

<p>6. a) Design and implement code converter I) Binary to Gray (II) Gray to Binary Code using basic gates.</p> <p>7. Design and verify the Truth Table of 3-bit Parity Generator and 4-bit Parity Checker using basic Logic Gates with an even parity bit.</p> <p>8. a) Realize a J-K Master / Slave Flip-Flop using NAND gates and verify its truth table.</p> <p>b) Design and develop the Verilog / VHDL code for D Flip-Flop with positive-edge triggering. Simulate and verify it's working.</p> <p>9. a) Design and implement a mod-n ($n < 8$) synchronous up counter using J-K Flip-Flop ICs and demonstrate its working.</p> <p>b) Design and develop the Verilog / VHDL code for mod-8 up counter. Simulate and verify it's working.</p> <p>10. Design and implement an asynchronous counter using decade counter IC to count up from 0 to n ($n \leq 9$) and demonstrate on 7-segment display (using IC- 7447).</p> <p>11. Generate a Ramp output waveform using DAC0800 (Inputs are given to DAC through IC74393 dual 4-bit binary counter).</p> <p>Study experiment</p> <p>12. To study 4-bit ALU using IC-74181.</p>
--

<p>Course outcomes:</p> <p>On the completion of this laboratory course, the students will be able to:</p> <ul style="list-style-type: none"> • Demonstrate various Electronic Devices like Cathode ray Oscilloscope, Signal generators, Digital Trainer Kit, Multimeters and components like Resistors, Capacitors, Op amp and Integrated Circuit. • Design and demonstrate various combinational logic circuits. • Design and demonstrate various types of counters and Registers using Flip-flops • Make use of simulation package to design circuits. • Infer the working and implementation of ALU.
<p>Conduction of Practical Examination:</p> <ol style="list-style-type: none"> 1 . All laboratory experiments (1 to 11 nos) are to be included for practical examination. 2 . Students are allowed to pick one experiment from the lot. 3 . Strictly follow the instructions as printed on the cover page of answer script. 4 . Marks distribution: <ol style="list-style-type: none"> a) For questions having part a only- Procedure + Conduction + Viva: 15 + 70 +15 =100 Marks b) For questions having part a and b <p>Part a- Procedure + Conduction + Viva: 09 + 42 +09= 60 Marks</p> <p>Part b- Procedure + Conduction + Viva: 06 + 28 +06= 40 Marks</p> 5 . Change of experiment is allowed only once and marks allotted to the procedure part to be made zero.

(Effective from the academic year 2017 -2018) SEMESTER - III			
Laboratory Code	17CSL38	IA Marks	40
Number of Lecture Hours/Week	01I + 02P	Exam Marks	60
Total Number of Lecture Hours	40	Exam Hours	03
CREDITS - 02			
Descriptions (if any) Implement all the experiments in C Language under Linux / Windows environment.			
Laboratory Experiments: <ol style="list-style-type: none"> Design, Develop and Implement a menu driven Program in C for the following Array operations <ol style="list-style-type: none"> Creating an Array of N Integer Elements Display of Array Elements with Suitable Headings Inserting an Element (ELEM) at a given valid Position (POS) Deleting an Element at a given valid Position(POS) Exit. Support the program with functions for each of the above operations. Design, Develop and Implement a Program in C for the following operations on Strings <ol style="list-style-type: none"> Read a main String (STR), a Pattern String (PAT) and a Replace String (REP) 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. Design, Develop and Implement a menu driven Program in C for the following operations on STACK of Integers (Array Implementation of Stack with maximum size MAX) <ol style="list-style-type: none"> Push an Element on to Stack Pop an Element from Stack Demonstrate how Stack can be used to check Palindrome Demonstrate Overflow and Underflow situations on Stack Display the status of Stack Exit Support the program with appropriate functions for each of the above operations 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 alphanumeric operands. Design, Develop and Implement a Program in C for the following Stack Applications <ol style="list-style-type: none"> Evaluation of Suffix expression with single digit operands and operators: +, -, *, /, %, ^ Solving Tower of Hanoi problem with n disks 			

6. Design, Develop and Implement a menu driven Program in C for the following operations on **Circular QUEUE** of Characters (Array Implementation of Queue with maximum size **MAX**)
- Insert an Element on to Circular QUEUE
 - Delete an Element from Circular QUEUE
 - Demonstrate **Overflow** and **Underflow** situations on Circular QUEUE
 - Display the status of Circular QUEUE
 - Exit

Support the program with appropriate functions for each of the above operations

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**
- Create a **SLL** of **N** Students Data by using **front insertion**.
 - Display the status of **SLL** and count the number of nodes in it
 - Perform Insertion / Deletion at End of **SLL**
 - Perform Insertion / Deletion at Front of **SLL**(**Demonstration of stack**)
 - Exit

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**
- Create a **DLL** of **N** Employees Data by using **end insertion**.
 - Display the status of **DLL** and count the number of nodes in it
 - Perform Insertion and Deletion at End of **DLL**
 - Perform Insertion and Deletion at Front of **DLL**
 - Demonstrate how this **DLL** can be used as **Double Ended Queue**
 - Exit

9. Design, Develop and Implement a Program in C for the following operations on **Singly Circular Linked List (SCLL)** with header nodes
- Represent and Evaluate a Polynomial $P(x,y,z) = 6x^2y^2z - 4yz^5 + 3x^3yz + 2xy^5z - 2xyz^3$
 - 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

10. Design, Develop and Implement a menu driven Program in C for the following operations on **Binary Search Tree (BST)** of Integers
- Create a BST of **N** Integers: 6, 9, 5, 2, 8, 15, 24, 14, 7, 8, 5, 2
 - Traverse the BST in Inorder, Preorder and Post Order
 - Search the BST for a given element (**KEY**) and report the appropriate message
 - Exit

11. Design, Develop and Implement a Program in C for the following operations on **Graph(G)** of Cities
- Create a Graph of **N** cities using Adjacency Matrix.
 - Print all the nodes **reachable** from a given starting node in a digraph using DFS/BFS method

<p>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 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. Let the keys in K and addresses in L are Integers. Design and develop a Program in C that uses Hash function H: K →L as $H(K)=K \text{ mod } m$ (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.</p>
<p>Course outcomes: On the completion of this laboratory course, the students will be able to:</p> <ul style="list-style-type: none"> • Analyze and Compare various linear and non-linear data structures • Demonstrate the working nature of different types of data structures and their applications • Develop, analyze and evaluate the searching and sorting algorithms • Choose the appropriate data structure for solving real world problems
<p>Conduction of Practical Examination:</p> <ol style="list-style-type: none"> 1. All laboratory experiments (TWELVE nos) are to be included for practical examination. 2. Students are allowed to pick one experiment from the lot. 3. Strictly follow the instructions as printed on the cover page of answer script 4. Marks distribution: Procedure + Conduction + Viva:15 + 70 +15 (100) 5. Change of experiment is allowed only once and marks allotted to the procedure part to be made zero.