

Department of Computer Engineering

B. Tech. in Computer Engineering

Scheme - 2020

**Scheme and Curriculum of
B. Tech. Computer Engineering**

Scheme Summary

Semester	Credits	Hours	Marks	No. of Theory Heads	No. of Practical Heads	Total Heads
I	18	26	575	5	7	12
II	19	26	550	5	6	11
III	21	26	525	5	3	8
IV	24	30	600	6	5	11
V	22	29	600	6	5	11
VI	21	26	650	7	5	12
VII	20	25	525	5	2	7
VIII	15	27	475	1	1	2
Total	160	215	4500	40	34	74

Course Category Credits Summary

Semester	No. of Credits for Course Category						TOTAL
	Core (C)	Basic Science and Humanities (BS &H)	Skills (A)	Electives (EL)	Open Electives (OE)	PROJECT/ INTRNSHIP (P)	
I	9	6	3	0	0	0	18
II	6	11	2	0	0	0	19
III	14	4	3	0	0	0	21
IV	16	4	2	0	2	0	24
V	16	2	1	3	0	0	22
VI	9	5	2	3	2	0	21
VII	4	0	0	12	0	4	20
VIII	0	0	0	0	3	12	15
TOTAL	74	32	13	18	7	16	160

Scheme of B.Tech. In Computer Engineering

Course Code	Name of Course	Course Category	Teaching Scheme				Credits	Evaluation Scheme					
								Theory			Practical		Total Marks
			L	T	P	Total Hours		TAE	CAE	ESE	INT	EXT	
SEMESTER-I													
UBSL151	Matrices and Differential Calculus	BS1	2	1		3	3	10	15	50			75
UBSL101/ UBSP101	Engineering Physics	BS2	1	1	2	4	3	10	15	50	25		100
UCSL101/ UCSP101	Computer Programming	C1	2		4	6	4	10	15	50	50		125
UCSL102/ UCSP102	Foundations of Data Analytics	C2	1		2	3	2	10	15	50	25		100
UECL103/ UECP103	Introduction to Discrete Devices and Circuits	C3	2		2	4	3	10	15	50	25		100
UITP101	Problem Identification and Design Thinking	A1			2	2	1				25		25
UISP102	Introduction to Drones	A2			2	2	1				25		25
UISP101	Biomedical Engineering	A3			2	2	1				25		25
TOTAL			8	2	16	26	18	50	75	250	200	--	575

Scheme of B. Tech. in Computer Engineering

Course Code	Name of Course	Course Category	Teaching Scheme				Credits	Evaluation Scheme					
								Theory			Practical		Total Marks
								TAE	CAE	ESE	INT	EXT	
SEMESTER-II													
UBSL152	Integral Calculus and Differential Equations	BS3	2	1		3	3	10	15	50			75
UBSL153	Linear Algebra and Statistics	BS4	2	1		3	3	10	15	50			75
UBSL131/ UBSL131	Environmental Chemistry	BS5	1		2	3	2	10	15	50	25		100
UITP102	Programming for Problem Solving	C4			4	4	2				50		50
UECL104/ UECP104	Modeling of Digital Circuits	C5	3		2	5	4	10	15	50	25		100
UHUL101/ UHUP101	Communication Skills	H1	2		2	4	3	10	15	50	25		100
UECP105	Internet of Things	A4			2	2	1				25		25
UHUP102	Foreign Language	A5			2	2	1				25		25
TOTAL			10	2	14	26	19	50	75	250	175	--	550

Scheme of B. Tech. in Computer Engineering

Course Code	Name of Course	Course Category	Teaching Scheme				Credits	Evaluation Scheme					
								Theory			Practical		Total Marks
			L	T	P	Total Hours		TAE	CAE	ESE	INT	EXT	
SEMESTER-III													
UBSL255	Discrete Mathematics and Graph Theory	BS6	3	1		4	4	10	15	50			75
UCSL201/ UCSP201	Data Structures and Algorithms	C6	3		2	5	4	10	15	50	25	25	125
UCSL202	Computer Architecture and Organization	C7	3			3	3	10	15	50			75
UCSL203	Formal Languages and Automata	C8	3			3	3	10	15	50			75
UITL201/ UITP201	Object Oriented Programming	C9	3		2	5	4	10	15	50	25	25	125
UCSP204	Python for Data Science	A6-A7-A8			6	6	3				50		50
TOTAL			15	1	10	26	21	50	75	250	100	50	525

Scheme of B. Tech. in Computer Engineering

Course Code	Name of Course	Course Category	Teaching Scheme				Credits	Evaluation Scheme					
								Theory			Practical		Total Marks
			L	T	P	Total Hours		TAE	CAE	ESE	INT	EXT	
SEMESTER-IV													
UBSL256	Transforms and Numerical Methods	BS7	3	1		4	4	10	15	50			75
UITL202/ UITP202	Computer Networks	C10	3		2	5	4	10	15	50	25		100
UCSL205/ UCSP205	Design and Analysis of Algorithm	C11	3		2	5	4	10	15	50	25	25	125
UECL208/ UECP208	Applications of Microprocessors and Microcontrollers	C12	3		2	5	4	10	15	50	25		100
UCSL206/ UCSP206	Operating System	C13	3		2	5	4	10	15	50	25		100
UISL2XX	Open Elective - I	OE1	2			2	2	10	15	50			75
UITP203	Data Analysis using R	A9-A10			4	4	2				25		25
TOTAL			17	1	12	30	24	60	90	300	125	25	600

Skill Category Courses

Semester	Details	Name	Credits
I	A1	Problem Identification and Design Thinking	1
	A2	Introduction to Drones	1
	A3	Biomedical Engineering	1
II	A4	Internet of Things	1
	A5	Foreign Language	1
III	A6-A7-A8	Python for Data Science	3
IV	A9-A10	Data Analysis using R	2
V	A11	Aptitude	1
VI	A12	Employability Skills	1
	A13	Campus Recruitment Training	1

Course Title: Discrete Mathematics and Graph Theory										
Semester	III	Teaching Scheme				Evaluation Scheme				
						Theory			Practical	
Term	ODD	Th	Tu	Pr	Credits	TAE	CAE	ESE	INT	EXT
Course Category	BS	3	1	--	4	10	15	50	--	--
Course Code	UBSL255	4			Total	75			--	
						75				

Course Objectives	This course introduces size and kind of objects.
	It also skills to analyze objects meeting the criteria, finding "largest", "smallest", or "optimal" objects.
	It also introduces combinatorial structures and apply algebraic techniques to combinatorial problems
Course Outcomes	CO1: Know grouping of objects and operation, Relation, ordering of objects.
	CO2: Know Groups, their types and Applications.
	CO3: Know Rings, their types and Applications.
	CO4: Solve different kinds Graph, Trees.
	CO5: Know the basics of combinatorial structure
	CO6: Solve number theory applications.

Mapping of Course Outcomes with Program Outcomes and Program Specific Outcomes:

Course Outcomes	Program Outcomes and Program Specific Outcomes														
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1	3	3										3	2		
CO2	3	3										3	2		
CO3	3	3										3	2		
CO4	3	3										3	2		
CO5	3	3										3	2		
CO6	3	3										3	2		

Course Contents:

Unit	Contents	Hours
I	Set Theory Operations on sets, Laws of algebra of sets, Representation of sets on computer in terms of 0's & 1's. Partition & covering of a set, ordered pair, Product set, Relation–Different types of relations, Graph of relation, Matrix of relation, Transitive closure of relation, Properties of relations, Compatible relation. Functions, Partial ordering & partially ordered set, Hasse diagram of Poset, Totally ordered set, Peano axioms & Mathematical Induction.	7
II	Group Algebra or Algebraic systems like semigroup, monoid and examples. Homomorphism, Isomorphism of semigroup & monoid. Groups, properties of algebraic groups. Permutations groups, Subgroups, Cosets, Lagrange's theorem, properties of cyclic groups, generator of group, kernel of Homomorphism, quotient group, fundamental theorems & Homomorphism of groups, Residue classes & Fermats theorem.	8
III	Rings Rings, types of rings, Fields, subring, Integral domain. Simple properties of rings. Lattice as Poset & as algebraic system, Types of lattices, Hasse diagrams, Sublattice, direct product of Lattices, Lattice Homomorphism, complement of elements of lattices, Various lattices, composition tables, Lattice, Boolean algebra; Boolean Expressions, Equivalence of Boolean Expression by tables, Simplification of circuit & equivalent circuit by truth tables.	8
IV	Graph Theory Graphs and its types, Sub graph, Quotient graph, Euler path, complete path, in degree, out degree, reachability, cycle, matrix representation of graph. Transitive closure of graph, Adjacency matrix, Trees, Venn diagram, Representation of trees, binary trees, spanning trees, Prim's algorithm.	8
V	Combinatorics Definition of generating functions and examples, proof of simple combinatorial identities. Recursive relations: definitions & examples, explicitly formula for sequence, back tracking to find explicit formula of sequence, solving recurrence relations. Counting Theorem and application, multiplication principle of counting. Permutation & Combination with examples. The pigeon hole principle & extended pigeon hole principle and application of pigeon hole principle in solving simple problems.	8
VI	Number Theory Continued fractions, The study of continued fractions. Alpha has Infinite continued fraction if alpha is irrational. Alpha has periodic continued fractions if alpha is quadratic irrational. Application to approximation of irrationals by rationals. Hurwitz's theorem, Advanced topic on Combinatorial Theory.	8

Text Books	1.	Discrete Mathematical structure with application to computer science by Trembley & Manohar (Mc. Graw Hill)
	2.	Discrete Mathematical Structure by Kolmann , Busby & Ross (PHI)
	3.	Discrete mathematics by Lipschutz& Lipson , Schaum's outline, TMH
Reference Books	1.	Discrete Mathematics by Liu
	2.	Discrete Mathematics by John Truss (Addison Wesley, 2000)
	3.	Foundations of Discrete Mathematical by K. D. Joshi (Willey Eastern).
	4.	Set Theory by Lipschutz (Schaum Series, Asian Student Edition).
	5.	Modern Algebra by M. L. Khanna (Jai Prakash Nath& Company Meerut).
on line TL Material	1.	https://onlinecourses.nptel.ac.in/noc20_cs37/unit?unit=41&lesson=42

Course Contents:

Unit	Contents	Hours
I	Introduction –Common operations on data structures, Types of data structures, Data structures & Programming, Program Design, Complexities, Time Complexity, order of Growth, Asymptotic Notation. Sorting and Searching Introduction, Sorting, Insertion Sort, Selection Sort, Merging, Merge-Sort, Shell Sort, Radix Sort, Searching and Data Modification, Hashing	9
II	Arrays: Introduction, Linear Arrays, Arrays as ADT, Representation of Linear array in Memory, Traversing Linear Arrays, Inserting and deleting, Sorting; Bubble Sort, Searching; Linear Search, Binary Search, : Linked List: Introduction: Linked Lists, Representation of Linked Lists in Memory, Traversing a Linked List, Searching a Linked List, Memory Allocation; Garbage Collection, Insertion into a Linked List, Deletion from a Linked List, Header Linked List, Circularly Linked Lists, Two-Way Lists (or Doubly Linked Lists).	9
III	Stacks, Queue and Recursion- Introduction, Stacks ,Array Representation of Stacks ,Linked Representation of Stacks, Stack as ADT, Arithmetic Expression; Polish Notation, Application of Stacks, Recursion, Towers of Hanoi, Implementation of Recursive Procedures by Stacks, Queue, Linked Representation of Queues, Queues as ADT, Circular Queues, Deques, Priority Queues, Applications of Queues	9
IV	Trees and Binary Trees -Binary Trees • Representation, Operations: Insert, Delete, Traversal: Preorder, Inorder, Postorder, Traversal Algorithms Using Stacks, Header Nodes; Threads, Threaded Binary Trees, Binary Search Trees ,Searching and Inserting in Binary Search Trees, Deleting in a Binary Search Tree, Balanced Binary Trees, AVL Search Trees, Insertion in an AVL Search Tree, Deletion in an AVL Search Tree, m-way Search Trees ,Searching, Insertion and Deletion in an m-way Search tree, B-Trees ,Searching, Insertion and Deletion in a B-tree, B+-Trees Graph Algorithms	10
V	Graphs and their Applications -) Introduction, Graph Theory Terminology, Sequential Representation of Graphs, Adjacency Matrix; Path Matrix, Linked Representation of a Graph, Operations on Graphs, Traversing a Graph, Posets; Topological Sorting, Spanning Trees	8

Text Books	1.	AVAho, J Hopcroft, JD Ullman, Data Structures and Algorithms, Addison- Wesley, 1983.
	2.	THCormen, CF Leiserson, RL Rivest, C Stein, Introduction to Algorithms, 3rd Ed., MIT Press, 2009.
	3.	Sahni, S., "Data Structures, Algorithms, and Applications in C++", WCB/McGraw-Hill.
E--Books	1.	https://apps2.mdp.ac.id/perpustakaan/ebook/Karya%20Uum/Dsa.pdf
Reference Books	1.	Data Structures & Algorithms, 1e, Alfred V.Aho, Jeffery D. Ullman , Person.
	2.	MT Goodrich, R Tamassia, DM Mount, Data Structures and Algorithms in Java, 5th Ed., Wiley, 2010. (Equivalent book in C also exists.)

	3.	Wirth, N., "Algorithms and Data Structures", Prentice-Hall of India.
Online TL Material	1.	https://nptel.ac.in/courses/106/102/106102064/
	2.	http://cse01-iiith.vlabs.ac.in/
	3.	https://ds2-iiith.vlabs.ac.in/data-structures-2/

Data Structure and Algorithms lab	
Sr. No.	Name of Experiments / Mini Projects/ Case Studies
1	<p>Consider a student database of SY COMP class (at least 10 records). Database contains different fields of every student like Roll No, Name and SGPA.(array of objects of class)</p> <p>a) Design a roll call list, arrange list of students according to roll numbers in ascending order (Use Bubble Sort)</p> <p>b) Arrange list of students alphabetically. (Use Insertion sort)</p> <p>c) Arrange list of students to find out first ten toppers from a class. (Use Quick sort)</p> <p>d) Search students according to SGPA. If more than one student having same SGPA, then print list of all students having same SGPA.</p> <p>e) Search a particular student according to name using binary search without recursion.</p>
2	<p>Department of Computer Engineering has student's club named 'COMET'. Students of Second, third and final year of department can be granted membership on request. Similarly one may cancel the membership of club. First node is reserved for president of club and last node is reserved for secretary of club. Write program to maintain club member's information using singly linked list. Store student MIS registration no. and Name. Write functions to a) Add and delete the members as well as president or even secretary. b) Compute total number of members of club c) Display members d) Display list in reverse order using recursion e) Two linked lists exists for two divisions. Concatenate two lists</p>
3	Implement Stack using a linked list. Use this stack to perform evaluation of a postfix expression.
4	Queues are frequently used in computer programming, and a typical example is the creation of a job queue by an operating system. If the operating system does not use priorities, then the jobs are processed in the order they enter the system. Write a program for simulating job queue. Write functions to add job and delete job from queue.
5	A double-ended queue (deque) is a linear list in which additions and deletions may be made at either end. Obtain a data representation mapping a deque into a one-dimensional array. Write a program to simulate deque with functions to add and delete elements from either end of the deque
6	A book consists of chapters, chapters consist of sections and sections consist of subsections. Construct a tree and print the nodes. Find the time and space requirements of your method.
7	Implement binary tree using linked list and perform recursive traversals.
8	<p>Beginning with an empty binary search tree, Construct binary search tree by inserting the values in the order given. After constructing a binary tree</p> <p>i. Insert new node</p> <p>ii. Find number of nodes in longest path</p>

	iii. Minimum data value found in the tree iv. Change a tree so that the roles of the left and right pointers are swapped at every node v. Search a value
9	Implement graph using adjacency list or matrix and perform DFS or BFS.
10	You have a business with several offices; you want to lease phone lines to connect them up with each other; and the phone company charges different amounts of money to connect different pairs of cities. You want a set of lines that connects all your offices with a minimum total cost. Solve the problem by suggesting appropriate data structures.
Open Ended Experiments / New Experiments	
1	A classic problem that can be solved by backtracking is called the Eight Queens problem, which comes from the game of chess. The chess board consists of 64 square arranged in an 8 by 8 grid. The board normally alternates between black and white square, but this is not relevant for the present problem. The queen can move as far as she wants in any direction, as long as she follows a straight line, Vertically, horizontally, or diagonally. Write C++ program for generating all possible configurations for 4-queen's problem.
2	A Dictionary stores keywords & its meanings. Provide facility for adding new keywords, deleting keywords, updating values of any entry. Provide facility to display whole data sorted in ascending/ Descending order. Also find how many maximum comparisons may require for finding any keyword. Use Binary tree and find the complexity for finding a keyword.

Course Title: UCSSL202: Computer Architecture and Organization										
Semester	III	Course Title: Computer Architecture and Organization								
Term	Odd	Teaching Scheme				Evaluation Scheme				
						Theory			Practical	
Course Category	C7	Th	Tu	Pr	Credits	TAE	CAE	ESE	Int	Ext
Course Code	UCSSL202	3	--	--	3	10	15	50	--	--
		3			Total	75				

Course Objectives	To understand the design principles of digital computing systems
	To provide essential understanding of different subsystems of modern computer system and design aspects these subsystems ³
	To provide overview on performance enhancement methods in instruction execution
Course Outcomes	CO1: To describe the basic components and design of a computer system
	CO2: To examine the issues involved in the instruction execution and various stages of instruction life stage
	CO3: To apply the concept of various memories and interfacing technologies
	CO4: To analyze the different parallel processing technique
	CO5: To design different high performance computing architecture

Mapping of Course Outcomes with Program Outcomes and Program Specific Outcomes:

[illegible]

Course Contents:

Unit	Contents	Hours
I	Basic functional blocks of a computer: CPU, memory, input-output subsystems, control unit. Instruction set architecture of a CPU –registers, instruction execution cycle, RTL interpretation of instructions, addressing modes, instruction set, Instruction set architecture CISC, RISC, Case study –instruction sets of common CPUs	8
II	Fixed point Addition, Subtraction, Multiplication and Division. Floating Point arithmetic, High performance arithmetic, Sub word parallelism, Booth's algorithm, integer division <ul style="list-style-type: none"> • Data representation method • Booths multiplication, division algorithm and example • IEEE standard single and double precision format and examples 	8
III	Memory hierarchy, Memory Chip Organization, Cache memory, Virtual memory. Parallel Bus Architectures, Internal Communication Methodologies, Serial Bus Architectures, Mass storage, Input and Output Devices, Segmentation, TLB, Page replacement algorithms	8
IV	Pipelining: Basic concepts of pipelining, throughput and speedup, pipeline hazards Pipelining: Basic concepts of pipelining, Arithmetic and Instruction Pipeline, throughput and speedup, pipeline hazards, Introduction, Logic Design Conventions, Building a Datapath – A Simple Implementation scheme – An Overview of Pipelining – Pipelined Datapath and Control. Data Hazards: Forwarding versus Stalling, Control Hazards, Exceptions, Parallelism via Instructions	8
V	CPU control unit design: hardwired and micro-programmed design approaches, Case study - design of a simple hypothetical CPU.	8

Text Books	1.	Introduction to Parallel Computing by Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar in Pearson Publication
	2.	Advance computer Architecture by Kai Hwang under Tata McGraw Hill publications
	3.	Introduction to Parallel Processing: Algorithms & Architectures, Behrooz Parhami in Springer Shop
EBooks	1.	Computer Architecture and Organization by William Stallings http://home.ustc.edu.cn/~leedsong/reference_books_tools/Computer%20Organization%20and%20Architecture%2010th%20-%20William%20Stallings.pdf
Reference Books	1.	Introduction to Parallel Processing by P. Ravi Prakash, M. Sasikumar, Dinesh Shikhare By PHI Publications
	2.	Fundamentals of Parallel Processing by Jordan Harry, Alaghband Gita, PHI Publication
	3	Parallel Computers – Architecture and Programming by V. Rajaraman And C. Siva Ram Murthy.
	4.	Introduction to Parallel Programming by Steven Brawer
Online TL Material	1.	NPTL https://nptel.ac.in/courses/106/105/106105163/

Unit	Contents	Hours
I	Introduction- Basic Mathematical Notation and techniques- Finite State systems – Basic Definitions – Finite Automaton – DFA & NDFA – Finite Automaton with ϵ - moves – Regular Languages- Regular Expression – Equivalence of NFA and DFA – Equivalence of NDFA's with and without ϵ -moves – Equivalence of finite Automaton.	08
II	Regular Expressions- Finite Automata and Regular Expressions, Applications of Regular Expressions, Algebraic Laws for Regular Expressions, Properties of Regular Languages Pumping Lemma for Regular Languages, Applications of the Pumping Lemma, Closure Properties of Regular Languages, Decision Properties of Regular Languages.	08
III	Context-Free Grammars: Chomsky hierarchy of languages. Definition of Context-Free Grammars, Derivations Using a Grammar, Leftmost and Rightmost Derivations, the Language of a Grammar, Sentential Forms, Parse Tree, Applications of Context-Free Grammars, Ambiguity in Grammars and Languages.	08
IV	Push Down Automata- Definition of the Pushdown Automaton, the Languages of a PDA, Equivalence of PDA's and CFG's, Deterministic Pushdown Automata.	08
V	Definitions of Turing machines – Models – Computable languages and functions – Techniques for Turing machine construction – Multi head and Multi tape Turing Machines – The Halting problem.	08

Text Books	1	Introduction to Automata Theory, Languages, and Computation, 3rd Edition, John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Pearson Education.
	2	Introduction to the Theory of Computation, Michael Sipser, 3 rd edition, Cengage Learning.
Reference Books	1	Introduction to Languages and The Theory of Computation, John C Martin, TMH.
	2	Introduction to Computer Theory, Daniel I.A. Cohen, John Wiley.
	3	A Text book on Automata Theory, P. K. Srimani, Nasir S. F. B, Cambridge University Press.
	4	Introduction to Formal languages Automata Theory and Computation Kamala Krithivasan, Rama R, Pearson.

Course Title: UITL201: Object Oriented Programming																		
Semester	III	Teaching Scheme				Evaluation Scheme												
						Theory			Practical									
Term	ODD	Th	Tu	Pr	Credits	TAE	CAE	ESE	INT	EXT								
Course Category	PC	3	-	2	4	10	15	50	25	-								
Course Code	UITL201/ UITP201										5			TOTAL	75		25	
															100			

Course Objectives	1. This course introduces student's general idea and concepts of object oriented programming.
	2. It is also aimed at developing skills to implement these concepts.
	3. The course provide carrier opportunities in design of some applications as object oriented concepts plays dominant role in software development
Course Outcomes	Up on successful completion of this course, student will be able to:
	CO1: Articulate the principles of object oriented programming using C++
	CO2: Apply function overloading, constructor overloading, operator overloading & its uses in programming
	CO3: Implement inheritance and polymorphis concepts and its use for application development
	CO4: Implement static and dynamic memory allocation for software development
	CO5: Develop generic programming applications using templates

Mapping of Course Outcomes with Program Outcomes and Program Specific Outcomes:

[illegible]

Course Contents:

Unit	Contents	Hours
I	Principles Of Object Oriented Programming: Differences between C and C++. A look at procedure Oriented programming, object oriented programming paradigm, basic concepts of OOP, Benefits of OOP, OO languages, A sample program, structure of C++ program. Introduction to OOPS :The origins of C++, What is Object Oriented Programming?, Some C++ fundamentals, Headers & Name Spaces, Introducing C++ Classes, Function overloading, Operator overloading, Inheritance, Constructors & Destructors, Function & Operator Overloading:	8
II	Overloading Constructor functions, Localizing variables, Function overloading & Ambiguity, Finding the address of an overloaded function, this Pointer, Operator overloading, References, Using reference to overload a unary operator, Overloading [], overloading (), Applying operator overloading.	8
III	Inheritance, Virtual Functions and Polymorphism Inheritance and the access specifier, Constructors and Destructors in derived classes, Multiple Inheritance, Passing parameters to a base class, Pointers and references to derived types, Virtual Functions, Why virtual functions?, Pure virtual functions and abstract types, Early Vs Late binding.	8
IV	Static & Dynamic allocation Static & Dynamic allocation using new and delete, static class members, Virtual base classes, const member functions and mutable, volatile member functions, Using the asm keyword, linkage specification, The .* and ->* operators, Creating conversion functions, Copy constructors, Granting access, namespaces, Explicit constructors, typename and export.	8
V	Templates & Exception Handling Class templates, class templates with multiple parameters, function templates, function templates with multiple parameters, Exception Handling, fundamentals, options the uncaught exception (), Applying exception Handling, and RTTI, casting operators, Recent trends in Object Oriented Programming in C++ , Advanced topics & its Application	8

Text Books	1.	Object Oriented Programming in C++ -Robert Lafore, edition, Galgotia publications
	2.	The Complete Reference C++, Herbert Schildt, 4th Edition, TMH
E--Books	1.	Cryptography & network security by atul kahate https://www.pdfdrive.com/cryptography-network-security-by-atul-kahate-e124796757.html
Reference Books	1.	Let's C++ by Y. Kanetkar, BPB publications
	2	Object oriented programming with C++, E Balagurusamy, 4th edition, TMH

Sr. No.	Name of Experiments / Mini Projects/ Case Studies
1	Write a program to compute the area of triangle and circle by overloading the area() function.
2	Define a class to represent a bank account. Include the following members : Data members:- Name of depositor, Account number, Type of account, Balance amount in the account Member functions:- To assign initial values, To deposit an amount, To withdraw an amount after checking the balance, To display name & balance Write a main program to test program using class and object.
3	Create two classes DM and DB which stores values of distances. DM stores distances in meters and centimeters and DB in feet and inches. Write a program that can read values for the class objects and add one object of DM with another object of DB. Use a friend function to carry out addition operation
4	Create a class MAT of size $m * n$. Define all possible matrix operations for MAT type objects
5	Create Stud class to display student information using constructor and destructor. (Default constructor, Multiple constructor, Copy constructor, Overloaded constructor)
6	Consider class network of given figure. The class master derives information from both account and admin classes which in turn derive information from the class person. Define all the four classes and write a program to create, update and display the information contained in master objects.
7	A book shop sells both books and video tapes. Create a class media that stores the title and price of the publication. Create two derived classes, one for storing number of pages in the book and another for storing playing time of tape. A function display() must be defined in all classes to display class contents. Write a program using polymorphism and virtual function.
8	Write a program to show use of this pointer, new and delete.
9	Write a function template for finding the minimum value contained in an array
10	Write a program containing a possible exception. Use a try block to throw it and catch block to handle it properly.
Open Ended Experiments / New Experiments	
1	Write a class template to represent a generic vector. Include member functions to perform following tasks -To create a vector -To modify the value of given element -To multiply by scalar value. -To display vector.
2	Write a C++ program to design a simple calculator

Course Contents:

Unit	Contents	Hours
I	Introduction: Data Science and Business Buzzwords, Difference between Analysis and Analytics, Data Science Info graphic, Tools in Data Science, Applications	3
II	Basics Of Python Spyder (Tool): Introduction Spyder, Setting working Directory, Creating and saving a script file, File execution, clearing console, removing variables from environment, clearing environment, Commenting script files, Variable creation, Arithmetic and logical operators, Data types and associated operations	3
III	Data Structures: Lists, Tuples, Dictionary, Sets, Numpy, Array, Matrix and associated operations, Linear algebra and related operations Control Structures: Control structures using Toyota Corolla dataset, if-else, for loop, for loop with if break, while loop	3
IV	Pandas Dataframe: Pandas dataframe and dataframe related operations, Reading files, exploratory data analysis, Data preparation and preprocessing Data Visualization: Data visualization on Toyoto Corolla dataset using matplotlib and seaborn libraries Scatter plot, Line plot, Bar plot, Histogram, Box plot, Pair plot	3
V	Regression: Predictive Modeling, Linear Regression, Model Assessment, Diagnostic to implement linear fit model Classification: Cross validation, Classification, Logistic regression, Performance measure Clustering: KNN, K-means clustering Multiple Regression: Multiple Regression, Decision Tree Case Studies	3

Text Books	1.	Practical Data Science with Python 3, Varga , Ervin, ISBN 978-1-4842-4859-1, Publisher- Apress
	2.	Python for Data Science For Dummies 2nd Edition by John Paul Mueller, Luca Massaron, Wiley , ISBN 9781119547624
Reference Books	1.	Python Data Science Handbook, ISBN-13: 978-1491912058 , Publisher- O'Reilly Media
	2.	Python Data Science Essentials, By Boschetti Alberto, ISBN: 9781785280429, 9781785280429, Publisher: Packt Publishing Limited
on line TL Material	1.	The Data Science Course 2020: Complete Data Science Bootcamp https://www.udemy.com/course/the-data-science-course-complete-data-science-bootcamp/

Sr. No.	Name of Experiments
1	Apply for Input Output functions and operators in Python
2	Perform and implement various control structures in Python
3	Create and Perform various operations data types in Python

4	Apply the dataframes in python for data reading, preparation and preprocessing
5	Perform various visualization using matplotlib and seaborn
6	Perform the analysis of various dataset and plot histogram on it
7	Study and Implement regression for predictive modelling and linear regression
8	Study and Implement various classification models on data sets
9	Study and Implement various clustering models on data sets
10	Study and Implement multiple regression models on data sets
11	Given a case study: Perform Interactive Data analysis with SciPy

Extra expt. list

Group A			
Sr.No	Name of the Experiment	Assignment Mapped	Unit Mapped
1	Introduction to data Science and Python Spyder Tool with its some basic operation	-	1,2
2.	<p>a) Write a program that prompts the user to input the radius of a circle and outputs the area and circumference of the circle. The formula is</p> <p>b) Suppose a, b, and c denote the lengths of the sides of a triangle. Then the area of the triangle can be calculated using the formula:</p> $Area = \sqrt{s(s-a)(s-b)(s-c)}$ $s = \frac{a+b+c}{2}$ <p>where</p> <p>Write a program that asks the user to input the length of sides of the triangle and print the area.</p>	1	2
3.	Write python program to accept subject marks and to display their Grade card.	1,2	2,3
4.	<p>Functions</p> <p>a) Write a function reverse to reverse a list. Without using the reverse function.</p> <p>b) Write function to compute gcd, lcm of two numbers. Each function shouldn't exceed one line.</p>	1	2,3
5.	<p>Write a program in python that accepts a string to setup a passwords. Your entered password must meet the following requirements:</p> <p>The password must be at least eight characters long.</p> <p>It must contain at least one uppercase letter.</p> <p>It must contain at least one lowercase letter.</p> <p>It must contain at least one numeric digit.</p> <p>Your program should should perform this validation.</p>	1,2	2,3
6	<p>a. Write a program to count the numbers of characters in the string and store them in a dictionary data structure</p> <p>b. To write a python program to generate all permutations of a given string using list or tuple</p>	3	2,3

7	Write a program that generates a random number and asks the user to guess what the number is. If the user's guess is higher than the random number, the program should display "Too high, try again." If the user's guess is lower than the random number, the program should display "Too low, try again." The program should use a loop that repeats until the user correctly guesses the random number. Program should count and display number of tries to win the game.	1,2, 3	2,3
Group B			
Sr. No	Experiment Name	Course Assignment Mapped	Unit Mapped
8	Create a 2D numpy array of 6 soccer team players with their [height(meters), weight(kgs)] by reshaping an 1D numpy array. i) Find out the average height ii) median height iii) standard deviation on height iv) Calcute the bmi and print the bmis >25 v) Create two arrays by checking the bmis. a) Arr1: bmi >25 b) Arr2 : bmi <25 and print both the arrays	4	4
9	Write a python function that takes two numpy.ndarray objects, checks if they are the same shape (printing ERROR and aborting if they aren't), then raises the numbers b of the second array to the exponents a in the first array. Do this without using numpy functions like numpy.power in your function. Make sure that it works with differently sized numpy arrays arrays of 0 dimensions, 1 dimension and 2 dimensions. Do the same using the numpy.power function.	4	4
10	Write a python program to search for a string in an heterogenous array of 7 items by using a user defined function that accepts two arguments. i) If string found return the position of the string ii)if string not found, add the string to the list. Note: Do not use built in functions/methods.	4	4
11	a. Using Panads library read excel sample file of 100 records into Dataframe. Also write these 100 records into particular sheet of excel file.	4	4
12	Using Panads library read CSV/excel sample file and write into JSON file formation	4	4
13	Write a python program to create a DataFrame of 6 rows from Dictionary. First create a Tourism_visitors dictionary from four lists of "Cities", "Visitors", "signups" and "Weekdays". Convert the dictionary to Dataframe importing Pandas library. And create row_labels as the short forms of the country names. And display the DataFrame.	4	4
14	Download 100 CC Records csv from http://eforexcel.com/wp/downloads-17-sample-csv-files-data-sets-for-testing-credit-card/ . And read the CSV into a DataFrame. <ul style="list-style-type: none"> Display the card type, holder name, issuing bank 		

	and credit limit of the holders from 20 to 40(row indexes) using loc and iloc commands. <ul style="list-style-type: none"> Display all columns and table information. 		
Group C			
Sr. No	Experiment Name	Course Assignment Mapped	Unit Mapped
15	Using the sample dataset i. Handle the null values if any by removing them or perform imputation ii. Import the necessary package and perform the train and test split on the dataset.	4	5
16	Perform various data visualization using matplotlib library.	5	5
17	Plot line and scatter charts for students interests in programming against their year in the Engineering college. And derive a distribution of the same over 10 years using a histogram.	6	5
18	Study and Implement any one type of regression model (linear, multiple regression) using appropriate dataset	7,10	5
19	Using Naïve Bayes, Decision Tree, KNN predict if a customer with certain age and Salary will purchase a product or not	8	5
20	Study and Implement K-mean clustering using appropriate dataset.	9	5
21	Given a case study : Perform Interactive Data analysis on titanic dataset using data analysis tool.	11	5