feed-Forward Neural Networks-Gradient Descent-Delta Rule and Learning Rates- Gradient Descent with Sigmoidal Neurons- The Backpropagation Algorithm-Stochastic and Minibatch Gradient Descent – Test Sets – Validation Sets – and Overfitting- Preventing Overfitting in Deep Neural Networks – Implementing Neural Networks in Tensor Flow.

UNITII **8Hours**

Local Minima in the Error Surfaces of Deep Networks- Model Identifiability-Spurious Local Minima in Deep Networks- Flat Regions in the Error Surface – Momentum-Based Optimization – Learning Rate Adaptation.

UNITIII **15 Hours**

Convolutional Neural Networks(CNN) – Architecture -Accelerating Training with Batch Normalization- Building a Convolutional Network using Tensor Flow-Visualizing Learning in Convolutional Networks – Embedding and Representation Learning –Auto encoder Architecture-Implementing an Auto encoder in Tensor Flow –Denoising Sparsity in Auto encoders Models for Sequence Analysis – Recurrent Neural Networks- Vanishing Gradients Long Short-Term Memory (LSTM) Units- Tensor Flow Primitives for RNN Models-Augmenting Recurrent Networks with Attention.

UNIT IV **7 Hours**

Sequence Modeling: Recurrent Nets: Unfolding computational graphs, recurrent neural networks (RNNs), bidirectional RNNs, encoder-decoder sequence to sequence architectures, deep recurrent networks, LSTM networks.

Transaction Mode

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning**.**

Suggested Readings

- Nikhil Buduma, "Fundamentals of Deep Learning: Designing Next-Generation Machine Intelligence Algorithm", O'Reilly, 2017.
- Ian Goodfellow, YoshuaBengio and Aaron Courville, "Deep Learning", MIT Press, 2016.
- Bunduma, N. (2017). Fundamentals of Deep Learning
- Heaton, J.(2015). Deep Learning and Neural Networks, Heaton Research Inc.

Web Sources

- https://sscbs.du.ac.in/wp-content/uploads/2020/05/BHCS-18B-Deep-Learning-Update-Awaited.pdf
- https://www.studocu.com/in/document/jeppiaar-engineering-college/computer-science-and-engineering/1-unit-iv-notes/84195701
- https://www.turing.com/kb/mathematical-formulation-of-feed-forward-neural-network
- https://www.javatpoint.com/perceptron-in-machine-learning

Course Title: Data Network & Security
Course Code: GCA119

L	T	P	Credits
3	0	0	3

Total Hours: 45

Learning Outcomes:

After the Completion of the course the learner will be able to

1. Understand the fundamental concepts of data networks
2. Explain the different network security threats and vulnerabilities
3. Evaluate network security measures and technologies
4. Implement network security controls

Course Content

UNIT I **12 Hours**

Introduction to Computer networks and applications: Network Structure and Architecture, Network Hardware and Software (protocol hierarchies, design issues for layers, interfaces and services: connection oriented and connection less), Network structure and architecture-point to point, multicast, broadcast, Classification of networks on the basis of Geographical Span (PAN, LAN, MAN and WAN) , LAN topologies (Bus, Ring, Star, Mesh, Tree and Hybrid). Network Connecting Devices: Repeaters, Hubs, Bridges, Routers, Gateways and Switches, Network Reference models: OSI model, TCP / IP model. Comparison between OSI and TCP/IP.

UNIT II **10 Hours**

Introduction: Attacks, Services and Mechanisms, Security Attacks, Security Services, Integrity check, digital Signature, authentication, has algorithms.

Secret Key Cryptography: Block Encryption, DES rounds, S-Boxes IDEA: Overview, comparison with DES, Key expansion, IDEA rounds, Uses of Secret key Cryptography; ECB, CBC, OFB, CFB, Multiple encryptions DES.

UNIT III **13 Hours**

Hash Functions and Message Digests: Length of hash, uses, algorithms (MD2, MD4, MD5, SHS) MD2: Algorithm (Padding, checksum, passes.) MD4 and 5: algorithm (padding, stages, digest computation.) SHS: Overview, padding, stages.

Public key Cryptography: Algorithms, examples, Modular arithmetic (addition, multiplication, inverse, and exponentiation) RSA: generating keys, encryption and decryption. Other Algorithms: PKCS, Diffie-Hellman, El-Gamal signatures, DSS, Zero-knowledge signatures.

UNIT IV **10 Hours**

Authentication: Password Based, Address Based, Cryptographic Authentication. Passwords in distributed systems, on-line vs offline guessing, storing. Cryptographic Authentication: passwords as keys, protocols, KDC's Certification Revocation, Inter domain, groups, delegation. Authentication of People: Verification techniques, passwords, length of passwords, password distribution, smart cards, biometrics.

Security Policies and Security Handshake Pitfalls: What is security policy, high and low level policy, user issues? Protocol problems, assumptions, Shared secret protocols, public key protocols, mutual authentication, reflection attacks, use of timestamps, nonce and sequence numbers, session keys, one-and two-way public key based authentication.

Transaction Mode

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning**.**

Suggested Readings

- *Tanenbaum, A. S. (2002). Computer networks. Pearson Education India.*
- *Peterson, L. L., & Davie, B. S. (2007). Computer networks: a systems approach. Elsevier.*
- *Kiesler, S. (1986). The hidden messages in computer networks (pp. 46-47). Harvard Business Review Case Services.*
- *AtulKahate .Cryptography and Network Security ,TMH.*
- *Behourz A Forouzan,Data Communications and Networking*

Web Sources

- *https://www.geeksforgeeks.org/data-communication-definition-components-types-channels/*
- *https://www.studytonight.com/computer-networks/reference-models-in-computer-networks*
- *https://www.bing.com/ck/a?!&&p=2b949258678ed6ceJmltdHM9MTY4MzUwNDAwMCZpZ3VpZD0yYzYwNzgyMS05YmI2LTY0ZDItMDJmNi02OGJmOWEyZDY1NjkmaW5zaWQ9NTIxMQ&ptn=3&hsh=3&fclid=2c607821-9bb6-64d2-02f6-68bf9a2d6569&psq=cOMPUTER+nETWORKS&u=a1aHR0cHM6Ly93d3cuamF2YXRwb2ludC5jb20vY29tcHV0ZXItbmV0d29yay10dXRvcmlhbA&ntb=1*

Course Title: Software Project Management
Course Code: GCA120

L	T	P	Credits
3	0	0	3

Total Hours: 45

Learning Outcomes:

On completion of this course, the learner will able to

1. Identify the different project contexts and suggest an appropriate project management strategy.
2. Practice the role of project planning, risks associated in successful software development.
3. Understand the role of resource allocation and effort estimation in the project management process.
4. Learn to apply the concept of project management and planning to organize team and people's behavior.

Course Content

UNIT I **10 Hours**

Introduction to Software Project Management: Project Definition, Contract Management, Activities Covered by Software Project Management, Overview Of Project Planning, plan methods, methodology. Software project economics – objectives – Project Lie Cycle.

UNIT II **12 Hours**

Software scope and Feasibility – Risk Evaluation – Software estimation – Effort and cost estimation – COCOMO – Resource allocation.

Sequencing and scheduling activities – Objectives of planning – Forward pass and backward pass – Scheduling – PERT techniques – CRM.

UNIT III **11 Hours**

Monitoring and Control: Creating Framework, Collecting the Data, Visualizing Progress, Cost Monitoring, Earned Value analysis, Prioritizing Monitoring, Getting Project Back to Target, and Change Control.

UNIT IV **12 Hours**

Creating Framework – Decision making – cost Monitoring – Types of Risk – Risk managing - Risk Planning and controlling.

Team structure – Project tracking - Managing the contract – change control – Team management – Communication – Software Configuration Management.

Transactional modes

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning.

Suggested Readings

- *Bob Hughes, Mike Cotterell, Software Project Management, Tata McGraw Hill Publishing*

- *Ramesh, GopalaSwamy, Managing Global Projects, Tata McGraw Hill Publishing*
- *Royce, Software Project Management, Pearson Education Publishing*
- *Jalote, Software Project Management in Practice, Pearson Education Publishing*

Web Sources

- *https://www.javatpoint.com/software-project-management*
- *https://www.geeksforgeeks.org/software-engineering-software-project-management-spm/*
- *https://www.wrike.com/project-management-guide/faq/what-is-software-project-management/*
- *https://www.tutorialspoint.com/software_engineering/software_project_management.htm*

SEMESTER II

Course Title: Data Structures
Course Code: GCA201

L	T	P	Credits
4	0	0	4

Total Hours: 60

Learning Outcomes:

On the completion of this course, the learner will be able to

1. Algorithms and algorithm complexity.
2. Attain knowledge of tree and graph concepts.
3. Implement link list and its applications in data structures.
4. Apply the different linear data structures like stack and queue to various computing problems.

Course Content

UNITI **15Hours**

Basic concept and notations: data structures and data structures operations, mathematical notation and functions, algorithmic complexity, Big'O'notations and time space tradeoff.
Arrays: Linear array, representation of linear array in memory, Traversing linear array, insertion and deletion in an array, multi-dimensional array: row-major, column major order, sparse array.

UNIT II **16 Hours**

Stacks: Push and Pop in stack. Representation of stack in memory (linked and sequential) application so f Stack: conversion from infix notation to postfix notations, evolution of postfix notation, matching of Parentheses, recursion, Tower of Hanoi.

UNIT III **14 Hours**

Queue: Queues and Dequeue, Priority Queues, Operations on queues. Linked list: Representation of linked list using static and dynamic data structures, Comparison of Linear and non-lineardata structures, Insertion and deletion of a node from a linear linked list, Introduction to doubly and circular linked lists, Application of linked lists.

UNIT IV **15 Hours**

Searching and Sorting: Linear and binary search, Bubble Sort, Insertion Sort, Selection Sort, Merge Sort, Radix Sort and Quick sort comparison of various searching and sorting algorithms.

Transaction Mode

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning**.**

Suggested Readings

- *Samet, H. (1990).The design and analysis of spatial data structures (Vol.85,p.87).Reading, MA: Addison-Wesley.*
- *Wirth, N.(1985).Algorithms & data structures. Prentice-Hall, Inc.*
- *Samet, H.(1990). Applications of spatial data structures: computer graphics, image processing, and GIS. Addison-Wesley Longman Publishing Co. Inc.*

Web Sources

- *https://www.javatpoint.com/data-structure-introduction*
- *https://www.javatpoint.com/ds-linked-list*
- *https://www.geeksforgeeks.org/array-data-structure/*
- *https://www.programiz.com/dsa/bubble-sort*
- *https://www.geeksforgeeks.org/binary-search-tree-data-structure/*
- *https://www.programiz.com/dsa/bubble-sort*

Course Title: Digital Electronics
Course Code: GCA202

L	T	P	Credits
3	0	0	3

Total Hours: 45

Learning Outcomes

After the completion of his course, the learner will be able to:

1. Solve the conversions of various number systems.
2. Learn the basics of Logic Gates.
3. Analyze and Design various combinational and sequential circuits.
4. Analyzeandpreventvarioushazardsandtimingproblemsinadigitald esign.

Course Content

UNITI **12Hours**

Information Representation: Number systems, Integer and floating pointer presentation, character codes (ASCII, EBCDIC).

Digital IC's: Logic gates, flip-flops, clocks and timers, shift registers, counters.

UNIT II **12Hours**

Boolean Algebra & Circuit Design: Basic laws of Boolean algebra, circuit design using standard (NAND)Gates, Adder, coder/De-multiplexer, encoder/multiplexer design.

UNITIII **11Hours**

MOS & LSI Digital Systems: Semiconductor memory, static and dynamic devices, read only & random-access memory chips, PROMS and EPROMS. Address selection logic. Read and write control timing diagrams for memory ICs.

UNIT IV **10 Hours**

Logical Families: TTL, STTL, CMOS logic families.

Digital Peripherals: Keyboard, multiplexed seven segment display, CRT display schemes, Printers, Control interfaces (parallel and serial) for the peripheral units.

Transaction Mode

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning**.**

Suggested Readings

- *Maini,A. K. (2007).Digital electronics: principles, devices and applications .John Wiley &Sons.*
- *Cook, N.P. (2001). Digital electronics with PLD integration.*
- *Rosen berg, P.(2005).Audel Basic Electronics(Vol.29).John Wiley & Sons*

Web Sources

- *https://www.geeksforgeeks.org/digital-electronics-logic-design-tutorials/*
- *https://www.tutorialspoint.com/digital_circuits/index.htm*
- *https://youtu.be/DBTna2ydmC0*
- *https://youtu.be/XrSgsJ-28Do*
- *https://codescracker.com/digital-electronics/*
- *https://www.tutorialandexample.com/digital-electronics-tutorial*

Course Title: Programming using Python
Course Code: GCA219

L	T	P	Credits
4	0	0	4

Learning Outcomes

After the Completion of this course, the learner will be able to:

1. Understand basic of Python Programming
2. Apply conditional and looping constructs.
3. Learn basic algorithmic problem-solving techniques (decision structures, loops, functions).
4. Know the basics of Strings and Dictionaries of programming.

Course Content

UNITI **15Hours**

Introduction to Python Getting Started: Introduction to Python-an interpreted high-level language, interactive mode and script mode.
Variables, Expressions and Statements: Values, Variables and keywords; Operators and Operands in Python: (Arithmetic, relational and logical operators), operator precedence, Expressions and Statements (Assignment statement); Taking input (using raw input () and input ()) and displaying output (print statement); Putting Comments
Conditional constructs and looping: if else statement While, for (range function), break, continue, else, pass, Nested loops, use of compound expression in conditional constructs and looping

UNITII **15Hours**

Functions: Importing Modules (entire module or selected objects), invoking built in functions, functions from math module, using random () and randint() functions of random module to generate random numbers, composition.
Defining functions, invoking functions, passing parameters, scope of variables, void functions and functions returning values, flow of execution

UNITIII **16Hours**

Strings: Creating, initializing and accessing the elements; String operators: +, *, in, not in, range slice [n:m]; Comparing strings using relational operators; String functions & methods: len, capitalize, find, isalnum, isalpha, isdigit, lower, islower, isupper, upper, lstrip, rstrip, isspace, istitile, partition, replace, join, split,

count, decode, encode, swap case, Pattern Matching.

Lists: Concept of mutable lists, creating, initializing and accessing the elements, traversing, appending, updating and deleting elements; List Operations (joining, list slices); List functions & methods: len, insert, append, extend, sort, remove, reverse, pop

UNIT IV **14 Hours**

Dictionaries: Concept of key-value pair, creating, initializing and accessing the elements in a dictionary, traversing, appending, updating and deleting elements. Dictionary functions & Methods: cmp, len, clear(),get(), has_key() ,items(), keys(), update(), values()

Tuples: Immutable concept, creating, initializing and accessing the elements in a tuple; Tuple functions: cmp(), len(), max(), min(), tuple() Input and Output: Output Formatting, Reading and Writing Files

Errors and Exceptions: Syntax Errors, Exceptions, Handling Exceptions, Raising Exceptions, User-defined Exceptions, Defining Clean-up Actions, Predefined Clean-up Actions

Transaction Mode

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning**.**

Suggested Readings

- *Dawson Michael. Programming with python, Ausers Book Cengage Learning*
- *Beazley Davi. Python EssentialReference,ThirdEdition*

Web Sources

- *https://www.w3schools.com/python/python_syntax.asp*
- *https://www.pythontutorial.net/python-basics/*
- *https://www.geeksforgeeks.org/python-programming-language/*
- *https://www.programiz.com/python-programming*
- *https://www.tutorialspoint.com/python/index.htm*
- *https://www.javatpoint.com/python-functions*
- *https://www.guru99.com/python-tutorials.html*
- *https://www.learnpython.org/*

Course Title: Data Structures Lab using Python
Course Code: GCA220

L	T	P	Credits
0	0	4	2

Total Hours:30

Learning Outcomes

After the Completion of this course, the learner will be able to:

1. Demonstrate proficiency in writing Python programs.
2. Solve the algorithmic problems like insertion and deletion of data.
3. Summarize and describe the flow control structures (conditionals, loops) In Python.
4. UtilizePythonlibrariesandmodulestoextendthefunctionalityoftheirprograms.
5. DebugandfixerorsinPythonprogramsusingappropriatedebuggingtechniques.

Course Content

1. Program to install Python.
2. Program to print Hello Your Name in Python.
3. Program to add numbers and concatenate strings.
4. Program to make input from a user.
5. Programtomakingasumoffirst10naturalnumberthroughLoopsinpython.
6. Python program for class, Flower, that has three instance variables of type str, int, and float that respectively represent the name of the flower, its number of petals, and its price.
7. Python program to implement Method Overloading and Method Overriding.
8. Program for Linear Search and Binary search.
9. Program to implement Bubble Sort and Selection Sort.
10. Program to implement Merge sort and Quick sort.
11. Program to implement Stacks and Queues.
12. Program to implement Singly Linked List.
13. Program to implement Doubly Linked list.
14. Program to implement Binary Search Tree.
15. Program to implement shortest path methods.
16. Program to make Calculator through Functions.
17. Program to delete the file from the system through File Handling

Course Title: Machine Learning
Course Code: GCA210

L	T	P	Credits
3	0	0	3

Total Hours: 45

Learning Outcomes:

After the Completion of the course the learner will be able to

1. Recognize the basic concepts of Bayesian Decision Theory.
2. Apply structured thinking to unstructured problems.
3. Class conditional probability distributions.
4. Apply Multi-Layer Perceptions and Back Propagation learning.

Course Content

UNIT I **10 Hours**

Overview and Introduction to Bayes Decision Theory: Machine intelligence and applications, pattern recognition concepts classification, regression, feature selection, supervised learning class conditional probability distributions, Examples of classifiers bayes optimal classifier and error, learning classification approaches.

UNIT II **12 Hours**

Linear machines: General and linear discriminates, decision regions, single layer neural network, linear reparability, general gradient descent, perception learning algorithm, mean square criterion and widrow-Hoff learning algorithm; multi-Layer perceptions: two-layers universal approximates, back propagation learning, on-line, off-line error surface, important parameters.

UNIT III **11 Hours**

Learning decision trees: Inference model, general domains, symbolic decision trees, consistency, learning trees from training examples entropy, mutual information, ID3 algorithm criterion, C4.5 algorithm continuous test nodes, confidence, pruning, learning with incomplete data

Instance-based Learning: Nearest neighbor classification, k-nearest neighbor, nearest neighbor error probability

UNIT IV **12 Hours**

Machine learning concepts and limitations: Learning theory, formal model of the learnable, sample complexity, learning in zero-bayes and realizable case, VC-dimension, fundamental algorithm independent concepts, hypothesis class, target class, inductive bias, occam's razor, empirical risk, limitations of inference machines, approximation and estimation errors, Trade off.

Transaction Mode

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning**.**

Suggested Readings

- *Zhang, C., & Ma, Y. (Eds.). (2012). Ensemble machine learning: methods and applications. Springer Science & Business Media.*
- *Marsland, S. (2011). Machine learning: an algorithmic perspective. Chapman and Hall/CRC..*
- *C. M. Bishop.Pattern Recognition and Machine Learning, Springer, (2006).*

Web Sources

- *https://www.geeksforgeeks.org/machine-learning/*
- *https://www.javatpoint.com/machine-learning*
- *https://www.w3schools.com/python/python_ml_getting_started.asp*
- *https://www.simplilearn.com/tutorials/machine-learning-tutorial*
- *https://www.tutorialspoint.com/machine_learning/index.htm*
- *https://www.kaggle.com/learn/intro-to-machine-learning*

Course Title: Data Visualization
Course Code: GCA221

L	T	P	Credits
3	0	0	3

Total Hours:45

Learning Outcomes

After the Completion of this course learner will be able to

1. Build and maintain reliable, scalable, distributed systems with Apache Hadoop
2. Understand Spark framework and explore various ML tools for data processing
3. Apply HIVEQL, PIG techniques to solve big data queries
4. Understand conventional SQL query language and No SQL
5. Design, build and query Mongo DB
6. Visualize big data to perform decision making in real world problems

UNIT I **15 Hours**

Introduction to Big Data: Distributed file system– Big data and its importance, 3Vs of Data Volume, Velocity and Variety, Data sets, Data analysis, Data analytics, Business intelligence, KPI, Big data characteristics, Different types of data, Drivers for big data adoption. Big Data Analysis Techniques: Quantitative analysis, Qualitative analysis, Data mining, Statistical analysis, Machine learning, Semantic analysis, Visual analysis, Case studies.

UNIT II **8 Hours**

Hadoop Architecture: Overview of Distributed database Systems, Hadoop eco-system, Hadoopcore components, Hadoop distributions, Developing enterprise applications with Hadoop. Storing Datain Hadoop: Moving data in and out of Hadoop, HDFS architecture, HDFS files, Hadoop specific file types, HDFS federation and high availability, working with HDFS Commands, Fundamentals of HBASE, Zookeeper concepts and methods to build applications with Zookeeper.

UNIT III **15 Hours**

Introduction to SPARK: Introduction to Data Analysis with Spark, Downloading Spark and Getting Started, Programming with RDDs, Machine Learning with MLlib. HIVE, HIVQL and PIG: HIVE: Architecture and installation, Comparison with traditional database, HIVQL querying data, Sorting and aggregating, Joins & sub queries, HIVEVs PIG, PIG: Architecture and installation, Execution Mechanisms, load/store operator, Pig scripts.

UNIT IV **7 Hours**

No SQL and Mongo DB: Introduction, Types of NoSQL databases, Advantages of No SQL, Use of No SQL in industry, SQL VS No SQL, Mongo DB: Mongo DB Support for dynamic queries, Replications, Sharding, Create Database and Drop Database, Collections and Documents, MongoDB Query Language.

Transaction Mode

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning**.**

Suggested Readings

- Borislublinsky,Kevint.Smith,AlexeyYakubovich,"*ProfessionalHadoopSolutions*",Wiley, ISBN: 9788126551071, 2015
- ThomasErl*," BigDataFundamentals-Concepts,DriversandTechniques"*,Pearson publication,2016
- KyleBanker,PiterBakkum,ShaunVerch,"*MongoDBinAction",*SecondEdition,Dreamtech Press
- TomWhite,"*HADOOP:ThedefinitiveGuide*", O Reilly2012
- AlainF. Zuur,ElenaN. Ieno,ErikH.W.G.Meesters, "*Beginner'sGuidetoR*",Springer2009

Web Sources

- https://www.techtarget.com/whatis/definition/3Vs
- https://www.geeksforgeeks.org/hadoop-architecture/
- https://www.oreilly.com/library/view/learning-spark/9781449359034/ch01.html
- https://www.mongodb.com/resources/basics/databases/nosql-explained/nosql-vs-sql

Course Title: Natural Language Processing

Course Code: GCA222

L	T	P	Credits
3	0	0	3

Total Hours: 45

Learning Outcomes

After the completion of this course, the learner are expected to Develop interactive augmented reality applications for both PC based mobile devices using a variety of novel input devices

1. In depth understanding of the computational properties of natural languages and the commonly used algorithms for processing linguistic information
2. The course examines NLP models and algorithms using both the traditional symbolic and the more recent statistical approaches

UNIT I **15 Hours**

Introduction: Knowledge in speech and language processing, Ambiguity, Models and Algorithms, Brief History. Regular Expressions and Automata, Morphology and Transducers: Inflectional and derivational morphology, finite state morphological parsing, Combining FST Lexicon and rules. Lexicon free FST: Porter Stemmer N-grams: Counting Words in Corpora, Simple Unsmoothed n-grams, Smoothing, Entropy HMM and Speech Recognition: Speech Recognition Architecture, Overview of HMM, decoding.

UNIT II **15 Hours**

World Classes and Part-of-Speech Tagging: English word classes, Targets for English, Part of Speech tagging, Rule based part of speech Tagging, Transformation based tagging. Context Free Grammars for English: Constituency, Context Free rules and Trees, Sentence level construction, The Noun Phrase, Coordination, Agreement, The verb phrase and sub categorization. Spoken Language Syntax, Grammar Equivalence and Normal form, Finite state context free grammars, Grammar and human processing.

UNIT III **8 Hours**

Parsing with context free grammars: Parsing as Search, basic Top down Parser, Problems with basic top-down-parsers, the early Algorithm, Finite state parsing method. Features and Unifications: Feature Structures, Unification of Features Structures, Features Structures in the grammar, Implementing Unification. Lexicalized and probabilistic parsing: Probabilistic context free grammars, problems with probabilistic context free grammars

UNIT IV **7 Hours**

Semantics: (Representing Meaning): Computational Desiderata for representation, meaning structure of language, First order predicate calculus, linguistically relevant concept, Related Representational approaches, and Alternative

approaches to meaning. Semantic Analysis: Syntax driven semantic analysis, Attachment of Fragment of English, Robust Semantic Analysis Lexical Semantics: Relation among lexemes and their senses, Internal Structure of words

Transactional Mode

Project based learning, Team Teaching, flipped teaching, Open talk, Collaborative Teaching, Case Analysis, Panel Discussions, Group Discussions

Suggested Readings

- Speech and Language processing an introduction to Natural Language Processing, Computational Linguistics and speech Recognition by Daniel Jura sky and James H. Martin
- Natural Language Processing with Python by Steven Bird, Ewan Klein, Edward Lopper
- Handbook of Natural Language Processing, Second Edition—NitinIndurkhya, Fred J.Damerau,Fred J. Damerau

Web Sources

- https://www.scribd.com/document/475855485/NLP-Notes-Ch1-5-pdf
- https://www.geeksforgeeks.org/nlp-part-of-speech-default-tagging/
- https://ebooks.inflibnet.ac.in/csp10/chapter/top-down-parser-pre-processing/
- https://www.scaler.com/topics/nlp/computational-desiderata-for-representations/

Course Title: Software Engineering & Testing
Course Code: GCA216

L	T	P	Credits
3	0	0	3

Total Hours: 45

Learning Outcomes:

After the Completion of the course the learner will be able to

1. Analyze and model customer's requirements and model its software design.
2. Estimate cost and efforts required in building software.
3. Analyze and compute impact of various risks involved in software development.
4. Design and build test cases, and to perform software testing.

Course Content

UNIT I **12 Hours**

Introduction: Software Engineering – A Layered Approach; Software Process – Process Framework, Umbrella Activities; Process Models – Waterfall Model, Incremental Model, and Evolutionary process Model (Prototyping, Spiral Model); Introduction to Agile – Agility Principles, Agile Model – Scrum.

Software Requirements Analysis and Specifications: Use Case Approach, Software Requirement Specification Document, Flow oriented Modeling, Data Flow Modeling, Sequence Diagrams.

UNIT II **11 Hours**

Design Modeling: Translating the Requirements model into the Design Model, The Design Process, Design Concepts – Abstraction, Modularity and Functional Independence; Architectural Mapping using Data Flow.

Software Metrics and Project Estimations: Function based Metrics, Software Measurement, Metrics for Software Quality; Software Project Estimation (FP based estimations, COCOMO II Model); Project Scheduling (Timeline charts, tracking the schedule).

UNIT III **12 Hours**

Quality Control and Risk Management: Quality Control and Quality Assurance, Software Process Assessment and Improvement Capability Maturity Model Integration (CMMI); Software Risks, Risk Identification, Risk Projection and Risk Refinement, Risk Mitigation, Monitoring and Management.

UNIT IV **10 Hours**

Testing and maintenance: Software Testing Techniques, Software testing fundamentals: objectives principles, testability; test case design, Unit testing: white box testing, basis path testing: Control structure testing: Black box testing,

testing for specialized environments, Software Reliability and Quality Assurance: Quality concepts, Software quality assurance: SQA activities; Software reviews; cost impact of software defects, defect amplification and removal; formal technical reviews: The review meeting, review reporting record keeping, review guidelines; Formal approaches to SQA;

Transaction Mode

Lecture Method, E-Team Teaching, Video based learning, Demonstration, Peer Discussion, Open talk, Cooperative Teaching, Flipped Teaching, Collaborative Learning**.**

Suggested Readings

- *Pressman Roger S, Software Engineering - A Practitioner's Approach, MGH, New Delhi, New Delhi. Publications, New Delhi.*
- *Ian Sommerville, Software Engineering, Pearson Education, 5th Edition, New Delhi*
- *Jalote Pankaj, An Integrated Approach to Software Engineering, NarosaPublications,New Delhi.*
- *Mall Rajib, Fundamentals of Software Engineering, PHI, New Delhi.*
- *Ali Bethforooz, Frederick J. Software Engineering Fundamentals, Hudson Oxford University.*

Web Sources

- *https://www.tutorialspoint.com/software_engineering/index.htm*
- *https://www.javatpoint.com/software-engineering*
- *https://www.geeksforgeeks.org/software-engineering/*
- *https://www.tutorialsduniya.com/notes/software-engineering-notes/*
- *https://ecomputernotes.com/software-engineering/levels-of-software-testing*

Course Title: Data Visualization Lab
Course Code: GCA223

L	T	P	Credits
3	0	0	3

Total Hours:45

Learning Outcomes

After the completion of his course, the learner will be able to:

1. Use Python, R and Tableau for data visualization
2. Apply data visuals to convey trends in data over time using tableau
3. Construct effective data visuals to solve workplace problems
4. Explore and work with different plotting libraries
5. Learn and create effective visualizations

Course Content

1. Introduction to various Data Visualization tools
2. Basic Visualization in Python
3. Basic Visualization in R
4. Introduction to Tableau and Installation
5. Connection to Data and preparing data for visualization in Tableau
6. Data Aggregation and Statistical functions in Tableau
7. Data Visualizations in Tableau
8. Basic Dashboards in Tableau
9. Program for Word Count Using Map Reduce Programming
10. Program in Map Reduce for Group Sum operation.
11. Program in Map Reduce for Matrix Multiplication
12. Program in Map Reduce for Intersection operation.
13. Program in Map Reduce for Union operation.

Course Title: Natural Language Processing Lab
Course Code: GCA224

L	T	P	Credits
3	0	0	3

Total Hours:45

Learning Outcomes

After the Completion of this course, the learner will be able to:

1. Understand formal methods of knowledge representation, logic and reasoning
2. Understand foundational principles, mathematical tools and program paradigms of Artificial intelligence
3. Formulate NLP tasks as learning and inference tasks, and address the computational challenges involved

Course Content

1. To Change the text into lower case
2. To Search digit pattern
3. To Remove URLs
4. Remove punctuation
5. Spelling Correction
6. Removing Stop Words (a, the, of, are, my)
7. Remove Numbers
8. Remove whitespace from text
9. Sentence Tokenization
10. Word Tokenization
11. Checking the type and number of tokens
12. Frequency of tokens
13. Bigrams, ngrams
14. Stemming
15. Lemmatization
16. Parts of Speech Tagging