Course Information
MCA (Scheme 2020-2022)
Semester - I

DEPARTMENT OF COMPUTER APPLICATIONS

PES University

(Established under Karnataka Act No.16)
100 Feet Ring Road, BSK III Stage, Bengaluru- 560085Tele: + 91 82 2672 2108 Fax: + 91 80 2672 0886
www.pes.edu

I Semester (2020 – 22 BATCH)

Sl. No.	Course Code	Course Title	Hours / week				Credits	Course Type
			L	T	P	S		
1	UE20MC401	Programming with Python	4	0	0	4	4	PC
2	UE20MC402	Discrete Mathematics for Computer Applications	4	0	0	4	4	FC
3	UE20MC403	Data Structures	4	0	0	4	4	FC
4	UE20MC404	Operating Systems	4	0	0	4	4	FC
5	UE20MC405	Web Technology	4	0	0	4	4	FC
6	UE20MC406	Programming with Python Lab	0	0	2	1	1	PC
7	UE20MC407	Data Structures Lab	0	0	2	1	1	FC
8	UE20MC408A	Special Topic	1	0	2	2	2	FC
		Total	21	**0**	**6**	**24**	**24**	

PROGRAMMING WITH PYTHON

Course Code: UE20MC401 **Hours per week : 04**

Course In-charge : Santosh S Katti **Total Hours: 56 (4C)**

Course Objectives:

The objective of the course is to

- UE20MC401_Cobj1: Learn how to solve a given problem
- UE20MC401_Cobj2: Learn various procedural programming aspects
- UE20MC401_Cobj3: Explore the basic syntax and semantics of using functions
- UE20MC401_Cobj4: Learn various object-oriented programming aspects
- UE20MC401_Cobj5: Learn how to represent and use data in different file formats

Course Outcomes:

At the end of the course, the student will be able to

- UE20MC401_CO1: Think solutions for solving real world problems
- UE20MC401_CO2: Solve problems using simple constructs
- UE20MC401_CO3: Construct modular programs
- UE20MC401_CO4: Think using different paradigms of programming.
- UE20MC401_CO5: Implement database and file-based applications.

Course Overview:

The course introduces the essence of computational problem solving followed by computer algorithms, the heart of computational problem solving. Knowledge of Python aims to provide the know-how of the language from the basics of writing and running Python scripts to more advanced features such as file operations, regular expressions, working with binary data, and using the extensive functionality of Python modules. The aim of this course is to explore the basic syntax and semantics of python to enable the students to develop solutions for real-world applications.

Pre-Requisite: Logic building skills

Class #	**Chapter Title/Reference Literature**	**Topics to be Covered**	**% of Portions Covered**	
			Unit-wise	**Cumulative**
1.	**Unit-I** **Introduction:** **Types, operators, expressions and control structures** **T1:** **Pg. 1 – 37** **Pg. 39 – 75**	Introduction- Computation Problem Solving - Limits of Computational Problem Solving Algorithms	29	29
2.		Computer Hardware - - - Operating System, Digital Computer - Limits of IC technology Computer Software - Syntax, semantics and program translation		
3.		Process of Computation Problem Solving - Analysis, Design, Implementation, Testing		
4.		Translation models Introduction to Python Programming Language Programming paradigms		
5.		First program in Python Program Structure Running a program		
6.		Output function, Variables, types and id		

7.		Input function		
8.		Problem solving using output functions & input functions		
9.		Operators and expressions		
10.		Operators and expressions cont.		
11.		Precedence and Associativity		
12.	**Unit-II** **Procedural Python** **T1: Pg 79 - 124**	Control structures – selection and looping		
13.		Control Structures Cont.		
14.		Problem Solving using selection and looping		
15.		Concept of Library		
16.		Problem solving using different libraries		
17.		Lists & Tuples	17	46
18.		Operations on Lists and Tuples		
19.		Problem solving using Lists & Tuples		
20.		Dictionary & Sets		
21.		Operations on Dictionary & Sets		
22.		Problem solving using Dictionary & Sets		
23.		Strings		
24.		Operations on Strings		
25.		Assignment		
26.	**Unit-III** **Python Functions**	Functions – Definition, call, value returning functions and Non-value returning functions	20	66
27.		Nesting of functions		
28. .		Positional and keyword parameters, Default parameters		

29.	**T1 : Pg 168 - 205**	variable number of arguments, key value pairs as arguments Recursion		
30.		Problem solving using functions		
31.		Recursion		
32.		Callbacks		
33.		Problem solving using Recursion and callback		
34.		Closures		
35.		Decorators		
36.		Problem solving using closures and decorators		
37.				
38.	**Unit-IV** **Object Oriented Programming** **T1:** **Pg 206 – 245** **Pg 247 – 285** **Pg 383 - 480**	Functional programming - map, filter, reduce, max, min	15	81
39.		Lambda function		
40.		List comprehensions		
41.		Modules - Import mechanisms, Usage of __doc__ , __name__ , __call__		
42.		Generators and iterators		
43.		Problem solving using generators and iterators		
44.		Object oriented programming features		
45.		Python Classes and objects		
46.	**Unit-V** **File Processing** **T1:** **Pg 289 – 330** **Pg 489 - 510**	Exception & Types	19	100
47.		Exception handling constructs - try, except, else, finally and raise		
48.		Exceptions Cont.		
49.		Exception propagation and Problem solving		
50.		File processing- reading and writing files		

51.		Problem solving on file processing		
52.		JSON files		
53.		DB Handling		
54.		Assignment		
55.		Revision		
56.		Revision		

Type	Code	Title & Author	Publication Information		
			Edition	Publisher	Year
Text Book	T1	Charles Dierbach: Introduction to Computer Science Using Python: A Computational Problem-Solving Focus	Wiley India Edition	John Wiley	2015
Link	L1	www.python.org	-	-	-

DISCRETE MATHEMATICS FOR COMPUTER APPLICATIONS

Course Code	**:**	**UE20MC402**	**Hours / Week**	**:**	**04**
Course Incharge	**:**	**Ms. Prema Latha H M**	**Total Hours**	**:**	**56(4C)**

Course Objectives:
The objective of the course is to

- UE20MC402_Cobj1: Understand the basic principles of sets and set operations
- UE20MC402_Cobj2: Understand counting algorithms to solve problems
- UE20MC402_Cobj3: Understand the usage of logical notations
- UE20MC402_Cobj4: Understand the concepts of relations and functions.
- UE20MC402_Cobj5: Understand the concepts of graph theory

Course Outcomes:
At the end of the course, the student will be able to

- UE20MC402_CO1: Apply set theory in different algorithms, particularly those found in the area of computer science
- UE20MC402_CO2: Apply counting principles to determine permutations and combinations
- UE20MC402_CO3: Formulate mathematical arguments using logical notations
- UE20MC402_CO4: Relate the applications of relations, functions in computer science
- UE20MC402_CO5: Relate the applications of graph theory in computer science

Course Overview:

Discrete Mathematics course serves as a gateway to provide mathematical background and sufficient experience on various topics like counting techniques, Set Theory, logic and proofs, Relations, Functions and graphs. It helps computer engineers to solve problems that occur during the development of programming languages. This course is designed to introduce students to the discrete mathematical techniques, algorithms and reasoning processes involved during the development of real-time applications.

Class	Unit	Topics to be covered	% of portions to be covered	
			Unit-wise	Cumulative
1	**Unit-I** **Set Theory and Properties of Integers** T1: Pages 121-141, 223-260 , 273-292,440-452	Sets and Subsets	19.6	19.6
2		Set Operations and the Laws of Set Theory		
3		Counting and Venn Diagrams		
4		Principles of Inclusion and Exclusion		
5		Principles of Inclusion and Exclusion		
6		Mathematical Induction		
7		Mathematical Induction		
8		Primes and GCDs		
9		Integers and Algorithms		
10		Number System		
11		Matrices		
12	**Unit-II** **Counting Principles** T1:Pages 335-361	The Rules of Sum and Product	19.6	39.2
13		The Rules of Sum and Product		
14		Permutations		
15		Permutations		
16		Combinations		
17		Combinations		

18		Combinations with Repetition		
19		Combinations with Repetition		
20		Permutations & Combinations		
21		The Pigeonhole Principle		
22		The Pigeonhole Principle		
23	**Unit-III** **Fundamentals of Logic and Proofs** T1: Pages 1-89	Propositional Logic	19.6	58.8
24		Propositional Logic		
25		Propositional Equivalences		
26		Propositional Equivalences		
27		Predicates and Quantifiers		
28		Predicates and Quantifiers		
29		Rules of Inference		
30		Rules of Inference		
31		Introduction to Proofs		
32		Normal Forms		
33		Normal Forms		
34	**Unit-IV** **Relations and Functions** T1: Pages 459--520	Cartesian Products and Relations	21.6	80.4
35		Relations and their Properties		
36		Representing Relations, Closures of Relations		
37		Equivalence Relations		
38		Equivalence Relations		
39		Partial ordering		

40		Partial ordering		
41		Functions: Plain and One-to-One		
42		Onto Functions		
43		Special functions		
44		Function Composition		
45		Inverse Functions		
46	**Unit-V** **Graph Theory and its Applications** T1: Pages 527-584,623-648	Graphs --Introduction	19.6	100
47		Graph Models		
48		Special Types of Graphs		
49		Connectivity		
50		Euler and Hamilton Paths		
51		Euler and Hamilton Paths		
52		Euler and Hamilton Paths		
53		Introduction to trees		
54		Trees		
55		Applications of Trees		
56		Applications of Trees		

Book Type	Code	Title and Author	Publication Information		
			Edition	Publisher	Year
Text Book	T1	Discrete Mathematics & its Applications with Combinatorics and Graph Theory, Kenneth H Rosen	7th	McGraw-Hill	2012
Reference Book	R1	Discrete & Combinatorial Mathematics - An Applied Introduction, Ralph P Grimaldi, B.V.Ramana	5th	Pearson Education	2011
Reference Book	R2	Discrete Mathematics with Proof, Eric Gossett	2nd	John Wiley and Sons	2010
Reference Book	R3	Discrete Mathematical Structures, Kolman, Busby and Ross	4th	Prentice-Hall of India	2004

DATA STRUCTURES

Course Code	:	UE20MC403	**Hours / Week**	:	06
Course Incharge	:	Dr. Lekha A	**Total Hours**	:	56

Course Objectives:
The objective of the course is to

- UE20MC403_Cobj1: Understand primitive data structures in computer science
- UE20MC403_Cobj2: Introduce the concepts of recursion and linear data structures such as Stacks and Queues
- UE20MC403_Cobj3: Understand the applications of linear data structures in real world
- UE20MC403_Cobj4: Introduce the concepts of non-linear data structures such as Linked Lists, Trees
- UE20MC403_Cobj5: Understand the arrangement of data and its applications

Course Outcomes:
At the end of the course, the student will be able to

- UE20MC403_CO1: Design new data structures as per problem definition
- UE20MC403_CO2: Implement fundamental data structures viz., Lists, Stacks, Queues, Linked Lists, Binary Trees.
- UE20MC403_CO3: Develop a greater understanding of the importance of data storage
- UE20MC403_CO4: Choose the appropriate data structure for a specified application.
- UE20MC403_CO5: Choose a particular sorting technique for a given problem

Overview

At a very abstract level, computer science can be defined as the study of data, its representation and its transformation by a digital computer. In this course, students will be studying about different kinds of data structures. Each data structure has a way of representation and operations that may be efficiently performed on the represented data. The ability to devise alternative data representation and the ability to analyze the algorithms which operate on this data will be taught in this course.

Class #	Unit	Topics to be covered	% of portions to be covered	
			Unit-wise	Cumulative
1	**Unit-I** **Introduction to Data Structures** T1: Pages 1-40	Introduction – Data Types	19.64	19.64
2		Abstract Data Types		
3		Sequences as Value Definition		
4		An ADT for varying length Character Strings,		
5		Arrays in C		
6		The Array as an ADT		
7		Using 1-Dimensional Arrays		
8		Implementing 1-Dimensional Array, Arrays as Parameters		
9		2-Dimensional Arrays, Multi- Dimensional Arrays		
10		Pointers in C		
11		Structures in C		
12	**Unit-II** **Stacks** T1: Pages 77-108, 117-150, 171-173	Definition and Examples - Primitive Operations, Representing Stacks in C	19.64	39.28
13		Implementing the Push and Pop Operations		
14		An Example - Infix, Prefix, Postfix expressions		
15		Converting an expression from infix to postfix		
16		Converting an expression from infix to prefix		
17		Converting an expression from prefix to postfix		
18		Evaluating a Postfix Expression		
19		Recursive Definition and processes		
20		Multiplication of Natural numbers		
21		The Fibonacci sequence		
22		Tower of Hanoi, Efficiency of Recursion		

23	**Unit-III** **Queues** T1: Pages 174 – 184	The Queue and its sequential representation	19.64	58.92
24		The Queue and its sequential representation		
25		The Queue as an Abstract Data Type		
26		The Queue as an Abstract Data Type		
27		C Implementation of Queues		
28		C Implementation of Queues		
29		The Priority Queue		
30		Array Implementation of a Priority Queue		
31		Array Implementation of a Priority Queue		
32		Circular Queue		
33		Circular Queue Using Dynamically Allocated Arrays		
34	**Unit-IV** **Lists** T1: Pages 186- 218, 228-238	Linked List- Inserting and Removing from a List	21.42	80.34
35		Linked implementation of stacks		
36		The getnode and freenode operations		
37		Linked implementation of queues		
38		The Linked List as Data Structure		
39		Examples of List operations,		
40		Array Implementation of lists		
41		Linked List using Dynamic variables, Other List Structures		
42		Doubly Linked List		
43		Primitive operations on Circular List		
44		The Stack as a circular list		
45		The Queue as a circular list		
46	**Unit-V** **Binary Trees, Sorting and Searching** T1: Pages 249 -209, 212-220 T2: Pages 611 – 630	Introduction, Binary Trees	19.64	100
47		Binary Tree Traversals		
48		Binary Search Trees		
49		Insertion into a BST		
50		Deletion from a BST		
51		Sorting, Tree Sorting		
52		Shell Sort, Address Calculation Sort		
53		Radix Sort, Indexed Sequential Searching		
54		Tree Searching		
55		Hashing		
56		Hashing		

Book Type	Code	Title and Author	Publication Information		
			Edition	Publisher	Year
Text Book	T1	Data Structures Using C, Aaron M. Tanenbaum, Yedidyah Langsam, Moshe J. Augenstein	2	Pearson Education	2011
Text Book	T2	Data Structures – A Pseudocode Approach with C, Richard F Gilberg, Behrouz A Forouzan	2	Cengage Learning	2009

OPERATING SYSTEMS

Course Code: UE20MC404 **Hours/Week: 4 Hrs**

Course Incharge: Dr. S Thenmozhi **Total Hours: 56 (4C)**

Course Objectives

The objective of the course is to

- UE20MC451_Cobj1: Have a general understanding of structure of modern computers
- UE20MC451_Cobj2: Understand the process concepts and CPU scheduling policies
- UE20MC451_Cobj3: Understand the concepts of process synchronization
- UE20MC451_Cobj4: Understand the memory management strategies
- UE20MC451_Cobj5: Understand the File System architecture and Disk scheduling

Course Outcomes

At the end of the course, the student will be able to

- UE20MC451_CO1: Describe, contrast and compare differing structures of operating systems
- UE20MC451_CO2: Analyse the theory and implementation of processes and CPU Scheduling
- UE20MC451_CO3: Implement inter process communication and avoiding deadlocks
- UE20MC451_CO4: Comprehend the usage of physical and virtual memory
- UE20MC451_CO5: Decide appropriate Disk scheduling algorithm

Course Overview

Operating system (OS) provides the crucial interface between a computer's hardware and the applications that run on it. This course will introduce the core concepts of operating systems, such as processes management, process scheduling, synchronization, memory management, File System and disk management. Students will learn strategies to design a modern operating system and bundle the services pertaining to specific architecture style.

Class #	Unit	Topics to be covered	% of portions to be covered	
			Unit-wise	Cumulative
1	**Unit-I Introduction to Operating Systems and its Structure T1: Chapter 1, Chapter 2 (pg: 49-75)**	Course Overview & Introduction	19.6	19.6
2		What Operating System Does? OS Services		
3		Computer System Organization		
4		Types of Operating Systems		
5		Types of Operating Systems		
6		Operating System Functions		
7		Operating System Operations System Calls		
8		Types of System Calls & System Programs		
9		Operating System Structures		
10		Hands -On: Virtual box, Booting, init services		
11		Basic commands		
12	**Unit - II Process Management and Process Scheduling T1: Chapter 3, Chapter 5 (Pg: 183-198)**	Process Concept	21.4	41.0
13		Process Scheduling		
14		Inter-process Communication		
15		Basic Concepts, Scheduling Criteria		
16		Scheduling Algorithms		
17		Scheduling Algorithms		
18		Scheduling Algorithms		
19		Multilevel Queue Scheduling		
20		Programming Scheduling Algorithms		
21		Process and Process Attributes		
22		Process metrics and process controls		
23		listing processes, starting processes		
24	**Unit – III Process Synchronization and**	Background, The Critical Section Problem	19.6	60.6
25		Petersons Solution		
26		Synchronization hardware		
27		Semaphores		
28		Classic Problems of Synchronization		

29	**deadlock** **Chapter 6 (Pg: 225-243)** **Chapter 7**	Classic Problems of Synchronization		
30		System Model, Deadlock Characterization, Methods for handling deadlocks		
31		Deadlock Prevention, Avoidance		
32		Deadlock Detection, Recovery from Deadlock		
33		Basic Bash Shell Scripting		
34		Basic Bash Shell Scripting		
35	**Unit - IV** **Memory Management** **T1: Chapter 8,** **Chapter 9 (Pg: 357-386)**	Swapping, Contiguous Memory Allocation	19.6	80.2
36		Contiguous Memory Allocation		
37		Paging		
38		Structure of Page table		
39		Segmentation		
40		Demand paging		
41		Page Replacement		
42		Page replacement		
43		Allocation of Frames, Thrashing		
44		Linux: Memory Management Commands		
45		Linux: Memory Management Commands		
46	**Unit - V** **Storage Management** **T1: Chapter 10 (Pg:421-443)** **Chapter 12 (Pg: 505-515)**	File Concept	19.8	100
47		Access Methods		
48		Access Methods		
49		Disk Structure		
50		Disk Scheduling.		
51		Disk Scheduling.		
52		Linux-Files and FS		
53		Linux File types		
54		File and Directory Permissions		
55		File System Commands		
56		Discussion and Clarification		

Type	Code	Title & Author	Publication Info		
			Edition	Publisher	Year
Text Book	T1	Operating System Concepts – Silberschatz, Galvin, Gagne	8th	Wiley India	2009
Reference Book	R1	Operating Systems- A Concept Based Approach- Dhamdhere D M	3rd	Tata McGraw-Hill	2012
Reference Book	R2	Operating Systems- Internals and Design Principles - William Stallings	6th	Pearson Education	2012
Website	R3	https://courses.edx.org/courses/course-v1:LinuxFoundationX+LFS101x+1T2020/course/	-	-	-

WEB TECHNOLOGY

Course Code: UE20MC405 **Hours/Week**: 04

Course Instructor: Mr. Tamal Dey **Total hours:** 56 (4C)

Course Objectives:

The objective of the course is to

- UE20MC455_Cobj1: To understand the basic concepts of the World Wide Web
- UE20MC455_Cobj2: To get an overview of HTML5 and CSS mark-up
- UE20MC455_Cobj3: To understand the JavaScript programming, Document Object Model and Dynamic Scripting
- UE20MC455_Cobj4: To understand the basics of PHP Programming
- UE20MC455_Cobj5: To understand the need of interactive and dynamic web applications using PHP

Course Outcomes:

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

- UE20MC405_CO1: Get an insight into basics of web technology
- UE20MC405_CO2: Design attractive web pages using style-sheets
- UE20MC405_CO3: Compose interactive web pages using JavaScript
- UE20MC405_CO4: Develop server-side programs using basic concepts of PHP
- UE20MC405_CO5: Design attractive and interactive web applications

Course Overview:

This course provides knowledge of styling html pages using advanced styling concept and server-side scripting language to communicate with front end elements. End of the course students will be able to acquire knowledge to develop real world web applications with effective user interface which works with sessions and storing and retrieving values in database.

Class no.	Chapter Title/ Reference Literature	Topics to be covered	% of portions to be covered	
			Unit-wise	Cumulative
1	**Unit-I: Fundamentals of Web** (T1: Chapter 1- Page# 1-32, Chapter 2 Page# 36-87) (T2: Chapter 2- Page# 41-97)	Introduction to Web	19.6	19.6
2		Basic Syntax		
3		Standard HTML Document Structure		
4		Basic Text Markup		
5		Basic Text Markup		
6		Images, Hyperlinks		
7		HTML Document Structure - Lists		
8		HTML Document Structure- Tables		
9		HTML Document Structure- Frames		
10		HTML Document Structure- Forms		
11		HTML Controls		
12	**Unit-II: Designing webpages using Stylesheet** (T1: Chapter 3- Page# 95-125) (T2: Chapter 2- Page# 98-104)	HTML5 - Semantic Elements	19.6	39.2
13		HTML5 - Semantic Elements		
14		HTML5 Media - Audio and Video files, iframe		
15		Introducing Cascading Style Sheets, Style Specification Formats		
16		Selector Forms		
17		Property Value Forms, Font Properties, List Properties		
18		Color, Alignment of Text		
19		The Box Model, Background Images		

20		The <span> and <div> Tags		
21		CSS3		
22		CSS3		
23	**Unit-III: Programming with JavaScript** (T1: Chapter 4- Page# 95-173, Chapter 5- Page# 189-222, Chapter 6 – Page# 230-257) (T2: Chapter 2- Page# 107-152)	Overview of JavaScript Basics	21.4	60.6
24		The JavaScript Execution Environment ,The Document Object Model		
25		Element Access in JavaScript		
26		Events and Event Handling		
27		Handling Events from Body Elements, Button Elements		
28		Handling Events from TextBox and Password Element		
29		The DOM2 Event Model, Dynamic Documents		
30		Positioning Elements, Moving Elements, Element Visibility		
31		Changing Colors and Fonts,		
32		Dynamic Content, Stacking Elements		
33		Locating the Mouse Cursor		
34		Reacting to a Mouse Click, Slow Movement of Elements		
35	**Unit-IV: Server-side Programming - PHP** (T1: Chapter 11- Page# 431-456) (T2: Chapter 3- Page# 156-160, Chapter 4- Page# 166-176, Chapter 5- Page# 182-193 Chapter 6- Page# 200-230)	Introducing to PHP	19.7	80.3
36		Working with Variables and Constants		
37		Controlling the Program Flow		
38		Working with Functions		
39		Working with Functions		
40		Working with Arrays		

41		Working with Arrays		
42		**Global Variables**		
43		**PHP Files Permissions**		
44		**PHP Files Operations**		
45		**PHP Include and Directories**		
46	**Unit-V: Advanced PHP** (T1: Chapter 11- Page# 460-472, Chapter 13- Page# 522-534, 539-547) (T2: Chapter 7- Page# 235-256, Chapter 8- Page# 262-267, 275,276)	**Form handling using GET and POST**	19.7	100
47		**Form Validation**		
48		Working with Database operations		
49		**Retrieving table values**		
50		**Storing and Deleting records**		
51		**Prepared statement**		
52		Limit data selection		
53		Exploring Cookies		
54		Session Management		
55		PHP Security		
56		PHP Security		

Book Type	Code	Title & Author	Publication info		
			Edition	Publisher	Year
Text Book	T1	"Programming the World Wide Web" by Robert.W.Sebesta,	4th	Pearson Education,	2011
Text Book	T2	“Web Technologies – BlackBook” by Kogent	--	Dreamtech Press	--
Reference Book	R1	HTML5 & CSS3 All-In-One for Dummies", Andy Harris, Wiley Brand	3rd		2013
Reference Websites	R2	"Web Technology – Theory and Practice" by Akshi Kumar	--	CRC Press	--

PYTHON PROGRAMMING LAB

Course Code: UE20MC406 **Hours per week : 02**

Course In-charge : Santosh S Katti **Total Hours: 26 (1C)**

Course Objectives:
The objective of the course is to

- UE20MC406_Cobj1: Learn how to implement data structures and functions available in Python to solve problems
- UE20MC406_Cobj2: Learn Lists, Dictionaries in Python
- UE20MC406_Cobj3: Learn how to handle strings in Python
- UE20MC406_Cobj4: Learn to implement modular programming
- UE20MC406_Cobj5: Learn how to handle files in Python

Course Outcomes:
At the end of the course, the student will be able to

- UE20MC406_CO1: Illustrate problem solving using Python Programming
- UE20MC406_CO2: Implement dynamic interactive real-world Python Applications using language idioms, data structures
- UE20MC406_CO3: Implement python applications with standard library, dictionaries
- UE20MC406_CO4: Implement reusable components
- UE20MC406_CO5: Implement python applications with files

Course Overview:

The aim of this course is to explore the basic syntax and semantics of python to enable the students to work with control structures, loops, different datatypes, functions and handling of files.
Pre-Requisite: Knowledge of syntax and semantics of Python language

1.	Getting started with Introduction (Basic Concepts of Programming)	2 Hours
2.	Program to demonstrate Input Output Functions, Operators and Expressions.	2 Hours
3.	Program to demonstrate the Usage of Libraries.	2 Hours
4.	Program to demonstrate Control Structures.	2 Hours
5.	Program to demonstrate Control Structures.	2 Hours
6.	Program to demonstrate Lists and Tuples.	2 Hours
7.	Program to demonstrate Sets and Dictionaries.	2 Hours
8.	Program to demonstrate String Related Operations.	2 Hours
9.	Program to demonstrate the Usage of Functions.	4 Hours
10.	Final Lab Test	2 Hours
11	Program to demonstrate Functional Programming.	2 Hours
12.	Program to demonstrate File Handling in Python.	2 Hours

DATA STRUCTURES LABORATORY

Course Code	:	UE20MC407	**Hours / Week**	:	02
Course Incharge	:	Dr. Lekha A	**Total Hours**	:	26

Course Objectives:
The objective of the course is to

- UE20MC407_Cobj1: Introduce the concepts of data structures and its significance in solving problems
- UE20MC407_Cobj2: Design and implement a data structure as per storage requirements
- UE20MC407_Cobj3: Implement different operations on each data structure
- UE20MC407_Cobj4: Write programs to understand the concept of recursion
- UE20MC407_Cobj5: Solve problems using data structures such as linear lists, stacks, queues, hash tables, binary trees, binary search trees

Course Outcomes:
At the end of the course, the student will be able to

- UE20MC407_CO1: Develop a greater understanding of the importance of data storage
- UE20MC407_CO2: Design and implement a new data structures as per need
- UE20MC407_CO3: Solve problems using recursion
- UE20MC407_CO4: Solve problems using data structures such as linear lists, stacks, queues, binary trees, binary search trees
- UE20MC407_CO5: Understand the usage of different data structures

Course Overview:

Each data structure has a way of representation and operations that may be efficiently performed on the represented data. The ability to work with different data representation in a particular language will be taught in this course.

1.	Getting started with Introduction (Basic Concepts of Programming)	2 Hours
2.	Write a program to compute address of matrix using Row-Major and Column-Major Addressing.	2 Hours
3.	Write a program to find the Saddle point of a matrix. ***Note:** Saddle point M(i, j) of a matrix M, is the smallest element in row 'i' and largest element in column 'j' .	2 Hours
4.	Write a program implement to Stack Operations i) Using Arrays ii) Using Structures	2 Hours
5.	Write a program to implement the conversion of an expression from prefix to postfix	2 Hours
6.	Write a program to implement recursion to find a) Factorial of a number N b) Generate Fibonacci Series upto N c) To solve Tower of Hanoi problem	2 Hours
7.	Write a program to implement the functions of circular queue	2 Hours

8.	Write a program to implement priority queue	2 Hours
9.	Write a program to realize Stacks and Queues using Linked List. Perform different operations on Singly Linked Lists (use self-referential structures)	4 Hours
10.	Final Lab Test	2 Hours
11	Write a program to implement insertion before a particular element and delete an element at specified position in doubly linked list. (Operations on DLL)	2 Hours
12.	Write a program to construct a BST and perform inorder, preorder and postorder traversal and deletion of nodes	2 Hours

SPECIAL TOPIC -WEB APPLICATION DEVELOPMENT

Course Code: UE20MC408A **Hours/Week**: 01 +02

Course Instructor: Mr. Tamal Dey **Total hours:** 42 (2C)

Course Objectives:

The objective of the course is to

- UE20MC408A_Cobj1: To get an overview of HTML and CSS styling for designing attractive webpages
- UE20MC408A_Cobj2: To use java script for designing interactive web interface
- UE20MC408A_Cobj3: To learn the programming constructs of PHP
- UE20MC408A_Cobj4: To develop simple PHP applications depicting the concepts of Session and Cookies
- UE20MC408A_Cobj5: Get to know how to interact with the database from PHP

Course Outcomes:

At the end of the course, the student will be able to

- UE20MC408A_CO1: Design attractive webpages comfortably using HTML and CSS
- UE20MC408A_CO2: Design interactive web applications
- UE20MC408A_CO3: Develop server-side scripts using PHP
- UE20MC408A_CO4: Apply session tracking and cookies while developing web applications
- UE20MC408A_CO5: Develop a full-fledged application with Database Interaction.

Course Overview:

Web Technology is one of the most important domain areas with respect to any application development. The course encompasses building an attractive user interface using HTML and CSS. Interactivity can also be added using the JavaScript programming which includes Event Handling and Dynamic Scripting. Possessing PHP knowledge in order to develop full-fledged web applications is vital to any enthusiastic learner. At the end of this course, the students will be able build rich UI with interactive features which can interact with the database.

Class no.	Chapter Title/ Reference Literature	Topics to be covered	% of portions to be covered	
			Unit-wise	Cumulative
1	Unit-I Designing Webpage	Tutorial:Basic Text Markup	19	19
2		Hands-on:Webpage Designing using basic tags		
3		Tutorial:Document Structuring using List		
4		Tutorial:Document Structuring using Tables		
5		Hands-on: Webpage Designing using List, and table		
6		Tutorial:Document Structuring using Frames and Forms		
7		Hands-on: Restructuring the webpage using frames		
8		Hands-on:Restructuring the webpage using interactive form		
9	Unit-II Adding Stylesheet	Project Identification:Team formation and project title allocation Tutorial:Semantic elements	19	38
10		Hands-on:Webpage designing with semantic elements		
11		Tutorial:adding audio and video using iframe		
12		Hands-on:Webpage designing with multimedia content		
13		Tutorial:Design webpage using stylesheet		

14		Hands-on:Working on stylesheet with borders and background		
15		Tutorial:Design webpage Box Model and Background Images		
16		Hands-on:Working on borders and background		
17	**Unit-III** **Client-side Scripting**	Project Discussion	21.5	59.5
18		Tutorial: DOM manipulation with validation		
19		Hands-on: Body element access in JavaScript		
20		Tutorial: Event Handling using mouse and keyboard		
21		Hands-on: User event handling using textbox elements		
22		Hands-on: User event handling using textbox elements		
23		Hands-on: Dynamic Documents using JavaScript		
24		Tutorial: Dynamic Content creation by Stacking		
25		Tutorial: Positioning and Moving Elements		
26	**Unit-IV** **Server-side Scripting**	Project Discussion Tutorial: PHP Variables and Constants	19	78.5
27		Hands-on: Controlling the program flow using PHP variables		
28		Hands-on: GLOBALS		
29		Tutorial: Functions and Arrays		
30		Hands-on: Working with Functions and Arrays		
31		Tutorial: Cookies and Sessions		
32		Hands-on : Cookies storage		

33		Hands-on : Sessions management		
34	**Unit-V** **Accessing** **Database**	Tutorial: Form data retrieval	21.5	100
35		Hands-on: Form handling using GET and POST		
36		Tutorial: Form validation		
37		Hands-on: Validating retrieved values		
38		Tutorial: Database operations		
39		Hands-on Database creation and access using CURD operation		
40		Hands-on: Database access using CURD operation		
41		Hands-on: Database access using CURD operation		
42		Project Discussion		