DEPARTMENT OF COMPUTER APPLICATIONS

PREAMBLE

UG : Programme Profile, list of Courses Offered to the other Departments and the Syllabi of Courses in the I & II Semesters along with Evaluation Components III & IV (With Effect from 2021-2024 Batch Onwards)

PROGRAMME PROFILE BCA

(LEARNING OUTCOMES-BASED CURRICULUM FRAMEWORK)

PSO1 : Understanding of the Key Concepts and Principles of Programming Languages.

PSO2 : Capacity to Analyze a Problem, Identify the Computing Requirements and using Procedures find a Solution.

PSO3 : Development of Practical Skills to Solve Problems and Provide Solutions using Current Trends in the Discipline of Computer Applications.

PSO4 : Ability to apply the Algorithmic Principles, Mathematical Foundations and Computer Science Theory for Designing Computer-Based Systems.

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	UCEL101/ UCEL102	Communicative English-/ English/AECC-I (2 Levels)	UENL107/ UENL108	5	3/4
	III	Major Core (DSC) - I	UCAM110	Principles of Information Technology	-	5	4
	III	Major Core (DSC) - II	UCAM111/ UCSM109	Programming Methodology	-	4	4
	III	Major Core (DSC) - III	UCAR106/ UCSR110	Programming Methodology - Practical	-	3	2
	III	Allied (GE) - I	UMAA110	Mathematical Methods I	-	6	4
	III	Professional English	UPEM101	Professional English I	-	6	4
	IV	Value Education (SEC)				2	1
Total						**30**	**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	UCEL201/ UCEL202	Communicative English-/ English/AECC-I (2 Levels)	UENL207/ UENL208	5	3/4
	III	Major Core (DSC) - IV	UCAM206/ UCSM207	Data Structures	UCAM205	4	4
	III	Major Core (DSC) - V	UCAM207/ UCSM208	Python Programming	UCAM407	4	4
		Major Core (DSC) - VI	UCAR205/ UCSR207	Data Structures using Python - Practical	-	3	2
	III	Allied (GE) - II	UMAA216	Mathematical Methods-II	-	6	4
	III	Professional English	UPEM201	Professional English II	-	6	4
	IV	NME(Skill Enhancement Course)				3	2
	IV	Soft skill				2	1
	V	Extension Programme / Physical Education/NCC				-	1/2
Total						**30**	**28/31**
III	III	Major Core (DSC) - VII	UCAM310/ UCSM305	Java Programming	UCAM307	5	4
	III	Major Core (DSC) - VIII	UCAM312	Software Engineering	UCAM509	5	4
	III	Major Core (DSC) - IX	UCAM311	Data Communication Networks	UCAM309/ UCAM405	5	4
	III	Major Core (DSC) - X	UCAR304/ UCSR308	Java Programming - Practical	UCAR303	4	2
	III	Allied (GE) - III	UCOA303	Financial Accounting	-	6	4
	IV	Online course		NPTEL/Spoken Tutorial/Swayam		3	1/2
	IV	Value Education				2	1
Total						**30**	**20/21**
IV	III	Major Core (DSC) - XI	UCAM404	Database Management System	-	4	4
	III	Major Core (DSC) - XII	UCAM408	Operating System	UCAM507	5	4
	III	Major Core (DSC) - XIII	UCAM403	Object Oriented Analysis and Design	-	4	4
	III	Major Core (DSC) - XIV	UCAR405	Database Modeling - Practical	UCAR402	3	2
	III	Major Core (DSC) - XV	UCAR406	Operating System- Practical	-	3	2

Semester	Part	Category	Course Code	Course Title	Previous Course Code	Contact Hrs/ Week	Credit Min/Max
IV	III	Allied (GE) - V	UCOA403	Accounting Package	-	3	2
	III	Allied (GE) - VI	UCOR403	Accounting Package - LAB	-	3	2
	IV	NME(Skill Enhancement Course)				3	2
	IV	Soft skill				2	1
	V	Extension Programme/ Physical Education				-	1/2
Total						**30**	**24/25**
V	III	Major Core (DSC) - XVI	UCAM510	Cloud Computing	UCAO604	4	4
	III	Major Core (DSC) - XVII	UCAM511	R Programming	-	4	4
	III	Major Core (DSC) - XVIII	UCAM508	Open Source Technology	UCAM501	4	4
	III	Major Core (DSC) - XIX	UCAR506	Open Source Technology - Practical	UCAR504	3	2
	III	Major Core (DSC) - XX	UCAR507	R Programming - Practical	-	3	2
	III	MAJOR ELECTIVE (Discipline Specific Elective) - XXI	UCAO501/ UCAO502/ UCAO503	Computer Ethics/ Artificial Intelligence / Software Testing	-	5	4
		Major Core (DSC) - XXII	UCAP501	Project	UCAP601	5	5
	IV	Value Education				2	1
Total						**30**	**26**
VI	III	Major Core (DSC) - XXIII	UCAM609	Data Mining	UCAM606	5	4
	III	Major Core (DSC) - XXIV	UCAM612	Computer Graphics and Image Processing	UCAM610	5	5
	III	Major Core (DSC) - XXV	UCAM613	Internet of Things	UCAM611	5	4
	III	Major Core (DSC) - XXVI	UCAR603	Data Mining - Practical	UCAR602	4	3
	III	Major Core (DSC) - XXVII	UCAR604	Computer Graphics and Image Processing - Practical	-	4	3
	III	MAJOR ELECTIVE (Discipline Specific Elective) - XXVIII	UCAO607/ UCAO608/ UCAO609	Data Analytics/ Mobile Computing / Network Security	-	5	4
	III	Viva-Voce	UCAM601	Comprehensive Viva Voce	-	-	1
	IV	Soft Skill			-	2	1

Semester	Part	Category	Course Code	Course Title	Previous Course Code	Contact Hrs/ Week	Credit Min/Max
VI	V	Extension Programme/ Physical Education/NCC			-	-	0/2
					Total	**30**	**25/26**
					Grand Total	**180**	**148/156**

EXTRA CREDIT EARNING PROVISION

Semester	Part	Category	Course Code	Course Title	Contact/ Week	Credit	
						Min	Max
II	III	Summer Internship	UCAI201	Summer Internship	-	-	1
IV	III	Summer Internship	UCAI401	Summer Internship	-	-	1
V	III	Self Study	UCAS503	IOT Projects	2	-	2
V	III	Self Study	UCSS502/ UCAS502	Android Applications	2	-	2
VI	III	Self Study	UCSS601/ UCAS601	Angular JS	2	-	2
VI	III	Self Study	UCSS602/ UCAS602	Green Computing	2	-	2

NON-MAJOR ELECTIVES-UG

Semester	Part	Category	Course Code	Course Title	Contact/ Week	Credit
II	IV	NME	UCAE207	Data Science using R	3P	2
			UCAE208	Cyber Forensics	3T	2
			UCAE209	PyMOL	3P	2
			UCAE210	Qlik View	3P	2
			UCAE211	Internet Lab	3P	2
			UCAE212	Data Analytics Tools	3P	2
IV	IV	NME	UCAE401	Multimedia Programming	3P	2
			UCAE402	MATLAB Programming	3P	2
			UCAE403	Mobile App Development	3P	2

Inclusion of Experiential Learning

A. Experiential Learning (Mandatory)

Course Mapping				Collaborating Agency - MSME		
Semester	**Course Code**	**Course Title**	**Assessment**	**Course Title**	**Hour/Days/ Month**	**Mode of Evaluation**
II	UCAM310	Java Programming	Component III	Java Programming	4 Days	Reflection
II	UCAM407	Python Programming	Component III	Python Programming Training	4 Days	Reflection
III	UCAM505	Web Programming	Component III	Web designing Certification	4 Days	Reflection
III	UCAM610	Computer Graphics	Component III	Computer Graphics Certification	4 Days	Reflection

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

Semester	Category	Course Code	Course Title	Collaborating Agency	Hour/ Days/Month	Mode of Evaluation	Credits (Min/Max)
V	Core	UCAT501	Excel Analytics with R-Language	**MSME**	4 Days	Reflection	1

PRINCIPELS OF INFORMATION TECHNOLOGY

UCAM110/UCSM108

Semester : I **Credit : 4**
Category : Major Core (DSC) - I **Hour/Week: 5**
Class &Major : I BCA **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

UCAM111/UCSM109

Semester	**: I**	**Credit : 4**
Category	**: Major Core (DSC) - II**	**Hour/Week : 4**
Class &Major	**: I BCA**	**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

UCAR106/UCSM110

Semester : I **Credit : 2**
Category : Major Core (DSC) - III **Hour/Week: 3**
Class &Major : I BCA **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

UCAM206/UCSM207

Semester : II **Credit: 4**
Category : Major Core (DSC) - IV **Hour/Week: 4**
Class &Major : I BCA **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 Basant Agarwal. (2018). *Hands-On Data Structures and Algorithms with Python.* (2ndEd.).
- Horowitz, E. Sahni, S. and Susan Anderson-Freed. *Fundamentals of Data Structure*. (2nd Ed.) Universities Press.

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

UCAM207/UCSM208

Semester : II **Credit: 4**

Category : Major Core (DSC) - V **Hour/Week: 4**

Class &Major : I BCA **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 BCA**	**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. Quick Sort
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.

CYBER FORENSICS

UCAE208

Semester	**: II**	**Credit : 2**
Category	**: NME**	**Hour/Week : 3T**
Class & Major	**: I UG**	**Total Hour : 52**

Objectives:

To enable the Students

- Demonstrate Data Recovery from Hardware.
- Understand various Software Threats.
- Learn the Working of Surveillance Tools.

Learning Outcomes:

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

- Define and Discuss the Concepts of Computer Forensics.
- Explain the Career of a Computer Forensics Professional.
- Explain and Apply the Concepts of Computer Investigations.

UNIT- I TYPES OF COMPUTER FORENSICS 10 Hour

Computer Forensics Fundamentals – Types of Computer Forensics Technology – Types of Vendor and Computer Forensics Services.

UNIT- II DATA RECOVERY 12 Hour

Data Recovery – Evidence Collection and Data Seizure – Duplication and Preservation of Digital Evidence – Computer Image Verification and Authentication.

UNIT - III ELECTRONIC EVIDENCE 10 Hour

Discover of Electronic Evidence – Identification of Data – Reconstructing Past Events – Networks.

UNIT- IV THREATS 10 Hour

Fighting against Macro Threats – Information Warfare Arsenal – Tactics of the Military – Tactics of Terrorist and Rogues – Tactics of Private Companies.

UNIT - V SURVEILLANCE 10 Hour

The Future – Arsenal – Surveillance Tools – Victims and Refugees – Advanced Computer Forensics

Text Books

- Majid, Yar. (2013). *Cybercrime and Society.* Sage Publications.
- Chad, Steel. (2006). *Windows Forensics.* Wiley India.

Reference Book

- John R. Vacca(2005). *Computer Forensics.* Firewall Media.

PyMOL

UCAE209

Semester	**: II**	**Credit: 2**
Category	**: NME**	**Hour/Week: 3P**
Class & Major	**: I UG**	**Total Hour : 52**

Objectives:

To enable the Students

- Understand the Installation Steps of Pymol.
- Implement Simple Pymol Commands.
- Write Python Script to Interact Pymol.

Learning Outcomes:

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

- Understand the Physical Movements of Atoms.
- Implement in 3D Visualization of Molecule.
- Learn to Apply the View of useful Drug Design Tools and their Functions in Pymol.

Lab Exercises

1. Install PyMOL
2. Load Protein from Public Structure Repository (Pdb Format) using Load and Fetch Commands.
3. Change the Color of a Protein Chain
4. Change the View of a Protein (Rotate And Move) using Rota and Move Commands.
5. Save an Image
6. Selecting Parts of an Object
7. Write Simple Python Script to Rotate a Molecule.
8. Write a Simple Python Code Interacts with Pymol to Show Animated Molecule.

QLIK VIEW

UCAE210

Semester	**: II**	**Credit : 2**
Category	**: NME**	**Hour/Week : 3P**
Class & Major	**: I UG**	**Total Hour : 52**

Objectives:

To enable the students

- Learn Business Intelligence Solution.
- Understand the Data Visualization Technique using Qlik View.
- Apply Qlik View function for Data Projection.

Learning Outcomes:

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

- Appreciate How Key Fields are Identified and Form Associations between Tables.
- Understand How Visualizations are Created and Configured.
- Understand various ways to select Data within Qlik Sense.

Lab Exercises

1. Install Qlik View
2. Load Data from Different Sources in Qlik View

3. Apply Visualization Techniques
    a. Gauge Chart
    b. Waterfall Chart
    c. Cyclic and Drill Groups
4. Apply Data Transformation
    a. Loading Cross Table
    b. Loading Inline Table
    c. Loading Data from Already Stored Data in Qlik View (Resident Load)
    d. Joins, Concatenation of Tables
    e. Use of Mapping Tables
5. Apply Aggregate Function
6. Apply Access Restriction (Section Access)

INTERNET LAB

UCAE211

Semester	**: II**	**Credit**	**: 2**
Category	**: NME (SKILL ENHANCEMENT COURSE)**	**Hour/Week**	**:3P**
Class &Major :	**I UG**	**Total Hour**	**: 52**

Objectives

To enable the Students

- Analyze a Webpage and Identify its Elements and Attributes.
- Create Webpage's using HTML and Cascading Style Sheets.
- Build Dynamic Webpage using Javascript.

Learning Outcomes

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

- Implement Interactive Web Page(s) using HTML and Javascript.
- Design a Responsive Web Site using HTML and CSS
- Demonstrate Rich Internet Application.

List of Programs

1. Get to know your way around your Web Browser.
2. Understanding Web Browser and Searching the Web.
3. E-Mail Inner Working (Sending and Receiving).
4. Introduction to Chat Rooms, How to Connect and Chat.
5. Create a Simple HTML Page by using some of the Basic Tags.
6. To Create Time-Table using Table Tag.
7. Creation of Frames in Browser Window using HTML.
8. Working with Java Script and Creation of Dialogue Boxes using Alert.

9. Program to Perform four Arithmetic Operations viz. Addition, Subtraction, Multiplication and Division on Two Numbers using Java Script.
10. To Create a Web Site of our College.

DATA ANALYTICS TOOLS

UCAE212

Semester	**: II**	**Credit : 2**
Category	**: NME (SKILL ENHANCEMENT COURSE)**	**Hour/Week : 3P**
Class & Major:	**I UG**	**Total Hour : 52**

Objectives

To enable the Students

- Perform Basic Operations and Formatting and use Different Formulae and Functions.
- Summarize and Visualize Result of Data Analysis.
- Apply the above Skills to Analyze Various Kinds of Data.

Learning Outcomes

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

- Understand How to Align Data in Cell Locations.
- Examine How to Enter Multiple Lines of Text in a Cell Location.
- Examine How to use the Feature of Data Analysis.

List of Programs

1. Creating Data in MS-Excel and do Addition, Subtraction, Multiplication and Division.
2. To Calculate Descriptive Statistics in Excel using the Data Analysis Tools.
3. To Generate Comparative Statistics in Excel using the Data Analysis Tools.
4. How to Run a Linear Regression Analysis in Excel using the Data Analysis Tools.
5. Exploring Data using Pivottable.
6. Creating a Power View Report.
7. Preparing Data for Consolidation.
8. Importing Data from Microsoft Access Database.
9. Importing Data from a Web Page.
10. Exploring Data with Creating Combination Charts.

E-Resources

- https://www.tutorialspoint.com/excel_data_analysis/excel_data_analysis_tutorial.pdf
- https://www.csusm.edu/qc/facultydocuments/biofolder/bio353.pdf

III & IV EVALUATION COMPONENTS OF CIA

Semester	Part	Category	Course Code	Course Title	Component III	Component IV
I	III	Major Core (DSC) - I	UCAM110	Principles of Information Technology	Assignment	Assignment
	III	Major Core (DSC) - II	UCAM111	Programming Methodology	Assignment	Problem Solving
	III	Major Core (DSC) - III	UCSR106	Programming Methodology - Practical	DPA	Viva-voce
II	III	Major Core (DSC) - IV	UCAM206	Data Structures	Assignment	Problem Solving
	III	Major Core (DSC) - V	UCAM207	Python Programming	Assignment	Problem Solving
	III	Major Core (DSC) - VI	UCAR205	Data Structures using Python - Practical	DPA	Viva-voce

NON-MAJOR ELECTIVES

Semester	Part	Category	Course Code	Course Title	Component III	Component IV
II	IV	Non Major Elective	UCAE207	Data Science using R	DPA	Viva-Voce
			UCAE208	Cyber Forensics	Assignment	Case Study
			UCAE209	PyMOL	DPA	Viva-Voce
			UCAE210	Qlik View	DPA	Viva-Voce
			UCSE211	Internet Lab	DPA	Viva-voce
			UCAE212	Data Analytics Tools	DPA	Viva-voce