DEPARTMENT OF COMPUTER SCIENCE

PREAMBLE

UG : Programme Profile- List of Courses offered to other Departments and Syllabi of Courses in the I and II Semesters along with Evaluation Components III and IV (with Effect from 2021-2024 Batch onwards) and

PG : Programme Profile- List of Courses offered and Syllabi of Courses in the I and II Semesters along with Evaluation Components III and IV (with Effect from 2021-2023 Batch onwards).

PROGRAMME PROFILE B.Sc. (Computer Science)

(LEARNING OUTCOMES-BASED CURRICULUM FRAMEWORK)

PSO:1 Ability to Understand, Analyze, Design, Develop and Optimize Solutions related to Computer Programming Languages.

PSO:2 Application of concepts in Core Areas related to Computer Programming for Efficient Design of Computer-Based Systems of Varying Complexity.

PSO:3 Ability to test the Technical issues in Software Engineering and Deliver a Quality Product for Business Success.

PSO:4 Ability to Innovate and Develop New Technologies.

Semester	Part	Category	Course Code	Course Title	Previous Course Code	Contact Hrs/ Week	Credit Min/Max
I	I	Language	UTAL107/ UTAL108	Languages/ AECC-II Tamil-I/ Hindi-I/ French-I (2 Levels)	UTAL105/ UTAL106/ UHIL101/ UFRL101	5	3/4
	II	English	UENL109/ UENL110	English for Communication (Stream-I)/ English for Communication (Stream-II)	UENL107/ UENL108	5	3/4
	III	Major Core (DSC) - I	UCSM110/ UCAM110	Principles of Information Technology	UCSM108	5	4
	III	Major Core (DSC) - II	UCSM109/ UCAM111	Programming Methodology	-	4	4
	III	Major Core (DSC) - III	UCSR110/ UCAR106	Programming Methodology - Practical	-	3	2
	III	Allied (GE) - I	UMAA114	Mathematics for Computer Science	-	6	4
	III	Professional English	UPEM101	Professional English I	-	6	4
	IV	Value Education (SEC)				2	1
					Total	**36**	**25/27**

Semester	Part	Category	Course Code	Course Title	Previous Course Code	Contact Hrs/ Week	Credit Min/Max
II	I	Language	UTAL207/ UTAL208	Languages/ AECC-II Tamil-II/ Hindi-II/ French-II (2 Levels)	UTAL205/ UTAL206/ UHIL201/ UFRL201	5	3 /4
	II	English	UENL209/ UENL210	English for Communication (Stream-I)/ English for Communication (Stream-II)/	UENL207/ UENL208	5	3/4
	III	Major Core (DSC) - IV	UCSM207/ UCAM206	Data Structures	UCSM206	4	4
	III	Major Core (DSC) - V	UCSM208/ UCAM207	Python Programming	-	4	4
		Major Core (DSC) - VI	UCSR207/ UCAR205	Data Structures using Python - Practical	UCSR206	3	2
	III	Allied (GE) - II	UMAA218	Mathematics for Computer Science	-	6	4
	III	Professional English	UPEM201	Professional English II	-	6	4
	IV	NME (Skill Enhancement Course)					
	V	Extension Programme / Physical Education/ NCC				-	1/2
					Total	**36**	**27/30**
III	I	Language	UTAL307/ UTAL308	Languages/ AECC-II Tamil-III/ Hindi-III/ French-III(2 Levels)	UTAL305/ UTAL306 UHIL301/ UFRL301	5	3 /4
	II	English	UENL309/ UENL310	English for Communication (Stream-I)/ English for Communication (Stream-II)	UENL307/ UENL308	5	3 /4
	III	Major Core (DSC) - VII	UCSM305	Java Programming	UCSM304	5	5
	III	Major Core (DSC) - VIII	UCSM307	Software Engineering	UCSM511	4	4
	III	Major Core (DSC) - IX	UCSR308	Java Programming – Practical	UCSR305	3	2
	III	Allied (GE) - III	UPHA304	Digital Electronics for Computer Science	UPHA303	3	2
	III	Allied (GE) -IV	UPHR304	Digital Electronics for Computer Science – Practical	-	3	2
	IV	Value Education (SEC)			-	2	1
					Total	**30**	**22/24**

Semester	Part	Category	Course Code	Course Title	Previous Course Code	Contact Hrs/ Week	Credit Min/Max
IV	I	Language	UTAL407/ UTAL408	Languages/ AECC-II Tamil-IV/ Hindi-IV/ French-IV(2 Levels)	UTAL405/ UTAL406/ UHIL401/ UFRL401	5	3 /4
	II	English	UENL409/ UENL410	English for Communication (Stream-I)/ English for Communication (Stream-II)	UENL407/ UENL408	5	3/ 4
	III	Major Core (DSC) - X	UCSM409	Operating Systems		5	5
	III	Major Core (DSC) - XI	UCSR412	Operating System Practical	UCSR411	4	3
	III	Allied (GE) - V	UPHA403	Electronics for Computer Science		3	2
	III	Allied (GE) - VI	UPHR403	Electronics for Computer Science– Practical		3	2
	IV	NME (Skill Enhancement Course)				3	2
	IV	Online Courses		NPTEL/SPOKEN TUTORIAL/SWAYAM		3	1/2
	IV	Soft Skill (SEC)				2	1
	V	Extension Programme / Physical Education				-	0/2
					Total	30	22/26
V	III	Major Core (DSC) - XII	UCSM506	Middleware Technologies	-	5	5
	III	Major Core (DSC) - XIII	UCSM510	Computer Networks		**5**	**4**
	III	Major Core (DSC) - XIV	UCSM512	Database Management System	UCSM509	4	4
	III	Major Core (DSC) - XV	UCSR512	Middleware Technologies - Practical	UCSR509	4	3
	III	MAJOR ELECTIVE (Discipline Specific Elective)– XVI	UCSO501/ UCSO502/ UCSO503	Computer Ethics/ Computer Graphics/ Data Mining	-	5	4
	III	Major Core (DSC) - XVII	UCSP501	Project	UCSP601	5	5
	IV	Value Education				2	1
					Total	30	26

Semester	Part	Category	Course Code	Course Title	Previous Course Code	Contact Hrs/ Week	Credit Min/Max
VI	III	Major Core (DSC) - Core XVIII	UCSM612	Cloud Computing	-	**5**	**5**
	III	Major Core (DSC) - XIX	UCSM614	Bigdata Tools	UCSM610	5	4
	III	Major Core (DSC) - XX	UCSM615	Internet of Things	UCSO608	5	4
	III	Major Core (DSC) - XXI	UCSR608	Bigdata Tools Practical	-	4	4
	III	Major Core (DSC) - XXII	UCSR609	Cloud Computing-Practical	UCSR508	4	3
	III	MAJOR ELECTIVE (Discipline Specific Elective)– XXIII	UCSO609/ (UCSO610/ UCSM613)/ UCSO606	Artificial Intelligence/ Open Source Technology/ Network Security	-	5	4
	III	Viva – Voce	UCSM611	Comprehensive Viva Voce	-	-	1
	IV	Soft Skill (SEC)				2	1
	V	Extension Programme / Physical Education/NCC				-	0/2
					Total	**30**	**26/28**
					Grand Total	**192**	**148/161**

ALLIED COURSES OFFERED TO OTHER DEPARTMENTS

Class & Major	Semester	Category	Course Code	New Course Title	Previous Course Code	Contact Hrs/ Week	Credit Min/Max
B.Com with Computer Applications	I	Allied	UCSA105	Multimedia	UCSA303	3	3
	I	Allied Practical	UCSR111	Multimedia Lab	UCSR306	3	2
	II	Allied	UCSA205	C Programming	UCSA104	3	3
	II	Allied Practical	UCSR208	C Programming Lab	UCSR110	3	2
	III	Allied	UCSA306	Object Oriented Programming	UCSA204	3	3
	III	Allied Practical	UCSR310	Object Oriented Programming – Lab	UCSR207	3	2
	IV	Allied	UCSA408	Fundamentals of Blockchain Technology	UCSA305	3	3
	IV	Allied Practical	UCSR414	Blockchain Technology Using Solidity – Lab	UCSR309	3	2
	V	Allied	UCSA510	Digital Marketing Analytics	UCSA406	3	3
	V	Allied Practical	UCSR513	Web Design using Microsoft Expression Web4 - Lab	UCSR412	3	2
Class &	**Semes**	**Category**	**Course Code**	**New Course Title**	**Previous**	**Contact**	**Credit**

Major	ter				Course Code	Hrs/ Week	Min/Max
BBA, B.Com and B.COM(IAT)	IV	Allied	UCSA409	Business Analytics and Intelligence.	UCSA509	3	3
	IV	Allied Practical	UCSR415	Business Analytics and Intelligence - Lab	UCSR512	3	2
Tamil	V	Allied	UCSA505	Tamil Kanini	-	3T + 2P	5
Maths	III	Allied	UCSA304	Mathematical Programming using C	-	3	3
	III	Allied Practical	UCSR307	Mathematical Programming using C – Lab	-	3	2
	V	Allied	UCSA507	Object Oriented Programming using Java	-	3	3
	V	Allied Practical	UCSR508	Object Oriented Programming using Java - Lab	-	3	2
Physics	III	Allied	UCSA306	Computational Physics with Python	-	3	3
	III	Allied Practical	UCSR310	Computational Physics with Python – Lab	-	3	3

NON-MAJOR ELECTIVE

Semester	Part	Category	Course Code	Course Title	Previous Course Code	Contact Hrs/week	Credit Min/Max
II	IV	Non Major Elective	UCSE206	Tableau Programming	UCSE202	1T+2P	2
			UCSE207	Python Programming	UCSE203	3P	2
			UCSE208	R Programming	UCSE204	3P	2
			UCSE209	Arduino Programming	UCSE205	3P	2
			UCSE210	Go Programming	-	3P	2
IV	IV	Non Major Elective	UCSE406	IOT Projects	-	3P	2
			UCSE407	Mobile Application Development	-	3P	2

EXTRA CREDIT EARNING PROVISION

Semester	Part	Category	Course Code	Course Title	Contact Hrs/week	Credit	
						Min	Max
II	III	Core	UCSI201	Summer Internship / Working Model	-	-	1
IV	III	Core	UCSI401	Summer Internship	-	-	1
V	III	Self Study Paper	UCSS501	Python Programming	2	1	1
V	III	Self Study Paper	UCSS502/ UCAS502	Android Applications	2	1	1
VI	III	Self Study Paper	UCSS601/ UCAS601	Angular JS	2	1	1
VI	III	Self Study Paper	UCSS602/ UCAS602	Green Computing	2	1	1

A. **Experiential Learning (Mandatory)**

Course Mapping				Collaborating Agency - MSME		
Sem	**Course Code**	**Course Title**	**Assessment**	**Course Title**	**Hour/Days/ Month**	**Mode of Evaluation**
VI	UCSM614	Big Data Tools	Component IV	Data Analytics certification	4 Days	Reflection

B. **Skill Orientation Programme (Only for Interested students) – Extra Credit Earning Provision**

Sem	Category	Course Code	Course Title	Collaborating Agency	Hour/ Days/Month	Mode of Evaluation	Credits (Min/Max)
V	Core	UCST501	Robotics Process Automation	MSME	4 Days	Reflection	1

***MSME Courses will be select at that time of availability.**

PRINCIPLES OF INFORMATION TECHNOLOGY

UCSM110/UCAM110

Semester : I **Credit : 4**
Category : Major Core (DSC) - I **Hour/Week: 5**
Class & Major: I B.Sc Computer Science **Total Hour: 65**

Objectives:

To enable the Students

- Obtain Knowledge on Object Oriented Programming Concepts.
- Understand the Basics of Microprocessor and Compiler.
- Acquire Knowledge on Information Security and Open Source Software.

Learning Outcomes:

On Completion of the course, the students will be able to

- Develop Logic for Assembly Language Programming.
- Analyze the Performance of Commercially available Computers.
- Examine the Construction of CPU, know Registers and Bus Systems.

UNIT - I PROGRAMMING LANGUAGES 13 Hour

Introduction - Evolution of Programming Languages- Classification of Programming Languages - Generations of Programming Languages - Features of a Good Programming Language- Selection of a Programming Language

UNIT - II FUNDAMENTALS OF COMPUTER ARCHITECTURE 13 Hour

Introduction- Central Processing Unit (CPU) Memory- Communication between Various Units of a Computer System- The Instruction Format- Instruction Set- Processor Speed- Multiprocessor Systems. Primary Memory Introduction- Memory Hierarchy- Random Access Memory (RAM)- Types of RAM- Read Only Memory (ROM)- Types of ROM. Secondary Storage Introduction- Classification of Secondary Storage Devices- Magnetic Tape- Magnetic Disk- Optical Disk- Magneto Optical disk. Input Devices - Output Devices.

UNIT - III MICROPROCESSOR 13 Hour

Introduction to Microprocessor – Microcontroller - 8085 Microprocessor and Architecture - Opcode fetch - Machine cycle - Memory Read Machine Cycle - Memory Write Machine Cycle - IO Read Machine Cycle - IO Write Machine Cycle - Execution time of the Instruction Cycle.

UNIT - IV INFORMATION SECURITY **13 Hour**

Introduction to Information Security - Components of Information System - Balancing Information Security and Access - The Systems Development Life Cycle - The Security Systems Development Life Cycle - Security Professionals and Organization.

UNIT - V OPEN SOURCE SOFTWARES **13 Hour**

Introduction to Open sources – Need of Open Sources – Advantages of Open Sources – Application of Open Sources. Open Source Operating Systems: LINUX. Introduction: MySQL - PHP – Python.

Text Books

- Arvind Kumar Bansal. (2014). *Introduction to Programming Languages*. CRC PRESS. Taylor and Francis Group.
- Michael, E. Whitman. Herbert, J. Mattord. (2012). *Principles of Information Security.* Course Technology. *(*4th Ed.). Cengage Learning.
- Alexis Leon. Mathews Leon. (2009). *Fundamentals of Information Technology.* Vikas Publishing House Pvt. Ltd.
- Rasmus Lerdorf. Levin Tatroe. (2012). *Programming in PHP.* Reilly.
- Ramesh, S. Goankar. (2011). *Microprocessor Architecture Programming and Applications with 8085.* Penram International. (5th Ed.).

Reference Books

- Dennis, P. Curtin. Kim Foley. Kunal Sen and Cathleen Morin. (2005). *Information Technology - the Breaking Wave.* Tata-McGraw Hill Publications. (7th Reprint).
- Alexis Leon. Mathews Leon. (2004). *Fundamentals of Information System.* Co-Published by Vijay Nicole Imprints Pvt Ltd.

e-Resource

- http://indexof.es/Computer/Fundamentals%20of%20Computer%20Organization%20and%20Architecture.pdf

PROGRAMMING METHODOLOGY

UCSM109/UCAM111

Semester	**: I**	**Credit: 4**
Category	**: Major Core (DSC) - II**	**Hour/Week: 4**
Class &Major: I B.Sc Computer Science		**Total Hour: 52**

Objectives:

To enable the Students

- Develop Simple Algorithms and Flow Charts to Solve a Problem.
- Acquire Knowledge on Functions, Arrays and Structures.
- Understand the concepts of File Management.

Learning Outcomes:

On Completion of the course, the students will be able to

- Be familiar with Good Programming Practice, and apply it in various programs.
- Know about Insecure Functions to be avoided.
- Understand the Compilation Process in File concepts.

UNIT – I INTRODUCTION TO PROGRAMMING 10 Hour

Introduction to Programming, Program Concept, Characteristics of Programming, Stages in Program Development, Algorithms, Notations, Design, Flowcharts, Types of Programming Methodologies, Introduction to C++ Programming - Basic Program Structure In C++, Variables and Assignments, Input and Output, Selection and Repetition Statements.

UNIT – II FUNCTIONS 10 Hour

Top-Down Design, Predefined Functions, Programmer -defined Function, Local Variable, Function Overloading, Functions with Default Arguments, Call -By-Value and Call-By-Reference Parameters, Recursion.

UNIT – III ARRAYS, STRUCTURES & UNION 12 Hour

Introduction to Arrays, Declaration and Referring Arrays, Arrays in Memory, Initializing Arrays. Arrays in Functions, Multi-Dimensional Arrays. Structures - Member Accessing, Pointers to Structures, Structures and Functions, Arrays of Structures, Unions.

UNIT – IV STRINGS 10 Hour

Declaration and Initialization, Reading and Writing Strings, Arrays of Strings, String and Function, Strings and Structure, Standard String Library Functions.

UNIT – V FILES 10 Hour

Files- File Streams - Creating File Streams - Open Modes - Closing Files - Reading and Writing Blocks.

Text Books

- Dale, N. and Weems, C. (2010). *Programming and Problem Solving with C++: Brief Edition.* Jones & Bartlett Learning.
- Kenrick Mock (2015). *Problem solving with C++ / Walter Savitch; Contributor.* (9th Ed.). ISBN-13: 978-0-13-359174-3

Reference Book

- Hanly, J.R. Koffman, E.B. (2015). *Problem Solving and Program Design*. Pearson.

E-Resource

- http://www.lmpt.univ-tours.fr/~volkov/C++.pdf

PROGRAMMING METHODOLOGY- PRACTICAL

UCSR110/UCAR106

Semester	**: I**	**Credit: 2**
Category	**:Major Core (DSC) - III**	**Hour/Week : 3**
Class &Major	**: I B.Sc Computer Science**	**Total Hour: 39**

Objectives:

To enable the students

- Acquire Knowledge on Basic Skills Coupled with Top Down Design Principles.
- Develop the Skills for Formulating Iterative Solutions to a Problem.
- Understand the Concepts of File Management.

Learning Outcomes

On Completion of the course, the students will be able to

- Apply Problem-Solving Knowledge and Skills to Write Effective C++ Programs.
- Appreciate the use of Simple Data Structure such as Array, Structures and Unions.
- Identify Opportunities to Write Modularized Code.

LIST OF PROGRAMS

1. Arithmetic Operators and Mathematical Expressions
2. Conditional Operators
3. Control Structures – Decision Making
4. Control Structures – Looping
5. Functions and Parameter passing in functions, writing Recursive Programs.
6. Arrays
7. Structures
8. Union.
9. Strings and string handling operations.
10. Files for data input and output.

DATA STRUCTURES

UCSM207/UCAM206

Semester	**: II**	**Credit : 4**
Category	**: Major Core (DSC) - IV**	**Hour/Week : 4**
Class &Major	**:I B.Sc Computer Science**	**Total Hour : 52**

Objectives:

To enable the Students

- Acquire Knowledge on Basic Operations like Insert, Delete, Search etc.,
- Design Programs using various Data Structures including Hash Tables, Binary and General Search Trees, Heaps, Graphs etc.
- Know and Implement the Applications of Algorithms for Sorting, Pattern Matching etc.

Learning Outcomes

On Completion of the course, the students will be able to

- Understand and Restate the Fundamentals of Basic Data Structures.
- Implement Basic Data Structures such as Stacks, Queues and Trees.
- Implement the Algorithms for Sorting and Searching.

UNIT – I INTRODUCTION TO ALGORITHM **11 Hour**

Basic Concepts- Algorithm Specification-Introduction, Recursive Algorithms, Data Abstraction Performance analysis, Linear and Non-Linear Data Structures, Singly Linked Lists-

Operations, Circularly linked Lists-Operations, Doubly Linked Lists- Operations. Representation of Single, Two Dimensional Arrays, Sparse Matrices-Array and Linked Representations.

UNIT - II STACK & QUEUE OPERATIONS 10 Hour

Stack- Operations, Array and Linked Implementations, Applications- Infix to Postfix Conversion, Postfix Expression Evaluation, Recursion Implementation. Queue- Definition and Operations, Array and Linked Implementations, Circular Queues - Insertion and Deletion Operations.

UNIT - III TREES 10 Hour

Trees, Representation of Trees, Binary tree, Properties of Binary Trees, Binary Tree Representations- Array and Linked Representations, Binary Tree Traversals, Threaded Binary Trees, Priority Queue- Implementation, Heap- Definition, Insertion, Deletion.

UNIT - IV GRAPHS 10 Hour

Graphs, Graph ADT, Graph Representations, Graph Traversals, Searching, Static Hashing- Introduction, Hash tables, Hash functions, Overflow Handling.

UNIT – V SORTING & SEARCHING ALGORITHMS 11 Hour

Sorting Methods: Bubble Sort – Insertion Sort – Quick Sort – Heap Sort. Searching Trees: Binary Search Trees, AVL Trees- Definition and Examples- Pattern Matching Algorithm.

Text Books

- Michael, T. Goodrich. Roberto Tamassia. Michael, H. Goldwasser. (2013). *Data Structures and Algorithms in Python.* Wiley.
- Dr. Kent, D. Lee, Dr. Steve Hubbard. (2015). *Data Structures and Algorithms with Python.* Springer Nature.
- Rance, D. Necaise. (2016). *Data Structures and Algorithms Using Python.*

Reference Books

- Benjamin Baka. Dr BasantAgarwal.(2018). *Hands-On Data Structures and Algorithms with Python.* (2nd Ed.).
- Horowitz, E,Sahni, S. and Susan Anderson-Freed. *Fundamentals of Data Structure*. Universities Press. (2ndEd.).

E-Resource

- file:///C:/Users/admins/AppData/Local/Temp/Data%20Structures%20and%20Algorithms%20in%20Python%20[Goodrich,%20Tamassia%20Goldwasser%202013-03-18]-1.pdf

PYTHON PROGRAMMING
UCSM208/UCAM207

Semester	**: II**	**Credit : 4**
Category	**: Major Core (DSC) - V**	**Hour/Week : 4**
Class &Major	**: I B.Sc Computer Science**	**Total Hour : 52**

Objectives:
To enable the Students

- Acquire Knowledge on Concepts of Functions & Illustrative Programs.
- Understand Python Lists, Tuples to Represent Compound Data.
- Develop and Execute Simple Python Programs.

Learning Outcomes:
On Completion of the course, the students will be able to

- Define and Demonstrate the Use of Built-in Data Structures "Lists" and "Dictionary".
- Design and Implement GUI Application and How to Handle Exceptions and Files
- Implement a Program to solve a Real World Problem.

UNIT - I INTRODUCTION TO PYTHON 11 Hour

Introduction to Python: Python, Features of Python, Execution of a Python, Program, Writing Our First Python Program, Data types in Python. Python Interpreter and Interactive Mode; **Values and Types**: int, float, Boolean, String, and List; Variables, Expressions, Statements, Tuple Assignment, Precedence of Operators, Comments; **Modules and Functions**: Function Definition and Use, Flow of Execution, Parameters and Arguments.

UNIT – II CONTROL STATEMENTS & FUNCTIONS 10 Hour

Control Statements: Boolean Values and operators - Conditional (if), Alternative (if-else), Chained Conditional (if-elif-else); **Iteration**: State, While, for, Break, Continue, Pass; **Fruitful Functions**: Return Values – Parameters - Local and Global Scope -Function Composition -Recursion.

UNIT - III ARRAYS, STRINGS &ILLUSTRATIVE PROGRAMS 11 Hour

Arrays: Lists as Arrays. **Strings:** String Slices – Immutability - String Functions and Methods - String Module; **Illustrative Programs:** Square Root –GCD – Exponentiation - Sum an Array of Numbers - Linear Search - Binary Search.

UNIT- IV LISTS & TUPLES 10 Hour

Lists: List Operations - List Slices - List Methods - List Loop – Mutability – Aliasing - Cloning Lists - List Parameters; **Tuples:** Tuple Assignment - Tuple as Return Value; **Dictionaries:** Operations and Methods; Advanced List Processing - List Comprehension; **Illustrative Programs:** Selection Sort - Insertion Sort - Merge Sort - Histogram.

UNIT- V FILES &EXCEPTION HANDLING 10 Hour

Files and Exception: Text Files, Reading and Writing Files, Format Operator; Command Line Arguments, Errors and Exceptions, Handling Exceptions, Modules, Packages; **Illustrative Programs:** Word Count, Copy File.

Text Books

- Mark Lutz. (2013). *Learning Python.* O'Reilly. (5th Ed.)
- Tony Gaddis. (2018). *Starting Out With Python.* Pearson. (4th Ed.)

Reference Books

- Kenneth A. Lambert. (2011). *Fundamentals of Python*.
- James Payne. (2010). *Beginning Python using Python.2.6 and Python 3.1*. wiley.

E-Resource

- http://www.sfu.ca/~eep2/Technology/Learning%20Python%205th%20Ed%202013.pdf

DATA STRUCTURES USING PYTHON PRACTICAL

UCSR207/UCAR205

Semester : II **Credit : 2**
Category : Major Core (DSC) - VI **Hour/Week : 3**
Class & Major : I B.Sc Computer Science **Total Hour :39**

Objectives:

To enable the Students

- Understand various Data Representation Techniques in the Real World.
- Implement Basic Concepts of Linear and Non-Linear Data Structures.
- Solve the Sorting and Searching Algorithms.

Learning Outcomes:

On Completion of the course, the students will be able to

- Understanding the writing Algorithms in solving Problems with the help of Fundamental Data Structures.
- Analyze the basic Concepts of Lists, Tuples, Trees and Graphs.
- Implement the Concepts of Searching and Sorting Techniques.

LIST OF PROGRAMS

1. Create a list of Elements where the Size of the List, Elements to be Inserted and Deleted are Dynamically given as Input.
2. Implement the Operations, Insertion, Deletion at a given Position in the List and Search for an Element in the list
3. Implement PUSH, POP Operations of Stack Operations.
4. Implement Add, Delete Operations of Queue.
5. Evaluate the Infix and Postfix Expression using Stack Operations
6. Implement the Graph Traversal Algorithms:
    a. Depth First Search.
    b. Breadth First Search
7. Binary Tree Traversal Using Linked List (In-order, Pre-order, Post-order).
8. Sorting Methods
    a. Bubble Sort
    b. Insertion Sort
    c. QuickSort
9. Searching Methods
    a. Linear Search
    b. Binary Search
    c. Fibonacci Search
10. Create a Binary Search Tree and Count the Number of Nodes in the Binary Search Tree.

ALLIED COURSES OFFERED TO OTHER DEPARTMENTS MULTIMEDIA

UCSA105

Semester	**: I**	**Credit : 3**
Category	**: Allied (GE-I)**	**Hour/Week: 3**
Class & Major:	**I B.Com (CA)**	**Total Hour :39**

Objectives:

To enable the students

- Understand the Concepts in Multimedia
- Apply Multimedia Concepts in Photoshop and Flash
- Develop Multimedia Applications with their Creative ideas.

Learning Outcomes:

On Completion of the course, the students will be able to

- Define Multimedia and Process.
- Understand Multimedia Components using various Tools and Techniques.
- Utilize the Different types of Media Format and their Properties.

UNIT - I INTRODUCTION 7 Hour

Definition - Classification - Multimedia Application - Multimedia Hardware - Multimedia Software - CDROM - DVD.

UNIT - II MULTIMEDIA AUDIO 8 Hour

Digital medium - Sound cards - Recording - Editing - MIDI fundamentals - Working with MIDI.

UNIT - III MULTIMEDIA TEXTAND GRAPHICS 8 Hour

Text in Multimedia - Multimedia Graphics**:** Coloring - Digital Imaging Fundamentals - Development and Editing.

UNIT - IV MULTIMEDIA ANIMATION AND VIDEO 8 Hour

Computer Animation Fundamentals - Animation s/w Tools and Techniques. Multimedia Video: How Video Works – Video Shooting – Video Capture Process.

UNIT - V MULTIMEDIA- PROJECT 8 Hour

Stages of Project - Design Concept - Authoring - Planning and Costing – Multimedia Team.

Text Books

- Gokul, S. (2008).*Multimedia Magic.* BPS Publication. (2nd Ed.). New Delhi.
- Tay Vaughan. (2010). *Multimedia: Making it Work.* Tata McGraw Hill. (9th Ed.). New Delhi.

Reference Books

- Ben Willmore. Dan Ablan. (2009). *Adobe Photoshop CS4 Studio Techniques*. Peachpit Publishers. (2nd Reprint). New Delhi.
- Nick Vandome. (2011). *Photoshop Elements 9.*jumpstart. McGraw Hill Ed. New Delhi.

e-Resources

- http://books.rediff.com/book/multimedia-magic/9788183330695
- https://www.google.co.in/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&cad=rja&uact=8&ved=0ahUKEwifpaWT74

MULTIMEDIA – LAB

UCSR111

Semester : I **Credit : 2**
Category : Allied (GE-II) **Hour/Week : 3**
Class &Major: I B.Com (CA) **Total Hour:39**

Objectives:

To enable the students

- Understand the Basic Concepts in Multimedia.
- Design Multimedia Projects in Photoshop and Flash.
- Develop Multimedia in Real Time Applications.

Learning Outcomes:

On Completion of the course, the students will be able to

- Use Basic Selection Tools and Edge Refinement to Isolate and Edit parts of an Image.
- Demonstrate Progress in Basic Drawing and Animation skills.
- Use Preset Brushes and Custom Brushes to Colorize Images, Enhance Images, and Build illustrations.

LIST OF PROGRAMS

1. Photo Effects: Image, Changing Cloth Texture and Pattern, Changing Background.
2. Create a Brick Wall Texture using Photoshop.
3. Photo Retouching: Color Correction, Blending Image.
4. Smooth Skin Effect.
5. Adding Blur Effect to Background.
6. Create Digital Banner using Text.
7. Bouncing a Ball in Flash.
8. Create Torn Paper Edge Effect.
9. Create a Masking Effect in Flash.
10. Create a Banner using Flash.

C PROGRAMMING

UCSA205

Semester : II **Credit : 3**
Category : Allied **Hour/Week: 3**
Class &Major : I B.Com CA **Total Hour :39**

Objectives:

To enable the students

- Understand the Basic Computer Knowledge
- Implement basic Concepts of the C Programming Language.
- Design, Build, Execute and Debug C Applications.

Learning Outcomes:

On Completion of the course, the students will be able to

- Understanding a Functional Hierarchical Code Organization.
- Ability to Define and Manage Data Structures Based on Problem Subject Domain.
- Ability to Work with Arrays of Complex Objects.

UNIT- I COMPUTER BASICS **8 Hour**

Introduction – Evolution, Generation & Classification of Computers – Computer system – Application of Computers. Input Devices, Output Devices, Storage Devices. **Information – Technology:** IT- Role of IT – IT and Internet – Careers in IT Industry. **Internet Tools**: Web

Browser – Browsing Internet – Email – Search Engines – Instant Messaging. E-Commerce – Electronic Data Interchange (EDI) – Mobile Communication – Bluetooth – Global Positioning System.

UNIT- II OVERVIEW OF C 8 Hour

Importance of C - C Program Structure - Sample C Program. Constants - Variables and Data Types - Character Set - C Tokens - Keywords and Identifiers - Declaration of Variables - Assigning Values to Variables - Operators – Expression - Arithmetic - Relational - Logical - Assignment - Increment - Decrement –Conditional - bitwise and Special Operators - Arithmetic Expressions - Operator Precedence - Type Conversions.

UNIT- III DECISION MAKING AND BRANCHING 7 Hour

Decision making with If - Simple IF - IF ELSE - Nested IF ELSE - ELSE IF Ladder – Switch - GOTO Statement. **Looping:** While - Do-While – For - Jumps in Loops.

UNIT- IV ARRAYS, STRINGS AND USERDEFINED FUNCTIONS 8 Hour

Declaration and Accessing of One &Two-dimensional Arrays - Initializing Two-Dimensional Arrays - Multidimensional Arrays.Declaring and Initializing String Variables – Reading Strings from Terminal – Writing Strings to Screen – Putting Strings Together – Comparison of Two Strings – String Handling Functions. User defined Functions -Recursion.

UNIT- VSTRUCTURES, UNIONS AND POINTERS 8 Hour

Defining - Giving Values to Members - Initialization and Comparison of Structure Variables - Arrays of Structure - Structures and Functions – Unions – Pointers.

Text Book

- Bala Gurusamy, E. (2012). *Programming in ANSI C.* Tata McGraw-Hill. (6th Ed.). New Delhi.

Reference Books

- Ashok N. Kamthane. (2006).*Programming in ANSI C and Turbo C.* Pearson Education. (3rd Ed). New Delhi.
- Yashavant Kanetkar, Y. (2010). *Let us C.*BPB Publication. (10th Ed.). New Delhi.

e-Resources

- http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-087-practical programming-in-c-january-iap-2010/lecture-notes/
- http://freevideolectures.com/Course/2519/C-Programming-and-Data-Structures/2
- http://www.powershow.com/view/d7c5Y2Y2N/OBJECT_ORIENTED_PROGRAMMING_powerpoint_ppt_presentation

C PROGRAMMING – LAB

UCSR208

Semester	**: II**	**Credit : 2**
Category	**: Allied**	**Hour/Week : 3**
Class &Major	**: I B.Com CA**	**Total Hour :39**

Objectives:

To enable the students

- Implement Basic Concepts of the C Programming Language.
- Develop Programs by using Control Structure, Arrays, Functions, Pointers and Files.
- Design, Build, Execute and Debug C Programs.

Learning Outcomes:

On Completion of the course, the students will be able to

- Recollect various Programming Constructs and to Develop C Programs.
- Understand the Fundamentals of C Programming.
- Choose the Right Data Representation Formats Based on the Requirements of the Problem.

LIST OF PROGRAMS

1. Write a C Program on Operators.
2. Write a C Program on Decision Making.
3. Write a C Program on Decision Looping.
4. Write a C Program on Arrays.
5. Write a C Program on Strings.
6. Write a C Program on Functions.
7. Write a C Program on Recursion.
8. Write a C Program on Structures.
9. Write a C Program on Union.
10. Write a C Program on Pointers.

NON-MAJOR ELECTIVES
TABLEAU PROGRAMMING

UCSE206

Semester	**: II**	**Credit : 2**
Category	**: NON MAJOR ELECTIVE**	**Hour/Week : 1T+ 2P**
Class &Major	**: I UG**	**Total Hour : 52**

Objectives:

To enable the students

- Learn Basic Concepts of Tableau Statistics and Tableau Interactive Dashboard.
- Acquire Knowledge in Master Tableau Reporting, Graphs, Maps, Table Calculation.
- Implementing the Concepts in Tableau.

Learning Outcomes:

On Completion of the course, the students will be able to

- Include Creating Advanced Visualizations, Formatting, Calculations, Tricks, and Tips to use Tableau.
- For Customer Analytic the Skills you will Gain as Predictive Analytics, Regression Analysis, Marketing Performer.

UNIT- I INTRODUCTION **5 Hour**

Introduction Tableau – Design Flow – File Types – Data Types - Connecting to Databases -Working with Data – Analyzing - Formatting.

UNIT- II CALCULATIONS **6 Hour**

Introduction to Calculations - Dashboard Development – Sharing - Data Calculations - Aggregate Calculations - User Calculations - Table Calculations - Logical Calculations -String Calculations - Number Calculations – LOD Expressions.

UNIT- III OPERATORS AND FUNCTIONS **5 Hour**

Type Conversion – Operators – Functions - Data Joining - Data Blending - Trendlines.

UNIT- IV SORTING AND FILTERING **5 Hour**

Add Worksheets – Paged Workbook – Sorting – Filtering Conditions - Filtering Measures - Grouping – Sets.

UNIT- V CHARTS **5 Hour**

Histograms - All Types of Charts - Tree Maps- Pareto Charts-Waterfall Charts-Bump Charts-Funnel Charts-Bollinger Bands.

LIST OF PROGRAMS

1. Data Visualization with Tableau - Tableau, Installation.
2. Basic Visualization Design - Exporting Data, Connecting Sheets, Loading into Tableau visualization Engine.
3. Visualizations Deep Dive - to Make Advance Charts and Graphs (Circle Plots, Side by Side Bars, Dual Charts, Area Charts, Tree Maps).
4. Data Organization - Calculated Metrics, Sorting, Filtering, Totals and Sub Totals, Various Aggregated Measures, Percentages.
5. Data Organization - Date and Time Functions, String Functions and logical Functions.
6. Playing with Time Dimension - Table Calculations, Moving Averages, Running Totals, Window Averages.
7. Incremental Loading and Blending - Custom SQL Queries, Creating Incremental Loads, Creating File Extractions.
8. Macros in Tableau – Parameters, Global Parameters.
9. Sharing Insights with Enterprise Dashboards - Creating Dashboards.

Text Book

- Joshua N. Milligan. (2015). *Learning Tableau.* Packt Publishing.

E-Resource

- https://www.tutorialspoint.com/tableau/

PYTHON PROGRAMMING

UCSE207

Semester : II **Credit: 2**
Category : NON MAJOR ELECTIVE **Hour/Week: 3**
Class &Major : I UG **Total Hour : 39**

Objectives:
To enable the students

- Implement Python Programs with Conditionals and Loops.
- Use Functions for Structuring Python Programs.
- Represent Compound data using Python Lists, Tuples, and Dictionaries.

Learning Outcomes:
On Completion of the course, the students will be able to

- Define and Demonstrate the use of Built-in Data Structures “Lists” and “Dictionary”.
- Design and Implement GUI Application and How to Handle Exceptions and Files.
- Implement a Program to Solve a Real World Problem.

LIST OF PROGRAMS:

1. Strings and Lists
    - To Calculate the Length of a String
    - To Get the Largest Number from a List
    - To Remove Duplicates from a List
2. Dictionary and Tuple
    - To Sort (Ascending and Descending) a Dictionary by Value
    - To Print a Dictionary Line by Line
    - To Create a Tuple with Different Data Types
3. Sets
    - To Create a Intersection, Union, and Difference of Sets
4. Array
    - To Append a New Item to the End of the Array.
    - To Remove the First Occurrence of a Specified Element from an Array
5. Conditional Statements
    - To Get the Fibonacci Series between 0 to 50.
    - To Accepts a String and Calculate the Number of Digits and Letters.
6. Functions
    - To Calculate the Factorial of a Number (a Non-negative Integer). The Function Accepts the Number as an Argument
    - To Reverse the Digits of an Integer
    - To Add Two Binary Numbers
7. Data structure
    - To Create an Enum Object and Display a Member Name and Value
    - To Compare Two Unordered lists (not sets).
    - To Push Three Items into the Heap and Print the Items from the Heap.
8. Searching and Sorting
    - Binary Search
    - Insertion Sort

R PROGRAMMING

UCSE208

Semester	**: II**	**Credit: 2**
Category	**: NON MAJOR ELECTIVE**	**Hour/Week: 3**
Class &Major	**: I UG**	**Total Hour : 39**

Objectives:

To enable the students

- Understand the Different Data Types in R.
- Use of Vectorized Calculations and Control Statements.
- Write User-Defined R Functions and Loop Constructs in R.

Learning Outcomes:

On Completion of the course, the students will be able to

- Understand the Basics in R Programming in terms of Constructs, Control Statements, String Functions.
- Understand the Use of R for Big Data Analytics.
- Learn to Apply R Programming for Text Processing.

LIST OF PROGRAMS

1. Write a Program on Vectors and Matrices.
2. Write a Program on Lists.
3. Write a Program on Factors.
4. Write a Program on Data Frame.
5. Write a Program on Array.
6. Write a Program on Time Series.
7. Write a Program on Storing data as Textual and Binary Format.
8. Write a Program on Reading And Writing Data in Files.
9. Write a Program on Functions.
10. Write a Program on Control Structures.
11. Write a Program on Debugging.
12. Write a Program on Simulations.

ARDUINO PROGRAMMING

UCSE209

Semester	**: II**	**Credit: 2**
Category	**: NON MAJOR ELECTIVE**	**Hour/Week: 3**
Class &Major	**: I UG**	**Total Hour :39**

Objectives:

To enable the students

- Understand the Basic of Arduino Programming.
- Develop a Basic Program in Arduino.
- Gain Knowledge in Arduino Software.

Learning Outcomes:

On Completion of the course, the students will be able to

- Provide Knowledge of Different Smart System Applications.
- Familiarize Students with Arduino as IDE, Programming Language & Platform.
- Provide Knowledge of Arduino Boards and Basic Components.

LIST OF PROGRAMS

1. Write a Program on Structure and Flow.
2. Write a Program on Variables.
3. Write a Program on Operators - Arithmetic Operators, Relational Operators, Logical Operators, Conditional Operator, and Increment Operator and Commenting.
4. Write a Program on Decision Statement - if Statement , if-else, and if-else-if.
5. Write a Program on Switch and Break.
6. Write a Program on Looping - For Loop and While Loop.
7. Write a Program on Functions – Calling Function and Returning a Value from a Function.
8. Write a Program on Arrays.
9. Write a Program on Strings.
10. Write a Program on Serial Input.

E-Resource

- https://startingelectronics.org/software/arduino/learn-to-program-course/

GO PROGRAMMING

UCSE210

Semester	**: II**	**Credit: 2**
Category	**: NME (SKILL ENHANCEMENT COURSE)**	**Hour/Week:3P**
Class & Major	**: I UG**	**Total Hour : 39**

Objectives:

To enable the students

- Learn Basics of Syntax of Go Programming.
- Understand Interfaces, Go Routines and Advanced concepts of Go.
- Implement a Good Distributed Application.

Learning Outcomes:

On Completion of the course, the students will be able to

- Understand and Use the Basic Programming Constructs of GO Language.
- Manipulate various GO Language Data types, such as Arrays, Strings, and Pointers.
- Write Faster and Modular Code, for Real-World.

LIST OF PROGRAMS

1. To Create an Operators using Go Programming.
2. To Create a Decision Making Statement using Go Programming.
3. To Create a Looping - For Loop and While Loop using Go Programming.
4. To Create a Functions using Go Programming.
5. To Create an Arrays and Strings using Go Programming.
6. To Create a Pointers using Go Programming.
7. To Create a Structures using Go Programming.
8. To Create a Slices & Ranges using Go Programming.
9. To Create a Interfaces using Go Programming.
10. To Create an Error Handling using Go Programming.

Text Book

- Alan, A. A. Donovan, Brian W. Kernighan. (2015). *The Go Programming Language.* Published in Paperback and Nov 20 in E-Book Addison-Wesley.

III AND IV EVALUATION COMPONENTS OF CIA

Semester	Part	Category	Course Code	Course Title	Component III	Component IV
I	III	Major Core (DSC) - I	UCSM110/ UCAM110	Principles of Information Technology	Assignment	Assignment
	III	Major Core (DSC) - II	UCSM109	Programming Methodology	Assignment	Problem Solving
	III	Major Core (DSC) - III	UCSR110	Programming Methodology - Practical	DPA	Viva-voce
II	III	Major Core (DSC) - IV	UCSM207	Data Structures	Assignment	Problem Solving
	III	Major Core (DSC) - V	UCSM208	Python Programming	Assignment	Problem Solving
	III	Major Core (DSC) - VI	UCSR207	Data Structures using Python - Practical	DPA	Viva-voce

ALLIED COURSES OFFERED TO OTHER DEPARTMENT

Semester	Part	Category	Course Code	Course Title	Component III	Component IV
I	III	Allied	UCSA105	Multimedia	Assignment	Poster Presentation
	III	Allied Practical	UCSR111	Multimedia Lab	DPA	Viva-voce
II	III	Allied	UCSA205	C Programming	Assignment	Problem Solving
	III	Allied Practical	UCSR208	C Programming Lab	DPA	Viva-voce

NON-MAJOR ELECTIVES

Semester	Part	Category	Course Code	Course Title	Component III	Component IV
II	IV	Non Major Elective	UCSE206	Tableau Programming	Assignment	Problem Solving
			UCSE207	Python Programming	DPA	Viva-voce
			UCSE208	R Programming	DPA	Viva-voce
			UCSE209	Arduino Programming	DPA	Viva-voce
			UCSE210	GO Programming	DPA	Viva-voce

PROGRAMME PROFILE M.Sc. (Computer Science)

PSO1: Demonstration of the Knowledge of Advanced Programming Skills and Distributed Environmental Need for Sustainable Development.

PSO2: Ability to Design and Develop Hardware and Software in Emerging Technology Environments.

PSO3: Ability to Solve Problems using the Techniques of Data Analytics like Pattern Recognition and Knowledge Discovery.

PSO4: Ability to Work out Effective and Efficient Real Time Solutions using Acquired Knowledge in Various Domains.

Semester	Category	Course Code	Course Title	Previous Course Code	Contact Hrs/ Week	Credit Min/Max
I	Core I	PCSM113	Principles of Concurrent Programming	-	5	4
	Core II	PCSM116	Digital Image Processing	PCSM404	4	4
	Core III	PCSM117	TCP / IP Networks	PCSM213	5	4
	Core IV	PCSM118	Compiler Design	-	4	4
	Core V	PCSM119	Mobile Computing	-	4	4
	Core VI	PCSR107	Digital Image Processing – Practical	-	4	3
	Core VII	PCSR108	TCP/IP Networks – Practical	-	3	2
	Extra Credit		Online Course (NPTEL/SWAYAM)	-	-	-/2
			Library	-	1	-
				Total	**30**	**25/27**
II	Core VIII	PCSM214	Big Data Analytics	PCSM315	4	3
	Core IX	PCSM215	Machine Learning	-	4	4
	Core X	PCSM216	Blockchain Technology	-	4	3
	Core XI	PCSM217	Software Testing	PCSM211	4	3
	Core XII	PCSR208	Big Data Analytics – Practical	PCSR306	4	3
	Core XIII	PCSR209	Machine Learning using Google CoLab – Practical	-	4	3
	Non Major Elective	PALE201/ PALE301		-	5	4
	Service Learning	PCSX201/ PCAX201		-	-	1
			Library	-	1	-
				Total	**30**	**24**

Semester	Category		Course Code	Course Title	Previous Course Code	Contact Hrs/ Week
III	Core XIV	PCSM316	Data Science	-	4	4
	Core XV	PCSM317	Augmented Virtual Reality	-	4	4
	Core XVI	PCSM318	Artificial Intelligence and Robotics	PCSM406	4	3
	Core XVII	PCSM313	Research Methodology	-	4	4
	Core XVIII	PCSI301	Fuzzy Set and Systems	-	5	4
	Core XIX	PCSR307	Data Science using Python Pandas - Practical	-	3	2
	Core XX	PCSR308	Augmented Virtual Reality using VRML - Practical	-	3	3
	Core XXI	PCSR303	Project	-	2	2
			Library	-	1	-
Total					**30**	**24**
IV	Core XXII	PCSM407	Fog computing		5	4
	Core XXIII	PCSM408	Cyber Security	PCSM314	4	3
	Core XIV	PCSP402	Project		20	10
			Library		1	-
Total					**30**	**17**
Grand Total					**120**	**90/92**

Minimum one MOOC (Compulsory Audit Course) has to complete during the first year.

EXTRA CREDIT EARNING PROVISION

Semester	Category	Course Code	Course Title	Hrs/Week	Credit	
					Min	Max
III	Self Study Paper	PCSS301/PCAS502	R-Programming	2	-	2
III	Self Study Paper	PCSS302/PCAS503	Rich Internet Applications	2	-	2
IV	Self Study Paper	PCSS401/PCAS601	Silver Light Applications	2	-	2
IV	Self Study Paper	PCSS402/PCAS602	Extreme Programming	2	-	2

COURSES OFFERED TO OTHER DEPARTMENTS
(Major and Major Elective)

Course	Semester	Category	Course Code	Course Title	Contact Hrs/Week	Credit
M.A. Tamil	IV	Major Elective	PTAM406	Kanini Payanpattiyal	5	3

NON-MAJOR ELECTIVE

Semester	Category	Course Code	Course Title	Contact Hrs/Week	Credit
II	Non Major Elective	PCSE206	Mobile Computing Practical	5P	4
		PCSE207	Web Based App Development	5P	4

PRINCIPLES OF CONCURRENT PROGRAMMING

PCSM113

Semester : I **Credit: 4**
Category : Major Core I **Hour/Week: 5**
Class &Major : IM.Sc Computer Science **Total Hour: 65**

Objectives:
To enable the students

- Understand the Concepts of Concurrent Programming.
- Design and Build Applications.
- Analyse the Concept and its Associated Package.

Learning Outcomes:
On Completion of the course, the students will be able to

- Understand the Conceptual Foundations of Concurrent Programming.
- Analyse the Effective ways of Structuring Concurrent and Distributed Programs.
- Implement the Concurrent Programming Abstractions Demonstrated by means of Functional Languages.

UNIT – I INTRODUCTION &CONCURRENT PROGRAMMING 13 Hour

Introduction - Concurrency as Abstract Parallelism – Multitasking - The Terminology of concurrency - The Challenge of Concurrent Programming - The Role of Abstraction - Justification of the Abstraction - Atomic Statements – Correctness – Fairness - Volatile and Non-Atomic Variables - Concurrency in Go.

UNIT – II CRITICAL SECTION PROBLEM, VERIFICATION OFCONCURRENT PROGRAMS & ADVANCE ALGORITHMS 13 Hour

The Definition of the Problem - First Attempt - Proving correctness with State Diagrams - Second Attempt, Third Attempt, Fourth Attempt - Dekker's Algorithm - Complex Atomic Statements - Logical Specification of Correctness Properties - Basic and Advanced Concepts of Temporal Logic - A Deductive Proof of Dekker's Algorithm - Spin and the Promela Modeling Language - The Bakery Algorithm - Fast Algorithms.

UNIT – III SEMAPHORES AND MONITORS 13 Hour

Definition of the Semaphore Type - Semaphore invariants - The Critical Section Problem for N Processes - Definitions of Semaphores - Barz's Simulation of General Semaphores - Udding's Starvation - Semaphores in Go - Monitors - Declaring and using Monitors - Condition variables - The Producer and Consumer Problem - The Problem of the Readers and Writers - Correctness of the Readers and Writers Algorithm - A Monitor Solution for the Dining Philosophers – Monitor in Go.

UNIT – IV CHANNELS, SPACE AND DISTRIBUTED ALGORITHMS 13 Hour

Models for Communications – Channels - Parallel Matrix Multiplication - Channels in Promela – Rendezvous - The Linda Model - Expressiveness of the Linda Model - Formal Parameters - The Master–Worker Paradigm - Implementations of Spaces - Distributed Algorithms - Implementations - Correctness of the Ricart – Agrawala Algorithm - The RA Algorithm in Promela - Token-Passing Algorithms - Tokens in Virtual Trees.

UNIT – V GLOBAL PROPERTIES, CONSENSUS AND REAL-TIME SYSTEMS 13 Hour

Global Properties - Distributed Termination - The Dijkstra–Scholten Algorithm - Credit-Recovery Algorithms. Consensus - The Problem Statement - A One-Round Algorithm - The Byzantine Generals Algorithm - Crash failures - Byzantine Failures with Three and Four Generals. Real-Time Systems - Reliability and Repeatability – Synchronous and Asynchronous Systems - The Mars Pathfinder in Spin - Simpson's Four-Slot Algorithm - The Ravenscar Profile – UPPAAL - Scheduling Algorithms for Real-Time Systems.

Text Book

- Ben-Ari, M. (2015). *Principles of Concurrent and Distributed Programming.* Addison-Wesley Publications. (2nd Ed.).

Reference Book

- Gregory, R. Andrews. (2012).*Concurrent Programming Principles and Practice.* Benjamin-Cummings Publishing Co. Inc. Subs. of Addison-Wesley Longman Publications. United States.

DIGITAL IMAGE PROCESSING

PCSM116

Semester : I	**Credit: 4**
Category : Major Core II	**Hour/Week: 4**
Class & Major: I M.Sc Computer Science	**Total Hour:52**

Objectives:

To enable the students

- Understand the Techniques of Processing Images in Different File Formats.
- Examine Different Image Enhancement and Segmentation Techniques.
- Implement Fundamental Image Processing Techniques with OpenCV.

Learning Outcomes:

On Completion of the course, the students will be able to

- Apply Image Enhancement and Restoration Techniques.
- Use Image Compression and Segmentation Techniques.
- Apply Hough Transform for Line, Circle, and Ellipse Detections.

UNIT - I INTRODUCTION 10 Hour

Introduction - What is Image Processing- Examples of Fields that uses DIP Fundamentals Step in DIP. Digital Image Fundamentals – Image Sensing and Acquisition, Image Sampling and Quantization – Basic relationship between Pixels.

UNIT- II IMAGE ENHANCEMENT 10 Hour

Spatial Domain - Gray level transformations - Histogram processing - Spatial filtering - Smoothing and Sharpening - Frequency Domain: Filtering in Frequency Domain - DFT, FFT, DCT - Smoothing and Sharpening Filters - Homomorphic Filtering.

UNIT–III IMAGE COMPRESSION AND IMAGE SEGMENTATION 10 Hour

Compression Methods: Huffman Coding – Arithmetic Coding – LZW Coding – Run Length Coding – Bit-Plane coding. EdgeLinking and Boundary Detection - Thresholding - Region Based Segmentation - Morphological Watersheds - Motion Segmentation, Feature Analysis and Extraction.

.

UNIT - IV INTRODUCTION TO OPENCV 11 Hour

An Introduction to OpenCV – Structure of OpenCV- Reading and Writing Image and Video Files. Image Processing Tools: BasicData Types-Pixel-Level Access-Common Operations with Images Arithmetic Operations-Histograms. Correcting and Enhancing Images: Image Filtering –Smoothing, Sharpening, Working with Image Pyramids. Morphological Operations - Geometric Transformation.

UNIT - V IMAGE PROCESSING IN OPENCV 11 Hour

Processing Color:-Color Spaces- Conversion between Color Spaces (cvt Color) RGB, Grayscale, YCrCb and HSV. Image Processing for Video: Video stabilization- Super resolution. Computational Photography: High-dynamic-range images-Creating HDR images - Tone mapping - Seamless cloning-Decolorization.

CASE Study: Exploring Structure from Motion Using OpenCV - Number Plate Recognition Using SVM and Neural Networks- Face Recognition using Eigen Faces or Fisher Faces.

Text Books

- Rafael, C. Gonzalez. Richard, E. Woods. (2017). *Digital Image Processing*. (4thEd). Pearson/Prentice Hall.
- Gloria Bueno Garcia. Oscar Deniz Suarez. (2015). *Learning Image Processing with OpenCV*. Packet Publishing Ltd. (1st Ed).

Reference Books

- Rafael C. Gonzalez. Richard, E. Woods. (2017). *Digital Image Processing*. Global Edition. PHI/Pearson Education. (4th Ed.)
- Rafael, C. Gonzalez. Richard, E. Woods. Steven, L. Eddins. *Digital Image Processing Using MATLAB*. (2nd Ed.).McGraw Hill Education.

e-Resources

- http://www.w3schools.com.
- http://www.youtube.com
- http://www.nptel.ac.in/courses/106105032/

TCP/IP NETWORKS

PCSM117

Semester : II	**Credit : 4**
Category : Core VIII	**Hour/Week : 5**
Class &Major: I M.Sc Computer Science	**Total Hour :65**

Objectives:

To enable the students

- Understand the Concepts of TCP/IP.
- Examine the Process of TCP/IP.
- Implement TCP/IP Concepts in Network.

Learning Outcomes:

On Completion of the course, the students will be able to

- Apply Programming Skills in TCP/IP Network Model.
- Understand and Configure IP Addresses.
- Analyse of Data Traffic on TCP/IP Networks.

UNIT- I INTRODUCTION 13 Hour

Internetworking Concepts and Architectural Model- Classful Internet Addresses – CIDR-Subnetting and Supernetting –ARP- RARP- IP – IP Routing –ICMP – Ipv6.

UNIT - II TCP 13 Hour

Services – Header – Connection Establishment and Termination- Interactive Data Flow-Bulk Data Flow- Timeout and Retransmission – Persist Timer - Keepalive Timer- Futures and Performance.

UNIT- III IP IMPLEMENTATION 13 Hour

IP Global Software Organization – Routing Table- Routing Algorithms-Fragmentation and Reassembly- Error Processing (ICMP) –Multicast Processing (IGMP).

UNIT - IV TCP IMPLEMENTATION - I 13 Hour

Data Structure and Input Processing – Transmission Control Blocks- Segment Format-Comparison-Finite State Machine Implementation-Output Processing- Mutual Exclusion-Computing the TCP Data Length.

UNIT- V TCP IMPLEMENTATION - II 13 Hour

Timers-Events and Messages- Timer Process- Deleting and Inserting Timer Event- Flow Control and Adaptive Retransmission-Congestion Avoidance and Control – Urgent Data Processing and Push Function.

Text Books

- Douglas, E. Comer. (2013). *Internetworking with TCP/IP Principles- Protocols and Architecture*-Vol.1 & 2. (6th Ed.). Pearson Education Asia.(Unit I in Comer Vol. I- Units II- IV & V – Comer Vol. II).
- Richard Stevens, W.(2011).*TCP/IP Illustrated.* (Volume 1- 6thEd.). Pearson Education. (Unit II).

Reference Books

- Forouzan. (2003). *TCP/IP Protocol Suite*. (2ndEd.). TMH.
- Richard Stevens, W. (2003). *TCP/IP Illustrated*- Volume 2- Pearson Education.

COMPILER DESIGN

PCSM118

Semester : I	**Credit : 4**
Category : Major Core IV	**Hour/Week: 4**
Class & Major: I M.Sc Computer Science	**Total Hour: 52**

Objectives:
To enable the students

- Understand the concept of Data Structure and Algorithm Design.
- Analyze the Context-Free Grammars, Lexical Analysis and Parsing Techniques**.**
- Implement the Process of Translating a Modern High-Level Language to Executable Code Required for Compiler Construction.

Learning Outcomes:
On Completion of the course, the students will be able to

- Acquire Knowledge of Modern Compiler & its Features.
- Learn & use Modern tools and Technologies for Designing New Compiler.
- Implement the Knowledge of Patterns, Tokens & Regular Expressions.

UNIT – I INTRODUCTION TO COMPILERS 10 Hour

Structure of a Compiler – Lexical Analysis – Role of Lexical Analyzer – Input Buffering – Specification of Tokens – Recognition of Tokens – Lex – Finite Automata – Regular Expressions to Automata – Minimizing DFA.

UNIT – II SYNTAX ANALYSIS 12 Hour

Role of Parser – Grammars – Error Handling – Context-Free Grammars – Writing a grammar, Top-Down Parsing – General Strategies Recursive Descent Parser – Predictive Parser-LL(1) Parser-Shift Reduce Parser-LR Parser-LR (0) Item Construction of SLR Parsing Table - Introduction to LALR Parser – Error Handling and Recovery in Syntax Analyzer-YACC.

UNIT – III INTERMEDIATE CODE GENERATION 10 Hour

Syntax Directed Definitions- Evaluation Orders for Syntax Directed Definitions- Intermediate Languages: Syntax Tree- Three Address Code- Types and Declarations- Translation of Expressions- Type Checking.

UNIT – IV RUN-TIME ENVIRONMENT AND CODE GENERATION 10 Hour

Storage Organization- Stack Allocation Space- Access to Non-local Data on the Stack-Heap Management – Issues in Code Generation – Design of a Simple Code Generator.

UNIT –V CODE OPTIMIZATION 10 Hour

Principal Sources of Optimization – Peep-Hole Optimization – DAG- Optimization of Basic Blocks-Global Data Flow Analysis – Efficient Data Flow Algorithm.

Text Books

- Godfrey Winster, S. Aruna Devi, R. Sujatha. (2018). *Compiler Design.* Yesdee Publications.
- Sudharani, S. Karthi Mand Rajkumar. Y. (2019). *Compiler Design.*.I.K.International.

Reference Book

- Aho, A.V. Monica, R. Sethi, J.D. Ullman. (2018). *Compilers, Principles, Techniques and Tools.* Pearson Education/Addison Wesley.

.

MOBILE COMPUTING

PCSM119

Semester :I	**Credits: 4**
Category : Major Core V	**Hour/Week :4**
Class &Major: I M.Sc. Computer Science	**Total Hour:52**

Objectives:
To enable the students

- Learn the Basic Concepts of GSM, SMS, and GPRS Architecture.
- Acquire Knowledge of Wireless Protocols -WLN, Bluetooth, WAP, Zig Beeissues.
- Implement the Concepts of Mobile Application Development Platform.

Learning Outcomes:
On Completion of the course, the students will be able to

- Understand the Infrastructures and Technologies of Mobile Computing Technologies.
- Impart Knowledge on Principles and Theories of Mobile computing Technologies.
- Analyse the Future of Mobile Computing Technologies and Applications.

UNIT- I WIRELESS COMMUNICATION FUNDAMENTALS ARCHITECTURE 10 Hour

Frequency Spectrum-Multiplexing-Spread Spectrum-GSM vs CDMA --Comparison of 2G, 3G, 4G -GSM Architecture-Entities-Call Routing-Address and Identifiers-GSM Protocol Architecture-Mobility Management-Frequency Allocation-Security –GPRS Architecture (Entity and Protocol).

UNIT- II MOBILE WIRELESS SHORT RANGE NETWORKS 10 Hour

Introduction-WLAN Equipment-WLAN Topologies-WLAN Technologies-IEEE 802.11 Architecture-WLAN MAC-Security of WLAN, Power Management-Standards-WAP Architecture-Bluetooth Enabled Devices Network-Layers in Bluetooth Protocol-Security in Bluetooth-IrDA-ZigBee.

UNIT- III MOBILE IP NETWORK LAYER, TRANSPORT LAYER 10 Hour

IP and Mobile IP Network Layer-Packet delivery and Handover Management-Location Management-Registration-Tunneling and Encapsulation-Route Optimization-Mobile Transport Layer-Conventional TCP/IP Transport Layer Protocol-Indirect, Snooping, Mobile TCP.

UNIT- IV MOBILE APPLICATION DEVELOPMENTUSING ANDROID 11 Hour

Mobile Applications Development -Understanding the Android Software Stack –Android Application Architecture –The Android Application Life Cycle –The Activity Life Cycle-Creating Android Activity -Views-Layout -Creating User Interfaces with basic Views-Linking Activities with Intents.

UNIT -V MOBILE APPLICATION DEVELOPMENT USING ANDROID 11 Hour

Services-Broadcast Receivers –Adapters –Data Storage, Retrieval and Sharing-Location Based Services-Development of Simple Mobile Applications.

Text Books

- Asoke, K. Talukder. Hasan Ahmed. Roopa, R. Yavagal. (2010). *Mobile Computing*. Tata McGraw Hill Pub. (2nd Ed.)
- Barry, A. Burd. (2015). *Android Application Development for Dummies All in One*. Wiley.
- Ed, Burnette. Hello. (2012). *Android: Introducing Google's Mobile Development Platform*. (3rd Ed.). Pragmatic Programmers.

Reference Books

- Maritn, Sauter. (2011). *From GSM to LTE: An Introduction to Mobile Networks and Mobile Broadband*. John Wiley and Sons.
- Raj Kamal. (2012). *Mobile Computing*. Oxford Higher Education. (2nd Ed.).

DIGITAL IMAGE PROCESSING – PRACTICAL

PCSR107

Semester : I	**Credit: 3**
Category : Major Core VI	**Hour/Week: 4**
Class & Major: I M.Sc. Computer Science	**Total Hour :52**

Objectives:

To enable the Students

- Develop Image Enhancement Techniques
- Design Algorithms to Solve Image Processing Problems
- Implement the Image Fundamentals and Mathematical Transforms.

Learning Outcomes:

On Completion of the course, the students will be able to

- Understand an Image Transformation and its Histogram.
- Apply Image Enhancement and Restoration Techniques.
- Implement Image Compression and Segmentation Techniques.

LIST OF PROGRAMS

1. Convert Color Image into Gray Scale Image.
2. Image Transformation using Fourier Transformation.
3. Histogram Equalization to Improve the Contrast of Images.

4. Water Shed Algorithm for Marker-based Image Segmentation.
5. Erosion and Dilation Operation in Image.
6. Sobel Filter in Edge Detection Method in Image
7. Canny Edge Detection method in Image.
8. Opening and Closing Operation in Image.
9. Hit and Miss Transformation in Morphological Image.
10. To find Objects in an Image using Template Matching Concepts

TCP/IP NETWORKS PRACTICAL

PCSR108

Semester : I	**Credit : 2**
Category : Major Core VII	**Hour/Week : 3**
Class & Major: I M.Sc. Computer Science	**Total Hour :39**

Objectives:

To enable the Students

- Implement Basic Concepts of TCP /IP Network.
- Develop Programs with Packet Tracer Software.
- Have Hands on Experience on Various Networking Commands.

Learning Outcomes:

On Completion of the course, the students will be able to

- Apply Programming Skills in TCP/IP Network Model.
- Understand and Configure IP Addresses.
- Analyse of Data Traffic on TCP/IP Networks.

LIST OF PROGRAMS

1. Socket Program for
    a. Ping Command in Java.
    b. TRACEROUTE command in Java
2. Implementation of IP address Configuration
3. Implementation of
    a. Stop and Wait Protocol
    b. Sliding Window Protocol
4. Simulation of ARP /RARP Protocols
5. Socket Program for NSlookupCommand in Java.
6. Simulation of Error Correction Code (like CRC).
7. Simulation of DNS using UDP Sockets

8. Implementation of TCP (like Packet Capturing and Filtering)
9. Implementation of
   a. go-back-n Protocol
   b. Selective Repeat Protocol.
10. Study of Wire Shark Tool for SDN and Hypervisor for Network.

BIG DATA ANALYTICS

PCSM214

Semester : II	**Credit: 3**
Category : Core VIII	**Hour / Week : 4**
Class & Major: I M. Sc Computer Science	**Total Hour : 52**

Objectives
To enable the students

- Understand the Concepts in Big Data and Apply Hadoop Ecosystem Components.
- Get Introduced to Tools like Pig, Hive, HBase, Elastic MapReduce etc.
- Realize the Hadoop Architecture and Implementation of MapReduce Application.
- Acquire Knowledge on Variety of NoSQL Databases.

Learning Outcomes:
On Completion of the course, the students will be able to

- Ability to Identify the Characteristics of Datasets and Compare the Trivial Data and Big Data for Various Applications.
- Ability to Solve Problems Associated with Batch Learning and Online Learning, and the Big data Characteristics.
- Ability to Integrate Machine Learning Libraries and Mathematical and Statistical Tools with Modern Technologies like Hadoop and Map reduce.

UNIT – I BIG DATA & HDFS 10 Hour

Big Data Definition and Taxonomy – Challenges for Processing Big Data - Technologies Support Big Data - Big Data Value for the Enterprise – Setting up the Environment – First Step with the Hadoop Ecosystem. HDFS Architecture – HDFS Concepts – Blocks – Name Node – Secondary Name Node – Data Node – HDFS Federation – Basic File System Operations – Data Flow – Anatomy of File Read – Anatomy of File Write.

UNIT – II HADOOP 10 Hour

Introduction to Hadoop – History of Hadoop – Hadoop Architecture Concepts- Components of Hadoop – Applications of Hadoop – Advantages/Disadvantages of Hadoop - Compression – Security – Enterprise Integration in Hadoop. Use cases of Hadoop – RDBMS vs Hadoop – Ecosystem Tour – Vendor comparison.

UNIT – III PIG 11 Hour

Introduction to Apache Pig – Map Reduce Vs. Apache Pig – SQL vs. Apache Pig – Different Data Types in Pig – Modes of Execution in Pig – Execution Mechanism- Local Mode- Map Reduce or Distributed Mode – Grunt Shell – Loading data – Exploring Pig – Latin commands - Embedded - Transformations in Pig - How to Write a Simple Pig Script - How to Develop the Complex Pig Script - Bags, Tuples and fields in PIG - UDFs in Pig - Need of using UDFs in PIG.

UNIT – IV HIVE AND HBASE 10 Hour

Hive Introduction – Hive Architecture – Hive vs RDBMS – HiveQL and the Shell - Managing Tables (External vs Managed) – Data Types and Schemas – Partitions and Buckets.
HBASE: Architecture and Schema Design - HBase vs. RDBMS- HMaster and Region Servers- Column Families and Regions- Write Pipeline- Read pipeline- HBase Commands.

UNIT – V FRAMEWORKS AND APPLICATIONS 10 Hour

IBM for Big Data – Map Reduce Framework – Algorithms using Map Reduce - Map Reduce Types and Formats- Map Reduce Features- Sharding – NoSQL Databases - S3– Hbase – Impala – Analyzing Big Data with Twitter – Big Data for E-Commerce – Big Data for Blogs.

Text Books

- Michael, Berthold. David, J. Hand. (2007). *Intelligent Data Analysis*. Springer.
- Anand Rajaraman and Jeffrey David Ullman. (2012). *Mining of Massive Datasets*. Cambridge. University Press.
- Paul Zikopoulos. Chris Eaton. Paul Zikopoulos. (2012). *Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data*. McGraw Hill.

Reference Books

- Jay Liebowitz. (2013). *Big Data and Business Analytics.* Auerbach Publications. CRC Press.
- EMC. (2015). *Education Services*, *Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data*. (1st Ed.).
- Arvind, Sathi. (2012). *Big Data Analytics: Disruptive Technologies for Changing the Game*. MC Press.

E-Resources

- http://postscapes.com
- http://www.bigdataanalysis.eu/what-is-bigdata

MACHINE LEARNING

PCSM215

Semester	**: II**	**Credit: 4**
Category	**: Major Core IX**	**Hour/Week: 4**
Class &Major	**: I M.Sc Computer Science**	**Total Hour: 52**

Objectives:

To enable the students

- Understand the Underlying Mathematical Relationships Across Various Machine Learning Algorithms.
- Analyse the Supervised, Unsupervised Machine Learning Approaches.
- Design and Implement Machine Learning Algorithms to Real World Applications.

Learning Outcomes:

On Completion of the course, the students will be able to

- Understand a Wide Variety of Learning Algorithms.
- Develop Learning Models from Data.
- Evaluate Models Generated from Data.

UNIT- I INTRODUCTION 10 Hour

Concept of Machine Learning, Applications of Machine Learning, Key elements of Machine Learning, Supervised vs. Unsupervised Learning.

UNIT- IISUPERVISED LEARNING ALGORITHMS 11 Hour

Learning a Class from Examples, Linear, Non-Linear, Multi-Class and Multi-Label Classification, Decision Trees: ID3, Classification and Regression Trees (CART), Regression: Linear Regression, Multiple Linear Regression, Logistic Regression.

UNIT- IIIADVANCED SUPERVISED LEARING ALGORITHMS 10 Hour

Neural Networks: Introduction, Perceptron, Multilayer Perceptron, Support Vector Machines: Linear and Non-Linear, Kernel Functions, K-Nearest Neighbors.

UNIT- IV UNSUPERVISED ALGORITHM 11 Hour

Introduction to Clustering, Hierarchical: AGNES, DIANA, Partitional: K-means Clustering, K-Mode Clustering, Self-Organizing Map, Expectation Maximization, Gaussian Mixture Models, Principal Component Analysis (PCA).

UNIT- V PROBABILISTIC LEARNING 10 Hour

Bayesian Learning, Bayes Optimal Classifier, Naïve Bayes Classifier, Mining Frequent Patterns.

Text Books

- Ethem Alpaydin. (2014). *Introduction to Machine Learning.* MIT Press. Prentice Hall of India. (3rd Ed.)
- Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar. (2012).*Foundations of Machine Learning.* MIT Press.
- Tom, M. Mitchell. (2013). *Machine Learning*. Tata McGraw-Hill Education.(1st Ed.).

Reference Books

- Mevin, P. Murphy. (2012). *Machine Learning: A Probabilistic Perspective*. The MIT Press.
- Christopher, M. Bishop. (2007).*Pattern Recognition and Machine Learning*. Springer.

BLOCKCHAIN TECHNOLOGY

PCSM216

Semester	**: II**	**Credit: 3**
Category	**: Major Core X**	**Hour/Week: 4**
Class &Major	**:I M.Sc Computer Science**	**Total Hour: 52**

Objectives:

To enable the students

- Acquire Knowledge on Types of Blockchain Technologies.
- Understand the Concept of Private and Public Blockchain
- Analyse Security and Applications in Blockchain Technology.

Learning Outcomes:

On Completion of the course, the students will be able to

- Evaluate Blockchain Technologies, their core Components, Protocols, and use Cases.
- Design and Build Blockchain Applications.
- Inculcate the State of the art and Emerging use cases of Blockchain.

UNIT- I FUNDAMENTALS OF BLOCK CHAIN 10 Hour

Introduction - Origin of Blockchain - Blockchain Solution - Components of Blockchain - Components of Blockchain - Block in Blockchain - The Technology and the Future.

UNIT- II BLOCKCHAIN TYPES AND CONSENSUS MECHANISM 11 Hour

Introduction - Decentralization and Distribution - Types of Blockchain - Consensus Protocol - CRYPTOCURRENCY - BITCOIN, ALTCOIN and TOKEN: Introduction - Bitcoin and Cryptocurrency Basics - Types of Cryptocurrency - Cryptocurrency Usage.

UNIT-III PUBLIC BLOCKCHAIN SYSTEM 10 Hour

Introduction - Public Blockchain - Popular Public Blockchains - The Bitcoin Blockchain– Etherum Blockchain.

UNIT- IV PRIVATE BLOCKCHAIN SYSTEM 11 Hour

Introduction - Key Characteristics of Private Blockchain - Private Blockchain Examples - Private Blockchain and Open Source - E-Commerce Site Examples - Various Commands in E-Commerce Blockchain - Smart Contract in Private Environment - State Machine - Different Algorithms of Permissioned Blockchain - Byzantine Fault - Multichain

UNIT- V SECURITY IN BLOCKCHAIN 10 Hour

Introduction - Security aspects in Bitcoin - Security and Privacy Challenges of Blockchain in General - Performance and Scalability - Identity Management and Authentication - Regularity Compliance and Assurance - Safeguarding Blockchain Smart Contract - Security Aspects in Hyper ledger Fabric.

Case Study: APPLICATIONS OF BLOCKCHAIN: Blockchain in Banking and Finance - Blockchain in Healthcare.

Text Book

- Chandramouli Subramaniam, Asha, A. George, Abhilash, K. A, Meera Karthikeyan. (2020). *Blockchain Technology* . University Press.

Reference Books

- Daniel Drescher, (2017). *Blockchain Basics: A Non-Technical Introduction.* Apress.
- Debajani Mohanty. (2018*). Blockchain From Concept to Execution BPB*.

E--Resourses

- https://www.pdfdrive.com/blockchain-books.html
- https://www.blockchain.com/

SOFTWARE TESTING

PCSM217

Semester : II	**Credit: 4**
Category : Major Core XI	**Hour/Week: 4**
Class &Major: I M.Sc Computer Science	**Total Hour: 52**

Objectives:

To enable the students

- Acquire Knowledge for Software Testing.
- Analyse Quality Assurance and Control.
- Evaluate the Quality of Project.

Learning Outcomes:

On Completion of the course, the students will be able to

- Create test Strategies and Plans, Design Test Cases, Prioritize and Execute them.
- Apply Modern Software Testing Processes in relation to Software Development.
- Manage Incidents and Risks within a Project.

UNIT- I INTRODUCTION 11 Hour

Introduction to Quality - Historical Perspective of Quality - Definitions of Quality - Core Components of Quality - Quality View - Customer, Suppliers and Processes - The Purpose of Testing. Basic Concepts of Software Testing: Introduction - Definition of Testing - Basic Principles of Testing - Work Bench - Test Policy - Test Strategy - Developing Test Strategy - Test Methodologies.

UNIT II TEST CASE DESIGN STRATEGIES 11 Hour

Test case Design Strategies – Using Black Box Approach to Test Case Design – Boundary Value Analysis – Equivalence Class Partitioning – State Based Testing – Cause-Effect Graphing – Compatibility Testing – User Documentation Testing – Domain Testing – Random Testing – Requirements Based Testing – Using White Box Approach to Test Design – Test Adequacy Criteria – Static Testing vs. Structural Testing – Code Functional Testing.

UNIT- III VERIFICATION AND VALIDATION 10 Hour

Software Verification and Validation: Introduction - Verification - Verification Work Bench - Methods of Verification - Types of Review on The Basis of Stage/Phase - Coverage in Verification - Concerns of Verification – Validation - Work Bench – Levels - Acceptance Testing - Software Development Verification and Validation Activities. V-Test Model - Analyzing and Reporting Test Results.

UNIT- IV TESTING TECHNIQUES AND TOOLS 10 Hour

Testing Techniques and Tools: Levels of Testing - Acceptance Testing: Introduction - Acceptance Criteria - Importance of Acceptance Criteria - Alpha Testing - Beta Testing - Gamma Testing - Acceptance Testing During Each Phase of Software Development - Software Development Methodologies - Developing Acceptance Plan.

UNIT-V TEST AUTOMATION 10 Hour

Software test automation – skills needed for automation – scope of automation – design and Architecture for Automation – Requirements for a Test Tool – Challenges in Automation – Test Metrics and Measurements – Project, Progress and Productivity Metrics.

Text Book

- Shen,J.J.(2019). *Software Testing: Techniques. Principles, and Practices*.
- Krishna Rungta. (2019). *Software Testing Learn Testing in 1 Day*. Kindle Edition.

Reference Book

- Dr. Anand Nayyar. (2019). In*stant Approach to Software Testing: Principles, Applications, Techniques*, *and Practices*.

BIG DATA ANALYTICS – PRACTICAL

PCSR208

Semester	**: II**	**Credits:**	**3**
Category	**: Core XII**	**Hour/Week**	**: 4**
Class & Major	**: I M.Sc Computer Science**	**Total/Hour**	**: 52**

Objectives

To enable the students

- Understand Analytical Concepts using PIG
- Gain Knowledge on Higher Level of Abstraction.
- Develop Programming Skills using HIVE Commands.

Learning Outcomes:

On Completion of the course, the students will be able to

- Perform Data Gathering of Large Data from a Range of Data Sources.
- Critically Analyse Existing Big Data Datasets and Implementations, Taking Practicality, and Usefulness Metrics into Consideration.
- Understand and Demonstrate the Role of Statistics in the Analysis of Large of Datasets.

LIST OF PROGRAMS

1. I/O Operations
2. Split and Union
3. Filters
4. Joins
5. Flattening
6. Sample and Parallel
7. Advanced Relational Operations
8. Perform Basic DDL - DML Operations using Hive Commands.
9. Perform Group by Operations using Hive Command.
10. Perform Order by vs. Sort by Operations using Hive Commands.
11. Demonstrate Join Operations using Hive Commands.

    (i) Left Outer Join (ii) Right Outer Join (iii) Full Outer Join
12. Calling out external program to perform Map and Reduce operations.

MACHINE LEARNING USING GOOGLE COLAB – PRACTICAL

PCSR209

Semester	**: II**	**Credit : 3**
Category	**: Major Core XIII**	**Hour/Week: 4**
Class & Major	**: I M.Sc Computer Science**	**Total Hour:52**

Objectives:

To enable the students

- Understand the Concepts and Techniques of Machine Learning.
- Acquire knowledge in recent Machine Learning Software.
- Implementation of Problem Solving.

Learning Outcomes:

On Completion of the course, the students will be able to

- Apply the Fundamental Concepts in Machine Learning.
- Evaluate the Scikit-Learn API.
- Develop Algorithms for Different Types of Dataset.

LIST OF PROGRAMS

1. Demonstrate the Working of the Decision Tree Based ID3 algorithm. Use an Appropriate Data set for Building the Decision tree and Apply this Knowledge to Classify a New Sample.
2. Apply k-Means Algorithm to Cluster a set of Data stored in a .CSV File.
3. House price Prediction using Linear Regression.
4. Diabetics Prediction using Logistic Regression.
5. Customer Churn Prediction using Decision Tree & Random Forest.
6. Predict Titanic Survivors using Artificial Neural Network Classification.
7. Implement k-Nearest Neighbors Algorithm from Scratch using IRIS Dataset
8. Implement the Naïve Bayesian Classifier for a Sample Training Data Set Stored as a .CSV file.
9. Covid-19 Data Analysis, Visualization Forecasting &Predication using ML
10. Implement Principle Component Analysis for Dimensionality Reduction.

NON-MAJOR ELECTIVE

MOBILE COMPUTING LAB

PCSE206

Semester	**: II**	**Credits**	**: 4**
Category	**: NON MAJOR ELECTIVE**	**Hour/Week**	**: 5**
Class &Major	**: I PG**	**Total Hour**	**: 65**

Objectives:
To enable the students

- Understand the Concepts Mobile Technologies
- Develop and Deploy Effective Mobile Applications.
- Impart Practical Training in Mobile Application Development.

Learning Outcomes:
On Completion of the course, the students will be able to

- Understand the Infrastructures and Technologies of Mobile Computing Technologies.
- Impart Knowledge on Principles and Theories of Mobile Computing Technologies.
- Analyse the Future of Mobile Computing Technologies and Applications.

LIST OF PROGRAMS (Any 10):

1. Create an Application which Deals with the Android Content Providers.
2. Create an Application using Android Layouts, Views and Events.
3. Create an Application which uses Files, Preferences and Notifications.
4. Create an Application to Create, Modify and Query an SQLite Database.
5. Create an Application for Querying Web services and Parsing response.
6. Create an Application which uses the concept of Services and Background Threats.
7. Creating Android Audio Video Application.
8. Create an Application which uses Map Activity and points the locations onto the Map Locations.
9. Create an Application with One-Time, Repeating Alarms, and Long-Running Background Task as Service.
10. Create an Application for Simple Mobile Game.
11. Develop an Application that uses GUI Components, Font and Colours.
12. Develop an Application that uses Layout Managers and Event Listeners.
13. Develop a Native Calculator Application.
14. Write an Application that Draws Basic Graphical Primitives on the Screen.
15. Develop an Application that makes use of Database.

16. Implement an Application that implements Multi Threading.
17. Develop a native Application that Uses GPS Location Information.
18. Implement an Application that Writes Data to the SD Card.
19. Implement an Application that creates an alert upon Receiving a Message.
20. Write a mobile Application that creates Alarm Clock.

WEB BASED APPLICATION DEVELOPMENT -LAB

PCSE207

Semester	**: II**	**Credit : 4**
Category	**: NME**	**Hour/Week :5P**
Class &Major	**: I PG**	**Total Hour :65**

Objectives:
To enable the students

- Understand the Concepts and Techniques of Web Development.
- Acquire Knowledge of RecentWeb Pages.
- Design and Implement JavaScript for Developing Web Site.

Learning Outcomes:
On Completion of the course, the students will be able to

- Understand the Fundamental Concepts of XHTML.
- Develop the Web Pages using Cascading Style Sheets.
- Publisha Web site using Java Scripts**.**

LIST OF PROGRAMS

1. XHTML and Web Pages
   a. Working with Text.
   b. Working with Lists, Tables and Frames.
   c. Working with Hyperlinks.
   d. Working with Images and Multimedia.
   e. Working with Forms and controls.
2. Cascading Style Sheets
   a. Styles (Background, Text Format, Controlling Fonts)
   b. Color Scheme
   c. Working with Block Elements and Objects
   d. Working with Lists and Tables
   e. Creating page Layout and Site Designs
3. Introduction to Web Publishing or Hosting
   a. Creating the Web Site
   b. Saving the Site
   c. Working on the Web site
   d. Creating Web site Structure
   e. Creating Titles for Web Pages

    f. Themes-Publishing web sites
4. Introduction to Java scripts
    a. Processing HTML Forms
    b. Conditions and Math Objects
    c. Loops and Arrays
    d. Return Functions and More Arrays
    e. Time Object and Image Roller
5. Introduction to Javascripts
    f. Processing HTML Forms
    g. Conditions and Math Objects
    h. Loops and Arrays
    i. Return Functions and More Arrays
    j. Time Object and Image Roller

III & IV EVALUATION COMPONENTS OF CIA

Semester	Category	Course Code	Course Title	Component III	Component IV
I	Core I	PCSM113	Principles of Concurrent Programming	Assignment	Seminar
	Core II	PCSM116/ PCSM404	Digital Image Processing	Assignment	Seminar
		PCSM117	TCP / IP Networks	Working Model	Seminar
	Core IV	PCSM118	Compiler Design	Problem Solving	Seminar
	Core V	PCSM119	Mobile Computing	Assignment	Seminar
	Core VI	PCSR107	Digital Image Processing – Practical	DPA	DPA
	Core VII	PCSR108	TCP/IP Networks – Practical	DPA	DPA
II	Core VIII	PCSM214	Big Data Analytics	Assignment	Seminar
	Core IX	PCSM215	Machine Learning	Assignment	Seminar
	Core X	PCSM216	Block Chain Technology	Assignment	Seminar
	Core XI	PCSM217	Software Testing	Working Model	Seminar
	Core XII	PCSR208	Big Data Analytics – Practical	DPA	DPA
	Core XIII	PCSR209	Machine Learning using Google CoLab – Practical	DPA	DPA

NON-MAJOR ELECTIVE

Semester	Category	Course Code	Course Title	Component III	Component IV
II	Non- Major Electives	PCSE206	Mobile Computing Practical	DPA	Viva voce
		PCSE207	Web Based Application Development Lab	DPA	Viva voce