PERIYAR UNIVERSITY

(NAAC A^{++} GRADE–STATE UNIVERSITY–NIRF RANK 63 ARIIA 10th RANK)

Salem-636 011

DEPARTMENT OF COMPUTER SCIENCE
(Supported by UGC-SAP)

M.Sc. DEGREE COMPUTER SCIENCE

[Choice Based Credit System (CBCS)]

OBE REGULATIONS AND SYLLABUS

(Effective from the academic year 2022-2023 and thereafter)

M. Sc. COMPUTER SCIENCE

OBE REGULATIONS AND SYLLABUS

(With effect from the academic year 2022-2023 onwards)

1. **Preamble**

   To Develop the Post Graduates in **Computer Science** with deep knowledge of theoretical Computer Science subjects who can be employed in research and development units of industries and academic institutions.

2. ***General Graduate Attributes***

   GA1 : **Apply Mathematical Knowledge**

   Graduates will be able to apply mathematics, and statistics to the design and development of software systems.

   GA 2 : Develop Softwares based on Software Engineering principles

   Graduates will be able to design and develop computer software systems based on the acquired knowledge in Programming Languages and based on Software Engineering.

   GA 3 : Develop Research skills

   Graduates will be able to exhibit the research skills in various areas and update their skills based on recent advances in research field.

   GA 4 : Understanding of Profession Ethics

   Graduates will exhibit an understanding of professional ethics and the roles of regulations and guidelines in the profession.

   GA 5: Solve computer science problems

   Be equipped with a range of fundamental principles of Computer Science that will provide the basis for future learning and enable them to adapt to the constant rapid development of the field. Be able to apply mathematics, logic, and statistics to the design, development, and analysis of software systems.

   GA 6: To apply algorithmic principles

   To identify the key intellectual themes of the field in algorithmic thinking, information representation, and computer programs.GA 7:

   GA 7 :To acquire the latest technical skills

   To enable the students to acquire the latest technical skills and build their carrier on the basis of continuous learning and adaptability.

GA 8: Leadership, initiative and teamwork:

To inculcate the Ability to work effectively in a team and lead in multidisciplinary environment.

GA 9: Kindle Creativity

To demonstrate critical thinking, imagination an intellectual agility and strive to be innovative and experimental in advancing knowledge and in creating solutions.

GA 10: Enhance the knowledge in Specialization area:

To enhance the knowledge in their specialist area and apply Analytical approach to identify and resolve problems.

GA 11: Persuade Intellectual Rigour

An ability to think clearly and deeply with rigour when faced with new knowledge and arguments and demonstrate the ability to apply research results to solve problems.

GA 12: Communication and social skills

To impart Good communication and social skills to widen the ability to listen to, as well as clearly express, information back to others in a variety of ways: oral, written, and visual - using a range of technologies.

3. **Programme Specific Qualification Attributes**

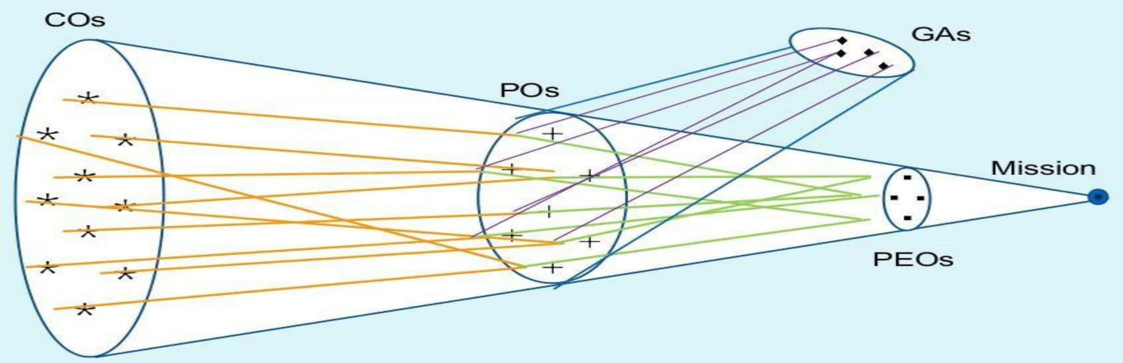
Mention the programme specific qualification attributes achieved through courses in the programme in terms of

- **Knowledge and understanding level (K1 and K2)**
  - Remember or recognize a term or a basic concept
  - Select an explanation for a statement related to the question topic
  - Understand the existing problems
- **Application level (K3)**
  - Be able to solve the problems using computing techniques.
- **Analytical level (K4)**
  - Be able to separate information related to a procedure or technique into its constituent parts for better understanding and can distinguish between facts and inferences.
- **Evaluation capability level (K5)**
  - Be able to make judgments based on criteria and standards. Detects inconsistencies or fallacies within a process or product, determines whether a process or product has internal consistency and detects the effectiveness of a procedure as it is being implemented.
- **Scientific or synthesis level (K6)**
  - A scientific way to analyze and solve the problems.
  -

4. Vision

Achieving excellence in Information Technology Enabled Services through Teaching, Research, Extension and Consultancy.

It must be linked like this



Mission is the Programme Specific Objectives, GAs – Graduate attributes (general)

5. Programme Objectives and Outcomes

Spelt the PEOs (Programme Educational Objectives), Programme Specific Objectives (PSOs) and Programme Outcomes (POs)

Programme Educational Outcomes (PEOs) for M. Sc Computer Science are as follows

PEO1: Apply algorithmic, mathematical and scientific reasoning to a variety of computational problems

PEO2: Implement software systems that meet specified design and performance requirements.

PEO3: Work effectively in teams to design and implement solutions to computational problems

PEO4: Communicate effectively, both orally and in writing. Design, correctly implement and document solutions to significant computational problems

Programme Specific Outcomes (PSOs) for M. Sc Computer Science are as follows:

PSO1: An ability to apply profound knowledge to analyze and design software and systems containing hardware and software components of varying complexity.

PSO2: An ability to apply mathematical model, algorithmic principles, and computer science theory in the design of real-time applications

Programme Outcomes (POs) for M. Sc Computer Science are as follows

PO1: Computational Knowledge
Gain knowledge in the theoretical foundations of Computer Science, Computing Fundamentals and Basic Mathematics.

PO2: Problem Analysis
To analyze and identify the customer requirements in multidisciplinary domains, create high level design and implement robust software applications using latest technological skills.

PO3: Design and Development
Design and develop solutions for complex problems in various domains. Serve as the Programmers or the Software Engineers with the sound knowledge of practical and theoretical concepts for developing software.

PO4: Research Activity
To understand the fundamentals of research and Inculcate the ability to undertake original research at the cutting edge of computer science & its related areas. Produce researchers who can investigate problems in different application domains and creatively develop, and evaluate computational solutions.

PO5: Software tool usage

To adapt and apply modern computing skills and tools to resolve problems with software development tools, software systems, and modern computing platforms.

PO6: Professional ethics
To understand professional ethics and Cyber regulations and develop youth with social commitments.

PO7: Personality development
To understand Management Principles and apply the principles to develop software as a team member and mange projects efficiently for multidisciplinary environments.

PO8: Communication and Presentation Efficacy
Communicate effectively with computing society in both verbal and written form. Improve communication and presentation skills, especially in providing technical support.

PO9: Social Responsibility
To access Social and Environmental issues for local and global needs and give relevant solutions for them. Gained the analytical ability to analyze the literature and social issues to appreciate the strength and to suggest the improvements for better results.

PO10: Entrepreneurship
Discover the opportunity for entrepreneurship and create and add value for the betterment of an individual and society at large.

PO11: Algorithmic principles and theory
An ability to apply mathematical foundations, algorithmic principles, and Computer Science theory in the modeling and design of computational systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

PO12: Team work
Solve the problems (programming networking database and Web design) in the Information Technology environment. Function effectively on teams to accomplish a common goal and demonstrate professional behavior.

PEO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
PEO1	▨	▨	▨	▨	▨		▨	▨		▨		
PEO2	▨	▨	▨	▨					▨		▨	▨
PEO3	▨	▨	▨		▨	▨	▨	▨		▨	▨	
PEO4	▨	▨	▨		▨	▨	▨	▨		▨	▨	

PO-GA Mapping

	GA1	GA2	GA3	GA4	GA5	GA6	GA7	GA8	GA9	GA10	GA11	GA12
PO1	▨											
PO2		▨										
PO3			▨									
PO4				▨								
PO5					▨							
PO6						▨						
PO7							▨					
PO8								▨				
PO9									▨			
PO10										▨		
PO11											▨	

PO12												

6. Candidate's eligibility for admission

A candidate who has passed B.Sc. Computer Science/B.C.A/B.Sc. Computer Technology / B.Sc. Information Science/Technology degree of this University or any of the degree of any other University accepted by the syndicate as equivalent thereto subject to such conditions as may be prescribed therefore shall be permitted to appear and qualify for the **M.Sc. Computer Science** degree examination of this University after a course of study of two academic years.

7. Duration of the programme

The programme for the degree of **Master of Science in COMPUTER SCIENCE** shall consist of **two Academic years** divided into four semesters. Each semester consist of 90 working days.

8. CBCS- Structure of the Programme

The programme structure comprises of two parts.
Option I :

Course Component	No. of Courses	Hours of Learning/ Week	Marks	Credits
Part A(Credit Courses)				
Core Courses	9	36	900	36
Elective Courses	4	12	400	12
Supportive Courses	1	04	100	04
Core –Practical	8	22	800	15
Elective - Practical	4	08	400	04
Mini-Project	1	02	100	03
Project	1	-	200	14
Fundamentals of Human Rights	1	02	100	02
Credit Seminar	1	02	100	02
Online Courses	1	01	100	02
Total	31	89	3200	94

Option II :

Course Component	No. of Courses	Hours of Learning Week	Marks	Credits
Part A(Credit Courses)				

Core Courses	10	40	1000	40
Elective Courses	04	12	400	12
Supportive Courses	01	04	100	04
Core –Practical	08	22	800	15
Elective - Practical	04	08	400	04
Mini-Project	01	02	100	03
Project	01	14	200	10
Fundamentals of Human Rights	01	02	100	02
Credit Seminar	01	02	100	02
Online Courses	01	01	100	02
Total	31	89	3200	94

Core Courses (CC): CURRICULUM AND SCHEME OF EXAMINATIONS
Two year M. Sc - Computer Science Programme
2022-23 onwards

Course	Number of Credits	Hours Per Week	Examination Duration (hrs)	Marks		
				I.A	ESE	Total
Semester - I						
Core Theory **22UPCSC2C01/ 22UPCSC1C01** Discrete Mathematics	4	4	3	25	75	100
Core Theory **22UPCSC2C02/ 22UPCSC1C02** Linux and Shell Programming	4	4	3	25	75	100
Core Theory **22UPCSC2C03/ 22UPCSC1C03** Python Programming	4	4	3	25	75	100
Core Theory **22UPCSC2C04** Theory of Computation	4	4	3	25	75	100
Core Lab **22UPCSC2C05** Linux and Shell Programming - Lab	2	3	3	40	60	100
Core Lab **22UPCSC2C06** Python Programming Lab	2	3	3	40	60	100

Core Lab **22UPCSC2C07** Soft Skill Development - Lab	2	2	3	100		100
Elective-01 Theory	3	3	3	25	75	100
Elective-01 Lab	1	2	3	40	60	100
	26	29				900

Course	**Number of Credits**	**Hours Per Week**	**Examination Duration (hrs)**	**Marks**		
				I.A	**ESE**	**Total**
Semester - II						
Core Theory **22UPCSC2C08** Design and Analysis of Algorithms	4	4	3	25	75	100
Core Theory **22UPCSC2C09** Big Data Analytics	4	4	3	25	75	100
Core Lab **22UPCSC2C10** Design and Analysis of Algorithms - Lab	2	3	3	40	60	100
Core Lab **22UPCSC2C11** Big Data Analytics Lab	2	3	3	40	60	100
Core Lab **22UPCSC2C12** Mini Project	3	2	3	40	60	100
22UPSOC2H01 Fundamentals of Human Rights	2	2	3	25	75	100
Elective-02 Theory	3	3	3	25	75	100
Elective-02 Lab	1	2	3	40	60	100
Elective-03 Theory	3	3	3	25	75	100
Elective-03 lab	1	2	3	40	60	100
SWAYAM	2	1			100	100
	27	**29**				**1000**

Course	Number of Credits	Hours Per Week	Examination Duration (hrs)	Marks		
				I.A	**ESE**	**Total**
Semester - III						
Core Theory **22UPCSC2C13** Advanced Java Programming	4	4	3	25	75	100
Core Theory **22UPCSC2C14** Web Technologies	4	4	3	25	75	100
Core Theory **22UPCSC2C15** Advanced Machine Learning Technologies	4	4	3	25	75	100
Core Lab **22UPCSC2C16** Advanced Java Programming-Lab	2	3	3	40	60	100
Core Lab **22UPCSC2C17** Web Technologies Lab	2	3	3	40	60	100
Core Lab **22UPCSC2C18** Advanced Machine Learning Lab	1	2	3	40	60	100
Supportive Course	4	4	3	25	75	100

Elective-04 Theory	3	3	3	25	75	100
Elective-04 lab	1	2	3	40	60	100
	25	**30**				**900**

Course	**Number of Credits**	**Hours Per Week**	**Examination Duration (hrs)**	**Marks**		
				I.A	**ESE**	**Total**
Semester - IV						
Option - 1						
Core **22UPCSC2C19** **(Option 1)** Credit Seminar	2	2	-	100		100
Core **22UPCSC2C20** **(Option 1)** Major Project and Viva-Voce	14	-	-	50	150	200
Total	16	2				300
	94					3100
Option - 2						
Core **22UPCSC2C19** **(Option 2)** Credit Seminar	2	2	-	100		100
Core Theory **22UPCSC2C20** **(Option 2)** Natural Language Processing	4	4	3	25	75	100
Core **22UPCSC2C21** **(Option 2)** Project and Viva-Voce	10	14	-	50	150	200
	16	20				400

Total	94					3200

LIST OF ELECTIVE COURSES FOR M. Sc Computer Science

Elective – 1

22UPCSC2E01 Data Engineering and Management
22UPCSC2E02 Data Engineering and Management Lab
22UPCSC2E03 Architecture and Frameworks
22UPCSC2E04 Architecture and Frameworks lab
22UPCSC2E05 Network Protocols
22UPCSC2E06 Network Protocols Lab
22UPCSC2E07 Dot Net Technologies
22UPCSC2E08 Dot Net Technologies lab

Elective – 2

22UPCSC2E09 Software Development Technologies
22UPCSC2E10 Software Development Technologies Lab
22UPCSC2E11 Soft Computing
22UPCSC2E12 Soft Computing Lab
22UPCSC2E13 Social Networks
22UPCSC2E14 Social Networks Lab
22UPCSC2E15 High Performance Computing
22UPCSC2E16 High Performance Computing Lab

Elective – 3

22UPCSC2E17 Internet of Things
22UPCSC2E18 Internet of Things Lab
22UPCSC2E19 Computer Vision
22UPCSC2E20 Computer Vision Lab
22UPCSC2E21 Cryptography and Network Security
22UPCSC2E22 Cryptography and Network Security Lab
22UPCSC2E23 Mobile Computing
22UPCSC2E24 Mobile Application Development Lab

Elective – 4

22UPCSC2E25 Cyber Security
22UPCSC2E26 Cyber Security Lab
22UPCSC2E27 Block chain Technologies
22UPCSC2E28 Block chain Technologies Lab
22UPCSC2E29 Optimization Techniques
22UPCSC2E30 Optimization Techniques Lab
22UPCSC2E31 Solution Architecture
22UPCSC2E32 Solution Architecture Lab

Option I

The students / Learners those who want to obtain the certificate for the job role / occupation in the name of Analytics Qualification Pack SSC/Q8108 of IT / ITeS Sector Skill Council should appear for the assessment to be conducted at the end of the fourth semester by the IT/ITeS Sector Skill Council of National Skill Development Corporation. In-order to get the above certificate, the students should choose the electives as suggested below:

Elective – 1

22UPCSC2E01 Data Engineering and Management
22UPCSC2E02 Data Engineering and Management Lab

Elective – 4

22UPCSC2E31 Solution Architecture
22UPCSC2E32 Solution Architecture Lab

The elective course contents cover the required contents in addition to the core courses such as Big Data Analytics, Advanced Machine Learning Techniques (Artificial Intelligence) and Soft Skill Development for the Qualification Pack SSC/Q8108 of IT / ITeS Sector Skill Council of NSDC.

Option II

The students / Learners those who want to obtain the certificate for the job role / occupation in the name of Analytics Qualification Pack SSC/Q0923 of IT / ITeS Sector Skill Council should appear for the assessment to be conducted at the end of the fourth semester by the IT/ITeS Sector Skill Council of National Skill Development Corporation. In-order to get the above certificate, the students should choose the electives as suggested below:

Elective – 1

22UPCSC2E05 Network Protocols
22UPCSC2E06 Network Protocols Lab

Elective – 3

22UPCSC2E21 Cryptography and Network Security
22UPCSC2E22 Cryptography and Network Security Lab

Elective – 4

22UPCSC2E25 Cyber Security
22UPCSC2E26 Cyber Security Lab

The elective course contents cover the required contents in addition to the core Soft Skill Development for the Qualification Pack SSC/Q8108 of IT / ITeS Sector Skill Council of NSDC.

09. Examinations

Examinations are conducted in semester pattern. The examination for the Semester I& III will be held in November/December and that for the Semester II and IV will be in the month of April/May.

Candidates failing in any subject (theory, practical and skill) will be permitted to appear for such failed subjects in the same syllabus structure at subsequent examinations with in next 3years after normal completion of the programme. Failing which, the candidate has to complete the course in the present existing syllabus structure.

10. Scheme for Evaluation and Attainment Rubrics

Evaluation will be done on a continuous basis and will be evaluated four times during the course work. The first evaluation will be in the 7th week, the second in the 11thweek, third in the 16thweek and the end– semester examination in the 19thweek. Evaluation maybe by objective type questions, short answers, essay so recombination of these, but the end semester examination is a University theory examination with prescribed question paper pattern.

Attainment of Rubrics for Theory Courses

THEORY EXAMINATION

Evaluation of Internal

Assessment

Test	:5 Marks (Best one out of Two Tests)
Model Examination	:5 Marks
Seminar	:5 Marks
Assignment	:5 Marks
Attendance	:5 Marks

Total:	25Marks

*** No Internal Minimum

Evaluation of End Semester Examinations

Question Paper Pattern (Theory)

Section	Question Pattern	Mark Pattern	K Level	CO Coverage
A	One word (Answer all questions)	20X1 = 20 (Multiple Choice Questions)	K1-K2	CO1-CO5
B	100 to 200 words (Answer any three out of five questions)	3X5 = 15 (Analytical type questions)	K4-K6	CO1-CO5
C	500 to 1000 words	5X8 = 40 (Essay type questions)	K2-K3	CO1-CO5

Attainment Rubrics for Lab Courses

PRACTICAL \ MINI PROJECTEXAMINATION

Evaluation of Internal Assessment

Test 1 : 20Marks

Test 2 : 20 Marks (Best one out o Two Tests)

Test 3 : 20Marks

---------

Total : 40 Marks

---------

*** No Internal Minimum

QUESTION PAPER PATTERN

Time duration: 3 Hours

Max. Marks: 60

Two Questions may be taken from the list of practical problems: 60 Marks

Distribution of the Marks

(i)Practical/Mini project

- Record Note Book- 10
- Problem Understanding- 10
- Implementation- 20
- Debugging and Modification- 10
- For correct output and viva - 10

(ii)Industrial Training

- Internal Assessment- 40
- Joint Viva-voce - 60

(Internal Examiner 30 and External Examiner 30) (iii)Dissertation

- Internal Assessment- 50
- Report Evaluation by External Examiner- 50
- Joint Viva-voce - 100

(Internal Examiner 50 and External Examiner 50)

REGULATIONS FOR DISSERTATION WORK

- Students should attach themselves with well reputed Industry/Company/Institutions to do their five months dissertation work.
- The Candidate should submit the filled in format to the department for approval during the First week of December during the even semester.
- The review of the dissertation will be carried out periodically.
- The student should submit three copies of their dissertation work.
- The students may use OHP/Power Point presentation during their dissertation Viva-Voce examinations.

PASSING MINIMUM

The candidate shall be declared to have passed in the theory/practical/Dissertation examination if the candidate secures:

(i) 50% marks in the ESE and

(ii) 50% in ESE and IA put together

11. Grading System

Evaluation of performance of students is based on ten-point scale grading system as given below.

Ten Point Scale			
Grade of Marks	Grade points	Letter Grade	Description
90-100	9.0-10.0	O	Outstanding
80-89	8.0-8.9	D+	Excellent
75-79	7.5-7.9	D	Distinction
70-74	7.0-7.4	A+	Very Good
60-69	6.0-6.9	A	Good
50-59	5.0-5.9	B	Average
00-49	0.0	U	Re-appear
ABSENT	0.0	AAA	ABSENT

Course Code: 22UPCSC2C01 / 22UPCSC1C01 Credits: 4

Discrete Mathematics

Course Objective

- To know the concepts of relations and functions
- To distinguish among different normal forms and quantifiers
- To solve recurrence relations and permutations & combinations
- To know and solve matrices , rank of matrix & characteristic equations
- To study the graphs and its types

Unit-I

Relations- Binary relations-Operations on relations- properties of binary relations in a set – Equivalence relations–– Representation of a relation by a matrix - Representation of a relation by a digraph – **Functions**-Definition and examples-Classification of functions-Composition of functions-Inverse function

Unit-II

Mathematical Logic-Logical connectives-**Well formed formulas** – Truth table of well formed formula –Algebra of proposition –Quine's method- **Normal forms of well formed formulas**- Disjunctive normal form-Principal Disjunctive normal form-Conjunctive normal form-Principal conjunctive normal form-**Rules of Inference for propositional calculus** – **Quantifiers**- Universal Quantifiers-Existential Quantifiers

Unit-III

Recurrence Relations- Formulation -solving recurrence Relation by Iteration-solving Recurrence Relations- Solving Linear Homogeneous Recurrence Relations of Order Two- Solving Linear Non homogeneous Recurrence Relations. **Permutations**-Cyclic permutation- Permutations with repetitions- permutations of sets with indistinguishable objects- **Combinations**- Combinations with repetition

Unit-IV

Matrices- special types of matrices-Determinants-Inverse of a square matrix-Cramer's rule for solving linear equations-Elementary operations-Rank of a matrix-solving a system of linear equations-characteristic roots and characteristic vectors-Cayley - Hamilton Theorem-problems

Unit-V

Graphs -Connected Graphs -Euler Graphs- Euler line-Hamiltonian circuits and paths –planar graphs – Complete graph-Bipartite graph-Hyper cube graph-Matrix representation of graphs

Text book

1. N.Chandrasekaran and M.Umaparvathi, Discrete mathematics, PHI Learning Private Limited, New Delhi, 2015.

Reference Book

1. Kimmo Eriksson & Hillevi Gavel, Discrete Mathematics & Discrete Models, Student litteratur AB, 2015.
2. Kenneth H. Rosen Discrete Mathematics and applications, Mc Graw Hill, 2012

Course Outcomes

On the successful completion of the course, students will be able

CO1	To understand the concepts of relations and functions distinguish among normal forms	K2	IO
CO2	To analyze and evaluate the recurrence relations	K4,K5	HO
CO3	To distinguish among various normal forms and predicate calculus	K5	HO
CO4	To solve and know various types of matrices	K1	LO
CO5	To evaluate and solve various types of graphs	K5	HO

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12

CO1	S	S	S	-	S	-	-	-	M	-	-	-
CO2	S	S	S	-	S	-	-	-	M	-	-	-
CO3	S	S	S	-	S	-	-	-	M	S	S	S
CO4	S	S	S	-	S	-	-	-	M	-	-	-
CO5	S	S	S	-	S	-	-	-	M	S	S	S

S- Strong; M-Medium; L-Low

Course Code-22UPCSC2C02 / 22UPCSC1C02 Credits: 4

Linux and Shell Programming

Course Objective

- To teach principles of operating system including File handling utilities, Basic Linux commands, Scripts and filters.
- To familiarize fundamentals of shell (bash), shell programming, pipes, Control structures, arithmetic in shell interrupt processing, functions, debugging shell scripts.
- To impart fundamentals of file concepts kernel support for file, File structure related system calls (file API's).
- To facilitate students in understanding Inter process communication, semaphore and shared memory.
- To explore real-time problem solution skills in Shell programming.

Unit-I

Basic bash Shell Commands: Interacting with the shell-Traversing the file system-Listing files and directories-Managing files and directories-Viewing file contents. **Basic Script Building:** Using multiple commands-Creating a script file-Displaying messages-Using variables-Redirecting input and output-Pipes-Performing math-Exiting the script. **Using Structured Commands:** Working with the if-then statement-Nesting ifs-Understanding the test command-Testing compound conditions-Using double brackets and parentheses-Looking at case.

(Book-1, Chapters: 3, 11, and 12)

Unit-II

More Structured Commands: Looping with for statement-Iterating with the until statement-Using the while statement-Combining loops-Redirecting loop output. **Handling User Input:** Passing parameters-Tracking parameters-Being shifty-Working with options-Standardizing options-Getting user input. **Script Control:** Handling signals-Running scripts in the background-Forbidding hang-ups -Controlling a Job-Modifying script priority-Automating script execution.

(Book-1, Chapters: 13, 14, and 16)

Unit-III

Creating Functions: Basic script functions-Returning a value-Using variables in functions-Array and variable functions-Function recursion-Creating a library-Using functions on the command line. **Writing Scripts for Graphical Desktops:** Creating text menus-Building text window widgets-Adding X Window graphics. **Introducing sed and gawk:** Learning about the sed Editor-Getting introduced to the gawk Editor-Exploring sed Editor basics.

(Book-1, Chapters: 17, 18, and 19)

Unit-IV

Regular Expressions: Defining regular expressions-Looking at the basics-Extending our patterns-Creating expressions. **Advanced sed:** Using multiline commands-Understanding the hold space-Negating a command-Changing the flow-Replacing via a pattern-Using sed in scripts-Creating sed utilities. **Advanced gawk:** Reexamining gawk-Using variables in gawk-Using structured commands-Formatting the printing-Working with functions.

(Book-1, Chapters: 20, 21, and 22)

Unit-V

Working with Alternative Shells: Understanding the dash shell-Programming in the dash shell-Introducing the zsh shell-Writing scripts for zsh. **Writing Simple Script Utilities:** Automating backups-Managing user accounts-Watching disk space. **Producing Scripts for Database, Web, and E-Mail:** Writing database shell scripts-Using the Internet from your scripts-Emailing reports from scripts. **Using Python as a Bash Scripting Alternative:** Technical requirements-Python Language-Hello World the Python way-Pythonic arguments-Supplying arguments-Counting arguments-Significant whitespace-Reading user input-Using Python to write to files-String manipulation.

(Book-1, Chapters: 23, 24, 25, and Book-2, Chapter: 14)

Text book:

1. Richard Blum, Christine Bresnahan, "Linux Command Line and Shell Scripting BIBLE", Wiley Publishing, 3rd Edition, 2015.**Chapters:** 3, 11 to14, 16 to 25.
2. Mokhtar Ebrahim, Andrew Mallett, "Mastering Linux Shell Scripting", Packt Publishing, 2nd Edition, 2018. **Chapter:** 14.

Reference Books:

1. ClifFlynt, SarathLakshman, ShantanuTushar, “Linux Shell Scripting Cookbook ”, Packt Publishing, 3rd Edition, 2017.
2. Stephen G.Kochan, Patrick Wood, “Shell Programming in Unix, Linux, and OS X”, Addison Wesley Professional, 4th Edition, 2016
3. Robert Love, “Linux System Programming”, O'Reilly Media, Inc, 2013
4. W.R. Stevens, “Advanced Programming in the UNIX environment”, 2nd Edition, Pearson Education, 2013
5. Graham Glass, King Ables, “ UNIX for Programmers and Users”, 3rd Edition, Pearson Education, 2003

Course Outcomes

On the successful completion of the course, students will be able

CO1	To understand, apply and analyze the concepts and methodology of Linux shell programming	K1-K6
CO2	To comprehend, impart and apply fundamentals of control structure and script controls	K1-K6
CO3	To understand, analyses and evaluate the functions, graphical desktop interface and editors	K1-K6
CO4	To collaborate, apply and review the concepts and methodology of regular expression and advanced gawk	K1-K6
CO5	To comprehend, use and illustrate the advance concepts such as alternate shell script, data connectivity and bash scripting using python	K1-K6

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	S	-	S	L	-	M	M	M	M	S
CO2	S	S	M	-	S	L	-	M	M	M	M	S
CO3	S	S	M	-	S	L	-	M	M	S	S	S
CO4	S	S	M	-	S	L	-	M	M	M	M	S
CO5	S	S	M	-	S	L	-	M	M	M	M	S

S- Strong; M-Medium; L-Low

Course Code-22UPCSC2C03 / 22UPCSC1C03 Credits: 4

Python Programming

Course Objectives:

- To acquire programming skills in core Python
- To learn Strings and function
- To develop object oriented skills in Python
- To comprehend various Python Packages
- To develop web applications using Django

Unit I

Introduction : Fundamental ideas of Computer Science **-** Strings, Assignment, and Comments - Numeric Data types and Character sets – Expressions – Loops and Selection Statements: Definite iteration: the for Loop - selection: if and if-else statements - Conditional iteration: the while Loop

Unit II

Strings and Text Files: Accessing Characters and substrings in strings - Data encryption-Strings and Number systems- String methods – Text - Lists and Dictionaries: Lists – Dictionaries – Design with Functions: A Quick review - Problem Solving with top-Down Design - Design with recursive Functions - Managing a Program’s namespace - Higher-Order Functions

Unit III

Design with Classes: Getting inside Objects and Classes – Data-Modeling Examples – Building a New Data Structure – The Two – Dimensional Grid - Structuring Classes with Inheritance and Polymorphism - Graphical User Interfaces - The Behavior of terminal-Based programs and GUI-Based programs - Coding Simple GUI-Based programs - Windows and Window Components - Command Buttons and responding to events.

Unit IV

Working with Python Packages: NumPy Library-Ndarray – Basic Operations – Indexing, Slicing and Iteration – Array manipulation - Pandas –The Series – The DataFrame - The Index Objects – Data Vizualization with Matplotlib – The Matplotlib Architecture – pyplot – The Plotting Window – Adding Elements to the Chart – Line Charts – Bar Charts – Pie charts

Unit V

Django: Installing Django – Building an Application – Project Creation – Designing the Data Schema - Creating an administration site for models - Working with QuerySets and Managers – Retrieving Objects – Building List and Detail Views

Text Book:

1. K.A. Lambert, “ Fundamentals of Python: first programs”, Second Edition, Cengage Learning, 2018 **(Unit - I, II and III)**
2. Fabio Nelli, “Python Data Analytics: With Pandas, NumPy, and Matplotlib”, Second Edition, Kindle Edition, 2018 **(Unit - IV)**
3. Antonio Mele, “Django 3 By Example”, Third Edition, 2020 **(Unit - V)**

Course Outcomes

On the successful completion of the course, students will be able to

CO1	Comprehend the programming skills in python and develop applications using conditional branches and loop	**K1- K6**
CO2	Create python applications with strings and functions	**K1- K6**
CO3	Understand and implement the Object Oriented Programming paradigm with the concept of objects and classes, Inheritance and polymorphism	**K1- K6**
CO4	Evaluate the use of Python packages to perform numerical computations and data visualization	**K1- K6**
CO5	Design interactive web applications using Django	**K1- K6**

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5 Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	M	S	M	S	S	S	S	M	S	S
CO2	S	S	S	M	S	S	S	S	S	S	M	S
CO3	S	M	S	S	M	S	M	S	S	M	S	S
CO4	S	S	S	S	S	S	S	M	S	S	M	S
CO5	S	S	S	S	S	S	S	S	S	M	M	S

S- Strong; M-Medium; L-Low

Course Code: 22UPCSC2C04 **Credits: 4**

Theory of Computation

COURSE OBJECTIVES:

- Introduce core concepts in Automata and Theory of Computation
- Identify different Formal language Classes and their Relationships
- Design Grammars and Recognizers for different formal languages
- Prove or disprove theorems in automata theory using their properties
- Determine the decidability and intractability of Computational problems

UNIT I

History of automata- grammar-Chomsky hierarchy-use of automata characteristics of automata-finite automata-graphical and tabular representation transactional system-DFA and NFA –conversion of NFA to DFA-Equivalence of DFA and NFA-Dead state-Finite automata with output-conversion of one machine to minimization of finite automata-Two way finite automata

UNIT II

Finite state machine-state equivalence and minimization of machine - incompletely specified machine-merger graph- merger table-finite memory and definite memory information lossless machine-inverse machine-minimal inverse machine-ardens theorem-construction of finite automata from regular expression

UNIT III

Equivalence of two finite automata- Equivalence of two regular expression-construction of regular grammar from an RE-constructing FA from regular grammar-Pumping lemma for regular expression--derivation and parse tree-Ambiguity in context free grammar-left recursion and left factoring-linear grammar-normal form – pumping lemma for CFL-Ogdens lemma for CFL

UNIT IV

Push down automata-acceptance PDA-DPDA and NPDA-Construction of PDA from CFG-construction of CFG equivalent to PDA-Graphical notation for PDA-Turing Machine-transactional representation of turing machine –non deterministic turing –conversion of regular expression to turing machine.

UNIT V

Variations of turing machine-turing machine as an integer function-Universal turing machine-linear bounded automata- undecidability- reducibility

TEXT BOOKS:

1. ShyamlenduKandar, "Introduction to automata theory, formal languages and Computation" First Edition, Pearson Education, 2013.

REFERENCE BOOK:

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D.Ullman, "Introduction to Automata Theory, Languages and Computation", 3rd Edition, Pearson Education, 2011.

COURSE OUTCOMES

On the successful completion of the course, students will be able to

CO1	To understand the concepts and methodology of automata	K1	LO
CO2	To analyze and construct automata, regular expressions and grammars	K4	IO
CO3	To analyses and evaluate the push down automata and various Turing machines	K4- K5	HO

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Course Code-22UPCSC2C05 **Credits: 2**

Linux and Shell Programming - Lab

Course Objective

- To enable the students to study and understand the efficiency of Linux shell script.
- To demonstrate the File Backup process.
- To develop and implement the shell script for GUI processing.
- To develop and implement the shell script for IPC and Networking.
- To demonstrate PostgreSQL.

List of Programs

1. Write a Shell Script program to calculate the number of days between two dates.
2. Write a Shell Script program to check systems on local network using control structures with user input.
3. Write a Shell Script program to check systems on local network using control structures with file input.
4. Write a Shell Script program to demonstrate the script control commands.
5. Write a Shell Script program to demonstrate the Shell script function.
6. Write a Shell Script program to demonstrate the Regular Expressions.
7. Write a Shell Script program to demonstrate the sed and awk Commands.
8. Write a Shell Script program to demonstrate the File Backup process through creating a daily archive location.
9. Write a Shell Script program to create a following GUI tools.
   a) Creating text menus
   b) Building text window widgets
10. Write a Shell Script program to demonstrate to connect a PostgreSQL database and performing CRUD operations.

Course Outcomes

On the successful completion of the course, students will be able to

CO1	To understand, apply and analyze the concepts and methodology of Linux shell programming	K1-K6
CO2	To comprehend, impart and apply fundamentals of control structure and script controls	K1-K6
CO3	To understand, analyses and evaluate the functions, graphical desktop interface and editors	K1-K6
CO4	To collaborate, apply and review the concepts and methodology of regular expression and advanced gawk	K1-K6

CO5	To comprehend, use and analyze the advance concepts such as alternate shell script, dy and bash scripting using Postgre SQL	K1-K6

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	S	-	S	-	-	-	M	-	-	-
CO2	S	S	S	-	S	-	-	-	M	-	-	-
CO3	S	S	S	-	S	-	-	-	M	S	S	S
CO4	S	S	S	-	S	-	-	-	M	-	-	-
CO5	S	S	S	-	S	-	-	-	M	S	S	S

S- Strong; M-Medium; L-Low

Course Code-22UPCSC2C06 **Credits: 2**

Python Programming Lab

Course Objectives:

This course enables the students:

- To master the fundamentals of writing python scripts
- To create program using elementary data items
- To implement Python programs with conditionals and loops
- To use functions for structuring Python programs
- To develop web programming with Django

Implement the following in Python:

1. Program using elementary data items, lists, dictionaries and tuples
2. Program using conditional branches, loops
3. Program using functions
4. Program using classes and objects
5. Program using inheritance
6. Program using polymorphism
7. Program using Numpy
8. Program using Pandas
9. Program using Matplotlib
10. Program for creating dynamic and interactive web pages using forms

Course Outcome:

On the successful completion of the course, students will be able to

CO1	Comprehend the programming skills in python and write scripts	**K1- K6**
CO2	Create python applications with elementary data items, lists, dictionaries and tuples	
CO3	Implement the Object Oriented Programming concepts such as objects and classes, Inheritance and polymorphism	
CO4	Assess the use of Python packages to perform numerical computations and perform data visualization	
CO5	Create interactive web applications using Django	

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	M	S	S	S	S	S	S	M	S	S
CO2	S	S	S	S	S	S	S	S	S	S	M	S
CO3	S	S	S	S	S	S	M	S	S	M	L	S
CO4	S	S	S	S	S	S	S	M	S	S	S	S
CO5	S	S	S	S	L	S	M	S	S	M	M	S

S- Strong; M-Medium; L-Low

Course Code: 22UPCSC2C07 **Credits: 2**

Soft Skill Development Lab

Course Objective

This course enables the students:

- To enable students to gain basic communication skills in professional and social contexts effectively.
- To acquire useful words and apply them in situational context.
- To develop listening and reading skills through comprehension passages
- To enrich the leadership qualities and interpersonal communication
- To enhance essential characteristics in writing

Exercises

1. Characteristics of Technical Writing
2. Development of Employability Skills
3. Vocabulary Development
4. Sentence Completion
5. Error Spotting
6. Interpretation of Verbal Analogy
7. Interpretation of Reading (Comprehension -Conception)
8. Interpretation of Reading (Comprehension -Reasoning)
9. Practice for writing E-mails/Technical Blogs/Forums
10. PPT Preparation / Demonstration of Technical Presentation
11. Preparation of Resume
12. Preparation for Job Interviews / Mock Interview Section
13. Group Discussion Skills
14. Developing Listening Skill(Comprehension)
15. Practice for Short Speeches / Situational Conversation
16. English through Mass Media
17. Essential Grammar
18. Communicating and collaborating with peer members
19. Team Empowerment
20. Persuasive Communication

Text Books

1. Uma Narula, "Development Communication: Theory and Practice", Revised Edition, Har-Aanad Publication, 2019.
2. Annette Capel and Wendy Sharp, "Cambridge English: Objective First", Fourth Edition, Cambridge University Press, 2013.
3. Emma Sue-Prince, "The Advantage: The 7 Soft Skills You Need to Stay One Step Ahead", First Edition, FT Press, 2013.
4. Guy Brook-Hart, "Cambridge English: Business Benchmark",Second Edition, Cambridge University Press, 2014.
5. Norman Lewis, "How to Read Better & Faster", Binny Publishing House, New Delhi, 1978.

Reference Books

1. Michael McCarthy and Felicity O'Dell, "English Vocabulary in Use:100 Units of Vocabulary Reference and Practice", Cambridge University Press,1996.
2. Murphy, Raymond, "Intermediate English Grammar", Second Edition, Cambridge University Press, 1999.

Course Code: 22UPCSC2C08 **Credits: 4**

DESIGN AND ANALYSIS OF ALGORITHMS

Course Objectives:

- Apply the algorithms and design techniques to solve problems
- Analyze the complexities of various problems in different domains.
- Analyze the performance of various algorithms.

Unit-I

Introduction – Notion of Algorithm - Fundamentals of algorithmic problem solving – Important problem types – Fundamentals of the analysis of algorithm efficiency – analysis frame work – Asymptotic Notations and Basic Efficiency Classes- Mathematical analysis of non-recursive Algorithms – Non-recursive solution to the Matrix Multiplication - Mathematical analysis of recursive algorithms – Recursive solution to the Tower of Hanoi Puzzle.

Unit-II

Divide and conquer Technique – Multiplication of large integers – Strassen's matrixmultiplication – Closest pair and Convex Hull Problems - Greedy method – Prim's algorithm – Kruskal's algorithm – Dijkstra's algorithm.

Unit-III

Dynamic Programming - Computing a binomial coefficient – Warshall's and Floyd' Algorithm – Application of Warshall's Algorithm to the digraph – Flyd's Algorithm for the all pairs shortest paths Problem - The Knapsack problem and Memory function.

Unit-IV

Backtracking – N-Queens problem – Hamiltonian circuit problem – Subset sum problem – Branch and bound – Assignment problem – Knapsack problem – Traveling salesman problem.

Unit-V

P, NP and NP-complete problems – Approximation algorithms for NP-hard problems – Traveling salesman problem – Knapsack problem.

Text Book:

1. Anany Levitin "Introduction to the Design and Analysis of Algorithms" Pearson Education 2011. (Chapters 1.1-1.3, 2.1, 2.2, 2.3, 2.4, 4.5, 4.6, 8.2, 8.4, 9.1-9.3, 11.3, 12.1,12.2, 12.3)

Reference Books:

1. Thomas H.Cormen, Charles E.Leiserson, Ronald L.Rivest, "Introduction to algorithms", Prentice Hall 1990.
2. S.K. Basu, "Design methods and Analysis of Algorithms", Prentice Hall, 2005.

Course Outcomes

On the successful completion of the course, students will be able

CO1	To understand and apply knowledge of computing and mathematics to find the algorithm efficiency.	K2,K3
CO2	To analyze a problem and identify the computing requirements appropriate for its solution.	K4
CO3	To create, implement, and evaluate a Dynamic Programming algorithm to meet desired needs.	K5,K6
CO4	To create, implement, and evaluate a Backtracking and Knapsack to meet desired needs.	K1-K6
CO5	To evaluate the algorithmic principles and efficiency of NP and NP-Complete problems – Approximation algorithms for NP-hard problems – Travelling salesman problem – Knapsack problem.	K6

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	S	-	S	L	-	M	M	M	M	S
CO2	S	S	M	-	S	L	-	M	M	M	M	S
CO3	S	S	M	-	S	L	-	M	M	S	S	S
CO4	S	S	M	-	S	L	-	M	M	M	M	S
CO5	S	S	M	-	S	L	-	M	M	M	M	S

S- Strong; M-Medium; L-Low

Course Code: 22UPCSC2C09 **Credits: 4**

Big Data Analytics

Course Objectives

- To introduce big data tools & Information Standard formats.
- To understand the basic concepts of big data.
- To learn Hadoop, HDFS and MapReduceconcepts.
- To teach the importance of NoSQL.
- To explore the big data tools such as Hive, HBase and Pig.

UNIT I

Big Data and Analytics: Classification of Digital Data: Structured Data- Semi Structured Data and Unstructured Data.
Introduction to Big Data: Characteristics – Evolution – Definition - Challenges with Big Data - Other Characteristics of Data - Big Data - Traditional Business Intelligence versus Big Data - Data Warehouse and Hadoop.
Environment Big Data Analytics: Classification of Analytics – Challenges - Big Data Analytics important - Data Science - Data Scientist - Terminologies used in Big Data Environments – Basically Available Soft State Eventual Consistency - Top Analytics Tools

UNIT II

Technology Landscape: NoSQL, Comparison of SQL and NoSQL, Hadoop - RDBMS Versus Hadoop - Distributed Computing Challenges – Hadoop Overview - Hadoop Distributed File System - Processing Data with Hadoop - Managing Resources and Applications with Hadoop YARN - Interacting with Hadoop Ecosystem

UNIT III

Mongodb and Mapreduce Programming: MongoDB: Mongo DB - Terms used in RDBMS and Mongo DB - Data Types - MongoDB Query Language.
MapReduce: Mapper – Reducer – Combiner – Partitioner – Searching – Sorting – Compression

UNIT IV

Hive: Introduction – Architecture - Data Types - File Formats - Hive Query Language Statements – Partitions – Bucketing – Views - Sub- Query – Joins – Aggregations - Group by and Having – RCFile - Implementation - Hive User Defined Function - Serialization and Deserialization.

UNIT V

Pig: Introduction - Anatomy – Features – Philosophy - Use Case for Pig - Pig Latin Overview - Pig Primitive Data Types - Running Pig - Execution Modes of Pig - HDFS Commands - Relational Operators - Eval Function - Complex Data Types - Piggy Bank - User-Defined Functions - Parameter Substitution – Diagnostic Operator - Word Count Example using Pig - Pig at Yahoo! - Pig Versus Hive

Text Book:

1. Seema Acharya, Subhashini Chellappan, "Big Data and Analytics", WileyPublications, First Edition, 2015

Reference Book:

1. Judith Huruwitz, Alan Nugent, Fern Halper, Marcia Kaufman, "Big data for dummies", John Wiley & Sons, Inc. (2013)
2. Tom White, "Hadoop The Definitive Guide", O'Reilly Publications, Fourth Edition, 2015
3. Dirk Deroos, Paul C.Zikopoulos, Roman B.Melnky, Bruce Brown, Rafael Coss, "Hadoop For Dummies", Wiley Publications, 2014

4. Robert D.Schneider, “Hadoop For Dummies”, John Wiley & Sons, Inc. (2012)
5. Paul Zikopoulos, “Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data, McGraw Hill, 2012 Chuck Lam, “Hadoop In Action”, Dreamtech Publications, 2010

Course Outcomes

On the successful completion of the course, students will be able to

CO1	To understand, illustrate and evaluate the concepts and techniques of Data Science, Big Data Analytics and its tools	K1-K6
CO2	To collaborate, apply and review the computing for big data in Hadoop, and NoSQL environment.	K1-K6
CO3	To comprehend, implement and review the concepts of data science and big data analytics projects using MapReduce, and MongoDB	K1-K6
CO4	To understand, use and anlayse the concepts of big data analytics projects using HIVE database.	K1-K6
CO5	To illustrate, develop and review the concepts of PIG database in Hadoop environment.	K1-K6

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	-	-	-	-	L	-	-	-	-	-	-
CO2	S	-	M	-	M	L	-	-	-	-	-	-
CO3	S	-	S	-	S	L	-	-	-	S	S	S
CO4	S	-	S	-	S	L	-	-	-	S	S	S
CO5	S	-	S	-	S	L	-	-	-	S	S	S

S- Strong; M-Medium; L-Low

Course Code: 22UPCSC2C10 **Credits: 2**

Design and Analysis of Algorithms Lab

Implement the following problems using Python Programming

1. Compute the transitive closure of any directed graph using Warshall's Algorithm.
2. Knapsack problem using backtracking
3. 0/1 knapsack problem using Dynamic programming
4. Apply the divide and conquer technique implement Strassen's matrix Multiplication Algorithm
5. Find minimum cost spanning Tree of a given undirected graph using Kruskal's Algorithm.
6. Find minimum cost spanning Tree of a given undirected graph using Prim's Algorithm.
7. All-pairs Shortest Paths algorithms
8. 8 Queen's problem using backtracking
9. DIjkstra's Algorithm using greedy technique
10. Sum of subset problem using backtracking

Course Code: 22UPCSC2C11 **Credits: 2**

Big Data Analytics Lab

Course Objectives

- To teach the fundamental techniques for handling the big data tools.
- To familiarize the tools required to manage big data.
- To analyse big data using Hadoop, MapReduce, Hive, and Pig
- To teach the fundamental principles in achieving big data analytics with scalability and streaming capability
- To enable students to have skills that will help them to solve complex.

Implement the following problems

1. Implement File System Shell Commands for HDFS in Hadoop Environment
2. Write a Mapreduce program using single reduce function for finding Maximum and Minimum Number
3. Write a Mapreduce program using multiple reduce function for Word Count in an given Text document
4. Implement the following using Pig Latin Input and Output Operations Relational Operations
5. Implement the following using Pig Latin User Defined Functions Advanced Relational Operations
6. Write a Word Count program using Pig Latin Script
7. Write a program to find a maximum temperature using Pig Latin Script
8. Implement the following using Hive commands Handling the Database Creating and Manipulating table
9. Implement Simple Queries for database using Mongo
10. Implement Simple Queries for collections using Mongo

Course Outcomes

On the successful completion of the course, students will be able to

CO1	Understand and develop conceptually how Big Data is stored and implement it using different tools	K1-K6
CO2	Comprehend and implement programs for data storage in HDFS and table manipulation using Big Data tools in Hadoop environment	K1-K6
CO3	Understand and Critically analyse existing Big Data datasets and implementations the solutions for it using MongoDB	K1- K6
CO4	Understand and examine existing Big Data datasets and implementations the solutions using HIVE database	K1- K6
CO5	Comprehend and review existing datasets and implementations the solutions to handle it using PIG	K1- K6

K1- Remember, K2- Understand, K3- Apply , K4- Analyze, K5- Evaluate,

K6- Create

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	M		M	S	-	-	-	-	-	-
CO2	S	M	S	S	S	M	-	-	-	-	-	-
CO3	S	S	S	S	S	S	-	-	-	-	-	-
CO4	S	M	S	S	S	M	-	-	-	-	-	-
CO5	S	S	S	S	S	S	-	-	-	-	-	-

S- Strong; M-Medium; L-Low

Course Code: 22UPCSC2C12 **Credit: 3**

Mini Project

The students are expected to collect the necessary data from neighboring industries during holidays and apply the concepts studied in the Semester I and Semester II to develop the project.

i. Introduction
ii. Data Collection
iii. System development
iv. Implementation
v. Conclusion

Course Code: 22UPSOC2H01 **Credits: 2**

Fundamentals of Human Rights

Unit I: Introduction: Meaning and Definitions of Human Rights – Characteristics and Importance of Human Rights – Evolution of Human Rights – Formation, Structure and Functions of the UNO - Universal Declaration of Human Rights – International Covenants – Violations of Human Rights in the Contemporary Era.

Unit II: Human Rights in India: Development of Human Rights in India – Constituent Assembly and Indian Constitution – Fundamental Rights and its Classification – Directive Principles of State Policy – Fundamental Duties.

Unit III: Rights of Marginalized and other Disadvantaged People: Rights of Women – Rights of Children – Rights of Differently Abled – Rights of Elderly - Rights of Scheduled Castes – Rights of Scheduled Tribes – Rights of Minorities – – Rights of Prisoners – Rights of Persons Living with HIVAIDS – Rights of LGBT.

Unit IV: Human Rights Movements: Peasant Movements (Tebhaga and Telangana) – Scheduled Caste Movements (Mahar and Ad-Dharmi) – Scheduled Tribes Movements (Santhal and Munda) – Environmental Movements (Chipko and Narmada Bachao Andolan) – Social Reform Movements (Vaikom and Self Respect).

Unit V: Redressal Mechanisms: Protection of Human Rights Act, 1993 (Amendment 2019) – Structure and Functions of National and State Human Rights Commissions – National Commission for SCs – National Commission for STs – National Commission for Women – National Commission for Minorities – Characteristics and Objectives of Human Rights Education.

References

1. Sudarshanam Gankidi, Human Rights in India: Prospective and Retrospective, Rawat Publications, Jaipur, 2019.
2. Satvinder Juss, Human Rights in India, Routledge, New Delhi, 2020.
3. Namita Gupta, Social Justice and Human Rights in India, Rawat Publications, Jaipur, 2021.
4. Mark Frezo, The Sociology of Human Rights, John Willy & Sons, U.K. 2014.
5. Chiranjivi J. Nirmal, Human Rights in India: Historical, Social and Political Perspectives, Oxford University Press, New York, 2000.
6. Dr. S. Mehartaj Begum, Human Rights in India: Issues and perspectives, APH Publishing Corporation, New Delhi, 2010.
7. Asha Kiran, The History of Human Rights, Mangalam Publications, Delhi, 2011.
8. Bani Borgohain, Human Rights, Kanishka Publishers & Distributors, New Delhi-2, 2007. Jayant Chudhary, A Textbook of Human Rights, Wisdom Press, New Delhi, 2011.

Course Code: 22UPCSC2C13 **Credit: 4**

Advanced Java Programming

Course Objectives

- To gain knowledge of Object Oriented Programming Concept in Java
- To understand usages of String functions in Java
- To familiarize with the applet and swing
- To grasp the concepts on Java Beans
- To comprehend the connection between Relational Database and Java.

Unit – I

An Overview of Java: Object Oriented Programming- Data Types, Variables, and Arrays: Primitive Types-Literals Variables - Type Conversion and Casting- Arrays- Operators: Control Statements-Classes and Methods – Inheritance- Exception Handling.

Unit – II

String Handling: The String Constructors - String Length - Special String Operations - Character Extraction - String Comparison - Searching Strings - Modifying a String - Input/Output: The I/O Classes and Interfaces – File - Byte Streams - Character Streams.

Unit – III

The Applet Class: Basic Architecture - Applet Skeleton - Display methods - Status Window – Passing Parameters. Introducing GUI Programming with Swing– Introducing Swing - Swing Is Built on the AWT- Two Key Swing Features - The MVC Connection - Components and Containers - The Swing Packages - A Simple Swing Application - Exploring Swing.

Unit- IV

Java Beans: Introduction - Advantages of Beans – Introspection - The JavaBeans API - A Bean Example. Servlets: Life Cycle Simple Servlet-Servlet API-Packages-Cookies session tracking.

Unit – V

Network Programming: Working with URLs- Working with Sockets - Remote Method Invocation. Introduction to Database Management Systems - Tables, Rows, and Columns - Introduction to the SQL SELECT Statement - Inserting Rows - Updating and Deleting Existing Rows - Creating and Deleting Tables - Creating a New Database with JDBC - Scrollable Result Sets.

Text Books:

1. Herbert Schildt, "Java the Complete Reference", 10th edition, McGraw Hill Publishing Company Ltd, New Delhi, 2017.
2. Tony Goddis, "Starting out with Java from Control Structures Through Objects" 6th Edition, Pearson Education Limited, 2016

Reference books :

1. Herbert Schildt, Dale Skrien, "Java Fundamentals – A Comprehensive Introduction", TMGH Publishing Company Ltd, New Delhi, 2013
2. John Dean, Raymond Dean, "Introduction to Programming with JAVA – A Problem Solving Approach", TMGH Publishing Company Ltd, New Delhi,2012.

Course Outcomes

On the successful completion of the course, students will be able

CO1	Understand the Object Oriented Program including classes and methods; inheritance and exception handling	K1-K6
CO2	Complete comprehension of String functions and I/O Streams	K1-K6
CO3	Creation of graphical representation using Applet	K1-K6
CO4	Application of Servlets for designing Web based applications	K1- K6
CO5	Usage of JDBC connectivity and implementation of the concept to get desired results from database	K1-K6

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	S	-	M	S	-	-	-	S	M	-
CO2	S	S	S	-	M	S	-	-	-	L	M	-
CO3	S	S	M	-	L	S	-	-	-	M	M	-
CO4	M	S	M	-	S	S	-	-	-	M	S	-
CO5	S	M	M	-	M	L	-	-	-	M	M	-

S- Strong; M-Medium; L-Low

Course Code: 22UPCSC2C14 **Credits 4**

Web Technology

COURSE OBJECTIVES:

At the end of the course, the student should be able to do:

- Explore the back bone of web page creation by developing .NET skill.
- Enrich knowledge about XHTML control and Cascading Style Sheets.
- Provide in- depth knowledge about Java script.
- Understand the need of usability, evaluation methods for web services.
- Provide in- depth knowledge about PHP, Angular JS, Jquery.

UNIT -I

INTRODUCTION TO XHTML AND CSS: Basic syntax, Standard structure, Basic text-markup, Images, Hypertext Links. Lists, Tables, Forms, Frames, syntactic differences between HTML and XHTML-Introduction, Levels of style sheets, Style specification formats, Selector forms, Property value forms, Font properties, List properties, Color, Alignment of text, The box model, Background images, The<span> and <div>tags, Conflict resolution.

UNIT - II

THE BASICS OF JAVASCRIPT: Overview of JavaScript, Object orientation and JavaScript, general Syntactic characteristics, Primitives, operations, and expressions, Screen output and keyboard input, Control statements, Object creation and modification, Arrays, Functions, Constructors, Pattern matching using regular expressions, Errors in scripts.

JAVASCRIPT AND XHTML DOCUMENTS: The JavaScript Execution Environment, The Document Object Model, Elements Access in Java Script, Events and Event Handling, Handling Events from Body Elements, Handling Events from Text Box and password Elements, The DOM2 Model

UNIT- III

DYNAMIC DOCUMENTS WITH JAVASCRIPT AND XML: Introduction, Positioning Elements, Moving Elements, Element Visibility, Changing Color and Fonts, Dynamic Content, Stacking Elements, Locating the Mouse Cursor, Reacting to a Mouse Click, Slow Movement of Elements, Dragging and Dropping Elements. Introduction to XML, Syntax of XML, XML Document Structure, Document type definitions, Namespaces, XML schemas, displaying raw XML documents, Displaying XML documents with CSS, XSLT Style Sheets, Web services.

UNIT - IV

PHP Introduction to PHP: Overview of PHP - General Syntactic Characteristics - Primitives, Operations, and Expressions - Output - Control Statements - Arrays - Functions - Pattern Matching - Form Handling - Cookies - Session Tracking.

Unit - V

ANGULAR JS AND JQUERY

Introduction to JQuery, Syntax, selectors, events, JQuery HTML, JQuery Effects, JQuery CSS. Introduction to Angular JS, Directives, Expressions, Controllers, Filters, Services, Events, Forms, Validations, Examples.

TEXT BOOKS:

1. Robert W. Sebesta: Programming the World Wide Web, Eighth Edition, Pearson education, 2015. **UNITS:** 1,2,3,4
2. Dayley Brad, Dayley Brendan ,"AngularJS, JavaScript, and jQuery All in One", Sams Teach Yourself 1st Edition, Kindle Edition, 2015. **UNIT**: 5

REFERENCE BOOKS:

1. M. Srinivasan: Web Programming Building Internet Applications, 3rdEdition, Wiley India, 2009.
2. Jeffrey C. Jackson: Web Technologies-A Computer Science Perspective, Pearson Education,7thImpression,2012.
3. Chris Bates: Web Technology Theory and Practice, Pearson Education, 2012.
4. Raj Kamal: Internet and Web Technologies, McGraw Hill Education.

COURSE OUTCOMES

On the successful completion of the course, students will be able

CO1	Design dynamic web pages using Java script, Jquery and Angular Java script	K1	LO
CO2	Develop Web pages using HTML, CSS and XML	K2	IO
CO3	Create web application using PHP and MySQL	K3, K4	HO
CO4	To design dynamic web pages using Angular java script	K2,K3	HO
CO5	Develop interactive web pages using Jquery	K4,K5	HO

MAPPING WITH PROGRAMME OUTCOMES

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	S	S	S	S	M	M	S	M	M	S	S
CO2	S	S	M	S	S	S	M	S	S	S	S	S
CO3	S	S	S	M	S	S	M	M	S	M	M	S
CO4	S	S	S	M	S	M	M	S	S	M	S	M
CO5	S	S	S	M	S	S	M	S	M	S	S	M

S- STRONG; M-MEDIUM; L-LOW

Course Code: 22UPCSC2C15 **Credit: 4**

Advanced Machine Learning Technologies

Course Objectives

- To understand the concepts of Machine Learning.
- To understand the theoretical and practical aspects of types of machine learning
- To teach and get familiarized with supervised learning and their applications.
- To teach and get familiarized with the concepts and algorithms of unsupervised learning.
- To appreciate the concepts and algorithms of deep learning.

Unit I

Introducing Machine Learning: The Origins of Machine Learning, Uses and Abuses of Machine Learning _ Basics of Machine Learning Algorithm Model Works - Steps to apply Machine Learning - Choosing a Machine Learning Algorithm - Using Machine Learning concepts.

Managing and Understanding Data: Data Structures, Vectors And Factors: Lists, Data frames, Matrixes and arrays - Managing Data - Exploring and Understanding Data: Exploring the Structure of Data, Exploring Numeric variables - Exploring Categorical Variables- Exploring Relationships between Variables.

Unit II

Lazy Learning – Classification Using Nearest Neighbors:The kNN Algorithm- Diagnosing Breast Cancer with the kNN Algorithm- Probabilistic Learning – Classification Using Naive Bayes: Basic concepts of Bayesian Methods- The Naïve Bayes Algorithm- Example – filtering Mobile Phone Spam with the Naive Bayes Algorithm.

Divide and Conquer – Classification Using Decision Trees and Rules: Understanding Decision Trees- Example – Identifying Risky Bank Loans using C5.0 Decision Trees- Understanding Classification Rules- Example – Identifying Poisonous Mushrooms with Rule Learners.

Unit III

Forecasting Numeric Data – Regression Methods: Understanding Regression- Example – Predicting Medical Expenses using Linear Regression- Understanding Regression Trees and Model Trees- Example – Estimating the Quality of Wines with Regression Trees and Model Trees.

Black Box Methods Neural Networks and Support Vector Machines: Understanding Neural Networks, from Biological to Artificial Neurons, Activation Functions, Network Topology, Training Neural Networks with Backpropagation - Modelling the Strength of Concrete with ANNs- Understanding Support Vector Machines- Performing OCR with SVMs- Finding Patterns – Market Basket Analysis Using Association Rules: Understanding Association Rules- Example – Identifying Frequently Purchased Groceries with Association Rules.

Unit IV

Finding Groups of Data – Clustering with K-Means: Understanding Clustering- The k-means Algorithm for clustering- Finding teen market segments using k-means Clustering- Evaluating Model Performance: Measuring Performance for Classification- Beyond Accuracy – other Measures of Performance, Visualizing Performance Tradeoffs.

Improving Model Performance: Tuning Stock Models for Better Performance-Using Caret for Automated Parameter Tuning- Creating a simple Tuned Model-Customizing the Tuning Process- Improving Model Performance with meta-learning- Understanding Ensembles- Bagging- Boosting- Random forests.

Unit V

Introduction to Deep Learning: Introduction to Deep Learning, Single Layer Perceptron Model (SLP), Multilayer Perceptron Model (MLP), Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs),Restricted Boltzmann Machines (RBMs).

Convolutional Neural Networks (CNNs): Structure and Properties of CNNs - Components of CNN Architectures- Convolutional Layer, Pooling Layer, Rectified Linear Units (ReLU) Layer, Fully Connected (FC) Layer, Loss Layer - Tuning Parameters ,Notable CNN Architectures, Regularization- Recurrent Neural Networks (RNNs): Fully Recurrent Networks, Training RNNs with Back-Propagation Through Time (BPPT)- Elman Neural Networks, Neural History Compressor, Long Short-Term Memory (LSTM), Traditional and Training LSTMs - Structural Damping Within RNNs, Tuning Parameter Update Algorithm.

Text Book

1. Brett Lantz, "Machine Learning with R", Addison-Wesley Packt Publishing, 2013.
2. Taweh Beysolow, "Introduction to Deep Learning Using R: A Step-by-Step Guide to Learning and Implementing Deep Learning Models Using R", San Francisco, California, USA, 2017.

Reference Book

1. Daniel T. Larose, Chantal D. Larose, "Data mining and Predictive analytics", Second Ed., Wiley Publication, 2015.
2. Bertt Lantz, "Machine Learning with R: Expert techniques for predictive modeling", 3rd Edition, April 15,2019,
3. Jason Bell, "Machine Learning: Hands-On for Developers and Technical Professionals", Wiley Publication,2015.

Course Outcomes

On the successful completion of the course, students will be able to

CO1	To understand, impart and analyze the concepts and of Machin Learning Techniques and types of data	**K1-K6**
CO2	To comprehend, apply and evaluate the classification technique for real-world applications	**K1-K6**
CO3	To understand, use and perform evaluation of Regressio methods	**K1-K6**
CO4	To recognize, implement and analyse the unsupervise techniques for real-world applications	**K1-K6**
CO5	To understand, identify, implement and review the deep learnin techniques for real-time applications	**K1-K6**

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	-	-	S	L	-	S	-	-	-	-
CO2	S	S	M	-	S	L	-	S	-	-	-	-
CO3	S	S	S	-	S	L	-	S	-	S	S	S
CO4	S	S	M	-	S	L	-	S	-	-	-	-
CO5	S	S	S	-	S	L	-	S	-	S	S	S

S- Strong; M-Medium; L-Low

Course Code: 22UPCSC2C16 **Credit: 2**

Advanced Java Programming Lab

1. Implementation and Exception handling concepts with different types of Exceptions.
2. Build a Swing application to implement metric conversion.
3. Use Grid Layout to design a calculator and simulate the functions of a simple calculator.
4. Create a Color palette with a matrix of buttons using Applet.
5. To invoke a servlet from HTML forms.
6. To invoke servlet from Applets.
7. To invoke servlet from JSP.
8. Implement message communication using Network Programming.
9. Write a program to connect databases using JDBC.
10. Implementation of Java Beans.

Course Code: 22UPCSC2C17 Credits: 2

Web Technology Lab

COURSE OBJECTIVES

At the end of the course, the student should be able to do:

- Learn how to create web pages using HTML, CSS and Javascript.
- Implement dynamic web pages using Javascript, Jquery and Angular Java script
- To create web applications using PHP and MySQL
- Create web pages using XML and Cascading Style Sheets
- Create XML documents and Schemas

Implement the following problems

1. Develop a web page to display your education details in a tabular format.
2. Develop a web page to display your CV on a web page.
3. Design a Homepage having three links: About Us, Our Services and Contact Us. Create separate web pages for the three links.
4. Design a web page to demonstrate the usage of inline CSS, internal CSS and external CSS.
5. Design an XML document and create a style sheet in CSS & display the document in the browser.
6. Develop a web page to Create image maps.
7. Design a web page to perform input validation using Angular Javascript.
8. Develop a web page in PHP to fetch details from the database.
9. Design a web page to hide paragraph using JQuery
10. Create a web page and add Javascript to handle mouse events and form Events.

COURSE OUTCOMES

On the successful completion of the course, students will be able

CO1	Design dynamic web pages using Javascript, Jquery and Angular Java script	**K1**	**LO**
CO2	Develop Web pages using HTML, CSS and XML	**K2**	**IO**
CO3	Create web application using PHP and MySQL	**K3, K4**	**HO**
CO4	Develop interactive web pages using Jquery	**K2,K3**	**HO**

CO5	To design dynamic web pages using Angular javascript	**K4,K5**	**HO**

MAPPING WITH PROGRAMME OUTCOMES

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	M	S	S	S	S	M	M	S	M	M	S	M
CO2	S	S	M	S	S	S	M	S	S	S	M	S
CO3	S	S	S	M	M	S	M	M	S	M	M	S
CO4	S	M	S	M	S	M	M	S	S	M	S	M
CO5	M	M	S	M	S	S	M	S	M	M	S	M

S- STRONG; M-MEDIUM; L-LOW

Course Code: 22UPCSC2C18 **Credits: 1**

Advanced Machine Learning Lab

Course Objectives

- To formulate machine learning problems corresponding to different applications.
- To understand a range of machine learning algorithms along with their strengths and weaknesses.
- To apply machine learning algorithms to solve problems of moderate complexity.
- To apply CNN to solve problems of moderate complexity.
- To apply LSTM and RNN to solve problems.

Implement the following problems

1. Write a program to compute the Central Tendency Measures: Mean, Median, Mode, Measure of Dispersion: Variance, Standard Deviation
2. Implement a Linear Regression and Multiple Linear Regression with a Real Dataset
3. Implementation of Logistic Regression using sklearn
4. Implement a binary classification model.
5. Classification with Nearest Neighbours and NavieBaye Algorithm
6. Implementation Decision tree for classification using sklearn and its parameter tuning
7. Implement the k-means algorithm.
8. Implement an Image Classifier using CNN.
9. Implement an Auto encoder.
10. Implement a Simple LSTM.

Course Outcomes

On the successful completion of the course, students will be able to

CO1	To understand and implement the mathematical and statistica prospective of machine learning algorithms through pythor programming	**K1-K6**
CO2	To recognize and develop the machine learning models through python in built functions	**K1-K6**
CO3	To understand, impart and develop the machine learning models for real-time dataset	**K1-K6**
CO4	To comprehend , impart and implement the deep learning models for real-time applications	**K1-K6**
CO5	To identify and evaluate the performance machine learning models for real-time dataset	**K1-K6**

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	S	-	S	-	-	-	M	-	-	-
CO2	S	S	S	-	S	-	-	-	M	-	-	-
CO3	S	S	S	-	S	-	-	-	M	S	S	S
CO4	S	S	S	-	S	-	-	-	M	-	-	-
CO5	S	S	S	-	S	-	-	-	M	S	S	S

S- Strong; M-Medium; L-Low

Course Code: 22UPCSC2C19 **Credits: 2**

Advanced Java Programming Lab

COURSE OBJECTIVES

- To implement object oriented concepts in JAVA
- Develop the program using concepts Network programme
- Learn how to create a program in java beans.
- Learn how to connect relational database to Java
- Develop the program using concepts Applet

List of Experiments

1. Implementation of and Exception handling concepts with different types of Exceptions.
2. Build a Swing application to implement metric conversion.
3. Use Grid Layout to design a calculator and simulate the functions of simple calculator.
4. Create a Color palette with a matrix of buttons using Applet.
5. To invoke a servlet from HTML forms.
6. To invoke servlet from Applets.
7. To invoke servlet from JSP.
8. Implement message communication using Network Programming.
9. Write a program to connect databases using JDBC.
10. Implementation of Java Beans.

Course Outcomes

On the successful completion of the course, students will be able to

CO1	Implement classes, objects, members of a class and the relationships among them needed for a finding the solution to specific problem	K1, K2	LO
CO2	Apply Applets and Swing programs	K3	IO
CO3	Develop Servlets and JSP for creating Web based applications using JDBC	K4, K5	HO

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												

S- Strong; M-Medium; L-Low

Course Code: 22UPCSC2C20 **Credits: 4**

Natural Language Processing

COURSE OBJECTIVES:

- To understand the NLP techniques.
- To be familiarized with the data structures and algorithms used in NLP.
- To understand and process raw text and apply categorizing tagging words for classification.
- To understand and apply text analytics grammar approaches.
- To know the basics of first order logic and propositional logic

UNITI

LANGUAGE PROCESSING AND PYTHON: Computing with Language: Texts and Words - A Closer Look at Python: Texts as Lists of Words -Computing with Language: Simple Statistics - Back to Python: Making Decisions and Taking Control -Automatic Natural Language Understanding- Accessing Text Corpora and Lexical Resources: Accessing Text Corpora- Conditional Frequency Distributions - More Python: Reusing Code - Lexical Resources –Word Net

UNITII

PROCESSING RAW TEXT -Accessing Text from the Web and from Disk - Strings: Text Processing at the Lowest Level- Text Processing with Unicode- Regular Expressions for Detecting Word Patterns- Useful Applications of Regular Expressions - Normalizing Text - Regular Expressions for Tokenizing Text - Segmentation –Formatting: From Lists to Strings - Writing Structured Programs : Back to the Basics- Sequences- Questions of Style - Functions: The Foundation of Structured Programming - Doing More with Functions- Program Development-Algorithm Design-A Sample of Python Libraries

UNIT III

CATEGORIZING AND TAGGING WORDS: Using a Tagger-Tagged Corpora-Mapping Words to Properties Using Python Dictionaries-Automatic Tagging- N-Gram Tagging-Transformation-Based Tagging- How to Determine the Category of a Word- Learning to Classify Text: Supervised Classification- Further Examples of Supervised Classification-Evaluation-Decision Trees-Naive Bayes Classifiers-Maximum Entropy Classifiers-Modeling Linguistic Patterns.

UNIT IV

EXTRACTING INFORMATION FROM TEXT: Information Extraction- Chunking-Developing and Evaluating Chunkers- Recursion in Linguistic Structure- Named Entity Recognition - Relation Extraction- Analyzing Sentence Structure: Some Grammatical Dilemmas-What’s the Use of Syntax? -Context-Free Grammar-Parsing with Context-Free Grammar-Dependencies and Dependency Grammar-Grammar Development.

UNIT V

BUILDING FEATURE-BASED GRAMMARS- Grammatical Features-Processing Feature Structures-Extending a Feature-Based Grammar-Analyzing the Meaning of Sentences: Natural Language Understanding- Propositional Logic- First-Order Logic- The Semantics of English Sentences- Discourse Semantics.

TEXT BOOKS

1. Steven Bird, Ewan Klein and Edward Loper, ―Natural Language Processing with Python‖, First Edition, O‗Reilly Media, 2009.
   UNIT I: Chapter 1:
   1.1,1.2,1.3,1.4,1.5,2.1,2.2,2.3,2.4,2.5
   UNIT II: Chapter 2:
   3.1,3.2,3.3,3.4,3.5,3.6,3.7,3.8,3.9,4.1,4.2,4.3,4.4,4.5,4.6,4.7,4.8
   UNIT III: Chapter 3:
   5.1,5.2,5.3,5.4,5.5,5.6,5.7,6.1,6.2,6.3,6.4,6.5,6.6,6.7
   UNIT IV: Chapter 4:
   7.1,7.2,7.3,7.4,7.5,7.6,8.1,8.2,8.3,8.4,8.5,8.6
   UNIT V: Chapter 5:
   9.1,9.2,9.3,10.1,10.2,10.3,10.4,10.5

REFERENCE BOOKS

1. Daniel Jurafsky and James H Martin, "Speech and Language Processing: An introduction to Natural Language Processing, Computational Linguistics and Speech Recognition", Prentice Hall, 2nd Edition, 2008.
2. Akshar Bharati, Vineet Chaitanya, Rajeev Sangal, "Natural Language Processing – A Paninian Prespective", Prentice Hall of India, 2004
3. Clark, A., Fox, C., & Lappin, S. (Eds.). (2012). The handbook of computational linguistics and natural language processing (Vol. 118). John Wiley & Sons.
4. Bender, E. M., & Lascarides, A. (2019). Linguistic fundamentals for natural language processing ii: 100 essentials from semantics and pragmatics. Synthesis Lectures on Human Language Technologies, 12(3), 1-268.
5. Hapke, H. M., Lane, H., & Howard, C. (2019). Natural language processing in action.
6. Indurkhya, N., &Damerau, F. J. (2010). Handbook of natural language processing. Chapman and Hall/CRC.
7. James Allen, Natural Language Understanding, Pearson Education, 3rd edt., 2005.

COURSE OUTCOMES

Upon completion of the course, the students are expected to

CO1	To understand how key concepts from NLP and linguistics are used to describe and analyze language.	**K1**	**HO**
CO2	To Learn about the data structures and algorithms used in NLP.	**K2**	**IO**
CO3	To Analyze data stored in standard formats.	**K3**	**HO**
CO4	To get familiar with the methods and algorithms used to process different types of textual data.	**K4**	**IO**
CO5	To know how to extract grammatical features and to understand the basics of first order logic and propositional logic.	**K5,K6**	**HO**

K1- REMEMBER, K2- UNDERSTAND, K3- APPLY, K4- ANALYZE, K5- EVALUATE, K6- CREATE

MAPPING WITH PROGRAMME OUTCOMES

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	M	M	S	M	S	M	S	M	S	M	S
CO2	S	M	S	M	S	S	M	M	M	M	S	M
CO3	M	M	S	S	M	M	S	S	S	S	S	M
CO4	S	M	S	M	S	M	M	S	S	M	S	M
CO5	S	S	M	M	S	S	M	S	M	M	S	M

S- STRONG; M-MEDIUM; L-LOW

ELECTIVE COURSES

Elective Course – 22UPCSC1E01 / 22UPCSC2E01 Credits: 3

Data Engineering and Management

Course Objectives:

- To understand Data Management concepts
- To get brief knowledge on Data Modelling
- To analyse the techniques used in Distributed Databases
- To assess Distributed database and Business Modeling
- To get familiar with CRM tools

Unit-I

DATABASE DEVELOPMENT: Database architecture of an information system-Overview of the database development process-Conceptual data modeling-Relational data analysis-Roles of a data model-Physical database design. DATA MANAGEMENT: Problems encountered without data management-Data management responsibilities-Data management activities-Roles within data management-Benefits of data management-Relationship between data management and enterprise

Unit-II

CORPORATE DATA MODELLING: Need for a corporate data model-Nature of a corporate data model- Develop a corporate data model - Corporate data model principles. DATA DEFINITION AND NAMING: Elements of a data definition-Data naming conventions. DATA QUALITY: Issues associated with poor data quality-Causes of poor data quality-Dimensions of data quality-Data model quality-Improving data quality. DATA ACCESSIBILITY: Data security-Data integrity-Data recovery

Unit-III:

USE OF PACKAGED APPLICATION SOFTWARE: Application software packages-Impact on data management. DISTRIBUTED DATA AND DATABASES: Rationale for distributing data-Perfect distributed database system-Top down fragmentation and partitioning. Bottom up integration-The management of replication. BUSINESS INTELLIGENCE: Data warehousing-Multidimensional model of data-Standard reporting tools-Online analytical processing OLAP-Relational schema for a data warehouse.

Unit-IV:

CRM: Three main pillars of CRM. GETTING TO KNOW YOUR CUSTOMER: 360-degree client view. UTILIZING ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING IN YOUR CRM STRATEGY: Evolution of AI-Current state of AI-Teaming up AI with people-Applying AI to your CRM solution-ethical aspects of AI-An example of AI in CRM processes.

Unit-V:

CLOUD VERSUS ON PREMISE VERSUS HYBRID: Factors influencing vendor selection-Hybrid deployment-what are your options. CRM DIFFERENTIATORS: It’s not about the feature list; it’s about the ecosystem-Fourth industrial revolution and CRM-AI and smart cloud-To cloud or not to cloud-Leveraging smart cloud into CRM-Big data-Social selling and advertising-Implementation tools-Sustainable CRM platform.

Text Books:

1. Keith Gordon, “Principles of Data Management Facilitating Information Sharing”, BCS Learning, 2013. (Chapters:1-5, 7,8,12,13,14)
2. Max Fatouretchi, “The Art of CRM”, Packt Publishing, 2019.(Chapters: 1,2,5,8,9)

Reference Books:

1. Peter Ghavami, “Big Data Management_ Data Governance Principles for Big Data Analytics”, De Gruyter, 2020.
2. Francis Buttle, Stan Maklan, Customer Relationship Management Concepts and Technologies, Routledge, 2019.

Course Outcome:

On the successful completion of the course, students will be able to,

CO1	Comprehend the Data Management concepts and analyse the relationship with the enterprise	
CO2	Analyze Data Modelling concepts and assess its quality	
CO3	Understand and implement business modelling techniques	
CO4	Evaluate the use of Artificial Intelligence and Machine Learning in CRM	
CO5	Develop CRM applications in cloud	

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	S	S	M	S	M	S	S	S	S	M	S	S
CO2	S	S	S	M	S	S	S	S	S	S	M	S
CO3	S	M	S	S	M	S	M	S	S	M	S	S
CO4	S	S	S	S	S	S	S	M	S	S	M	S
CO5	S	S	S	S	S	S	S	S	S	M	M	S

S- Strong; M-Medium; L-Low

Elective Lab – 22UPCSC1E02 **Credit: 1**

Data Engineering and Management Lab

Course Objectives:

- To acquire basic scripting knowledge in MongoDB
- To learn CRUD Operation on MongoDB database
- To realize MongoDB using DbVisualizer
- To be familiar with Zoho CRM features
- To customize your application using Zoho CRM

1. Write a script to create a MongoDB database and perform insert operation
2. Write a MongoDB script to perform query operations
3. Write a MongoDB Script to perform update operations
4. Write a MongoDB Script to update documents with aggregation pipeline
5. Write a MongoDB script to delete single and multiple documents
6. Write a MongoDB script to perform string aggregation operations
7. Design a Data Model for MongoDB using DbVisualizer
8. Perform CRUD operations using DbVisualizer
9. Create a Zoho CRM account and organize your Tasks, Meetings and Deals
10. Create and maintain a project using Zoho CRM features

Course Outcomes

On the successful completion of the course, students will be able to

CO1	Comprehend the scripting knowledge in MongoDB and perform basi operations in shell prompt	**K1 – K6**
CO1	Implement, Create, Read, Update and Delete Operations on MongoDB database	
CO3	Analyze MongoDB using DbVisualizer	
CO4	Assess Zoho CRM features for managing the customer relationships	
CO5	Create a customized application in Zoho CRM	

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	S	S	S	M	S	S	S	M	S	S
CO2	S	S	S	S	S	S	S	S	S	S	L	S
CO3	S	M	S	S	M	S	M	S	S	S	S	S
CO4	S	S	S	M	S	S	S	L	S	S	M	S
CO5	S	S	S	S	M	S	S	S	S	S	S	S

S- Strong; M-Medium; L-Low

Course Code-22UPCSC2E03 /22UPCSC1E03 **Credits: 2**

Architecture and Frameworks

Course Objectives

- To understand the basics, benefits and purpose of software architecture
- Understand the quality attributes to fulfil the software requirements and relates the software with an organization
- Explore the design patterns, best practice and paradigms of efficient software development
- Understand the performance and security measures of software architecture
- Enable the developers to advance their carrier in software domain

Unit – I

Software architecture introduction – Importance of Software architecture – Software architecture consumers – Architect role - software architecture in an organization – Types of software architects – Software development methodologies – Project management – Office politics – Software risk management – Configuration management – Software product lines

Unit – II

Domain Knowledge – Developing business acumen – Domain-driven design – requirement engineering – requirement elicitation –Software Quality attributes: Maintainability – Usability –Availability – Portability – Interoperability - Testability

Unit – III

Software Architectures design – Importance - Top-down Versus bottom-up design approaches – Architectural drivers – Documenting the Software architecture design – Systematic approach - Attribute-driven design – Microsoft's technique for architecture and design –Architecture-centric design method – Architecture development method – Tracking the progress of the software architecture's design.

Unit – IV

Designing orthogonal software systems – Minimizing complexity – SOLID design principles – Software architecture patterns – layered – Event-driven architecture – Model-View patterns – Service-oriented architecture

Unit – V

Architecting Modern Applications.- Importance of Performance – Performance improvement - Server side caching – Web application performance – Database performance -Securing software systems – Threat modelling – Secure by design

Text Book

1. Joseph Ingeno, "Software Architect's Handbook" Packt Publishing 2018.

Reference books

1. Oliver Vogel, Indo Arnold, ArifChughtai andTImoKehrer, "Software Architecture" Springer-Verlag, 2011.
2. Ian Gorton, "Essential Software architecture", Second Edition, Springer, 2011
3. Len Bass, Paul Clements and Rick Kazman, "Software architecture in practice", Third edition, Addison-Wesley, 2013

Course Outcomes

On the successful completion of the course, students will be able to

CO1	Understand, analyze and evaluate the purpose of Software architecture and development methodologies with consideration of risk management	K1-K6
CO2	Comprehend, apply and evaluate the domain knowledge for software development process and determine the impact of quality attributes.	K1-K6
CO3	Understand, track and examine the systematic approach for various software design models with effective document process	K1-K6
CO4	Illustrate and summarize the functions of orthogonal systems with complexity, design principles and design pattern for software architecture	K1-K6
CO5	Comprehend, analyze and evaluate the performance and security measures for Server, Web and Database applications in order to create the secure software systems for various domain applications	K1-K6

K1- Remember, K2- Understand, K3- Apply , K4- Analyze, K5- evaluate and K6- Create

Mapping with Programme Outcome

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	S	-	L	M	S	M	M	L	S	-	-
CO2	M	S	-	M	M	L	S	S	M	M	S	-
CO3	S	M	-	S	S	M	S	M	M	S	M	-
CO4	S	M	L	S	M	L	S	L	S	M	S	M
CO5	M	S	M	L	S	L	M	S	L	S	-	M

S- Strong; M-Medium; L-Low

Course Code - 22UPCSC2E04 / 22UPCSC1E04 Credits: 2

Architecture and Frameworks Lab

Course Objectives

- To understand and implement the basic concepts of Software architecture and its functions.
- To acquire programming skills to develop Implement various technologies and services associated with network protocols along with the challenges of data transfer.
- Implement the importance and functioning of Routing Protocols over communication service.
- To acquire skills to connect two routers and any two switches.
- To comprehend related to SSH protocols and accessing the remote device.

Note: Use the solid servers and client specification for implementation https://github.com/solid/specification/.

Implement the following using Linux / Windows environments

1. Find the WebID profile document and display the necessary attributes
2. Set and access the primary authentications with account recovery mechanisms
3. Set and access the secondary authentications with account recovery mechanisms
4. Design authorization and web access control
5. Find the content representation
6. Reading resources from HTTP REST API and WebSockets API
7. Writing resources from HTTP REST API and WebSockets API
8. Data notification using Social Web App protocol
9. Managing subscriptions and friends list using Social Web App protocol
10. Managing list of followers and following list using Social Web App protocol

Course Outcomes

On the successful completion of the course, students will be able to

CO1:	Comprehend the programming skills of Software architecture tools anc packages	K1-K6
CO2:	Understand and implement the user profiles and authentication witl recovery mechanism.	K1-K6
CO3:	Comprehend and evaluate the access control and conten representation use of FTP server	K1-K6
CO4:	Understand and implement reading and writing resources for variou applications	K1-K6
CO5:	Identify and examine the notifications, friends, and follower list of socia application protocols.	K1-K6

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5-Evaluate, K6-Create

Mapping Course outcomes with Programme outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	M	L	M	S	-	-	-	-	-	-
CO2	S	M	S	S	S	M	-	-	-	-	-	-
CO3	S	M	S	M	S	M	-	-	-	-	-	-
CO4	S	M	L	S	M	L	-	-	-	-	-	-
CO5	M	S	M	L	S	L	-	-	-	-	-	-

S- Strong; M-Medium; L-Low

Course Code - 22UPCSC2E05 / 22UPCSC1E05 Credits: 3

Network Protocols

Course Objectives

- To understand the basic concepts of Transmission Control Protocol/Internet Protocol and associated functions
- Explore to describe the internet architecture and its processes associated with the data transfer and to provide the quality of service
- To understand technologies and services associated with network protocols along along with the challenges of dat transfer.
- Learners will understand the importance and functioning of Routing Protocols over communication service.
- Empower the learners to comprehend and manage the issues associated with IP protocols like data traffic problems, security and mobility.

Unit I: Transmission Control Protocol/Internet Protocol : Fundamental Architecture - Internet Protocol Basics - Routing - Transport-Layer Protocols : Transmission Control Protocol - User Datagram Protocol - Stream Control Transmission Protocol - Real-Time Transport Protocol.

Unit II: Internet Architecture: Internet Exchange Point - History of Internet Exchange Points - Internet Service Provider Interconnection Relationships - Peering and Transit - IP Routing Protocols: Overview of Routing Protocols - Routing Information Protocol - Open Shortest Path First - Border Gateway Protocol - Multiprotocol Label Switching.

Unit III: IP Quality Of Service : Introduction - Quality of Service in IP Version 4 - Integrated Services - Differentiated Services - Quality of Service with Nested Differentiated Services Levels - IP Multicast and Any cast: Addressing - Multicast Routing - Routing Protocols –Any casting- IPv6 Any cast Routing Protocol: Protocol Independent Any cast—Sparse Mode - Transport over Packet: Draft-Martini Signaling and Encapsulation - Layer-2 Tunneling Protocol.

Unit IV: Virtual Private Wired Service - Types of Private Wire Services - Generic Routing Encapsulation - Layer-2 Tunneling Protocol - Layer-3 Virtual Private Network 2547bis, Virtual Router - IP and Optical Networking: IP/Optical Network Evolution - Challenges in Legacy Traditional IP/Optical Networks - Automated Provisioning in IP/Optical Networks - Control Plane Models for IP/Optical Networking - Next-Generation MultiLayer Network Design Requirements - Benefits and Challenges in IP/Optical Networking - IP Version 6: Addresses in IP Version 6 - IP Packet Headers - IP Address Resolution - IP Version 6 Deployment: Drivers and Impediments.

Unit V: IP Traffic Engineering: Models of Traffic Demands - Optimal Routing with Multiprotocol Label Switching - Link-Weight Optimization with Open Shortest Path First - Extended Shortest-Path-Based Routing Schemes - IP Network Security: Introduction - Detection of Denial-of-Service Attack - IP Trace back- Edge Sampling Scheme - Advanced Marking Scheme - Mobility Support for IP: Mobility Management Approaches - Security Threats Related to IP Mobility - Mobility Support in IPv6 - Reactive Versus Proactive Mobility Support - Relation to Multihoming - Protocols Supplementing.

Text Book :

1. "Advanced Internet Protocols, Services and Applications", Eiji Oki, Roberto Rojas-Cessa, Mallikarjun Tatipamula, Christian Vogt, Copyright © 2012 by John Wiley & Sons, Inc.

Reference Books :

1. "TCP/IP Protocol Suite", BehrouzA. Forouzan, Fourth Edition, Tata Mcgraw-Hill Edition 2010.
2. "Computer Communications and Networking Technologies" - Michael A. Gallo & William M. Hancock- BROOKS&COLE
3. "Computer Networks and Internets" -Douglas E. Comer- PEARSON.
4. Data and Computer Communications- Eighth Edition- William Stallings- Pearson Education.
5. Network Security Bible, 2nd edition, Eric Cole, Wiley Publishers.
6. Data communication and networks –James Irvine and David Harley- Publishers: Wiley India

Course Outcomes:

On the successful completion of the course, students will be able to

CO1	Understand, analyse and examine the concepts of Communicatio Protocols with its architecture and functions	K1- K6
CO2	Illustrate and apply the appropriate internet architecture along with efficien protocol models for the user defined communication environment	K1- K6
CO3	Comprehend, categorize and formulate the appropriate IP routing protoco to establish a efficient data transfer	K1- K6
CO4	Comprehend, analyse and evaluate the concepts of Virtual wired servic and IP/optical networking with its functions and deployment	K1- K6
CO5	Elucidate, analyse and inspect the IP traffic engineering and its model along with the security mechanisms	K1- K6

K1- Remember, K2- Understand, K3- Apply , K4- Analyze, K5- evaluate and K6- Create

Mapping with Programme Outcome

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	S	-	L	M	S	M	M	-	S	-	-
CO2	S	M	-	S	M	L	M	S	-	M	-	-
CO3	S	S	-	M	S	S	S	M	-	S	-	-
CO4	S	M	L	S	M	L	S	L	-	M	L	-
CO5	M	S	M	L	S	L	M	S	-	S	L	

S- Strong; M-Medium; L-Low

Course Code - 22UPCSC2E06 / 22UPCSC1E06 Credits: 2

Network Protocols Lab

Course Objectives

- To understand and implement the basic concepts of Transmission Control Protocol/Internet Protocol and associated functions.
- To acquire programming skills in Implement various technologies and services associated with network protocols along with the challenges of data transfer.
- Implement the importance and functioning of Routing Protocols over communication service.
- To acquire skills to connect two routers and any two switches.
- To comprehend related to SSH protocols and accessing the remote device.

Implement the following using Linux / Windows environments

1. Implement the following commands
    a. ipconfig
    b. ping
    c. traceroute
    d. netsat
    e. nslookup
2. Implement the following server commands
    a. ifconfig
    b. ip
    c. tracepath
    d. ss
    e. tcpdum
3. Connect and place the given file in the FTP server
4. Install packet tracer and connect a computer to router, switch and get a Icmp request
5. Implement the SSH protocols and accessing the remote device
6. Connect any two switches and get the status of each switches
7. Connect two routers and get packets from the routers.
8. Get the access of the router by connecting with working computer
9. Identify the route password of server and get the connection using telnet
10. Install wire shark for capture and analyse the packets (TCP /UDP)

Course Outcomes

On the successful completion of the course, students will be able to

CO1:	Comprehend the programming skills the SSH protocols an accessing the remote device	K1-K6
CO2:	Understand and implement the various functioning of Routin Protocols over communication service.	K1-K6
CO3:	Evaluate the use of FTP server	K1-K6
CO4:	Design to Connect any two switches and get the status of eac switches	K1-K6
CO5:	Solve to Connect two routers and get packets from the routers.	K1-K6

K1- Remember, K2- Understand, K3- Apply , K4- Analyze, K5-Evaluate, K6- Create

Mapping Course outcomes with Programme outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	-	-	-	-	L	-	-	-	-	-	-
CO2	S	-	M	-	M	L	-	-	-	-	-	-
CO3	S	-	S	-	S	L	-	-	-	S	S	S
CO4	S	-	S	-	S	L	-	-	-	S	S	S
CO5	S	-	S	-	S	L	-	-	-	S	S	S

S- Strong; M-Medium; L-Low

Course Code - 22UPCSC2E07 / 22UPCSC1E07 **Credits: 3**

Dot NET Technologies

Course Objectives:

- To get strong understanding of .NET Framework and C# programming.
- To get advanced programming skills in Visual Studio with C# language.
- To get advanced methods of manipulating data using Microsoft SQL Server.
- To get clear idea of how to developing real-time standalone, web applications using .NET Technologies.
- To get clear understanding and get experience in Microsoft Azure.

Unit-I

Introducing C#: .NET Framework - C# language - Visual Studio 2017 - Writing a C# Program: Visual Studio 2017 Development Environment - Console Applications - Desktop Applications - Variables and Expressions: Basic C# Syntax - Basic C# Console Application Structure - Variables - Expressions - Flow Control: Boolean Logic – Branching - Looping.

Unit-II

More About Variables: Type Conversion - Complex Variable Types - String Manipulation – Functions: Defining and Using Functions - Variable Scope - The Main Function - Struct Functions - Overloading Functions - Using Delegates - Debugging and Error Handling: Debugging in Visual Studio - Error Handling - Introduction to Object Oriented Programming: Object-Oriented Programming - OOP Techniques - OOP in Desktop Applications.

Unit-III

Defining Classes: Class Definitions in C# - System.Object - Constructors and Destructors - OOP Tools in Visual Studio - Class Library Projects - Interfaces Versus Abstract Classes - Struct Types - Shallow Copying Versus Deep Copying - Defining Class Members: Member Definitions - Additional Class Member Topics - Interface Implementation - Partial Class Definitions - Partial Method Definitions - The Call Hierarchy Window - Basic Cloud Programming: Cloud, Cloud Computing, and the Cloud Optimized Stack - Cloud Patterns and Best Practices - Using Microsoft Azure C# Libraries to Create a Storage Container - Creating an ASP.NET 4.7 Web Site That Uses the Storage Container - Advanced Cloud Programming and Deployment: Creating an ASP.NET Web API - Deploying and Consuming an ASP.NET Web API on Microsoft Azure - Scaling an ASP.NET Web API on Microsoft Azure.

Unit-IV

.NET Standard and .NET Core: Cross-Platform Basics and Must Know Terms – Need of .NET - Referencing and Targeting Frameworks - .NET Core - Building and Packaging a.NET Standard Library - Building a .NET Core Application with Visual Studio - Porting from .NET Framework to .NET Core - ASP.NET and ASP.NET Core: Overview of Web Applications – Use of ASP.NET - ASP.NET Web Forms - Creating ASP.NET Core Web Applications – Files: File Classes for Input and Output –

Streams - Monitoring the File System - XML and JSON: XML Basics - JSON Basics - XML Schemas - XML Document Object Model - Converting XML to JSON - Searching XML with XPath.

Unit-V

LINQ: LINQ to XML - LINQ Providers - LINQ Query Syntax - LINQ Method Syntax - Ordering Query Results - Understanding the orderby Clause - Querying a Large Data Set -Using Aggregate Operators - Using the Select Distinct Query - Ordering by Multiple Levels -Using Group Queries - Using Joins – Databases: Using Databases - Installing SQL Server - Express - Entity Framework - Code First Database - Finding the Database - Navigating Database Relationships - Handling Migrations - Creating and Querying XML from an Existing Database - Universal Apps: Windows Universal Apps - App Concepts and Design - App Development - Common Elements of Windows Store Apps - Windows Store.

Text book:

1. Benjamin Perkins, Jacob Vibe Hammer, Jon D. Reid, "Beginning C#7 Programming with Visual Studio 2017", Wiley Publishing, 2018. Chapters: 1 to 10, 16 to 23, and 25.

Reference books:

1. Nagel, Christian, "Professional C 7 and .NET Core 2.0", Wrox Publishing, 2018.
2. Mehboob Ahmed Khan, Ovais, "C# 7 and .NET Core 2.0 High Performance", Packt Publishing, 2018

Course Outcome

On the successful completion of the course, students will be able to,

CO1	Understand and learn .NET Framework an C# .NET	K1, K2	LO
CO2	Apply the concepts to develop th applications for real-time problem in C# .NE and ASP .NET	K3	IO
CO3	Analyze the feasibility of using .NET for rea time problems	K4,K5	HO

K1- Remember, K2 - Understand, K3 - Apply, K4 - Analyze, K5 - Evaluate, K6 - Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	L	M	L	L	L	-	-	-	-	-	L
CO2	S	M	L	M	M	L	-	L	-	L	-	L
CO3	S	S	S	M	M	L	-	L	-	L	-	L

L - Low, M- Medium, S - Strong

Course Code - 22UPCSC2E08 /22UPCSC1E08 Credits: 1

Dot NET Technologies Lab

Course Objectives:

- To get strong understanding of .NET Framework and C# programming.
- To get advanced programming skills in C# .NET OOPs Concepts
- To get advanced methods of manipulating data using Microsoft SQL Server.
- To get clear idea of how to developing real-time standalone, web applications using ASP .NET.
- To get clear understanding and get experience in Microsoft Azure.

Implement the following problems using C# with Visual Studio 2017

1. Demonstrate method overloading and method overriding
2. Class and Objects
3. Multilevel Inheritance
4. Interfaces
5. Demonstrate multiple type of Exceptions
6. Azure Storage Container Using the Microsoft Azure Storage Client Library
7. Demonstrate Read and Write a Data using Random Access Files
8. Employee management database using LINQ
9. Student management system using ASP.NET
10. Demonstrates simple Universal App.

Course Outcome:

- Get a strong understanding of .NET Visual Studio platform
- Become a strong knowledge in C# .NET.
- Getting real-time application developing using .NET Cloud Technologies.

Course Code - 22UPCSC2E09 / 22UPCSC1E09 Credits: 3

Software Development Technologies

Course Objectives

- To learn and Implementing Micro services
- To analysing the Azure Kubernetes Service
- To learn and anlyse .NET DevOps for Azure and its applications
- To building code for .NET core applications
- To get familiarized with Azure pipelines

Unit -I

Implementing Microservices: Client to microservices communication, Interservice communication, data considerations, security, monitoring, microservices hosting platform options. Azure Service Fabric: Introduction, core concepts, supported programming models, service fabric clusters, develop and deploy applications of service fabric. Monitoring Azure Service Fabric Clusters: Azure application, resource manager template, Adding Application Monitoring to a Stateless Service Using Application Insights, Cluster monitoring, Infrastructure monitoring.

Unit-II

Azure Kubernetes Service (AKS): Introduction to kubernetes and AKS, AKS development tools, Deploy applications on AKS. Monitoring AKS: Monitoring, Azure monitor and analytics, monitoring AKS clusters, native kubernetes dashboard, Prometheus and Grafana. Securing Micro services: Authentication in micro services, Implanting security using API gateway pattern, Creating application using Ocrlot and securing APIs with Azure AD. Database Design for Micro services: Data stores, monolithic approach, Micro services approach, harnessing cloud computing, database options on MS Azure, overcoming application development challenges. Building Micro services on Azure Stack: Azure stack, Offering IaaS, PaaS on-premises simplified, SaaS on Azure stack.

Unit-III

.NET DevOps for Azure: DevOps introduction, Problem and solution. Professional Grade DevOps Environment: The state of DevOps, professional grade DevOps vision, DevOps architecture, tools for professional DevOps environment, DevOps centered application. Tracking work: Process template, Types of work items, Customizing the process, Working with the process. Tracking code: Number of repositories, Git repository, structure, branching pattern, Azure repos configuration, Git and Azure.

Unit-IV

Building the code: Structure of build, using builds with .NET core and Azure pipelines, Validating the code: Strategy for defect detection, Implementing defect detection. Release candidate creation: Designing release candidate architecture, Azure artifacts workflow for release candidates, Deploying the release: Designing deployment pipeline, Implementing deployment in Azure pipelines. Operating and monitoring release: Principles, Architectures for observability, Jumpstarting observability.

Unit-V

Introduction to APIs: Introduction, API economy, APIs in public sector. API Strategy and Architecture: API Strategy, API value chain, API architecture, API management. API Development: Considerations, Standards, kick-start API development, team orientation. API Gateways: API Gateways in public cloud, Azure API management, AWS API gateway. API Security: Request-based security, Authentication and authorization.

Text Book

1. Harsh Chawla and Hemant Kathuria, Building Microservices Applications on Microsoft Azure- Designing, Developing, Deploying, and Monitoring, Apress, 2019.
2. Jeffrey Palermo , NET DevOps for Azure A Developer's Guide to DevOps Architecture the Right Way, Apress, 2019.
3. Thurupathan and Vijayakumar, Practical API Architecture and Development with Azure and AWS - Design and Implementation of APIs for the Cloud, Apress, 2018.

Reference Books

1. Karl Matthias and Sean P. Kane, Docker: Up and Running, O'Reilly Publication, Second Edition 2018.
2. Len Bass,Ingo Weber,Liming Zhu,"DevOps, A Software Architects Perspective", AddisonWesley-Pearson Publication, First Ediiton 2015.
3. John Ferguson Smart,"Jenkins, The Definitive Guide", O'Reilly Publication, First Ediiton 2011.

Course Outcomes

On the successful completion of the course, students will be able to

CO1	To understand, apply and summarize the basic concepts of Micro services communication Microsoft Azure and Dev Ops for software development life cycle	K1-K6
CO2	To illustrate, and implement Azure Kubernetes Service tools for software development life cycle	K1-K6
CO3	To recognize, analyse and summarize the functionalities of .NET Dev Ops for Azure applications	K1-K6
CO4	To understand, design and evaluate the principles and architecture service tools for software development life cycle.	K1-K6
CO5	To comprehend, implement and review the functionalities of API and API gateways for cloud and Azure applications	K1-K6

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	M	-	M	-	-	-	S	-	-	-
CO2	S	S	M	-	M	-	-	-	S	-	-	-
CO3	S	S	S	-	S	-	-	-	S	S	S	S
CO4	S	S	M	-	M	-	-	-	S	-	-	-
CO5	S	S	M	-	M	-	-	-	S	-	-	-

S- Strong; M-Medium; L-Low

Course Code-22UPCSC2E10 / 22UPCSC1E10 **Credits: 2**

Software Development Technologies Lab

Course Objectives

- To understand the concept of DevOps with associated technologies and methodologies.
- To be familiarized with Jenkins, which is used to build & test software Applications
- To understand Continuous integration in Devops environment.
- To understand Docker to build, ship and run containerized images
- To use Docker to deploy and manage Software applications running on Container.

List of Programs

1. Deploy Version Control System / Source Code Management, install git and create a GitHub account.
2. Perform various GIT operations on local and Remote repositories using GIT Cheat-Sheet
3. Continuous Integration: install and configure Jenkins with Maven/Ant/Gradle to setup a build Job.
4. Build the pipeline of jobs using Maven / Gradle / Ant in Jenkins, create a pipeline script to Test and deploy an application over the tomcat server.
5. Implement Jenkins Master-Slave Architecture and scale your Jenkins standalone implementation by implementing slave nodes.
6. Setup and Run Selenium Tests in Jenkins Using Maven.
7. Implement Docker Architecture and Container Life Cycle, install Docker and execute docker commands to manage images and interact with containers.
8. Implement Dockerfile instructions, build an image for a sample web application using Dockerfile.
9. Install and Configure Pull based Software Configuration Management and provisioning tools using Puppet.
10. Implement LAMP/MEAN Stack using Puppet Manifest.

Course Outcomes

On the successful completion of the course, students will be able to

CO1:	To Understand and analyse the importance of Jenkins to Build Deploy and Test Software Applications	K1-K6
CO2:	To synthesis and summarize the importance of Software Configuration Management in DevOps	K1-K6
CO3:	To identify, analyze and illustrate the Containerization of OS images and deployment of applications over Docker	K1-K6
CO4	To design, analyze and develop the Pull based Software Configuration Management	K1-K6
CO5	To design, analyze and develop Puppet Manifest	K1-K6

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate,

K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	M	-	M	-	-	-	S	S	S	S
CO2	S	S	M	-	M	-	-	-	S	S	S	S
CO3	S	S	M	-	M	-	-	-	S	S	S	S
CO2	S	S	M	-	M	-	-	-	S	S	S	S
CO3	S	S	M	-	M	-	-	-	S	S	S	S

S- Strong; M-Medium; L-Low

COURSE - 22UPCSC2E11 / 22UPCSC1E11 CREDITS: 4

SOFT COMPUTING

COURSE OBJECTIVES:

- Develop the skills to gain a basic understanding of neural network theory and fuzzy logic theory.
- To understand supervised and unsupervised learning algorithms
- To enable the students to gain a basic understanding of neural networks.
- To know about fuzzy logic, fuzzy inference systems, and their functions.
- To impart basic knowledge on Genetic algorithms and their applications.

UNIT-I

INTRODUCTION TO SOFT COMPUTING: Artificial Neural Networks- Biological Neurons- Basic Models of Artificial Neural Networks-Connections-Learning-Activation Functions- Important Terminologies of ANNs- Muculloch and Pitts Neuron-Linear Separability- Hebb Network-Flowchart of Training Process-Training Algorithm.

UNIT - II

SUPERVISED LEARNING NETWORK : Perceptron Networks–Perceptron Learning Rule-Architecture-Flowchart for Training Process-Perceptron Training Algorithms for Single Output Classes-Perceptron Training Algorithm for Multiple Output Classes-Perceptron Network Testing Algorithm - Adaptive Linear Neuron-Delta Rule for Single Output Unit-Flowchart for training algorithm-Training Algorithm – Testing Algorithm - Multiple Adaptive Linear Neurons-Architecture-Flowchart of Training Process-Training Algorithm-Back Propagation Network-Architecture-Flowchart for Training Process-Training Algorithm-Learning Factors of Back-Propagation Network-Radial Basis Function Network- Architecture-Flowchart for Training Process-Training Algorithm.

UNIT-III

UNSUPERVISED LEARNING NETWORK: Associative Memory Networks - Auto Associative Memory Network**-**Architecture-Flowchart for Training Process-Training Algorithm-Testing Algorithm- Bidirectional Associative Memory- Architecture-Discrete Bidirectional Associative Memory-Iterative Auto Associative Memory Networks - Linear AutoAssociative Memory-Kohonen Self-Organizing Feature Map-Architecture-Flowchart for Training Process-Training Algorithm.

UNIT-IV

INTRODUCTION TO FUZZY LOGIC: Classical Sets –Operations on Classical Sets-Fuzzy sets - Fuzzy Sets- Properties of Fuzzy Sets- Fuzzy Relations –Membership Functions: Fuzzification- Methods of Membership Value Assignments – Defuzzification – Lambda-Cuts for Fuzzy sets and Fuzzy Relations – Defuzzification

Methods–Max-Membership Principle-Centroid Method-Weighted Average Method-Mean Max Membership-Center of Sums-Center of Largest Area-First of Maxima - Fuzzy Set Theory - Fuzzy Arithmetic And Fuzzy Measures: Fuzzy Measures – Belief and Plausibility Measures-Probability Measures-Possibility and Necessity Measures-Formation of Rules –Fuzzy Inference Systems (FIS) – Fuzzy Decision Making – Fuzzy Logic Control Systems.

UNIT-V

GENETIC ALGORITHM: Introduction - Biological Background - Traditional Optimization and Search Techniques -Gradient Based Local Optimization Method-Random Search-Stochastic Hill Climbing-Simulated Annealing-Symbolic Artificial Intelligence-Operators in Genetic Algorithm -Encoding-Selection-Crossover-Mutation - Stopping Conditions for Genetic Algorithm Flow-Genetic Programming-Working of Genetic Programming-Characteristics of Genetic Programming-Data Representation.

TEXT BOOKS

1. Principles of Soft Computing, S.N. Sivanandam, S.N.Deepa, Wiley, Third Edition, 2019.

    UNIT I: Chapter 1: 2.1,2.3,2.4,2.5,2.6,2.7
    UNIT II: Chapter 2: 3.2,3.3,3.4,3.5,3.6
    UNIT III: Chapter 3: 4.3,4.4,4.7,5.3
    UNIT IV: Chapter 4: 7.2,7.3,8.4,9.3,9.4,10,10.2,10.3,10.4,11.4,12.8,14
    UNIT V: Chapter 5: 15,15.2,15.3,15.4,15.9,15.10

REFERENCE BOOKS

1. Das, A. (2018). Artificial Intelligence and Soft Computing for Beginners.
2. Amit, K. (2018). Artificial intelligence and soft computing: behavioral and cognitive modeling of the human brain. CRC press.
3. Rajasekaran, S., &Pai, G. V. (2011). Neural networks, fuzzy logic and genetic algorithm: synthesis and applications (with cd). PHI Learning Pvt. Ltd.
4. Jang, J. S. R., Sun, C. T., &Mizutani, E. (2004). Neuro-fuzzy and soft computing-a computational approach to learning and machine intelligence [Book Review]. IEEE Transactions on automatic control, 42(10), 1482-1484.
5. Gupta, M. M. (2004). Soft computing and intelligent systems: theory and applications. Elsevier.
6. Jang, J. S. R., Sun, C. T., &Mizutani, E. (1997). Neuro-fuzzy and soft computing-a computational approach to learning and machine intelligence [Book Review]. IEEE Transactions on automatic control, 42(10), 1482-1484.

COURSE OUTCOMES:

On the successful completion of the course, students will be able to

CO1	To provide an introduction to the basic principles, techniques, and applications of soft computing	K- 1 K2	LO
CO2	To get familiar with Neural network architectures and supervised learning algorithms	K3	IO
CO3	To understand the architectures and algorithms of Unsupervised Learning techniques	K3- K4	HO
CO4	Develop the skills to gain a basic understanding of fuzzy logic theory and fuzzy inference systems	K4	IO
CO5	Ability to learn traditional optimization and search techniques and genetic programming	K5	HO

MAPPING WITH PROGRAMME OUTCOMES

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	S	M	S	M	S	S	S	M	S	M	S	S
CO2	M	S	M	S	M	M	M	S	M	S	S	M
CO3	M	M	S	S	S	M	M	S	S	S	S	S
CO4	S	S	M	M	M	S	S	S	S	M	M	M
CO5	S	S	S	S	S	M	S	M	M	S	S	M

S- STRONG; M-MEDIUM; L-LOW

COURSE-22UPCSC2E12 /22UPCSC1E12 Credits: 1

Soft Computing Lab

COURSE OBJECTIVES:

- To implement various Supervised Neural Network-based approaches
- To apply the fuzzy-based logical operations and arithmetic operations
- To implement unsupervised neural network approaches
- To solve a problem using a simple genetic algorithm
- To implement logic gates.

Program List

1. Implementation of Logic gates using Artificial Neural Network.
2. Implementation of Perception Algorithm.
3. Implementation of Back Propagation Algorithm.
4. Implementation of Self Organizing Maps.
5. Implementation of Radial Basis Function Network.
6. Implementation of De-Morgan’s Law.
7. Implementation of McCulloch Pits Artificial Neuron model
8. Implementation of Simple genetic algorithm
9. Implementation of fuzzy based Logical operations
10. Implementation of fuzzy based arithmetic operations

COURSE OUTCOMES

On the successful completion of the course, students will be able to

CO1	To apply supervised learning algorithms for real datasets	K- 1K2	LO
CO2	To implement Unsupervised Learning techniques	K3	IO
CO3	To apply fuzzy based arithmetic and logical operations	K3- K4	HO
CO4	To find solutions for problems using Genetic algorithm	K4	IO
CO5	To implement DeMorgan's Law	K5	HO

MAPPING WITH PROGRAMME OUTCOMES

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	M	M	S	M	S	S	S	M	S	M	S	S
CO2	M	S	M	M	M	M	M	S	M	S	M	M

CO3	M	M	S	S	S	M	M	S	S	S	S	S
CO4	S	S	S	M	M	S	S	M	S	S	S	M
CO5	S	S	S	S	S	M	M	M	M	S	M	M

S- STRONG; M-MEDIUM; L-LOW

Course Code: 22UPCSC2E13 / 22UPCSC1E13 Credits: 3

Social Networking

Course Objectives

- To learn about Social media, Social networking and Webcasts
- To understanding and building a Word Press Powered Website
- To analysis the Social Networking & Micro-Blogging.
- To learn and analysis the Widgets & Badges.
- To explore the importance of Website optimization.

UNIT I: Introduction: Social Media Strategy-Important First Decisions -Websites, Blogs - RSS Feeds Mapping -Preparation - Multimedia Items Gathering Content for Blog Posts RSS Feeds & Blogs-RSS Feeds-The Feed Reader-The Feed-Options for Creating an RSS Feed-Planning Feed-Blogs-Options for Starting. Blog and RSS Feed-Feed or Blog Content-Search Engine Optimization (SEO)-Feed Burner-RSS Feed and Blog Directories-An Optimization Plan for Blog or RSS Feed

UNIT II: Building a Word Press Powered Website: Word Press as A CMS - Diversity of Word Press Sites-The Anatomy of a Word Press Site -a Brief Look at the Word Press Dashboard Planning - Site Themes Plug-ins setting up Sidebars Building Pages- Posting Blog Entries. Podcasting, Vidcasting, & Webcasting- Publishing Options for Podcast- Creating and Uploading Podcast Episodes- Publishing Podcast Optimizing Podcast- Webcasting

UNIT III: Social Networking & Micro-Blogging: Facebook-The Facebook Profile - Myspace LinkedIn-Twitter-Niche Social Networking Sites-Creating Own Social Network-Promoting Social Networking Presence- Social Bookmarking & Crowd-Sourcing - Social Bookmarking-A Social Bookmarking Strategy- Crowd-Sourced News Sites- Preparation And Tracking Progress Media Communities-Image Sharing Sites-Image Sharing Strategy-Video Sharing Sites-Video Sharing Strategy-Searching And Search Engine Placement-Connecting With Others.

UNIT IV: Widgets & Badges: Highlighting Social Web Presence-Sharing And Syndicating Content Making Site More Interactive-Promoting Products And Making Money-Using Widgets In Word Press-Widget Communities And Directories-Working Widgets Into Strategy Social Media Newsrooms-Building Social Media Newsroom - Populating The Newsroom-Social Media News Releases-Social Media Newsroom Examples. More Social Tools-Social Calendars-Social Pages Wikis-Social Search Portals-Virtual Worlds.

Unit V: Website optimization: A Website Optimization Plan-Streamlining Web Presence-An Integration Plan- Looking to the Future-Life streaming: The Future of Blogging-Distributed Social Networking-Social Ranking, Relevancy, and —Defriending-Web 3.0 or The Semantic Web-Mobile Technology- Measuring Your Success-A Qualitative Framework-A Quantitative Framework-Tools to Help You Measure-Come To Your Own Conclusions

Text Book:

1. Deltina hay —A Survival Guide To social Media and Web 2.0 Optimization‖, Dalton Publishing, 2009

Reference Books:

1. Miriam Salpeter —Social Networking for Career Success‖ Learning Express, 2011.
2. Miles, Peggy, —Internet world guide to webcasting‖ Wiley, 2008 Professionals", Wiley Publication,2015.

Course Outcomes

On the successful completion of the course, students will be able to

CO1:	To understand, impart and summarize the concepts of Social media, Social networking and Webcasts	K1-K6
CO2:	To comprehend, design and develop a Word Press Powered Website	K1-K6
CO3:	To understand, implement and perform evaluation of Social Networking and Micro-Blogging	K1-K6
CO4	To collaborate, implement and analyse the Widgets and Badges in social networking environment	K1-K6
CO5	To understand, illustrate and perform evaluation of web optimization for social networks	K1-K6

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	S	-	S	L	-	S	-	S	S	S
CO2	S	S	S	-	S	L	-	S	-	S	S	S
CO3	S	S	S	-	S	L	-	S	-	S	S	S
CO4	S	S	S	-	S	L	-	S	-	S	S	S
CO5	S	S	S	-	S	L	-	S	-	S	S	S

S- Strong; M-Medium; L-Low

Course Code: 22UPCSC2E14 / 22UPCSC1E14 Credits: 1

Social Networks Lab

Course Objectives

- To familiarize the tools required to manage social network applications
- To analyze social networks like Facebook, LinkedIn, Google+, GitHub
- To teach the fundamental techniques and principles in achieving social networking environment.
- To enable students to have skills that will help them to solve real time applications.
- To get explore in the Github API.

List of Programs

1. Creating and Exploring Twitter's API
2. To analyzing and visualizing tweets and tweet entities with frequency analysis
3. Creating and Exploring Facebook's Social Graph API
4. To analyzing the Facebook's Social Graph connections
5. Creating and Exploring LinkedIn API
6. To downloading LinkedIn connections as a CSV file
7. Creating and Exploring Google+ API
8. To create and querying Human Language Data with TF-IDF
9. Creating and Exploring GitHub's API
10. To analyzing GitHub interest graph

Course Outcomes

On the successful completion of the course, students will be able to

CO1:	To understand , implement and review the fundamental technique and principles for social networks.	K1-K6
CO2:	To design and develop the programs using the tools required t develop and manage social network like Facebook, LinkedIn Google+, GitHub	K1-K6
CO3:	To create and explore the functionality of social networking tool such as GitHub	K1-K6
CO4	To understand , implement and review the fundamental principle for social network graph.	K1-K6
CO5	To comprehend and critically analyse the existing API for socia networks	K1-K6

K1- Remember, K2- Understand, K3- Apply , K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	M		M	S	-	-	-	S	-	-
CO2	S	M	S	S	S	M	-	-	-	S	-	-
CO3	S	S	S	S	S	S	-	-	-	S	S	S
CO4	S	M	S	S	S	M	-	-	-	S	-	-
CO5	S	S	S	S	S	S	-	-	-	S	S	S

S- Strong; M-Medium; L-Lo

Course Code: 22UPCSC2E15 / 22UPCSC1E15 Credits: 3

High Performance Computing

Course Objectives:

- To get a clear idea of High Performance Computing concept.
- To get brief knowledge about how to function the HPC systems.
- To get idea of what techniques used in HPC models.
- To understand a Parallel computing concepts.
- To get familiar with OpenMP technology that is widely used in HPC technology.

Unit-I

Modern processors: Stored-program computer architecture-General purpose cache based microprocessor architecture-Memory hierarchies-Multi core processors-Multithreaded processors-Vector processors. **Basic optimization techniques for serial code:** Scalar profiling-Common sense optimizations-Simple measures, large impact-The role of compilers-C++ optimizations.

Unit-II

Data access optimization: Balance analysis and light speed estimates-Storage order-Algorithm classification and access optimizations-The Jacobi algorithm-Algorithm classification and access optimizations-Sparse matrix-vector multiply. **Parallel computers:** Taxonomy of parallel computing paradigms-Shared-memory computers-Distributed memory computers-Hierarchical systems-Networks.

Unit-III

Basics of parallelization: Introduction to Parallelism -Parallel scalability. **Shared memory parallel programming with OpenMP:** Short introduction to OpenMP-OpenMP-parallel Jacobi algorithm.

Unit-IV

Efficient OpenMP programming: Profiling OpenMP programs-Performance pitfalls-Parallel sparse matrix-vector multiply. **Locality optimizations on ccNUMA architectures:** Locality of access on ccNUMA-ccNUMA optimization of sparse MVM-Placement pitfalls-ccNUMA issues with C++.

Unit-V

Distributed-memory parallel programming with MPI: Message passing-A short introduction to MPI-MPI parallelization of a Jacobi solver. **Efficient MPI programming:** MPI performance tools-Communication parameters-Synchronization, serialization, contention-Reducing communication overhead-Understanding intranode point-to-point communication.

Text book:

1. Georg Hager, Gerhard Wellein "Introduction to High Performance Computing for Scientists and Engineers", CRC Press, 2011. **Chapters:** 1 to 10.

Reference books:

1. Michael W. Berry, Kyle A. Gallivan, Efstratios Gallopoulos, Ananth Grama, Bernard Philippe, Yousef Saad, Faisal Saied, "High-performance scientific computing: algorithms and applications", Springer, 2012.
2. Victor Eijkhout, "Introduction to High Performance Scientific Computing", MIT Press, 2011.

Course Outcome:

On the successful completion of the course, students will be able to,

CO1	Understand of the HPC and ccNUMA concepts	K1 - K6
CO2	Design and develop a parallel programming with modern C, C++ and new version of FORTRAN	
CO3	Apply with parallel computing	
CO4	Develop an efficient OpenMP programming	
CO5	Evaluate an efficient MPI programming	

K1- Remember, K2 - Understand, K3 - Apply , K4 - Analyze, K5 - Evaluate, K6 -Create

Mapping with Programme Outcomes:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	L	M	L	L	L	S	S	S	S	M	L
CO2	S	M	L	M	M	L	S	L	S	L	S	L
CO3	S	S	S	M	M	L	M	L	M	L	S	L
CO4	S	S	S	M	S	L	M	L	M	S	S	S
CO5	S	S	S	M	M	L	M	M	M	M	S	L

L - Low, M- Medium, S - Strong

Course Code: 22UPCSC2E16 /22UPCSC1E16 Credits: 1

High Performance Computing Lab

Course Objectives:

- To understand concepts of High Performance Computing.
- To get brief knowledge about PB and Slurm.
- To understand techniques of OpenMP and OpenMPI.
- To understand Parallel computing concepts.
- To get familiar with CUDA.

(Implemented either PB, Slurm,OpenMP, OpenMPI, and CUDA)

1. Demo: - Access and best practices on HPC
2. Matrix multiplication with Job scheduling (PB or Slurm)
3. Vectors add with malloc shared
4. Vector add program with MPI
5. Hello world task for Multithreading with openMP
6. openMP shared memory on Host and Device
7. openMP Matrix Multiplication with parallelism and Barrier
8. openMP with Reduction on operands and aggregate functionality
9. Vector and Matrix multiplication on CUDA
10. Feed forward computing on CUDA

Course Outcome:

On the successful completion of the course, students will be able to,

CO1	Apply and Evaluate the HPC Programs	K1 - K6
CO2	Design and Develop a MPI Programs	
CO3	Design and Develop a different programming concepts of OpenMP	
CO4	Develop an efficient PB and Slurm programming	
CO5	Evaluate an efficient CUDA programming	

1- Remember, K2 - Understand, K3 - Apply , K4 - Analyze, K5 - Evaluate, K6 - Create

Mapping with Programme Outcomes:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	L	M	L	L	L	S	S	S	S	M	L
CO2	S	M	L	M	M	L	S	L	S	L	S	L
CO3	S	S	S	M	M	L	M	L	M	L	S	L
CO4	S	S	S	M	S	L	M	L	M	S	S	S
CO5	S	S	S	M	M	L	M	M	M	M	S	L

L - Low, M- Medium, S - Strong

Course Code: 22UPCSC2E17 / 22UPCSC1E17 Credits:3

Internet of Things

Course Objectives:

- To get familiar with the evolution of IOT with its design principles
- To outline the functionalities and protocols of internet communication
- To analyze the hardware and software components needed to construct IOT applications
- To identify the appropriate protocol for API construction and writing embedded code
- To realize various business models and ethics in Internet of Things

UNIT I FUNDAMENTALS OF IOT

Evolution of Internet of Things – Enabling Technologies – IOT Architectures: oneM2M, IOT World Forum (IOTWF) and Alternative IOT models – Simplified IOT Architecture and 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 IOT PROTOCOLS

IOT Access Technologies: Physical and MAC layers, topology and Security of IEEE 802.15.4, 802.15.4g, 802.15.4e, 1901.2a, 802.11ah and LoRaWAN – Network Layer: IP versions, Constrained Nodes and Constrained Networks – Optimizing IP for IOT: From 6LoWPAN to 6Lo, Routing over Low Power and Lossy Networks – Application Transport Methods: Supervisory Control and Data Acquisition – Application Layer Protocols: CoAP and MQTT.

UNIT – III DESIGN AND DEVELOPMENT

Prototyping Embedded Devices: Electronics - Embedded Computing Basics – Arduino - Raspberry Pi - Beagle Bone Black - Electric Imp. Prototyping the Physical Design: Non digital Methods - Laser Cutting - 3D printing - CNC Milling - Repurposing/Recycling.

UNIT – IV

Prototyping Online Components: Getting started with an API - Writing a New API - Real-Time Reactions - Other Protocols. Techniques for Writing Embedded Code: Memory Management - Performance and Battery Life – Libraries - Debugging.

UNIT – V

Business Models: History of Business Models – Model – Internet of Starting up – Lean Startups. Moving to Manufacture: Designing Kits - Designing Printed circuit boards – Certification – Costs - Scaling Up Software. Ethics: Privacy – Control – Environment – Solutions.

Text Books:

1. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Rob Barton and Jerome Henry, —IoT Fundamentals: Networking Technologies, Protocols and Use Cases for Internet of Things, Cisco Press, 2017 (UNIT I and II)
2. Adrian McEwen and Hakim Cassimally, "Designing the Internet of Things", Wiley, 2014. (UNIT III, IV and V)

Reference Books:

1. Ovidiu Vermesan and Peter Friess, "Internet of Things – From Research and Innovation to Market Deployement" , River Publishers, 2014.
2. Peter Waher, "Learning Internet of Things" ,Packt Publishing, 2015.
3. Donald Norris, "The Internet of Things: Do-It-Yourself at Home Projects for Arduino, Raspberry Pi and BeagleBone Black",McGraw Hill, 2015.

Course Outcomes

On the successful completion of the course, students will be able to

CO1:	Comprehend the IoT evolution with its architecture and sensors	K1- K6
CO2:	Understand the networking concepts for communication an underlying IoT protocols	
CO3:	Assess the embedded technologies and develop prototypes for th IoT products	
CO4:	Evaluate the use of Application Programming Interface and design a API for IoT in realitime	
CO5:	Recognize the ethics of business models and perform securit analysis	

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	M	S	S	S	S	S	S	S	S	S
CO2	S	S	S	S	S	S	S	M	S	S	M	S
CO3	S	M	S	S	S	S	M	S	S	M	S	S
CO4	S	S	S	S	S	S	S	S	S	S	M	S
CO5	S	S	S	S	M	S	S	S	S	M	M	S

S- Strong; M-Medium; L-Low

Course Code: 22UPCSC2E18 / 22UPCSC1E18 **Credits:1**

Internet of Things Lab

Course Objectives:

- To create IoT program to turn ON/OFF LED
- To implement IoT program for object detection
- To develop IoT programs for agricultural purpose
- To create web server program for local hosting
- To design IoT application for health monitoring

1. To develop an IoT program to turn ON/OFF LED light (3.3V)
2. To develop an IoT program using IR sensor (Smart Garbage Monitoring, Detecting Parking Availability, etc.)
3. To develop an IoT program using Humidity and Temperature Monitoring (Forest fire Detection, Weather Monitoring)
4. To develop an IoT web server program for local hosting
5. To develop an IoT program using Soil Moisture Sensor
6. To develop an IoT program using Ultrasonic Sensor (Distance Measurement, etc.)
7. To develop an real-time IoT program using Relay Module (Smart Home Automation with 230V)
8. To develop an IoT program for Fire Detection (Home, Industry,etc.)
9. To develop an IoT program for Gas Leakage detection (Home, Industry, etc.)
10. To develop an IoMT program using Heartbeat Sensor

Course Outcomes

On the successful completion of the course, students will be able to,

CO1	Implement IoT programs to turn ON/OFF LED	K1 - K6
CO2	Develop IoT programs for object detection	
CO3	Create IoT programs for agricultural purpose	
CO4	Implement web server program for local hosting	
CO5	Design IoT application for health monitoring	

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	M	S	S	S	M	S	S	S	S	S
CO2	S	S	S	S	S	S	S	M	S	S	M	S
CO3	S	M	S	S	S	S	M	S	S	M	S	S
CO4	S	S	S	S	S	S	S	S	S	L	M	S
CO5	S	S	S	S	M	S	L	S	S	M	M	S

S- Strong; M-Medium; L-Low

Course Code: 22UPCSC2E19 /22UPCSC1E19 Credits: 3

Computer Vision

Course Objectives:

- To get understanding about Computer vision techniques behind a wide variety of real- world applications.
- To get familiar with various Computer Vision fundamental algorithms and how to implement and apply.
- To get an idea of how to build a computer vision application with Python language.
- To understand various machine learning techniques that are used in computer vision tasks.
- To incorporate machine learning techniques with computer vision systems.

Unit-I

Basic Image Handling and Processing: PIL – the Python Imaging Library-Matplotlib-NumPy-SciPy-Advanced example: Image de-noising. **Local Image Descriptors:** Harris corner detector-SIFT - Scale-Invariant Feature Transform-Matching Geotagged Images.

Unit-II

Image to Image Mappings: Homographies-Warping images-Creating Panoramas. **Camera Models and Augmented Reality:** The Pin-hole Camera Model-Camera Calibration-Pose Estimation from Planes and Markers-Augmented Reality.

Unit-III

Multiple View Geometry: Epipolar Geometry-Computing with Cameras and 3D Structure-Multiple View Reconstruction-Stereo Images. **Clustering Images:** K-means Clustering-Hierarchical Clustering-Spectral Clustering.

Unit-IV

Searching Images: Content based Image Retrieval-Visual Words-Indexing Images-Searching the Database for Images-Ranking Results using Geometry-Building Demos and Web Applications. **Classifying Image Content:** K-Nearest Neighbors-Bayes Classifier-Support Vector Machines-Optical Character Recognition.

Unit-V

Image Segmentation: Graph Cuts-Segmentation using Clustering-Variational Methods. **OpenCV:** Python Interface-OpenCV Basics-Processing Video-Tracking.

Course Outcome:

On the successful completion of the course, students will be able

CO1	To understand and recall computer vision and its application areas	K1-K6
CO2	To develop build a computer vision system	
CO3	To apply and analyze a design range of algorithms for image processing and computer vision	
CO4	To develop incorporate machine learning techniques with computer vision system	
CO5	To apply and analyze image segmentation and image registration	

K1- Remember, K2 - Understand, K3 - Apply , K4 - Analyze, K5 - Evaluate, K6 -Create

Mapping with Programme Outcomes:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	L	M	L	L	L	M	M	M	S	S	L
CO2	S	M	L	M	M	L	S	L	S	L	S	L
CO3	S	S	S	M	M	L	M	L	M	L	M	L
CO4	S	S	S	M	M	L	M	L	M	L	S	L
CO5	S	S	S	M	M	L	S	L	S	L	S	L

L - Low, M- Medium, S - Strong

Course Code: 22UPCSC2E20 / 22UPCSC1E20 **Credits: 1**

Computer Vision Lab

Course Objectives:

- To get an idea of how to build a computer vision application with Python language.
- To learn the basic image handling and processing
- To get familiar with various Computer Vision fundamental algorithms and how to implement and apply.
- To get an idea of how to implement the image transforms.
- To understand various image segmentation algorithms.

Implement the following problems using Python with OpenCV

1. Image Loading, Exploring, and displaying an Image.
2. Access and Manipulate of Image Pixels.
3. Image Transformations.
   i) Resizing
   ii) Rotation
4. Addition operation of Two Images.
5. Image filtering operations
   i) Mean Filtering
   ii) Gaussian Filtering
6. Image Binarization Using Simple Thresholding method.
7. Edge Detection operation using Sobel and Scharr Gradients.
8. Find Grayscale and RGB Histograms of an Image.
9. Segment an Image using K-means Clustering algorithm.
10. Write a program to classify an Image using KNN Classification algorithm.

Course Outcome:

On the successful completion of the course, students will be able to

CO1	To develop and implement the image loading and exploring	K1-K6
CO2	To Evaluate the image transforms	
CO3	To apply and analyze for image processing denoising algorithms	
CO4	To design and develop the Image Segmentation using Edge detection and Histograms	
CO5	To apply and analyze image clustering and classification algorithms	

K1- Remember, K2 - Understand, K3 - Apply , K4 - Analyze, K5 - Evaluate, K6 -Create

Mapping with Programme Outcomes:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	L	M	L	L	L	M	M	M	S	S	L
CO2	S	M	L	M	M	L	S	L	S	L	S	L
CO3	S	S	S	M	M	L	M	L	M	L	M	L
CO4	S	S	S	M	M	L	M	L	M	L	S	L
CO5	S	S	S	M	M	L	S	L	S	L	S	L

L - Low, M- Medium, S - Strong

Course Code: 22UPCSC2E21 / 22UPCSC1E21 **Credits: 3**

Cryptography and Network Security

Course Objectives

- To familiarize classical encryption techniques and advanced encryption standards
- To explore the working principles and utilities of various cryptographic algorithms including secret key cryptography, hashes and message digests, and public key algorithms
- To recognize different encryption and decryption techniques to solve problems related to confidentiality and authentication
- To develop the ability to use existing cryptographic utilities to build programs for secure communication.
- To learn the need of digital signatures to secure the document with key management

Unit-I

Overview: Computer Security Concepts – The OSI Security Architecture – Security Attacks – Security Services – Security Mechanisms –A Model for Network Security – Classical Encryption Techniques: Symmetric Cipher Model – Substitution Techniques – Transposition Techniques – Rotor Machines – Steganography.

Unit-II

Block Ciphers and the Data Encryption Standard: Traditional Block Cipher Structure –The Data Encryption Standard – The DES Example – The Strength of DES – Block Cipher Design Principles –Basic Concepts in Number Theory and Finite Fields: Divisibility and the Division Algorithm – The Euclidean Algorithm – Modular Arithmetic – Groups, Rings, and Fields – Finite Fields of the Form GF(p) – Polynomial Arithmetic.

Unit-III

Advanced Encryption Standard: Finite Field Arithmetic– AES Structure – AES Transformation Functions – AES Key Expansion –Block Cipher Operation: Multiple Encryption and Triple DES – Stream Ciphers – RC4 – Public-Key Cryptography and RSA: Principles of Public-Key Cryptosystems – The RSA Algorithm –Diffe-Hellman Key Exchange – Elgamal Cryptographic System – Elliptic Curve Arithmetic – Elliptic Curve Cryptography – Pseudorandom Number Generation Based on an Asymmetric Cipher.

Unit-IV

Cryptographic Hash Functions: Applications of Cryptographic Hash Functions – Two Simple Hash Functions – Requirements and Security – Hash Functions Based on Cipher Block Chaining – Secure Hash Algorithm(SHA) – SHA-3 – Message Authentication Codes: Requirements – Functions – Security of MACs – MACs Based on Hash Functions: HMAC – MACs based on Block Ciphers: DAA and CMAC – Authenticated Encryption: CCM and GCM – Key Wrapping.

Unit-V

Digital Signatures – Elgamal Digital Signature Scheme – Schnorr Digital Signature Scheme – NIST Digital Signature Algorithm – Elliptic Curve Digital Signature Algorithm – RSA-PSS Digital Signature Algorithm – Key Management and Distribution: Symmetric

Key Distribution Using Symmetric Encryption – Symmetric Key Distribution Using Asymmetric Encryption – Distribution of Public Keys – X.509 Certificates – Public-Key Infrastructure.

Text Books

1. William Stallings, "Cryptography and Network Security – Principles and Practices", Pearson Education / PHI, 7th Edition.
2. Behrouz A Forouzan, Debdeep Mukhopadhyay, "Cryptography And Network Security", McGraw Hill Education, 3rd Edition.

Reference Books

1. Bernard Menezes, "Network Security and Cryptography", Cengage,1st Edition, 2010.
2. William Stallings, "Cryptography and Network Security", Pearson Education India, Sixth Edition, 2016.
3. V.K. Jain, "Cryptography and Network Security", Khanna Book Publishing, New Delhi, 2016.
4. C.K. Shyamala, N. Harini, Dr. T. R. Padmanabhan, "Cryptography and Security", Wiley India Pvt. Ltd.,2011

Course Outcomes :

On the successful completion of the course, students will be able to:

CO1:	Comprehend and analyze the security concepts to apply and evaluate the encryption techniques in various models	K1-K6
CO2:	Understand and examine the various data encryption standards and number theory. Illustrate and evaluate the various techniques in different applications	K1-K6
CO3:	Grasp the knowledge of AES techniques and apply to evaluate the performance with different key types	K1-K6
CO4:	Comprehend and analyse the basics of hash function and MAC tha helps to develop the encryption models in various application	K1-K6
CO5:	Understand and illustrate the need of digital signature to examine the method of providing good security to the document. And also learn the concept of key management	K1-K6

K1- Remember, K2- Understand, K3- Apply , K4- Analyze, K5- evaluate and K6- Create

Mapping with Programme Outcomes:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	S	-	L	M	S	M	M	-	S	-	-
CO2	M	S	-	M	M	L	M	S	-	M	-	-
CO3	S	S	-	M	S	M	S	M	-	S	-	-
CO4	S	M	L	S	M	L	S	M	-	M	-	-
CO5	M	S	M	L	S	L	M	S	-	S	-	-

S – Strong, M – Medium, L - Low

Course Code: 22UPCSC2E22 /22UPCSC1E22 **Credits: 1**

Cryptography and Network Security Lab

Course Objective

- To develop in classical encryption techniques and advanced encryption standards.
- To acquire programming skills in Implement various cryptographic algorithms including secret key cryptography.
- To develop hashes, message digests and public key algorithms.
- Implement different encryption and decryption techniques.
- To comprehend related to confidentiality and authentication techniques.

Implement the following

1. Write a program that contains a string (char pointer) with a value 'Hello world'. The program should XOR each character in the string with 0 and display the result
2. Write a program to perform encryption and decryption using the Ceaser Cipher
3. Write a program to perform encryption and decryption using the Hill Cipher
4. Write a program to perform encryption and decryption using the Substitution Cipher
5. Write a program to perform encryption and decryption using the DES algorithm
6. Connect to switch with a computer and enable the port security
7. Defeating malware using Building Trojans and Rootkit hunter
8. Implement signature scheme – Digital Signature Standard
9. Identify and capture the user name and password in a same network using wireshark
10. Implement Man-in-the-middle attack and Session hijacking

Course Outcomes

On the successful completion of the course, students will be able to

CO1	Comprehend the programming skills in classical encryption techniques and to develop advanced encryption standards	K1-K6
CO2	Understand and implement the various cryptographic algorithms including secret key cryptography, hashes and message digests	K1-K6
CO3	Evaluate the use of different encryption and decryption techniques	K1-K6
CO4	Design to Solve related confidentiality and authentication problems	K1-K6
CO5	Create public key algorithms	K1-K6

K1- Remember, K2- Understand, K3- Apply , K4- Analyze, K5-Evaluate, K6-Create

Mapping Course outcomes with Programme outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	-	-	-	-	L	-	-	-	-	-	-
CO2	S	-	M	-	M	L	-	-	-	-	-	-
CO3	S	-	S	-	S	L	-	-	-	S	S	S
CO4	S	-	S	-	S	L	-	-	-	S	S	S
CO5	S	-	S	-	S	L	-	-	-	S	S	S

S- Strong; M-Medium; L-Low

Course Code: 22UPCSC2E23 / 22UPCSC1E23 Credits: 3

Mobile Computing

Course Objective:

- To introduce the concepts of wireless devices with signal, Antenna, Radio Frequencies, Signal Propagation.
- To introduce wireless communication and networking principles, that support connectivity to cellular networks, Wireless LAN, GSM, CDMA.
- To introduce the WAP Architecture, MANET and Routing

Unit-I

Introduction – Applications – History of wireless communication – A Simplified reference model - Wireless transmission – Frequencies for radio transmission – Regulations – Signals –Antennas - Signal propagation: Path loss of radio signals - Additional signal propagation effects - Multi-path propagation – Multiplexing –Modulation Chapters: 1, 2.1 to 2.6

Unit-II

Spread spectrum – Direct sequence spread spectrum – Frequency hopping spread spectrum – Cellular systems. Medium access control: Hidden and exposed terminals – Near and far terminals – SDMA, FDMA, TDMA, Fixed TDM, Classical Aloha, slotted Aloha, Carrier sense multiple access – Reservation TDMA – Multiple access with collision avoidance – Polling – CDMA – Spread Aloha multiple access. Chapters: 3.1 to 3.3, 3.4.1 to 3.4.4, 3.4.7 to 3.4.9, 3.5.1

Unit-III

GSM - Mobile services – System architecture – Radio interface – Protocols – Localization and calling – Handover – Security – New Data services. UMTS and IMT-2000 - Satellite Systems: Applications – Basics – Routing – Localization – Handover. Chapters: 3.6, 4.1.1 to 4.1.8, 4.4, 5.2 to 5.6

Unit-IV

Wireless LAN: Infra red vs. radio transmission – Infrastructure and ad-hoc network – IEEE 802.11 – System architecture – Protocol architecture – Physics layer – Medium access control layer – MAC management – Blue tooth. Mobile network layer: Mobile IP: Goals, assumptions and requirements – entities and terminology – packet delivery – Agent discovery – Registration – Tunneling and encapsulation Recent technologies Chapters: 7.1 to 7.3.5, 7.5, 8.1.1 to 8.1.6

Unit-V

WAP: Architecture – wireless datagram Protocol, Wireless transport layer security, Wireless transaction protocol, Wireless session protocol, Wireless application environment, Mobile ad-hoc networks – MANET Characteristics – Classification of MANETs, Routing of MANETs, Proactive Routing Protocol - DSDV, Reactive Routing Protocols – DSR, AODV.Chapter10.3.1 to 10.3.6 (Text Book 2- 6.1, 6.2, 6.4, 6.5, 6.6)

Course Outcomes:

On the successful completion of the course, students will be able to:

Understanding the basic concepts of Wireless Communication
Understanding the basic concepts of Spread Spectrum
Analyzing the concepts of Medium Access Control
Analyzing the concepts of Global System for Mobile Communication
Understanding the basic concepts of Wireless LAN
Understanding the basic concepts of Mobile Network Layer
Understanding the basic concepts of Wireless Application Protocol
Analyzing the concepts of Routing Protocols in MANET

Text Book:

1. Jochen Schiller, "Mobile Communications", Second Edition, Pearson Education, 2013.
2. Kum Kum Garg, "Mobile Computing Theory and Practice", Pearson Education, 2014.

Reference Books:

1. Rifaat A. Dayen, "Mobile Data & Wireless LAN Technologies", Prentice Hall,1997.
2. Steve Mann and Scoot Schibli, "The Wireless Application Protocol", John Wiley & Inc., 2000.

Course Outcomes:

On the successful completion of the course, students will be able to

CO1	Understanding the basic concepts of Mobile and Wireless Communication	K1, K2	LO
CO2	Understanding the basic concepts of Spread Spectrum. Analyzing the concepts of Medium Access Control.	K3	IO
CO3	Analyzing the concepts of Global System for Mobile Communication and Satellite Communications Understanding the basic concepts of Wireless LAN	K4	HO
CO4	Understanding the basic concepts of Wireless LAN. Evaluate the performance of Mobile Network Layer	K2, K5	HO
CO5	Understanding the basic concepts of Wireless Application Protocol and create a Moile App with real time application. Analyzing the concepts of Routing Protocols in MANET	K2, K4, K6	HO

K1- Remember, K2- Understand, K3- Apply , K4- Analyze, K5- Evaluate, K6- Create

Mapping Course outcomes with Programme outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	L	-	-	-	-	-	-	-	-	-	-	-
CO2	S	M	M	M	M	-	M	-	-	-	-	-
CO3	S	M	M	M	M	-	M	-	-	L	-	M
CO4	S	M	M	M	M	-	M	-	-	L	-	M
CO5	S	M	M	M	M	-	M	-	-	L	-	M

3. S- Strong; M-Medium; L-Low

CourseCode-22UPCSC2E24 / 22UPCSC1E24 **Credits: 2**

Mobile Computing and Application Development Lab

Course Objective

- To enable the students practice the concepts of Mobile application and develop solutions for real world problems.
- Understand how to work with various mobile application development frameworks.
- Comprehend the capabilities and limitations of mobile devices.
- To get clear understanding of mobile application development with WML/J2ME.
- To get advanced methods for mobile application that makes use of any database.

List of programs

1. Implement the WML tags and Image using WML/J2ME.
2. Design of simple Calculator having +, -, * and / using WML/J2ME.
3. Design of Calendar for any given month and year using WML/J2ME.
4. Design a Timer to System Time using WML/J2ME.
5. Design of a simple game using WML/J2ME.
6. Animate an image using WML/J2ME.
7. Design a personal phone book containing the name, phone no., address, e-mail, etc
8. Browsing the Internet using a Mobile phone simulator.
9. Develop a Mobile application to view the live streaming using video view.
10. Develop a mobile application that makes use of any database.

Course Outcomes

On the successful completion of the course, students will be able to

CO1:	Apply the knowledge of mobile application development with WML/J2ME.	K1-K6
CO2:	Design real life situational problems and think creatively abou solutions of them.	K1-K6
CO3:	Appraise the best features Programs for creating dynamic anc interactive web pages using forms.	K1-K6
CO4	Create a Mobile application to view the live streaming using vide view.	K1-K6
CO5	Create a mobile application that makes use of any database.	K1-K6

K1- Remember, K2- Understand, K3- Apply , K4- Analyze, K5- Evaluate, K6- Create

Mapping Course outcomes with Programme outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	-	-	-	-	L	-	-	-	-	-	-
CO2	S	-	M	-	M	L	-	-	-	-	-	-
CO3	S	-	S	-	S	L	-	-	-	S	S	S
CO4	S	-	S	-	S	L	-	-	-	S	S	S
CO5	S	-	S	-	S	L	-	-	-	S	S	S

S- Strong; M-Medium; L-Low

Course Code: 22UPCSC2E25 / 22UPCSC1E25 **Credits: 3**

Cyber Security

Course Objectives:

- To understand the basics of Cybercrime and Computer forensics with protecting mechanism
- To explore the working principles of WLAN, Email and Smartphone along with security mechanism and guidelines
- To gain the ability to understand the importance of cyber investigations with its functioning role and learn the basics of Wi Fi and its security measures
- To understand and learn the method of seize the digital evidence
- To learn and analyze the concepts of digital forensics with cybercrime prevention techniques

Unit – I

Introduction to cybercrime: Classification of cybercrimes – reasons for commission of cybercrime – malware and its type – kinds of cybercrime – authentication – encryption – digital signatures – antivirus – firewall – steganography – computer forensics – why should we report cybercrime – introduction counter cyber security initiatives in India – generating secure password – using password manager-enabling two-step verification – security computer using free antivirus.

Unit – II

Tips for buying online: Clearing cache for browsers – wireless LAN-major issues with WLAN-safe browsing guidelines for social networking sites – email security tips – introduction-smart phone security guidelines – purses, wallets, smart phones – platforms, setup and installation-communicating securely with a smart phone.

Unit – III

Cyber investigation roles**:** Introduction – role as a cybercrime investigator – the role of law enforcement officers – the role of the prosecuting attorney – incident response: introduction-post mortem versus live forensics – computer analysis for the hacker defender program-network analysis – legal issues of intercepting Wi-Fi transmission – Wi-Fi technology – Wi-Fi RF-scanning RF – eavesdropping on Wi-Fi – fourth amendment expectation of privacy in WLAN.

Unit – IV

Seizure of digital information: introduction – defining digital evidence – digital evidence seizure methodology – factors limiting the wholesale seizure of hardware – other options for seizing digital evidence – common threads within digital evidence seizure – determining the most appropriate seizure method– conducting cyber investigations– demystifying computer/cyber crime – IP addresses – the explosion of networking – interpersonal communication.

Unit – V

Digital forensics and analyzing data: introduction – the evolution of computer forensics– phases of digital forensics-collection – examination-analysis – reporting – Cyber crime prevention: Introduction – crime targeted at a government agency.

Text books:

1. Dr.JeetendraPande, "Introduction to Cyber Security" Published by Uttarakhand Open University, 2017.(Chapter: 1.2-6.4,9.3-12.2)
2. Anthony reyes, Kevin o'shea, Jim steele, Jon R. Hansen, Captain Benjamin R. Jean Thomas Ralph, "Cyber-crime investigations" - bridging the gaps between security professionals, law enforcement, and prosecutors, 2007.(Chapter: 4, 5, 6, 7, 8, 9,10)

Reference Books:

1. Sebastian Klipper, "Cyber Security" EinEinblickfur Wirtschafts wissens chaftler Fachmedien Wiesbaden,2015
2. John G.Voller Black and Veatch, "Cyber Security" Published by John Wiley & Sons, Inc., Hoboken, New Jersey Published simultaneously in Canada ©2014.

Course Outcomes

On the successful completion of the course, students will be able to

CO1	Understand, describe, analyze and examine the basics of Cybe security concepts and its implementation in India	K1- K6
CO2	Comprehend and demonstrate the security tips in browsers, WLAN social networks, Email security and Smart phone. Apply the investigations in post mortem and Forensics	K1- K6
CO3	Understand, apply and evaluate the various investigation roles and W Fi protecting mechanisms.	K1- K6
CO4	Understand, illustrate and evaluate the method of seize the digita information and evidences forensics data and evaluate the forensic reports	K1- K6
CO5	Comprehend, apply and appraise the methods digital forensics wit cybercrime prevention techniques	K1- K6

K1- Remember, K2- Understand, K3- Apply , K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	L	-	L	M	L	M	M	-	M	-	-
CO2	M	S	-	L	M	L	M	M	-	M	-	-
CO3	M	S	L	L	M	L	M	M	-	M	M	L
CO4	S	M	L	S	M	L	S	M	-	M	-	-
CO5	M	S	M	L	S	L	M	S	-	S	-	-

S- Strong; M-Medium; L-Low

Course Code: 22UPCSC2E26 /22UPCSC1E26 **Credits: 2**

Cyber Security Lab

COURSE OBJECTIVES

- To learn and implement to Change the wireless device mode as monitor mode
- To develop in multiple vulnerabilities web server
- To understand and implement the open ports in the network
- To acquire programming skills in Implement various wireless device modes
- To comprehend related to find the sub domains of webpage

Implement the following using any cyber security tools

1. Install virtual box (kali Linux)
2. Generate a secure password using keepass
3. Change the wireless device mode as monitor mode
4. Find the known and open vulnerabilities of system using metaspolit
5. Identify the multiple vulnerabilities webserver using nikto tool
6. Identify the open ports in the network using nmap tools
7. List all the network around us and display the information about the networks
8. Sniff and capture the packet sent over HTTP requests
9. Find the owners of internet resources using Whois Lookup tool
10. Find the sub domains of webpage using knock tool

Course Outcomes

On the successful completion of the course, students will be able to

CO1:	Comprehend the programming skills in Change the wireless devic mode as monitor mode	K1-K6
CO2:	Understand and implement multiple vulnerabilities web server	K1-K6
CO3:	Evaluate the use of different wireless device modes	K1-K6
CO4:	Design to Solve related to find the sub domains of webpage	K1-K6
CO5:	Create and apply open ports in the network	K1-K6

K1- Remember, K2- Understand, K3- Apply , K4- Analyze, K5-Evaluate, K6-Create

Mapping Course outcomes with Programme outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	-	-	-	-	L	-	-	-	-	-	-
CO2	S	-	M	-	M	L	-	-	-	-	-	-
CO3	S	-	S	-	S	L	-	-	-	S	S	S
CO4	S	-	S	-	S	L	-	-	-	S	S	S
CO5	S	-	S	-	S	L	-	-	-	S	S	S

S- Strong; M-Medium; L-Low

Course Code: 22UPCSC2E27 / 22UPCSC1E27 Credits 3

Block chain Technologies

COURSE OBJECTIVES

- To understand about Blockchain is an emerging technology platform for developing decentralized applications and data storage.
- To comprehend fundamentals of Public Key Cryptography technology and Consensus Algorithms.
- To familiarize with Bitcoin Network, Bitcoin Clients, APIs and Payments technology of blockchain operations.
- To engage with Components of the Ethereum ecosystem.
- To grasp about Development Tools and Frameworks.

Unit I: Blockchain, Decentralization

Blockchain : The growth of blockchain technology - Distributed systems - The history of blockchain and Bitcoin - Blockchain - Consensus - CAP theorem and blockchain. **Decentralization:** Decentralization using blockchain - Methods of decentralization - Routes to decentralization - Blockchain and full ecosystem decentralization - Pertinent terminology - Platforms for decentralization - Innovative trends.

Unit II: Public Key Cryptography, Consensus Algorithms and Smart Contracts

Public Key Cryptography: Asymmetric cryptography - Cryptographic constructs and blockchain technology. **Consensus Algorithms:** Introducing the consensus problem - Analysis and design - Classification - Algorithms - Choosing an algorithm. **Smart Contracts:** History - Definition - Ricardian contracts - Smart contract templates – Oracles - Deploying smart contracts - DAO

Unit III: Bitcoin

Bitcoin: Bitcoin—an overview - Cryptographic keys - Transactions - Blockchain – Mining. **Bitcoin Network and Payments:** The Bitcoin network - Wallets - Bitcoin payments - Innovation in Bitcoin - Advanced protocols - Bitcoin investment and buying and sellingBitcoin. **Bitcoin Clients and APIs:** Bitcoin client installation - Experimenting further with bitcoin-cli - Bitcoin programming.

Unit IV: Alternative Coins

Alternative Coins: Theoretical foundations - Difficulty adjustment and retargeting algorithms - Bitcoin limitations - Extended protocols on top of Bitcoin -Development of altcoins.**Ethereum: Ethereum –** an overview - Ethereum network - Components of the Ethereum ecosystem - EthereumVirtual Machine (EVM) - Smart contracts. - Blocks and blockchain - Wallets and client - Nodes and miners - APIs, tools, and DApps - Supporting protocols - Programming languages.

Unit V: Development Tools and Frameworks, Use Cases & Security

Development Tools and Frameworks : Languages - Compilers - Tools and libraries - Frameworks - Contract development and deployment - Layout of a Solidity source code file - Solidity language. **Use Cases:** IoT – Government - Health -Finance – Media. **Scalability and Other Challenges:** Scalability - Privacy - Security - Other challenges.

TEXT BOOKS

1. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, Steven Goldfeder. Bitcoin and Crypto currency Technologies. Princeton University Press, 2016. ISBN 978-0691171692

REFERENCES

1. Andreas Antonopoulos. Mastering Bitcoin: Programming the open block chain. Oreilly Publishers, 2017. ISBN 978-9352135745

Course Outcomes

On the successful completion of the course, students will be able to

CO1:	Understand, apply and examine the characteristics of blockchain bitcoin and consensus algorithm in centralized and decentralize methods.	K1-K6
CO2:	Comprehend and demonstrate the application of hashing and publi key cryptography in protecting the blockchain.	K1-K6
CO3:	Understand and analyse the elements of trust in a Blockchain validation, verification, and consensus.	K1-K6
CO4	Comprehend and evaluate the alternate coin, Ethereum and smar contract.	K1-K6
CO5	Grasp and apply the knowledge of Tools and languages fo applications	K1-K6

K1- Remember, K2- Understand, K3- Apply , K4- Analyze, K5-Evaluate, K6- Create

Mapping Course outcomes with Programme outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	-	-	-	-	L	-	-	-	-	-	-
CO2	S	-	M	-	M	L	-	-	-	-	-	-
CO3	S	-	S	-	S	L	-	-	-	S	S	S
CO4	S	-	S	-	S	L	-	-	-	S	S	S
CO5	S	-	S	-	S	L	-	-	-	S	S	S

S- Strong; M-Medium; L-Low

Course Code: 22UPCSC2E28 / 22UPCSC1E28 **Credits: 1**

Blockchain Technologies Lab

Course Objectives

- To learn the basics of Blockchain and apply cryptographic algorithms
- To design, build, and deploy smart contracts and distributed applications,
- To deploy Private Blockchain and smart contracts on Ethereum.
- To understand and deploy crypto currencies and their functions in applications
- To implement Blockchain for various use cases.

Implement the following

1. Create a Public Ledger and Private Ledger with the various attributes like Access, Network Actors, Native token, Security, Speed and examples.
2. Building and Deploying MultiChain private Blockchain
3. Write Hello World smart contract in a higher programming language (Solidity)
4. Construct the Naïve block chain
5. Construct and deploy your contract (Use deploy method)
6. Set up a Regtest environment
7. Build a payment request URI
8. Hashcash implementation
9. Develop a toy application using Blockchain
10. Create simple wallet transaction from one account to another account using Metamask.

Course Outcomes

On the successful completion of the course, students will be able to:

CO1	Enable to setup your own private Blockchain and deploy smar contracts on Ethereum.	K1-K6
CO2	Gains familiarity and implement with cryptography and Consensu algorithms.	K1-K6
CO3	Create and deploy projects using Web3j.	K1-K6
CO4	Recall and deploy the structure and mechanism of Bitcoin, Ethereum Hyperledger	K1-K6
CO5	Implement Blockchain for various use cases	K1-K6

Mapping with Programme Outcomes:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	M	L	M	S	-	-	-	-	-	-
CO2	S	M	S	S	S	M	-	-	-	-	-	-
CO3	S	S	S	S	S	S	-	-	-	-	-	-
CO4	S	M	L	S	M	L	-	-	-	-	-	-
CO5	M	S	M	L	S	L	-	-	-	-	-	-

S – Strong, M – Medium, L - Low

Course Code: 22UPCSC2E29 / 22UPCSC1E29 **Credits: 4**

Optimization Techniques

Course Objectives

- To understand the concept of Linear optimization
- To develop mathematical models of transportation and assignment Problems
- To understand the Networking models
- To study non-linear optimization models
- To develop optimization algorithms based on Evolutionary concepts

Unit – I : Linear Programming Problem

Linear Programming Problem (LPP): Mathematical Formulation of Linear Programming Problem - Graphical Solution of LPP - canonical and standard forms of linear programming problem- Simplex method for solving LPP

Unit – II : Transportation and Assignment Problems

Transportation Model: North West corner Method, Least cost method, and Vogel's Approximation Method. Assignment Model : Hungarian assignment model – Travelling Sales Man Problem.

Unit – III : CPM/PERT

Project Scheduling PERT/CPM Networks – Fulkerson's Rule – Measure of Activity –PERT Computation – CPM Computation – Resource Scheduling.

Unit – IV : Non-Linear Optimization Models

Simplex Method – Gradient of function – Steepest Descent method – Conjugate Gradient method

Unit – V : Evolutionary Algorithms

Particle Swarm Optimization method – Ant Colony optimization algorithm – Fruit Fly method – Fire Fly method

Text Book

1. KantiSwarup, P. K. Gupta and Man Mohan, Operations Research, Sultan Chand and Sons, New Delhi, 2014. (Unit 1, 2, and 3)

2. S. S. Rao, Engineering Optimization: Theory and Practice, JOHN WILEY &amp; SONS, INC., 2009. (Unit 4)

3. Bo Xing and Wen-Jing Gao, Innovative Computational Intelligence: A Rough Guide to Clever Algorithms, Springer, 2014.(Unit 5)

Reference Book

1. Hamdy A. Taha, Operations Research: An Introduction, Pearson, 2010

Course outcome:

CO1 Develop Linear Programming models for industrial problems
CO2 Formulation of Transportation and Assignment problems for real life problems

CO3 Critical path estimation can be done for a project

CO4 Non-Linear optimization models can be solved

CO5 Evolutionary Optimization algorithms can be used to solve any optimization problems to overcome the issues in the traditional optimization models

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	M	S	S	S	M	S	S	S	S	S
CO2	S	S	S	S	S	L	S	M	S	S	M	S
CO3	S	M	S	S	M	S	M	S	S	L	S	M
CO4	S	S	S	S	S	S	S	S	S	S	M	S
CO5	S	S	S	S	M	L	S	S	S	M	M	S

S- Strong; M-Medium; L-Low

Course Code: 22UPCSC2E30 /22UPCSC1E30 **Credits: 1**

Optimization Techniques Lab

Course Objectives

- To understand the concept of optimization
- To develop mathematical model of real life cases
- To study Optimization algorithms

Exercises

Implement the following algorithms using Python

1. Simplex algorithm
2. North-West Corner algorithm
3. Least Cost algorithm
4. Vogel's Approximation Algorithm
5. Modi algorithm
6. Hungarian assignment Algorithm
7. Travelling Sales man algorithm
8. Steepest Descent algorithm
9. Conjugate gradient algorithm
10. Ant Colony algorithm

Course Code: 22UPCSC2E31 / 22UPCSC1E31 **Credits: 3**

Solution Architecture

Course Objectives:

- To get familiar with the evolution of solution architecture
- To outline the functionalities of Solution Architecture Design
- To assess the Performance Considerations of Solution Architecture
- To analyze the Architectural Reliability Considerations
- To identify the design principles for cost optimization

UNIT I

Evolution of solution architecture - The benefits of solution architecture - Solution architecture in the public cloud - Solution Architects in an Organization - Types of solution architect role - Understanding a solution architect's responsibilities - Attributes of the Solution Architecture - Scalability and elasticity - Security and compliance - Cost optimization and budget

UNIT II

Principles of Solution Architecture Design - Scaling workload - Building resilient architecture - Design for performance - Using replaceable resources - Cloud Migration and Hybrid Cloud Architecture Design - Benefits of cloud-native architecture - Creating a cloud migration strategy - Steps for cloud migration - Creating a hybrid cloud architecture - Designing a cloud-native architecture

UNIT III

Solution Architecture Design Patterns - Building an n-tier layered architecture - Creating multi-tenant SaaS-based architecture - Building stateless and stateful architecture designs - Understanding SOA - Performance Considerations - Design principles for architecture performance - Technology selection for performance optimization - Managing performance monitoring

UNIT IV

Architectural Reliability Considerations - Design principles for architectural reliability - Technology selection for architecture reliability - Improving reliability with the cloud - Operational Excellence Considerations - Designing principles for operational excellence - Selecting technologies for operational excellence - Achieving operational excellence in the public cloud

UNIT V

Cost Considerations - Design principles for cost optimization - Techniques for cost optimization - Cost optimization in the public cloud - Architecting Legacy Systems -

Learning the challenges of legacy systems - Defining a strategy for system modernization - Looking at legacy system modernization techniques - Defining a cloud migration strategy for legacy systems

Text Books:

1. Saurabh Shrivastava and Neelanjali Srivastav, " Solutions Architect's Handbook", Packt Publishing, 2020

Reference Books:

1. Alan McSweeney , "Introduction to Solution Architecture", Kindle Edition, 2019
2. Bernard, Scott A. An introduction to holistic enterprise architecture. Author House, 2020.

Course Outcomes

On the successful completion of the course, students will be able to

CO1	Comprehend the types, benefits and attributes of solution architecture	**K1- K6**
CO2	Assess the cloud architecture and create hybrid cloud architecture	
CO3	Analyze the design patterns for solution architecture	
CO4	Understand Architecture Reliability and Operational Excellence	
CO5	Optimize the cost in cloud and assess the legacy system	

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	S	S	S	S	S	S	S	M	S	S
CO2	S	S	S	S	S	S	S	S	S	S	M	S
CO3	S	M	S	S	S	S	M	S	S	S	S	S
CO4	S	S	S	M	S	S	S	S	S	S	M	S
CO5	S	S	S	S	M	S	S	S	S	M	M	S

S- Strong; M-Medium; L-Low

Course Code: 22UPCSC2E32 /22UPCSC1E32 **Credits: 1**

Solution Architecture Lab

Course Objectives:

- To get familiar with AWS cloud services
- To design secure, scalable, and well structured cloud solutions
- To create EC2 instance and configure options
- To connect EC2 with Linux instance
- To connect VPN server to securely access instances

1. Managing Virtual Private Cloud
2. Creating and Configuring Internet Gateways
3. Configuring Routing Tables
4. Working with Amazon Elastic Cloud Compute (EC2)
5. Connecting EC2 Linux instance using PuTTY, Gitbash and Console
6. Recovering and connecting EC2 instance if the SSH key is lost
7. Creating and Configuring Elastic Load Balancer
8. Scheduling Auto Snapshot of volumes
9. Configuring Centralized Log Management using Cloud Watch Log
10. Connecting OpenVPN server

Course Outcomes

On the successful completion of the course, students will be able to

CO1	Understand AWS cloud services and manage the cloud data	**K1- K6**
CO2	Develop secure, scalable, and well structured cloud solutions	
CO3	Implement EC2 instance and configure the instance	
CO4	Connect EC2 with Linux instance and perform operations	
CO5	To connect VPN server to access instances with more security	

K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create

Mapping with Programme Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	S	S	S	S	S	S	S	M	S	S
CO2	S	S	S	S	S	S	S	S	S	S	M	S
CO3	S	M	S	S	S	S	M	S	S	S	S	S
CO4	S	S	S	M	S	S	S	S	S	S	M	S
CO5	S	S	S	S	M	S	S	S	S	M	M	S

S- Strong; M-Medium; L-Low