Table of Contents

Note : Remove “Table of Content” before including in CP Book

18CSL38: Data Structure Laboratory

A. LABORATORY INFORMATION

1. Lab Overview

Degree:	B.Tech	*Program:*	CS
Year / Semester :	2/ 3	*Academic Year:*	2018-19
Course Title:	Design and Analysis of Algorithms Lab	*Course Code:*	18CSL38
Credit / L-T-P:	2/ 0-0-2	*SEE Duration:*	180 Minutes
Total Contact Hours:	40Hrs	*SEE Marks:*	60 Marks
CIA Marks:	40	*Assignment*	
Course Plan Author:	Akshatha Kamath/Sowmya C v	*Sign*	Dt :
Checked By:		*Sign*	Dt :

2. Lab Content

Unit	**Title of the Experiments**	**Lab Hours**	**Concept**	**Blooms Level**
1	Design, Develop and Implement a menu driven Programin C for the following Array operations a.Creating an Array of NInteger Elements b.Display of Array Elements with Suitable Headings c.Inserting an Element (ELEM) at a given valid Position (POS) d.Deleting an Element at a given valid Position(POS) e.Exit. Support the program with functions for each of the above operations		Data Manipulation	L4 Analyze
2	Design, Develop and Implement a Program in C for the following operations on Strings a.Read a main String (STR), a Pattern String (PAT) and a Replace String (REP) b.Perform Pattern Matching Operation: Find and Replace all occurrences of PAT in STR with REP if PAT exists in STR. Report suitable messages in case PAT does not exist in STR Support the program with functions for each of the above operations. Don't use Built-in functions.		Data Manipulation	L4
3	Design, Develop and Implement a menu driven Programin C for the following operations on STACK of Integers (Array Implementation of Stack with maximum size MAX) a.Push an Element on to Stack b.Pop an Element from Stack c.Demonstrate how Stack can be used to check Palindrome d.Demonstrate Overflow and Underflow situations on Stack e.Display the status of Stack f.Exit Support the program with appropriate functions for each of the above operations		Stack Operations	L4
4	Design, Develop and Implement a Program in C for converting an Infix Expression to Postfix Expression. Program should support for both parenthesized and free parenthesized expressions with the operators: +, -, *, /, %(Remainder), ^(Power) and alphanumericoperands.		Stack Operations	L4
5	Design, Develop and Implement a Program in C for the following Stack Applications		Stack Operatio	L4

	a.Evaluation of Suffix expressionwith single digit operands andoperators: +, -, *, /, %, ^ b.Solving Tower of Hanoi problem with n disks		ns	
6	Design, Develop and Implement a menu driven Program in C for the following operations on Circular QUEUE of Characters (ArrayImplementation of Queue with maximum size MAX) a.Insert an Element on to Circular QUEUE b.Delete an Element from Circular QUEUE c.Demonstrate Overflow and Underflow situations on Circular QUEUE d.Display the status of Circular QUEUE e.Exit Support the program with appropriate functions for each of the above operations		Queue Features	L4
7	Design, Develop and Implement a menu driven Program in C for the following operations on Singly Linked List (SLL) of Student Data with the fields: USN, Name, Branch, Sem, PhNo a.Create a SLL of NStudents Data by using front insertion. b.Display the status of SLLand count the number of nodes in it c.Perform Insertion / Deletion at End of SLL d.Perform Insertion / Deletion at Front of SLL(Demonstration of stack) e.Exit		Linked List Characteristics	L4
8	Design, Develop and Implement a menu driven Program in C for the following operations on Doubly Linked List (DLL) of Employee Data with the fields: SSN, Name, Dept, Designation, Sal, PhNo a.Create a DLL of NEmployees Data by using end insertion. b.Display the status of DLL and count the number of nodes in it c.Perform Insertion and Deletion at End of DLL d.Perform Insertion and Deletion at Front of DLL e.Demonstrate how this DLLcan be used as Double Ended Queue f.Exit		Linked List Characteristics	L4
9	Design, Develop and Implement a Program in C for the following operationson Singly Circular Linked List (SCLL) with header nodes a.Represent and Evaluate a Polynomial P(x,y,z) = 6x2y2z-4yz+3x3yz+2xy5z-2xyz3 b.Find the sum of two polynomials POLY1(x,y,z) and POLY2(x,y,z) and store the result in POLYSUM(x,y,z) Support the program with appropriate functions for each of the above operations.		Linked List Characteristics	L4
10	Design, Develop and Implement a menu driven Programin C for the following operations on Binary Search Tree (BST) of Integers a. Create a BST of N Integers: 6, 9, 5, 2, 8, 15, 24, 14, 7, 8, 5, 2 b. Traverse the BST in Inorder, Preorder and Post Order c. Search the BST for a given element (KEY) and report the appropriate message e. Exit		Traversal Method	L4
11	Design, Develop and Implement a Program in C for the following operations onGraph(G) of Cities a.Create a Graph of N cities using Adjacency Matrix. b.Print all the nodes reachable from a given starting node in a digraph using DFS/BFS method		Traversal Method	L4
12	Given a File of N employee records with a set K of Keys(4- digit) which uniquely determine the records in file F. Assume that file Fis maintained in memory by a Hash Table(HT) of m memory locations with Las the set of memory addresses (2- digit) of locations in HT. Let the keys in K and addresses in L are Integers.		File Organization	L4

	Design and develop a Program in C that uses Hash function H: K →→→→L as H(K)=Kmod m (remainder method), and implement hashing technique to map a given key K to the address space L.Resolve the collision (if any) using linear probing.			

3. Lab Material

Unit	Details	Available
1	Text books	
	Higher Engineering Mathematics, B S Grewal, Khanna Publishers, Latest edition, 2015. Title, Author, Publisher, Edition, Publication Year	In Lib
2	Reference books	
	1. Higher Engineering Mathematics, B V Ramana, Tata Mc. Graw Hill, 3 Ed., 2014	In dept
3	Others (Web, Video, Simulation, Notes etc.)	
	A.A .PutAmbekar	Not Available
	PadmaReddy	

4. Lab Prerequisites:

-	-	Base Course:		-	-
SNo	Course Code	Course Name	Topic / Description	Sem	Remarks
1		C Programing	Knowledge on Data Structures	2	

Note: If prerequisites are not taught earlier, GAP in curriculum needs to be addressed. Include in Remarks and implement in B.5.

5. General Instructions

SNo	Instructions	Remarks
1	Observation book and Lab record are compulsory.	
2	Students should report to the concerned lab as per the time table.	
3	After completion of the program, certification of the concerned staff in-charge in the observation book is necessary.	
4	Student should bring a notebook of 100 pages and should enter the readings /observations into the notebook while performing the experiment.	
5	The record of observations along with the detailed experimental procedure of the experiment in the Immediate last session should be submitted and certified staff member in-charge.	
6	Should attempt all problems / assignments given in the list session wise.	
7	It is responsibility to create a separate directory to store all the programs, so that nobody else can read or copy.	
8	When the experiment is completed, should disconnect the setup made by them, and should return all the components/instruments taken for the purpose.	
9	Any damage of the equipment or burn-out components will be viewed seriously either by putting penalty or by dismissing the total group of students from the lab for the semester/year	
10	Completed lab assignments should be submitted in the form of a Lab Record in which you have to write the algorithm, program code along with	

	comments and output for various inputs given	

6. Lab Specific Instructions

SNo	Specific Instructions	Remarks
1	Start computer	
2	Open the turbo c++ window	
3	Select new file.	
4	Write the program	
5	Save the program with .c extension.	
6	Compile the program F9	
7	Execute the program F10	

B. OBE PARAMETERS

1. Lab / Course Outcomes

#	COs	Teach. Hours	Concept	Instr Method	Assessment Method	Blooms' Level
1	Choose the Data manipulation functions for array and strings using memory allocation methods	6	Data Manipulation	Demonstration	Labs	L5
2	Demonstration of stack operations on the expression using stacks	12	Stack Operations	Demonstration	Labs	L5
3	Determine the queue features on the problem using queue methods	3	Queue Features	Demonstration	Labs	L5
4	Compare linked list classification using linked list method	9	Linked List Characteristics	Demonstration	Labs	L5
5	Decide the hierarchical organization of data using binary search tree method	4	Hierarchical Organization	Demonstration	Labs	L5
6	Explain the traversal method on node and edges using graph operation	3	Traversal Method	Demonstration	Labs	L5
7	Importance of file organization on files and records using hash function.	3	File Organization	Demonstration	Labs	L5
-	**Total**	**40**	-	-	-	-

Note: Identify a max of 2 Concepts per unit. Write 1 CO per concept.

2. Lab Applications

SNo	Application Area	CO	Level
1	Analyze the memory allocation method	CO1	L5
2	Code and debug the operations of stack	CO2	L5
3	Demonstrate the working of the data structure in queues	CO3	L5
4	Evaluate the operations of linked list	CO4	L5
5	Analyze hierarchical linear and non linear data-structures	CO5	L5
6	Implement the traversal methods	CO6	L5
7	Evaluate the searching & sorting method by organizing the file structures	CO7	L5

Note: Write 1 or 2 applications per CO.

3. Articulation Matrix

(CO – PO MAPPING)

-	Course Outcomes	Program Outcomes												
#	COs	PO1	PO2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	Level
18CSL38.1	Choose the Data manipulation functions for array and strings	2	-	-	-	-	-	-	-	-	-	-	-	L5

	using memory allocation methods													
18CSL38.2	Demonstration of stack operations on the expression using stacks	-	-	3	2	-	-	-	-	-	-	-	-	L5
18CSL38.3	Determine the queue features on the problem using queue methods	2	-	3	-	-	-	-	-	-	-	-	-	L5
18CSL38.4	Compare linked list classification using linked list method	1	-	2	3	-	-	-	-	-	-	-	-	L5
18CSL38.5	Decide the hierarchical organization of data using binary search tree method	1	2	2	1	-	-	-	-	-	-	-	-	L5
18CSL38.6	Explain the traversal method on node and edges using graph operation	-	-	3	2	-	-	-	-	-	-	-	-	L5
18CSL38.7	Importance of file organization on files and records using hash function.	-	-	3	2	-	-	-	-	-	-	-	-	L5
18CSL38	Average	2	2	3	2									

Note: Mention the mapping strength as 1, 2, or 3

4. Mapping Justification

Mapping		Mapping Level	Justification
CO1	PO1	L5	The knowledge of structure and abstract data type can be applied to solve complex problems.
CO2	PO2	L5	These fundamental concepts of CS can be applied to solve complex problems
CO2	PO3	L5	Efficient algorithms can be designed based on their time complexity.
CO3	PO2	L5	These fundamental concepts of CS can be applied to solve complex problems
CO4	PO4	L5	Analysis of algorithms helps to select suitable algorithms and reach valid conclusions.
CO5	PO1	L5	The knowledge of structure and abstract data type can be applied to solve complex problems.
CO6	PO5	L5	Complexity analysis can be applied in research and other innovative areas.
CO7	PO3	L5	The knowledge about the various data structures can be applied to solve complex engineering problems.
CO8	PO4	L5	This knowledge helps in suitable representations and thereby interpretation of data can be done efficiently
CO9	PO2	L5	These fundamental concepts of CS can be applied to solve complex problems
CO10	PO5	L5	Complexity analysis can be applied in research and other innovative areas.
CO11	PO3	L5	The knowledge about the various data structures can be applied to solve complex engineering problems.
CO12	PO5	L5	Complexity analysis can be applied in research and other innovative areas.

Note: Write justification for each CO-PO mapping.

Table 39: CO-PO Mapping

					Program Outcomes														
	Course Outcomes	B leves	Hrs	CO Attn	PO1	PO2	PO 3	PO 4	PO 5	PO6	PO7	PO8	PO9	PO1 0	PO11	PO12	POS1	POS2	POS3
	18CSL3	L5	6	-	√	-	-	-	-	-	-	-	-	-	-	-	-	-	-

	8.1																		
	18CSL38.2	L5	12	-	-	-	√	√	-	-	-	-	-	-	-	-	-	-	-
	18CSL38.3	L5	3	-	√	-	√	-	-	-	-	-	-	-	-	-	-	-	-
	18CSL38.4	L5	9	-	√	√	√	√	-	-	-	-	-	-	-	-	-	-	-
	18CSL38.5	L5	4	-	√	√	√	√	-	-	-	-	-	-	-	-	-	-	-
	18CSL38.6	L5	3	-	-	-	√	√	-	-	-	-	-	-	-	-	-	-	-
	18CSL38.7	L5	3	-	-	-	√	√	-	-	-	-	-	-	-	-	-	-	-
A	**Sum**	**L5**	**40**	**-**	**4**	**2**	**6**	**5**											
B	**%**	**-**	**100**	**-**	**57**	**29**	**86**	**71**											

5. Curricular Gap and Content

SNo	Gap Topic	Actions Planned	Schedule Planned	Resources Person	PO Mapping
1					
2					
3					
4					
5					

Note: Write Gap topics from A.4 and add others also.

6. Content Beyond Syllabus

SNo	Gap Topic	Actions Planned	Schedule Planned	Resources Person	PO Mapping
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					

Note: Anything not covered above is included here.

C. COURSE ASSESSMENT

1. Course Coverage

Unit	Title	Teaching	No. of question in Exam							CO	Levels
			CIA-1	CIA-2	CIA-3	Asg-1	Asg-2	Asg-3	SEE		

		Hours									
1	Arrays	03	1	-	1	-	-	-	1	CO1	L5
2	Strings	03	1	-	1	-	-	-	1	CO1	L5
3	Stack	03	1	-	1	-	-	-	1	CO2	L5
4	Conversion Of Expressions	03	1	-	1	-	-	-	1	CO2	L5
5a	Evaluation Of Expressions	02	1	-	1	-	-	-	1	CO2	L5
5b	Tower of Hanoi	01	1	-	1	-	-	-	1	CO2	L5
6	Queues	03	1	-	1	-	-	-	1	CO3	L5
7	Singly Linked List	03	-	1	1	-	-	-	1	CO4	L5
8	Doubly Linked List	03	-	1	1	-	-	-	1	CO4	L5
9	Circular Linked List	03	-	1	1	-	-	-	1	CO4	L5
10	Binary Search Tree	03	-	1	1	-	-	-	1	CO5	L5
11	Depth First Search	03	-	1	1	-	-	-	1	CO6	L5
12	Hash Functions	03	-	1	1	-	-	-	1	CO7	L5
-	**Total**	**36**	**6**	**6**	**12**	**0**	**0**	**0**	**12**	**-**	**-**

Note: Write CO based on the theory course.

2. Continuous Internal Assessment (CIA)

Evaluation	Weightage in Marks	CO	Levels
CIA Exam – 1	20	CO1, CO2, CO3	L23, L3
CIA Exam – 2	20	Co4, CO5, CO6, CO7,	L5
CIA Exam – 3	40	CO1, CO2, CO3,Co4, CO5, CO6, CO7	L5
Assignment - 1	00	-	L5
Assignment - 2	00	-	L5
Assignment - 3	00	-	L5
Seminar - 1	00	-	L5
Seminar - 2	00	-	L5
Seminar - 3	00	-	L5
Other Activities – define – Slip test	-	-	-
Final CIA Marks	**40**	**-**	**-**

-

SNo	Description	Marks
1	Observation and Weekly Laboratory Activities	05 Marks
2	Record Writing	15 Marks for each Expt
3	Internal Exam Assessment	20 Marks
4	Internal Assessment	40 Marks
5	SEE	60 Marks
-	**Total**	**100 Marks**

D. EXPERIMENTS

Experiment 01 : Arrays

-	**Experiment No.:**	1	**Marks**		**Date Planned**		**Date Conducted**	
1	Title	Arrays						
2	Course Outcomes	Choose the Data manipulation functions for array and strings using memory allocation methods						
3	Aim	Exercise on memory allocation						
4	Material / Equipment Required	Lab Manual						
5	Theory, Formula, Principle, Concept	Learn data structure classification for array with the memory allocation functions						
6	Procedure, Program, Activity, Algorithm, Pseudo Code	• step 1: start • step 2: write programming • step 3: save the program • step 4: compile • step 5:if error then correct the errors • step 6:run • step 7:stop						
7	Block, Circuit, Model Diagram, Reaction Equation, Expected Graph	• - • - • -						
8	Observation Table, Look-up Table, Output	• SAMPLE OUTPUT: • --------Menu----------- • 1.Create • 2.Display • 3.Insert • 4.Delete • 5.Exit • Enter your choice: • 1 • Enter the size of the array elements: 5 • Enter the elements for the array: • 10 • 20 30 • 40 • SAMPLE OUTPUT: --------Menu----------- • 1.Create • 2.Display • 3.Insert • 4.Delete • 5.Exit • Enter your choice: 1 • Enter the size of the array elements: 5 • Enter the elements for the array: 10 20 30 40 50						
9	Sample Calculations	• - Creating an array • - Displaying an array elements • - Inserting an element in to an array • - Deleting an array element						

10	Graphs, Outputs	• Enter the elements for the array: 10 20 30 40 50
11	Results & Analysis	• - • -
12	Application Areas	• Analyze the memory allocation method
13	Remarks	
14	Faculty Signature with Date	

Experiment 02 : String

-	**Experiment No.:**	2	**Marks**		**Date Planned**		**Date Conducted**	
1	Title	String						
2	Course Outcomes	Choose the Data manipulation functions for array and strings using memory allocation methods						
3	Aim	Choose the String manipulation functions for array and strings using memory allocation methods						
4	Material / Equipment Required	Lab Manual						
5	Theory, Formula, Principle, Concept	Learn data structure organization for strings with the memory allocation functions						
6	Procedure, Program, Activity, Algorithm, Pseudo Code	Step 1: Start. Step 2: Read main string STR, pattern string PAT and replace string REP. Step 3: Search / find the pattern string PAT in the main string STR. Step 4: if PAT is found then replace all occurrences of PAT in main string STR with REP string. Step 5: if PAT is not found give a suitable error message. Step 6: Stop.						
7	Block, Circuit, Model Diagram, Reaction Equation, Expected Graph							
8	Observation Table, Look-up Table, Output	Enter a main string This is Data Structure lab Enter a pattern string Data Structure Enter a replace string Data structure with C The resultant string is This is Data structure with C lab						
9	Sample Calculations	Enter a text String enter pattern String Enter the replacing Sting						
10	Graphs, Outputs	The resultant string is This is Data structure with C lab						
11	Results & Analysis							
12	Application Areas	Analyze the memory allocation method						
13	Remarks							

14	Faculty Signature with Date	

Experiment 03 : Stack

-	**Experiment No.:**	3	**Marks**		**Date Planned**		**Date Conducted**	
1	Title	Stack						
2	Course Outcomes	Demonstration of stack operations on the expression using stacks						
3	Aim	Implementation of stack operations						
4	Material / Equipment Required	Lab Manual						
5	Theory, Formula, Principle, Concept	Push Operations Pop Operations Display Operations						
6	Procedure, Program, Activity, Algorithm, Pseudo Code	Step 1: Start. Step 2: Initialize stack size MAX and top of stack -1. Step 3: Push integer element on to stack and display the contents of the stack. if stack is full give a message as 'Stack is Overflow'. Step 3: Pop element from stack along with display the stack contents. if stack is empty give a message as 'Stack is Underflow'. Step 4: Check whether the stack contents are Palindrome or not. Step 5: Stop.						
7	Block, Circuit, Model Diagram, Reaction Equation, Expected Graph	 						
8	Observation Table, Look-up Table, Output	----MAIN MENU---- 1. PUSH (Insert) in the Stack 2. POP (Delete) from the Stack 3. PALINDROME check using Stack 4. Exit (End the Execution) Enter Your Choice: 1 Enter an element to be pushed: 1 The stack contents are: ----- | 1 |						
9	Sample Calculations	Pushing the elements Poping the elements Checking the stack content form Palindrome Check overflow and underflow conditions						
10	Graphs, Outputs	----MAIN MENU---- 1. PUSH (Insert) in the Stack 2. POP (Delete) from the Stack 3. PALINDROME check using Stack 4. Exit (End the Execution) Enter Your Choice: 1 Enter an element to be pushed: 1 The stack contents are: ----- | 1 |						
11	Results & Analysis							
12	Application Areas	Code and debug the operations of stack						
13	Remarks							

14	Faculty Signature with Date	

Experiment 04 : Conversion Of Expression

-	**Experiment No.:**	4	**Marks**		**Date Planned**		**Date Conducted**	
1	Title	Conversion Of Expression						
2	Course Outcomes	Demonstration of stack operations on the expression using stacks						
3	Aim	Exercise on Keywords and identifiers						
4	Material / Equipment Required	Lab Manual						
5	Theory, Formula, Principle, Concept	Identify infix ,postfix,prefix Expressions						
6	Procedure, Program, Activity, Algorithm, Pseudo Code	Step 1: Start. Step 2: Read an infix expression with parenthesis and without parenthesis. Step 3: convert the infix expression to postfix expression. Step 4: Stop						
7	Block, Circuit, Model Diagram, Reaction Equation, Expected Graph							
8	Observation Table, Look-up Table, Output	Enter a valid infix expression (a+(b-c)*d) The infix expression is: (a+(b-c)*d) The postfix expression is: abc-d*+						
9	Sample Calculations	Precedence calculation comparing input character with the stack top character						
10	Graphs, Outputs	Enter a valid infix expression (a+(b-c)*d) The infix expression is: (a+(b-c)*d) The postfix expression is: abc-d*+						
11	Results & Analysis							
12	Application Areas	Code and debug the operations of stack						
13	Remarks							
14	Faculty Signature with Date							

Experiment 05 a: Evaluation of expressions

-	**Experiment No.:**	5a	**Marks**		**Date Planned**		**Date Conducted**	
1	Title	Evaluation of expressions						
2	Course Outcomes	Demonstration of stack operations on the expression using stacks						

3	Aim	Evaluate the Suffix Expression using stack operations
4	Material / Equipment Required	Lab Manual
5	Theory, Formula, Principle, Concept	Evaluate the suffix Expression with single digit operands and operators +,-,*,/,%,^
6	Procedure, Program, Activity, Algorithm, Pseudo Code	Step 1: Start. Step 2: Read the postfix/suffix expression. Step 3: Evaluate the postfix expression based on the precedence of the operator. Step 4: Stop.
7	Block, Circuit, Model Diagram, Reaction Equation, Expected Graph	
8	Observation Table, Look-up Table, Output	Enter the postfix expression: 23+ The result is: 5.000000
9	Sample Calculations	Computations of the operands with stack top elements with the operators
10	Graphs, Outputs	Enter the postfix expression: 23+ The result is: 5.000000
11	Results & Analysis	
12	Application Areas	Demonstrate the working of the data structure in queues
13	Remarks	
14	Faculty Signature with Date	

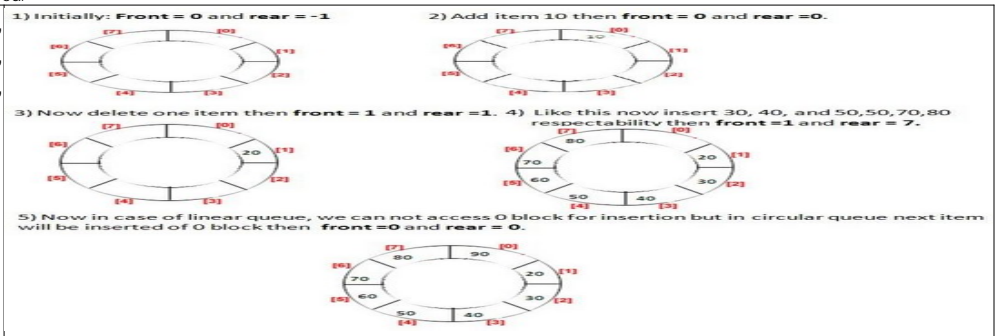
Experiment 05 b: Tower of Hanoi

-	**Experiment No.:**	5b	**Marks**		**Date Planned**		**Date Conducted**	
1	Title	Tower of Hanoi						
2	Course Outcomes	Demonstration of stack operations on the expression using stacks						
3	Aim	Moving the disk from first peg to third peg using auxilary peg						
4	Material / Equipment Required	Lab Manual						
5	Theory, Formula, Principle, Concept	Perform the tower of hanoi using recursion method						
6	Procedure, Program, Activity, Algorithm, Pseudo Code	Step 1: Start. Step 2: Read N number of discs. Step 3: Move all the discs from source to destination by using temp rod. Step 4: Stop.						
7	Block, Circuit, Model Diagram, Reaction Equation, Expected Graph							
8	Observation Table,	Enter the number of discs:						

	Look-up Table, Output	3 Move disc 1 from A to C Move disc 2 from A to B Move disc 1 from C to B Move disc 3 from A to C Move disc 1 from B to A Move disc 2 from B to C Move disc 1 from A to C Total Number of moves are: 7"
9	Sample Calculations	2^n-1 where n l number of disk
10	Graphs, Outputs	Enter the number of discs: 3 Move disc 1 from A to C Move disc 2 from A to B Move disc 1 from C to B Move disc 3 from A to C Move disc 1 from B to A Move disc 2 from B to C Move disc 1 from A to C Total Number of moves are: 7"
11	Results & Analysis	
12	Application Areas	Demonstrate the working of the Tower of Hanoi
13	Remarks	
14	Faculty Signature with Date	

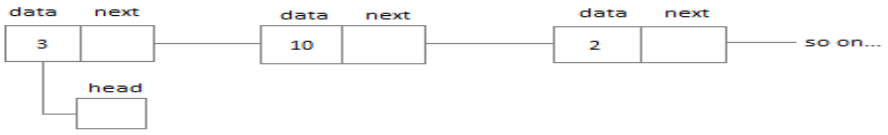
Experiment 06: Circular queues

-	**Experiment No.:**	6	**Marks**		**Date Planned**		**Date Conducted**	
1	Title	Circular queues						
2	Course Outcomes	Determine the queue features on the problem using queue methods						
3	Aim	Circular Queue Implementation						
4	Material / Equipment Required	Lab Manual						
5	Theory, Formula, Principle, Concept	Array Implementation of Queue with Maximum size						
6	Procedure, Program, Activity, Algorithm, Pseudo Code	Step 1: Start. Step 2: Initialize queue size to MAX. Step 3: Insert the elements into circular queue. If queue is full give a message as 'queue is overflow" Step 4: Delete an element from the circular queue. If queue is empty give a message as 'queue is underflow'. Step 5: Display the contents of the queue. Step 6: Stop.						

7	Block, Circuit, Model Diagram, Reaction Equation, Expected Graph	
8	Observation Table, Look-up Table, Output	1. Insert 2. Delete 3. Display 4. Exit Enter the choice: 1 Enter the character / item to be inserted: A 1. Insert 2. Delete 3. Display 4. Exit Enter the choice: 1
9	Sample Calculations	Insertion of elements Deletion of element
10	Graphs, Outputs	1. Insert 2. Delete 3. Display 4. Exit Enter the choice: 1 Enter the character / item to be inserted: A 1. Insert 2. Delete 3. Display 4. Exit Enter the choice: 1
11	Results & Analysis	
12	Application Areas	Demonstrate the working of the data structure in queues
13	Remarks	
14	Faculty Signature with Date	

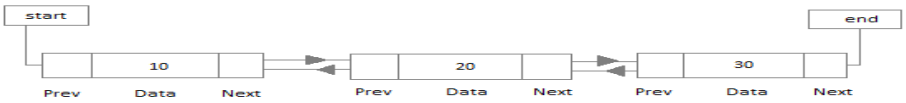
Experiment 07: Singly Linked List

-	**Experiment No.:**	7	**Marks**		**Date Planned**		**Date Conducted**	
1	Title	Singly Linked List						
2	Course Outcomes	Compare linked list classification using linked list method						
3	Aim	Singly linked list implementation						
4	Material / Equipment Required	Lab Manual						
5	Theory, Formula, Principle, Concept	Implement Menu driven with student data						
6	Procedure, Program, Activity, Algorithm, Pseudo Code	Step 1: Start. Step 2: Read the value of N. (N student's information) Step 2: Create a singly linked list. (SLL) Step 3: Display the status of SLL. Step 4: Count the number of nodes. Step 5: Perform insertion at front of list. Step 6: Perform deletion at the front of the list. Step 7: Perform insertion at end of the list. Step 8: Perform deletion at the end of the list. Step 9: Demonstrate how singly linked list can be used as stack. Step 10: Demonstrate how singly linked list can be used as queue. Step 11: Stop.						

7	Block, Circuit, Model Diagram, Reaction Equation, Expected Graph	
8	Observation Table, Look-up Table, Output	1. Create 2. Display 3. Insert 4. Delete 5. Stack 6.Queue 7. Exit Enter your choice: 1 How many student data you want to create: 2 Enter USN, Name, Branch, Sem, Ph.No 1kt12cs001 kumar cs 3 9900099000 Enter USN, Name, Branch, Sem, Ph.No 1kt12is002 ravi is 3 9900099111
9	Sample Calculations	Create front insertion status informations deletion at end and front
10	Graphs, Outputs	1. Create 2. Display 3. Insert 4. Delete 5. Stack 6.Queue 7. Exit Enter your choice: 1 How many student data you want to create: 2 Enter USN, Name, Branch, Sem, Ph.No 1kt12cs001 kumar cs 3 9900099000 Enter USN, Name, Branch, Sem, Ph.No 1kt12is002 ravi is 3 9900099111
11	Results & Analysis	
12	Application Areas	Evaluate the operations of linked list
13	Remarks	
14	Faculty Signature with Date	

Experiment 08: Doubly Linked List

-	**Experiment No.:**	8	**Marks**		**Date Planned**		**Date Conducted**	
1	Title	Doubly Linked List						
2	Course Outcomes	Compare linked list classification using linked list method						
3	Aim	Implementation of Doubly linked list						
4	Material / Equipment Required	Lab Manual						
5	Theory, Formula, Principle, Concept	Menu driven Employee data storage						
6	Procedure, Program, Activity, Algorithm, Pseudo Code	Step 1: Start. Step 2: Read the value of N. (N student's information) Step 3: Create a doubly linked list. (DLL) Step 4: Display the status of DLL. Step 5: Count the number of nodes. Step 6: Perform insertion at front of list. Step 7: Perform deletion at the front of the list. Step 8: Perform insertion at end of the list. Step 9: Perform deletion at the end of the list. Step 10: Demonstrate how doubly linked list can be used as double ended queue. Step 11: Stop						

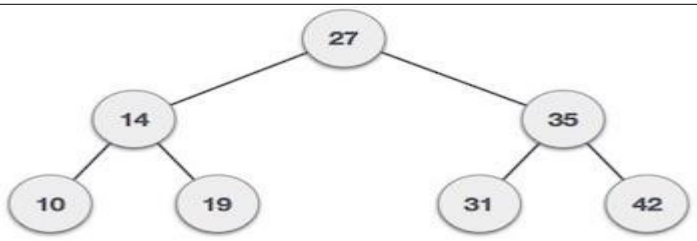
7	Block, Circuit, Model Diagram, Reaction Equation, Expected Graph	
8	Observation Table, Look-up Table, Output	1. Create 2. Display 3. Insert 4. Delete 5. Queue 7. Exit Enter your choice: 1 How many employees data you want to create: 2 Enter SSN, Name, Dept, Designation, Sal, Ph.No 1 KUMAR CSE INSTRUCTOR 8000 900099000
9	Sample Calculations	Create front insertion status informations deletion at end and front
10	Graphs, Outputs	1. Create 2. Display 3. Insert 4. Delete 5. Queue 7. Exit Enter your choice: 1 How many employees data you want to create: 2 Enter SSN, Name, Dept, Designation, Sal, Ph.No 1 KUMAR CSE INSTRUCTOR 8000 900099000
11	Results & Analysis	
12	Application Areas	Evaluate the operations of linked list
13	Remarks	
14	Faculty Signature with Date	

Experiment 09: Circular Linked List

-	**Experiment No.:**	9	**Marks**		**Date Planned**		**Date Conducted**	
1	Title	Circular Linked List						
2	Course Outcomes	Compare linked list classification using linked list method						
3	Aim	Evaluation of polynomial expressions using Circular linked list						
4	Material / Equipment Required	Lab Manual						
5	Theory, Formula, Principle, Concept	Represent and Evaluate polynomial expression						
6	Procedure, Program, Activity, Algorithm, Pseudo Code	Step 1: Start. Step 2: Read a polynomial. Step 3: Represent the polynomial using singly circular linked list. Step 3: Evaluate the given polynomial Step 4: Read two polynomials and find the sum of the polynomials. Step 5: Stop						
7	Block, Circuit, Model Diagram, Reaction Equation, Expected Graph	**1) Write in Standard form** $(3y^5 + y^4 + 2y^3 - 2y + 5) + (2y^5 + 3y^3 + 7y + 2)$ **2) Arrange in columns of like terms and then add** © mathwarehouse.com $3y^5 + y^4 + 2y^3 - 2y + 5$ $2y^5 + 3y^3 + 7y + 2$ $5y^5 + y^4 + 5y^3 + 5y + 7$						
8	Observation Table, Look-up Table, Output	1. Evaluate polynomial P(x,y,z) = 6x2y2z-4yz5+3x3yz+2xy5z-2xyz3 2. Add two polynomials 3. Exit Enter your choice: 1						

		Enter polynomial to evaluate: Enter coeff: 6 Enter x, y, z powers (0-indiacate NO term: 2 2 1 If you wish to continue press 1 otherwise 0: 1 Enter coeff: -4
9	Sample Calculations	P(x,y,z) = 6x2y2z-4yz5+3x3yz+2xy5z-2xyz3
10	Graphs, Outputs	1. Evaluate polynomial P(x,y,z) = 6x2y2z-4yz5+3x3yz+2xy5z-2xyz3 2. Add two polynomials 3. Exit Enter your choice: 1 Enter polynomial to evaluate: Enter coeff: 6 Enter x, y, z powers (0-indiacate NO term: 2 2 1 If you wish to continue press 1 otherwise 0: 1 Enter coeff: -4
11	Results & Analysis	
12	Application Areas	Evaluate the operations of linked list
13	Remarks	
14	Faculty Signature with Date	

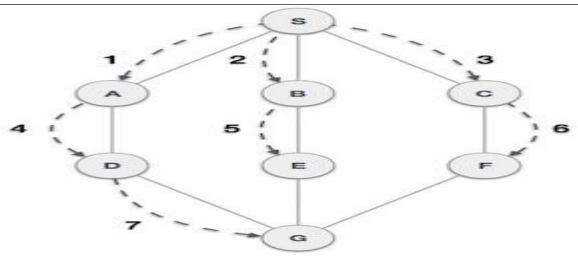
Experiment 10: Binary Search Tree

-	**Experiment No.:**	10	**Marks**		**Date Planned**		**Date Conducted**	
1	Title	Binary Search Tree						
2	Course Outcomes	Decide the hierarchical organization of data using binary search tree method						
3	Aim	Implementing Tree operation using Binary Search tree						
4	Material / Equipment Required	Manual						
5	Theory, Formula, Principle, Concept	Travers the Binary tree in inorder,preorder and post order						
6	Procedure, Program, Activity, Algorithm, Pseudo Code	Step 1: Start. Step 2: Create a Binary Search Tree for N elements. Step 3: Traverse the tree in inorder. Step 4: Traverse the tree in preorder Step 6: Traverse the tree in postorder. Step 7: Search the given key element in the BST. Step 8: Delete an element from BST. Step 9: Stop						
7	Block, Circuit, Model Diagram, Reaction Equation, Expected Graph							
8	Observation Table, Look-up Table, Output	1. Insertion in Binary Search Tree 2. Delete Element in Binary Search Tree 3. Inorder 4. Preorder 5. Postorder						

		6. Exit Enter your choice: 1 Enter N value: 12 Enter the values to create BST like(6,9,5,2,8,15,24,14,7,8,5,2) 6 9 5 2 8 15 24 14 7 8 5 2 1. Insertion in Binary Search Tree 2. Delete Element in Binary Search Tree 3. Inorder 4. Preorder 5. Postorder 6. Exit Enter your choice: 3
9	Sample Calculations	left_subtree (keys) ≤ node (key) ≤ right_subtree (keys)
10	Graphs, Outputs	1. Insertion in Binary Search Tree 2. Delete Element in Binary Search Tree 3. Inorder 4. Preorder 5. Postorder 6. Exit Enter your choice: 1 Enter N value: 12 Enter the values to create BST like(6,9,5,2,8,15,24,14,7,8,5,2) 6 9 5 2 8 15 24 14 7 8 5 2 1. Insertion in Binary Search Tree 2. Delete Element in Binary Search Tree 3. Inorder 4. Preorder 5. Postorder 6. Exit Enter your choice: 3
11	Results & Analysis	
12	Application Areas	Analyze hierarchical linear and non linear data-structures

13	Remarks	
14	Faculty Signature with Date	

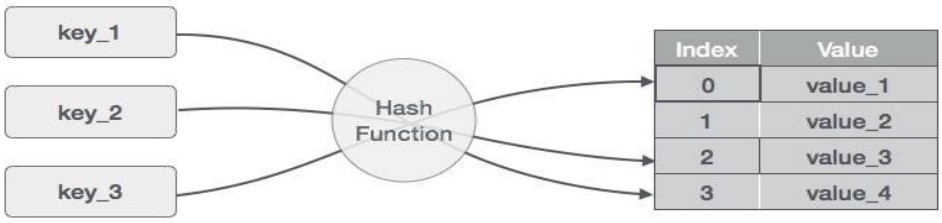
Experiment 11: Breadth First Search

-	**Experiment No.:**	11	**Marks**		**Date Planned**		**Date Conducted**	
1	Title	Breadth First Search						
2	Course Outcomes	Explain the traversal method on node and edges using graph operation						
3	Aim	Traverse the graph using breadth first search methods						
4	Material / Equipment Required	Manual						
5	Theory, Formula, Principle, Concept	A graph G = (V, E) where v= {0, 1, 2, . . .n-1} can be represented using two dimensional integer array of size n x n						
6	Procedure, Program, Activity, Algorithm, Pseudo Code	Step 1: Start. Step 2: Input the value of N nodes of the graph Step 3: Create a graph of N nodes using adjacency matrix representation. Step 3: Print the nodes reachable from the starting node using BFS. Step 4: Check whether graph is connected or not using DFS. Step 5: Stop.						
7	Block, Circuit, Model Diagram, Reaction Equation, Expected Graph							
8	Observation Table, Look-up Table, Output	1. Create Graph 2.BFS Enter your choice: 1 3.Exit Enter the number of vertices of the digraph: 4 Enter the adjacency matrix of the graph: 0 0 1 1 0 0 0 0 0 1 0 0 0 1 0 0 1. Create Graph 2.BFS						

		Enter your choice: 2 3.Exit Enter the source vertex to find other nodes reachable or not: 1 3 4 2
9	Sample Calculations	Initially all vertices are unvisited. DFS starts in arbitrary vertex and runs as follows: ﾌﾑ Mark vertex u as visited. ﾌﾑ For each edge (u, v), where u is unvisited , run depth-first search for u recursively. ﾌﾑ Mark vertex u as DFS has finished processing the vertex. and backtrack to the parent.
10	Graphs, Outputs	1. Create Graph 2.BFS Enter your choice: 1 3.Exit Enter the number of vertices of the digraph: 4 Enter the adjacency matrix of the graph: 0 0 1 1 0 0 0 0 0 1 0 0 0 1 0 0 1. Create Graph 2.BFS Enter your choice: 2 3.Exit Enter the source vertex to find other nodes reachable or not: 1 3 4 2
11	Results & Analysis	
12	Application Areas	Implement the traversal methods
13	Remarks	
14	Faculty Signature with Date	

Experiment 12: Hashing Functions

-	**Experiment No.:**	12	**Marks**		**Date Planned**		**Date Conducted**	
1	Title	Hashing Functions						
2	Course Outcomes	Importance of file organization on files and records using hash function.						

3	Aim	Organizing the employee records in a hash table by setting Keys
4	Material / Equipment Required	Lab Manual
5	Theory, Formula, Principle, Concept	Hash Table is a data structure which store data in associative manner. In hash table, data is stored in array format where each data values has its own unique index value.
6	Procedure, Program, Activity, Algorithm, Pseudo Code	Step 1: Start. Step 2: Given a File of N employee records with a set K of Keys (4-digit) which uniquely determine the records in file F. Step 3: Assume that file F is maintained in memory by a Hash Table(HT) of m memory locations with L as the set of memory addresses (2-digit) of locations in HT. Step 3: Let the keys in K and addresses in L are Integers Step 4: Hash function H: K ®L as H(K)=K mod m (remainder method) Step 5: Hashing as to map a given key K to the address space L, Resolve the
7	Block, Circuit, Model Diagram, Reaction Equation, Expected Graph	
8	Observation Table, Look-up Table, Output	Enter the data: 2 Enter emp id: 100 Enter emp name: Anand Do you wish to continue? (1/0): Enter the data: 4 Enter emp id: 101 Enter emp name: Kumar Do you wish to continue? (1/0): 1 0 1.Display ALL 2.Filtered Display Enter the choice: 1 The hash table is: HTKey EmpID 0 0 1 0 2 100 3 0 4 101 5

		0 6 0 7 0 8 0 9 0 EmpName Anand Kumar
9	Sample Calculations	(1,20) (2,70) (42,80) (4,25) (12,44) (14,32) (17,11) (13,78) (37,98) S.n. Key Hash Array Index 1 1 1 % 20 = 1 1 2 2 2 % 20 = 2 2 3 42 42 % 20 = 2 2
10	Graphs, Outputs	Enter the data: 2 Enter emp id: 100 Enter emp name: Anand Do you wish to continue? (1/0): Enter the data: 4 Enter emp id: 101 Enter emp name: Kumar Do you wish to continue? (1/0): 1 0 1.Display ALL 2.Filtered Display Enter the choice: 1 The hash table is: HTKey EmpID 0 0 1 0 2 100 3 0 4 101 5 0 6 0 7 0 8 0 9 0 EmpName Anand

		Kumar
11	Results & Analysis	
12	Application Areas	Evaluate the searching & sorting method by organizing the file structures
13	Remarks	
14	Faculty Signature with Date	