

Problem Solving using Computers Lab (B. Tech. I Sem)

LIST OF EXPERIMENTS

Lab 1. Sequential Instruction Based C Programs

1. Program to Print the Message “Hello” on the Screen.
2. Write a program to take an input of two integer numbers and print the sum of that numbers.
3. Convert the time in seconds to hours, minutes and seconds. (1 hr =3600 sec).
4. Find the sum of the digits of a four-digit number (ex 1234 sum=10) (without using a loop).
5. Convert temperature given in Fahrenheit to Centigrade and Centigrade to Fahrenheit. Hint: $C=5/9(F-32)$.
6. Converting distance in mm to cm, inch, feet (1 cm =10mm, 1inch=2.5cm, 1 feet =12 inches).
7. Find out the distance between two points e.g. (x1, y1) and (x2, y2).
Hint: $Distance=\sqrt{(x2-x1)^2+(y2-y1)^2}$
8. Evaluate the area of the circle $Area = \pi * R^2$
9. Interchange values of two variables using a third variable.
10. Interchange values of two variables without using a third variable.

Lab 2. Control Structures: If, If-Else and Switch-Case statements

1. Check whether the given number is odd or even.
2. Check whether a given year is a leap year or not.
3. Write a program to take two numbers as an input and find whether one number is multiple of other or not.
4. Write a program that returns a letter grade based on a quiz score. The input will be the integer score from a ten-point quiz.
 - a. The letter grades are assigned by:
 - b. 9 - 10 “A” 7 - 8 “B” 5 - 6 “C” 3 - 4 “D” < 3 “F”
5. Write a program to check whether given character is vowel, consonant or digit.

Control Structures: Switch-Case

1. Program to calculate an area of a circle, a rectangle or a triangle depending on user's choice.
2. Write a program to design a calculator that performs addition, subtraction, minus and division operation. This program inputs two operands and an operator and then displays the calculated results.
3. Write a program to calculate a bill of internet browsing. The conditions are given below:
Minimum Rs. 200 for up to 100 calls
Plus, Rs. 0.60 per call for next 50 calls.

Plus, Rs. 0.50 per call for next 50 calls.

Plus, Rs. 0.40 per call for any call beyond 200 calls

Lab 3. Control Structures: Loops and Nested Loops

1. Write a program to print the sum of N natural numbers.
2. Write a program to take N as input print the odd numbers in descending order.
3. Write a program to print the Nth Fibonacci number.

Hint: (Fibonacci series is 0, 1, 1, 2, 3, 5, 8)

4. Find whether given number is prime or not.
5. Convert the decimal number into binary to decimal.

Ex: $1101 = 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 13$

6. Reverse a given number

Ex: $1234 \text{ reverse} = 4 \cdot 10^3 + 3 \cdot 10^2 + 2 \cdot 10^1 + 1 \cdot 10^0 = 4321$

7. Check whether given number is Armstrong or not. An Armstrong number of three digits is an integer such that the sum of the cubes of its digits is equal to the number itself. For example, 371 is an Armstrong number since $3^3 + 7^3 + 1^3 = 371$.
8. Print the Fibonacci numbers that fall in given range.
9. Print the prime numbers that fall in given range.

Lab 4. 1-D and 2-D Array

1. Find whether given number is available in an array or not (Linear Search).
2. Find the largest and smallest element in an array.
3. Find the number of positive numbers, negative numbers, odd numbers, even numbers and number of 0 of an array.
4. Reverse an array with an auxiliary array.
5. Arrange the elements of an array in ascending order by bubble sort method.

Using 2-D Array

6. Write a program to take 3 student marks of 5 subjects. Print the total marks of each student and average marks of each subject.
7. Multiplication of two matrices.
8. Find whether a given matrix is symmetric or not. (Hint: $A = A^T$)

Lab 5. Strings

1. Print number of vowels and consonant in a string.
2. Change all lower-case letters into an upper-case in a sentence.
3. Count the number of words in a sentence.
4. Reverse a string.
5. Find the string length of a string without using the predefined `strlen()` function.

6. Find the sub-string of a given string.
7. Check whether the given string is a palindrome or not.

Lab 6. Pointers

1. Access two integers using pointers and add them.
2. Write a program to find out the greatest and the smallest among the three numbers using pointers.
3. Determine the length of a character string using a pointer.
4. Compute the sum of all elements stored in an array using a pointer.
5. Determine whether a substring (string 1) is in the main string or not. If present, return the pointer of the first occurrence.

Lab 7. Functions

1. Find the factorial of a number using a function.
2. Find the maximum of a given set of numbers using functions.
Find GCD of two numbers recursively.
3. Check whether the given number is prime or not. Using this function generate first n prime numbers using the above function.
4. Write a function to generate the nth Fibonacci term using recursion. Print first N Fibonacci terms using this function.
5. Check if the given string is a palindrome or not, using string handling function.
6. Write a function **Sort** for sorting a list of names which will use a function **compare** to compare two names. (bubble Sort may be used).

References

1. E. Balagurusamy, "Programming in ANSI C", 7th Edition, McGraw Hill Publication, 2016.
2. Y. P. Kanetkar, "Let us C", 12th Edition, BPB Publication, 2014.
3. B. W. Kernighan, D. M. Ritchie, "The C Programming Language", 2nd Edition, Prentice Hall of India, 2014.
4. Gottfried, "Schaum's Outline Series: Programming with C", 3rd Edition, McGraw Hill Publication, 2012.

ADDITIONAL EXERCISE

1. Write a program to find the determinant of a square matrix.
2. Write a program to check whether a saddle point exists in a given matrix.(i.e element which is minimum in row and maximum in column)
3. Write a program, which generates all the possible combinations of a 3-digit number.
4. Given a list of N numbers in the range 1 to N (not necessarily in the order) and out of that one number is missing (i.e. only N-1 inputs), write a program to find the missing number.
5. Two sets are represented by arrays A and B. Find AUB, A^B and A-B.
6. Given 4 numbers 1, 2, 3 and 4. Find all possible 4-digit numbers that can be formed using these 4-digits. From these numbers, generate the permutation of the 4 digits (i.e. 4 digits are distinct)
7. Write a program to find the median of N numbers.
8. Write a program to check whether the given number is Armstrong Number.
9. Write a program that examines all the numbers from 1 to N, displaying all those for which the sum of the cubes of all the digits equals the number itself.
10. Input N numbers and place only those numbers whose P'th position (specified by user) is divisible by 3 into an array. (Eg: The numbers input : 1223, 1243, 1232; position P=4 (left to right) then the resultant array will { 1223,1243})
11. A number has all its digits squared and added together to make a new number. This process is repeated until a 1 is obtained, then the original number is described as "HAPPY". If a 1 is never obtained, then the original number is said to be "SAD". { eg.: $19 = 1^2 + 9^2 = 82 \Rightarrow 8^2 + 2^2 = 68 \Rightarrow 6^2 + 8^2 = 100 \Rightarrow 1^2 = 1$ }
12. Write a program to delete blanks from a given string.
13. Write a program to display following output for N number of lines.

```

                1
              1  2  1
            1  2  3  2  1
          1  2  3  4  3  2  1
        1  2  3  2  1
      1  2  1
    1

```

14. Find the inverse of a matrix A.
15. Find the solution to the system of simultaneous equations using Guass- Jordan elimination method.
16. Display the list of perfect squares in the given mXn matrix.
17. Given a string S1="This is a way to do this" and a string S2="is" find the positions of occurrence of S2 in S1.
18. Replace odd numbers in a matrix by zeroes.
19. Display all the Pythagorean triplets up to n. (Eg: $3^2 + 4^2 = 5^2$).

20. Write a program to find the transpose of a matrix without using another matrix.
21. WAP to print the abbreviation of a given name. (Eg: Central Computing Facility => C.C.F)
22. WAP to simulate an arithmetic calculator.
23. Replace each element of the matrix by its factorial.
24. Find the odd numbers in a given matrix and display its row and column number.
25. Convert 3X3 matrix to a 4X4 matrix.

Eg: Original Matrix:

1	2	3
1	2	3
1	2	3

New Matrix:

1	2	3	6
1	2	3	6
1	2	3	6
3	6	9	18

26. WAP to print all combinations of a given word. (Eg: Rose => oser, osre.... Etc)
27. Write an n digit number in words.(236=two three six)
28. Find the volume of a cylinder and a cube.
29. Find the LCM of two numbers.
30. Find all the primes in the multiplication tables ranging from 2 to 5 and store them in array without having duplicates.
31. Check all the rows, columns, principal diagonal, and secondary diagonal for palindromes in an integer matrix of any order.
32. WAP to find the digital roots of numbers for multiplication tables ranging from 2 to 5 and store those forty resultant numbers in a matrix of order 4X10. (Digital root of number is a single digit derived from adding all the individual digits of the number repeatedly. (Eg: digital root of 2456= 2+4+5+6=17, again 1+7=8---> digital root)
33. Generate:

			1			
			1		1	
		1		2		1
	1		3		3	
		1		3		1
1		4		6		4
			4		6	
				1		1

34. Write a program to read a line, encode the line and display the original and encoded form.

The encode should be:

a	b	c	d	z
z	y	x	w	a