

COMPUTING LAB ASSIGNMENTS

SUBJECT: INTRODUCTION TO COMPUTING LAB (ES-CS-291)
STREAM: CSE,IT 2ND SEMESTER (2021)

Session No.	List of Lab Assignments
1 Operators & Expression I	1. WACP to print a message on the Screen. 2. WACP to add two numbers. 3. WACP to read a character and print its ASCII value. 4. WACP to calculate Simple and Compound interest. 5. WACP to swap two integer numbers without using 3 rd Variable. 6. WACP to swap two integer numbers using 3 rd Variable. 7. WACP to find the area and circumference of a circle. (value of radius must be given by user and use the concept of symbolic constants)
2 Operators & Expression II	8. WACP to print the sizes and range of various data types. 9. WACP to calculate sum of digits of a five digit numbers. (do not use any loop construct) 10. WACP to find the average of five numbers 11. WACP to convert temperature in degree F to degree C and vice versa 12. WACP to read in the age of a person as an integer number in years and print the three immediate integral range of the age entered. For example if the user enters 25, then the output will print the three immediate integral ranges: 24 to 26, 23 to 27 and 22 to 28. 13. WACP to swap two integer numbers using bitwise operators.
3 Flow of Control I	14. WACP to check whether a number is odd or even. (using simple if statement) 15. WACP to find the roots of a Quadratic equation. (using if else statement) 16. WACP to determine whether the character entered is a capital, small case letter, a digit or a special symbol. (using else if ladder) 17. WACP to add, subtract, multiply and divide two numbers using switch case. 18. Any year is entered through keyboard. WACP to determine whether the year is a leap year or not. (using conditional operators). 19. Find greatest of three numbers.(using ternary operator).
4 Flow of Control II	20. WACP to calculate the sum of first n odd numbers. (using while loop) 21. WACP to calculate factorial of a number without using recursion. (using do while loop) 22. WACP to print the fibonacci series without using recursion. (using for loop) 23. WACP to find whether a number is Armstrong or not. (using while and if statement) 24. WACP to check whether a number is palindrome or not. (using while and if statement) 25. WACP to find whether a number is prime or not. (using for and if statements) 26. WACP to print the first n integer numbers divisible by 7(using for and if statement) 27. WACP to print the first n integer numbers not divisible by 7(using for and if statement)

	Hint: Inputs to the program are lower limit(A), upper limit(B) and the number of trapezoids
7 Arrays	37. WACP to multiply two matrices and then find the transpose of the resultant matrix. (use separate functions to multiply two matrices and to find the transpose) 38. WACP to reverse a given string (of characters). 39. Write your own strcpy() and strcat() functions in C 40. WACP to delete all vowels from a sentence. 41. Implement Binary Search and Linear Search 42. Implement Bubble sort. 43. Implement Insertion sort. 44. Implement Selection sort. 45. WACP to illustrate passing of an array of strings(names) to a function. 46. WACP to find the sum of all integers that lie between 1 and n and are divisible by 7.
Session No.	List of Lab Assignments
8 Pointers	47. Write a C function to swap two numbers using pointers. 48. WACP which will read a text and count all occurrence of a particular word. 49. Write a C function to find the largest of three numbers using pass by reference. The function to find the largest should return a pointer. 50. Write your own strcpy() and strlen () functions in C 51. WACP using pointers to read in an array of integers and print its elements in reverse order. 52. WACP to sort strings abc, deh, bac in ascending order using array of pointers. 53. Write a program that will read C source code from the standard input and ensure that the braces are paired correctly.Note: You need not worry about braces that appear within comments, string literals, or character constants.
9 Recursion	54. WACP using a recursive function to print the fibonacci series. 55. WACP using a recursive function to find the factorial of a number. 56. WACP using a recursive function to print the non-fibonacci series. 57. Implement quick sort using recursion. 58. Implement Merge sort using recursion.
10 Structures and Unions	59. WACP that can maintain the name, roll, number and marks of a class of students. The size of the class is variable. Include function to compute the average marks of the class. 60. Define a structure called "cricket" that will describe the following information - Player name Team name Batting average using "cricket" declare an array "player" with 50 elements and write a program to read the information about all the 50 players and print a team wise list containing names of players with their batting average.
11 File Handling & Macros	61. WACP that will receive a file name and a line of text as command line arguments and write the text to the file. 62. WACP to remove all comments from a C file. 63. WACP to remove all blank from a C file. 64. WACP to copy the contents of one file into another. 65. Write a nested macro that gives the minimum of three values. 66. Define a macro "Print-Value" which can be used to print two values of arbitrary type.

12
EXTRA

67. WACP In England the currency is made up of pound, £, and pence, p, and there are eight coins in general circulation:

1p, 2p, 5p, 10p, 20p, 50p, £1 (100p) and £2 (200p).

It is possible to make £2 in the following way:

$$1 \times £1 + 1 \times 50p + 2 \times 20p + 1 \times 5p + 1 \times 2p + 3 \times 1p$$

How many different ways can £2 be made using any number of coins?

68. Given an array A[] consisting 0's, 1's and 2's, give an algorithm for sorting A[].The algorithm should put all 0's first, then all 1's and all 2's in the last.(using single for or while loop(O(n)))

Input={0,1,1,0,1,2,1,2,0,0,1} Output={0,0,0,0,1,1,1,1,2,2}

69. Given an array of positive integers, all numbers occurs even number of times except one number which occurs odd number of times. Find the number in O(n) time and in constant space.

70. Find the kth smallest element in an array S of n-elements in best possible way.(O(n)).