

B. Tech Computer Science and Engineering (Artificial Intelligence and Machine Learning)

Semester-III

S. No.	Course No./ Code	Subject	L:T:P	Hours/ Week	Credits	Examination Schedule(Marks)				Duration of Exam(Hours)
						End Semester Exam	Internal assessment	Practical Exam	Total	
1	B24-CAM-201	Data Structures	4:0:0	4	4	70	30	0	100	3
2	B24-CAM-203	Modern Computer Architecture	3:0:0	3	3	70	30	0	100	3
3	B24-CSE-205	Object Oriented Programming	4:0:0	4	4	70	30	0	100	3
4	B24-CSE-207	IT Workshop(Python)	3:0:0	3	3	70	30	0	100	3
5	B24-BSC-205	Mathematical Concepts for AI	3:0:0	3	3	70	30	0	100	3
6	B24-CAM-209	Artificial Intelligence	3:0:0	3	3	70	30	0	100	3
7	B24-CAM-211	Data Structures Lab	0:0:3	3	1.5	0	40	60	100	3
8	B24-CSE-211	Object Oriented Programming Lab	0:0:3	3	1.5	0	40	60	100	3
9	B24-CSE-213	IT Workshop(Python)Lab	0:0:2	2	1	0	40	60	100	3
10	B24-MAC-202	Essence of IndianTraditional Knowledge	2:0:0	2	1	--	100	0	100	3
TOTAL				30	25	420	400	180	1000	

Lesson Plan:

Name of the Faculty:		Ms. Upasana Sood	
Discipline:		B.Tech (CSE)	
Semester:		3rd	
Subject:		Data Structures(B24-CAM-201)	
Course Outcome:		CO1	To introduce the basic concepts of Data structure, basic data types, searching and sorting based on array data types.
		CO2	To introduce the structured data types like Stacks and Queue and its basic operations implementation. CO 3
		CO3	To introduce dynamic implementation of linked list.
		CO4	To introduce the concepts of Tree and graph and implementation of traversal algorithms
WorkLoad(Lecture/Practical) perweek(Inhours):		Lecture-4	
S.No	Lecture No.	Theory	
		Topic(IncludingAssignment/ Test/)	Pedagogy (PPT/Chalk and Board/Activity/Case Study)
1.	L1	Unit1: Introduction to Data Structures	Chalk and Board
2.	L2	Data Types	Chalk and Board
3.	L3	Built in and User Defined Data Structures	Chalk and Board
4.	L4	Data Structure Operations,	Chalk and Board

5.	L5	Revision	Chalk and Board
6.	L6	Applications of Data Structure	Chalk and Board
7.	L7	Analysis of an Algorithm	Chalk and Board
8.	L8	Asymptotic Notations	Chalk and Board
9.	L9	Time-space trade off .	Chalk and Board
10.	L10	Arrays, Arrays(One Dimensional	Chalk and Board
11.	L11	Two Dimensional and Multi-Dimensional)	Chalk and Board,
12.	L12	Sparse Matrices	Chalk and Board,
13.	L13	Searching from Array using Linear	Chalk and Board
14.	L14	Binary Searching Algorithm	Chalk and Board
15.	L15	Sorting of Array using Selection	Chalk and Board
16.	L16	Insertion	Chalk and Board
17.	L17	Bubble Sort.	Chalk and Board
18.	L18	Unit2: Stacks: ADT Stack and Its Operations	Chalk and Board
19.	L19	Algorithm and their complexity analysis,	Chalk and Board
20.	L20	Applications of Stack, Conversion	Chalk and Board
21.	L21	Evaluation, Implementation of	Chalk and Board

		Merge Sort	
22.	L22	Quick Sort Algorithm	Chalk and Board
23.		Sessional	Offline
24.	L23	ADT Queue	Chalk and Board
25.	L24	Types of Queue: Simple Queue, Circular Queue, Priority Queue	Chalk and Board
26.	L25	Operations on each type of Queues, Applications of Queues.	Chalk and Board
27.		Revision	Chalk and Board
28.	L26	Unit 3: Linked Lists: Need of Dynamic Data Structures	Chalk and Board and PPT
29.	L27	Singly Linked List and Its Dynamic Implementation	Chalk and Board
30.	L28	Algorithms of several operations: Traversing, Searching, Insertion into Link List	Chalk and Board
31.	L29	Deletion from Linked Lists, Comparison between Static and Dynamic	Chalk and Board
32.	L30	Implementation of Linked List. Circular Link Lists and Doubly Link List	Chalk and Board
33.	L31	Dynamic Implementation of Primitive Operations on Doubly Linked Lists and Circular Linked List	Chalk and Board
34.	L32	Linked representations of Stacks and Queues.	Chalk and Board
35.		Sessional	Offline

36.	L33	Unit4: Trees: Definition, Basic Terminology	Chalk and Board
37.	L34	Binary Tree	
38.	L35	Static and Dynamic Implementation of a Binary Tree	
39.	L36	Primitive Operations on Binary Trees, Binary Tree Traversals	Chalk and Board
40.	L37.	Representation of Infix, Post-Fix and Prefix Expressions using Trees	Chalk and Board
41.	L38.	Different types of Trees: Binary Search Trees, B+ Trees	Chalk and Board
42.	L39.	AVL Trees, Threaded Binary Trees,	Chalk and Board
43.	L40	Balanced Multi- way Search Trees, Implementation of Heap Sort Algorithm.	Chalk and Board
44.	L41	Graphs: Basic Terminologies	Chalk and Board
45.	L42	Representations of Graphs	Chalk and Board
46.	L43	Revision	Offline
47.	L44	Graph Search	Chalk and Board
48.	L45	Traversal Algorithms and Complexity Analysis.	Chalk and Board
49.	L46	Sessional	Offline

Name of the Faculty:		Ms. Pratibha	
Discipline:		B.Tech (CSE)	
Semester:		3rd	
Subject:		Modern Computer Architecture (B24- CAM- 203)	
Course Outcome:		CO1	Be familiar with computer arithmetic and digital logic related to computer architecture
		CO2	Be familiar with computer architecture and instruction set architecture.
		CO3	Be familiar with understanding the concept of memory hierarchy and basic computer organization and design
		CO4	Be acquainted with the basic knowledge of I/O devices organization.
WorkLoad(Lecture/Practical) perweek(Inhours):		Lecture-3	
S.No	Lecture No.	Theory	
		Topic(IncludingAssignment/ Test/)	Pedagogy (PPT/Chalk and Board/Activity/Case Study)
1.	L1	Unit1: Digital Logic and Architecture: History of computer architecture, Von Neumann Architecture	Chalk and Board
2.	L2	Flynn's Classification of Computers, combinational vs sequential logic	Chalk and Board
3.	L3	physical constraints (gate delay, fan-in, fan-out, energy/power)	Chalk and Board
4.	L4	MIPS, MFLOPS	Chalk and Board
5.	L5	Computer Arithmetic: Representation of numeric data, signed and unsigned arithmetic; Range	Chalk and Board

6.	L6	Precision and errors in floating-point arithmetic	Chalk and Board
7.	L7	Digital arithmetic algorithms for Addition	Chalk and Board
8.	L8	Subtraction	Chalk and Board
9.	L9	Multiplication using Booth's algorithm.	Chalk and Board
10.	L10	Unit 2: Computer Architecture: Computer architecture for artificial intelligence applications,	Chalk and Board
11.	L11	Artificial intelligence applications	Chalk and Board,
12.	L12	AI-assisted Design for Architecture (AIDArc).	Chalk and Board,
13.	L13	AI-assisted Design for Architecture (AIDArc).	Chalk and Board
14.	L14	Common Bus System,	Chalk and Board
		Sessional	Offline
15.	L15	General register organization, stack	Chalk and Board
16.	L16	Organization Instruction Set Architecture	Chalk and Board
17.	L17	Basic organization of computing machine: fetch	Chalk and Board
18.	L18	Decode, and execute	Chalk and Board
19.	L19	Instruction set types	Chalk and Board
20.	L20	Instruction format	Chalk and Board

21.	L21	Addressing modes	Chalk and Board
22.	L22	Sub routine call and return mechanisms.	Chalk and Board
23	L23	Unit 3: Basic Computer Organization and Design: Storage systems,	Chalk and Board
24.	L24	Types of cache memory: address mapping, block size, replacement	Chalk and Board
25.	L25	Store policies	Chalk and Board
26.	L26	virtual memory system: page table and TLB	Chalk and Board
27.		Sessional	Offline
28.	L27	Advanced Optimizations of Cache Performance	Chalk and Board and PPT
29.	L28	Memory Technology and Optimizations,	Chalk and Board
30.	L29	Case Study: Memory Hierarchies in Intel Core i7 and ARM Cortex-A8.	Chalk and Board
31.	L30	CISC vs RISC Designs	Chalk and Board
32.	L31	Design of arithmetic and logic unit (ALU).	Chalk and Board
33.	L32	Control unit: hardwired realization vs micro-programmed realization	Chalk and Board
34.	L33	Multi-cycle implementation,	Chalk and Board
35.	L34	Instruction level parallelism, instruction pipelining, pipeline hazards.	Chalk and Board

36	L35	Unit 4: Input-output organization: I/O interface. I/O Bus and interface modules, I/O versus Memory Bus	Chalk and Board
37	L36	Asynchronous data transfer: Strobe control, Handshaking, Asynchronous serial transfer	Chalk and Board
38	L37	Modes of Transfer: Programmed I/O, Interrupt driven I/O	Chalk and Board
39	L38	Priority interrupt; Daisy chaining, Parallel Priority interrupt.	Chalk and Board
40	L39	Direct memory Access, DMA controller and transfer.	Chalk and Board
41		Sessional	Offline

Name of the Faculty:		Ms. Deepshikha	
Discipline:		B.Tech (CSE)	
Semester:		3rd	
Subject:		Object Oriented Programming (B24- CAM- 205)	
Course Outcome:		CO1	To introduce the basic concepts of object oriented programming language and the its representation.
		CO2	To allocate dynamic memory, access private members of class and the behavior of inheritance and its implementation
		CO3	To introduce polymorphism, interface design and overloading of operator
		CO4	To handle backup system using file, general purpose template and handling of raised exception during programming
WorkLoad(Lecture/Practical) perweek(Inhours):		Lecture-4	
S.No	Lecture No.	Theory	
		Topic(IncludingAssignment/ Test/)	Pedagogy (PPT/Chalk and Board/Activity/Case Study)
1.	L1	Unit1: Introduction to C++, C++ Standard Library, Illustrative Simple C++ Programs.	Chalk and Board
2.	L2	Header Files, Namespaces, Application of object oriented programming	Chalk and Board
3.	L3	Control flow, variables	Chalk and Board
4.	L4	Assignments statements, conditional execution, looping, function calls including recursion.	Chalk and Board
5.	L5	Object Oriented Concepts	Chalk and Board

6.	L6	Introduction to Objects and Object Oriented Programming Encapsulation, Polymorphism	Chalk and Board
7.	L7	Overloading , Inheritance	Chalk and Board
8.	L8	Abstract Classes	Chalk and Board
9.	L9	Accessifier (public/ protected/ private)	Chalk and Board
10.	L10	Class Scope	Chalk and Board
11.	L11	Accessing Class Members, Constant	Chalk and Board,
12.	L12	Class Member, Structure	Chalk and Board,
13.	L13	Class Macro vs Inline Functions.	Chalk and Board
14.	L14	Revision	Chalk and Board
		Sessional	Offline
15.	L15	Unit 2: Friend Function	Chalk and Board
16.	L16	Friend Classes, This Pointer	Chalk and Board
17.	L17	Dynamic Memory Allocation and Deallocation (New and Delete)	Chalk and Board
18.	L18	Static Class Members	Chalk and Board
19.	L19	Constructors	Chalk and Board
20.	L20	Parameter Constructors and Copy Constructors	Chalk and Board

21.	L21	Deconstructors	Chalk and Board
22.	L22	Introduction of inheritance	Chalk and Board
23	L23	Types of Inheritance, Overriding Base Class	Chalk and Board
24.	L24	Members in a Derived Class	Chalk and Board
25.	L25	Public, Protected and Private Inheritance	Chalk and Board
26.	L26	Effect of Constructors and Deconstructors of Base Class in Derived Classes.	Chalk and Board
27.		Sessional	Offline
28.	L27	Unit 3: Polymorphism	Chalk and Board and PPT
29.	L28	Pointer to Derived class	Chalk and Board
30.	L29	Virtual Functions	Chalk and Board
31.	L30	Pure Virtual Function	Chalk and Board
32.	L31	Abstract Base Classes	Chalk and Board
33.	L32	Static and Dynamic Binding	Chalk and Board
34.	L33	Virtual Deconstructors	Chalk and Board
35.	L34	Fundamentals of Operator Overloading , Rules for Operators Overloading	Chalk and Board
36	L35	Implementation of Operator Overloading Like <<, >>	Chalk and Board

		Unary Operators	
37	L36	Binary Operators	Chalk and Board
38	L37	Unit 4: Text Streams and binary stream	Chalk and Board
39	L38	Sequential and Random Access File creation and updation	Chalk and Board
40	L39	Stream Input/ Output Classes, Stream Manipulators	Chalk and Board
41	L40	Basics of C++ Exception Handling, Try, Throw, Catch, multiple catch	Chalk and Board
42	L41	Re-throwing an Exception, Exception specifications	Chalk and Board
43	L42	Templates: Function Templates, Overloading	Chalk and Board
44	L43	Template Functions, Class Template,	Chalk and Board
45	L44	Class Templates and Non-Type Template arguments.	Chalk and Board
46		Sessional	Offline

Name of the Faculty:		Ms. Ruchi Gupta	
Discipline:		B.Tech (CSE)	
Semester:		3rd	
Subject:		IT Workshop (Python) (B24- CAM- 207)	
Course Outcome:		CO1	To Study Fundamental concept of Python.
		CO2	To Study and implement tuples , list and dictionary operations.
		CO3	To Study and implement tuples , list and dictionary operations.
		CO4	To Study and implement exception handling and file operation.
WorkLoad(Lecture/Practical) perweek(Inhours):		Lecture-3	
S.No	Lecture No.	Theory	
		Topic(IncludingAssignment/ Test/)	Pedagogy (PPT/Chalk and Board/Activity/Case Study)
1.	L1	Unit1: Familiarization with the basics of Python programming	Chalk and Board
2.	L2	Introduction to Python, Features of Python,	Chalk and Board
3.	L3	Execution modes: interactive mode	Chalk and Board
4.	L4	script mode	Chalk and Board
5.	L5	Python character set, use of indentation	Chalk and Board
6.	L6	Python tokens(keyword, identifier, literal, operator, punctuator)	Chalk and Board
7.	L7	variables, use of comments	Chalk and Board

8.	L8	Knowledge of data types: Number(Integer, Floating point, Complex).	Chalk and Board
9.	L9	Errors: syntax errors, logical errors, and run-time errors	Chalk and Board
10.	L10	Unit 2: Expressions: Statement	Chalk and Board
11.	L11	Type conversion, and input/output: Precedence of Operators	Chalk and Board,
12.	L12	Arithmetic operators, relational operators	Chalk and Board,
13.	L13	logical operators, assignment operators, augmented assignment operators	Chalk and Board
14.	L14	Identity operators (is, is not), Expression, evaluation of an expression, type-conversion	Chalk and Board
		Sessional	Offline
15.	L15	Flow of Control, Conditional statements	Chalk and Board
16.	L16	Iterative Statements Strings: Introduction	Chalk and Board
17.	L17	string operations (concatenation, repetition, membership and slicing), traversing a string using loops	Chalk and Board
18.	L18	built-in functions/methods–len(), capitalize(), title(), lower(), upper()	Chalk and Board
19.	L19	count(), find(), index(), endswith()	Chalk and Board
20.	L20	startswith(), isalnum(), isalpha(), isdigit()	Chalk and Board
21.	L21	islower(), isupper(), isspace(), lstrip(), rstrip()	Chalk and Board

22.	L22	strip(), replace(), join(), partition(), split()	Chalk and Board
23	L23	Unit 3: Array: Access the Elements of an Array, Length of an Array,	Chalk and Board
24.	L24	Adding Array Elements, Removing Array Elements, Adds and remove the element at the specified position	Chalk and Board
25.	L25	Lists, Tuples	Chalk and Board
26.	L26	Dictionary: introduction, indexing, list operations, traversing a list using loops	Chalk and Board
27.		Sessional	Offline
28.	L27	built-in functions/methods–len(), list(), append(), extend(), insert(), count(), index(), remove(), pop(), reverse(), sort(), sorted(), min(), max(), sum()	Chalk and Board and PPT
29.	L28	Introduction to Python modules: Importing module using ‘import ’ and using from statement	Chalk and Board
30.	L29	importing math module (pi, e, sqrt(), ceil(), floor(), pow(), fabs(), sin(), cos(), tan())	Chalk and Board
31.	L30	random module (random(), randint(), randrange()), statistics module (mean(), median(), mode())	Chalk and Board
32.	L31	Functions and its types	Chalk and Board
33.	L32	(Built-in Functions ,Functions defined in Module, User Defined Functions)	Chalk and Board
34.	L33	Arguments, default parameters, positional parameter	Chalk and Board
35.	L34	Function Returning Value(s), Recursion, Scope of a	Chalk and Board

		Variable	
36	L35	Unit 4: Files: Introduction to files, types of files (Text file, Binary file, CSV file),	Chalk and Board
37	L36	Text file: opening a text file, file open modes (r, r+, w, w+, a, a+ etc),	Chalk and Board
38	L37	closing a text file, opening a file using with clause,	Chalk and Board
39	L38	writing/appendng data to a text file using write() and writelines()	Chalk and Board
40	L39	reading from a text file using read(), readline() and readlines	Chalk and Board
41		Sessional	Offline

Name of the Faculty:		Dr. Punam Kalra	
Discipline:		B.Tech (CSE)	
Semester:		3rd	
Subject:		Mathematical Concepts for AI (B24- BSC-205)	
Course Outcome:		CO1	To introduce the fundamental concepts of probability to analyze and predict outcomes in real- life situations.
		CO2	Probability theory provides models of probability distributions(theoretical models of the observable reality involving chance effects) to be tested by statistical methods which has various engineering applications.
		CO3	To make the students familiar about basic statistics to analyze data sets using various measures of central tendency and dispersion
		CO4	Upon completion of Unit IV, students will proficiently apply correlation and regression techniques, including calculating coefficients and determining lines of regression, to analyze relationships between variables in datasets.
WorkLoad(Lecture/Practical) perweek(Inhours):		Lecture-3	
S.No	Lecture No.	Theory	
		Topic(IncludingAssignment/ Test/)	Pedagogy (PPT/Chalk and Board/Activity/Case Study)
1.	L1	Unit1: Basic Probability: Introduction, additive law of probability	Chalk and Board
2.	L2	Conditional Probability, Independent Events	Chalk and Board

3.	L3	Bayes' Theorem.	Chalk and Board
4.	L4	Random Variables: Discrete random variables	Chalk and Board
5.	L5	probability distribution	Chalk and Board
6.	L6	Probability mass function	Chalk and Board
7.	L7	distribution function	Chalk and Board
8.	L8	Expectation, Moments	Chalk and Board
9.	L9	Variance	Chalk and Board
10.	L10	Standard deviation of discrete random variables.	Chalk and Board
11.	L11	Unit 2: Continuous Probability distribution	Chalk and Board,
12.	L12	Continuous random variables	Chalk and Board,
13.	L13	probability distribution	Chalk and Board
14.	L14	Revision	Chalk and Board
		Sessional	Offline
15.	L15	Probability density function	Chalk and Board
16.	L16	distribution function	Chalk and Board
17.	L17	Expectation, Moments	Chalk and Board
18.	L18	Variance and standard deviation of Continuous random variables	Chalk and Board

19.	L19	Variance and standard deviation of Continuous random variables	Chalk and Board
20.	L20	Probability distributions: Binomial	Chalk and Board
21.	L21	Poisson	Chalk and Board
22.	L22	Normal - evaluation of statistical parameters for these three distributions	Chalk and Board
23	L23	Unit 3: Basic Statistics: Measures of	Chalk and Board
24.	L24	Central tendency: Mean, median, quartiles, mode	Chalk and Board
25.	L25	Central tendency: Mean, median, quartiles, mode	Chalk and Board
26.	L26	Revision	Chalk and Board
27.		Sessional	Offline
28.	L27	Geometric mean	Chalk and Board and PPT
29.	L28	Harmonic mean	Chalk and Board
30.	L29	Measures of dispersion: Range	Chalk and Board
31.	L30	Quartile deviation	Chalk and Board
32.	L31	mean deviation	Chalk and Board
33.	L32	standard deviation	Chalk and Board
34.	L33	coefficient of variation, Moments	Chalk and Board
35.	L34	Skewness and Kurtosis	Chalk and Board

36	L35	Unit 4: Correlation & Regression: Introduction, Correlation	Chalk and Board
37	L36	Coefficient of correlation,	Chalk and Board
38	L37	methods of calculations	Chalk and Board
39	L38	Lines of regression	Chalk and Board
40	L39	Rank correlation	Chalk and Board
41		Sessional	Offline

Name of the Faculty:		Dr. Niti	
Discipline:		B.Tech (CSE)	
Semester:		3rd	
Subject:		Essence of Indian Traditional Knowledge (B24- MAC-202)	
Course Outcome:		CO1	The students will be able to understand , connect up and explain basics of Indian traditional knowledge in modern scientific perspective CO2
		CO2	The students will be able to understand Holistic Health using Indian Knowledge System
		CO3	The students will be able to Manage thoughts and Emotions , will learn positivity, self regulation and control
		CO4	The students will be able to Achieve Consciousness through Indian Knowledge System
WorkLoad(Lecture/Practical) perweek(Inhours):		Lecture-2	
S.No	Lecture No.	Theory	
		Topic(IncludingAssignment/ Test/)	Pedagogy (PPT/Chalk and Board/Activity/Case Study)
1.	L1	Unit1: Introduction to Indian Traditional knowledge:	Chalk and Board
2.	L2	Define traditional knowledge	Chalk and Board
3.	L3	Importance	Chalk and Board
4.	L4	Kinds of traditional knowledge.	Chalk and Board
5.	L5	Philosophical systems	Chalk and Board

6.	L6	, Basics of Rajyoga	Chalk and Board
7.	L7	Karamyoga, Benefits of Rajyoga and Karamyoga.	Chalk and Board
8.	L8	Unit2: Holistic Health using Indian Knowledge System:	Chalk and Board
9.	L9	Indian Knowledge System	Chalk and Board
10.		Sessional	Offline
11.	L10	Basic principles of natural life style,	Chalk and Board,
12.	L11	Benefits through five elements.	Chalk and Board,
13.	L12	Healing through food, Chakras and Mudras	Chalk and Board
14.	L13	Physical, Mental, Emotional	Chalk and Board
15.	L14	Spiritual health using traditional knowledge	Chalk and Board
16.	L15	Unit 3: Positivity: Traditional approaches.	Chalk and Board
17.	L16	Happiness: objective and subjective measures of wellbeing, life satisfaction. Resilience	Chalk and Board
18.		Sessional	Offline
19.	L17	Self-regulation and self-control, optimism, self-esteem	Chalk and Board
20.	L18	Managing thoughts and Emotions with the help of Rajyoga	Chalk and Board
21.	L19	Achieving Powers for Self Mastery.	Chalk and Board

22.	L20	Unit 4: Achieving Consciousness through Indian Knowledge System: Emotional intelligence,	Chalk and Board
23	L21	Indian approach to Psychology	Chalk and Board
24.	L22	Consciousness; levels, body-mind relationship	Chalk and Board
25	L23	self motivation	Chalk and Board
26	L24	Self and Identity in modern Psychology and Indian thought., Spirituality and well being.	
27		Sessional	Offline

Name of the Faculty:	Ms. Tamnna	
Discipline:	B.Tech (CSE)	
Semester:	3rd	
Subject:	Data Structures Lab (B24-CAM-211)	
Course Outcome:	CO1	To introduce the basic concepts of Data structure, basic data types, searching and sorting based on array data types.
	CO2	To introduce the structured data types like Stacks and Queue and its basic operation's implementation
	CO3	To introduce dynamic implementation of linked list
	CO4	To introduce the concepts of Tree and graph and implementation of traversal algorithms.
WorkLoad(Lecture/Practical) perweek(Inhours):	Lecture-3	

List of Program:

1. Write a program for Binary search methods.
2. Write a program that implements the following sorting: (i) Bubble Sort (ii) Selection Sort (iii) Insertion Sort.
3. Write a program to implement Stack and its operation using (i) Arrays (ii) Linked List.
4. Write a program that implements the following : (i) Quick Sort (ii) Merge Sort (iii) Heap Sort.
5. Write a program to implement Queue and its operation using (i) Arrays (ii) Linked List.
6. Write a program to implement Circular Queue and its operation using Arrays.
7. Write a program to implement singly linked list for the following operations: Create, Display, searching, traversing and deletion.
8. Write a program to implement doubly linked list for the following operations: Create, Display, inserting, counting, searching, traversing and deletion.
9. Write a program to implement circular linked list for the following operations: Create, Display, inserting, counting, searching, traversing and deletion.

10. Write a program to implement the tree traversal methods.
11. Write a program to perform the following operations : (i) Insert an element into a AVL tree. (ii) Delete an element from a AVL tree. (iii) Search for a key element in a AVL tree.

Name of the Faculty:	Ms. Deepshikha	
Discipline:	B.Tech (CSE)	
Semester:	3rd	
Subject:	Object Oriented Programming Lab (B24-CSE-211)	
Course Outcome:	CO1	To introduce the basic concepts of object oriented programming language and the its representation.
	CO2	To allocate dynamic memory, access private members of class and inheritance,Constructors.
	CO3	To introduce polymorphism, interface design and overloading of operator.
	CO4	To handle backup system using file, general purpose template and handling of raised exception during programming.
WorkLoad(Lecture/Practical) perweek(Inhours):	Lecture-3	

List of Program:

Q1. Raising a number n to a power p is the same as multiplying n by itself p times. Write a function called power () that takes a double value for n and an int value for p, and returns the result as double value. Use a default argument of 2 for p, so that if this argument is omitted, the number will be squared. Write a main () function that gets values from the user to test this function.

Q2. Write a C++ program that illustrates the order of execution of constructors and destructors when new class is derived from more than one base class. b) Write a Program to Invoking Derived Class Member Through Base Class Pointer.

Q3. Create the equivalent of a four function calculator. The program should request the user to enter a number, an operator, and another number. It should then carry out the specified arithmetical operation: adding, subtracting, multiplying, or dividing the two numbers. (It should use a switch statement to select the operation). Finally it should display the result. When it finishes the calculation, the program should ask if the user wants to do another calculation. The response can be 'Y' or 'N'. Some sample interaction with the program might look like this. Enter first number,

operator, and second number: 10/ 3 Answer = 3.333333 Do another (Y/ N)? Y Enter first number, operator, second number 12 + 100 Answer = 112 Do another (Y/ N) ? N

Q4. A phone number, such as (212) 767-8900, can be thought of as having three parts: the area code (212), the exchange (767) and the number (8900). Write a program that uses a structure to store these three parts of a phone number separately. Call the structure phone. Create two structure variables of type phone. Initialize one, and have the user input a number for the other one. Then display both numbers. The interchange might look like this: ☐ Enter your area code, exchange, and number: 415 555 1212 ☐ My number is (212) 767-8900 ☐ Your number is (415) 555-1212

Q5. Create two classes DM and DB which store the value of distances. DM stores distances in metres 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 the addition operation. The object that stores the results maybe a DM object or DB objects, depending on the units in which the results are required. The display should be in the format of feet and inches or metres and centimetres depending on the object on display.

Q6. Create a class rational which represents a numerical value by two double values- NUMERATOR and DENOMINATOR. Include the following public member Functions: ☐ constructor with no arguments (default). ☐ constructor with two arguments. ☐ void reduce() that reduces the rational number by eliminating the highest common factor between the numerator and denominator. ☐ Overload + operator to add two rational number. ☐ Overload >> operator to enable input through cin. ☐ Overload << operator to enable output through cout. Write a main () to test all the functions in the class.

Q7. Consider the following class definition class father { protected : int age; public; father (int x) {age = x;} virtual void iam () { cout << "I AM THE FATHER, my age is : "<< age<< endl;} }; Derive the two classes son and daughter from the above class and for each, define iam () to write our similar but appropriate messages. You should also define suitable constructors for these classes. Now, write a main () that creates objects of the three classes and then calls iam () for them. Declare pointer to father. Successively, assign addresses of objects of the two derived classes to this pointer and in each case, call iam () through the pointer to demonstrate polymorphism in action.

Q8. Write a program that creates a binary file by reading the data for the students from the terminal. The data of each student consist of roll no., name (a string of 30 or lesser no. of characters) and marks.

Q9. A hospital wants to create a database regarding its indoor patients. The information to store include i. Name of the patient ii. Date of admission iii. Disease iv. Date of discharge Create a structure to store the date (year, month and date as its members). Create a base class to store the above information. The member function should include functions to enter information and display a list of all the patients in the database. Create a derived class to store the age of the patients. List the information about all the to store the age of the patients. List the information about all the pediatric patients (less than twelve years in age).

Q10. Create a class Employee with a name and salary. Create a class Manager inherit from Employee. Add an instance variable, named department, of type string. Supply a method to to String that prints the manager's name, department and salary. Create a class Executive inherits

from Manager. Supply a method to String that prints the string “Executive” followed by the information stored in the Manager superclass object. Supply a test program that tests these classes and methods.

Q11. Imagine a tollbooth with a class called toll Booth. The two data items are a type unsigned int to hold the total number of cars, and a type double to hold the total amount of money collected. A constructor initializes both these to 0. A member function called payingCar () increments the car total and adds 0.50 to the cash total. Another function, called nopayCar (), increments the car total but adds nothing to the cash total. Finally, a member function called displays the two totals. Include a program to test this class. This program should allow the user to push one key to count a paying car, and another to count a nonpaying car. Pushing the ESC key should cause the program to print out the total cars and total cash and then exit.

Q12. Write a function called reversit () that reverses a string (an array of char). Use a for loop that swaps the first and last characters, then the second and next to last characters and so on. The string should be passed to reversit () as an argument. Write a program to exercise reversit (). The program should get a string from the user, call reversit (), and print out the result. Use an input method that allows embedded blanks. Test the program with Napoleon’s famous phrase, “Able was I ere I saw Elba”.

Q13. Create a class Student with a name and roll no. as data member. Create a Class Template for student class. The program should also implement template overloading.

Q14. Assume that a bank maintains two kinds of accounts for customers, one called as savings account and the other as current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed. Create a class account that stores customer name, account number and type of account. From this derive the classes cur_acct and sav_acct to make them more specific to their requirements. Include necessary member functions in order to achieve the following tasks: i. Accept deposit from a customer and update the balance. ii. Display the balance. iii. Compute and deposit interest. iv. Permit withdrawal and update the balance. v. Check for the minimum balance, impose penalty, necessary and update the balance. vi. Do not use any constructors. Use member functions to initialize the class members.

Name of the Faculty:	Ms. Ruchi Gupta	
Discipline:	B.Tech (CSE)	
Semester:	3rd	
Subject:	IT Workshop (Python) Lab (B24-CSE-213)	
Course Outcome:	CO1	To study fundamentals of python programming and implement basic programs.
	CO2	To implement the searching technique using python.
	CO3	To implement sorting techniques using python.
	CO4	To implement matrix multiplication using python.
WorkLoad(Lecture/Practical) perweek(Inhours):	Lecture-3	

List of Program:

1. Write a program to compute the GCD of two numbers.
2. Write a program to find the square root of a number
3. Write a program to find the Exponentiation (power of a number)
4. Write a program to find the maximum of a list of numbers
5. Write a program for Linear search and Binary search
6. Write a program for Selection sort
7. Write a program for Insertion sort
8. Write a program to find first n prime numbers
9. Write a program to multiply matrices
10. Write a program that take command line arguments (word count)
11. Write a program to find the most frequent words in a text read from a file