- **Formula based 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).

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.

- **Control Structures: If statement**
1. Check whether the given number is odd or even.
2. Find the absolute value of an integer.
3. Check whether a given year is a leap year or not.
4. Find the roots of a quadratic equation.
    a. Hint: root $= -b +/- sqrt(b^2-4ac)/2a$
5. Find no. of days of particular input month
6. Find total no. of days for a given number of months counting from January month.
    a. Example: m=3, days=31+ (28 0r 29) + 31.
7. Write a program to take two numbers as an input and find whether one number is multiple of other or not.
8. 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"
9. Write a program which takes three sides of a triangle input and calculates its area, if these conditions are satisfied a+b>c, b+c>a, a+c>b, calculate area=(a+b+c)/2
10. Write a program to check whether given character is vowel, consonant or digit.

- **Control Structures: Switch**
1. Program to calculate an area of a circle, a rectangle or a triangle depending on user's choice.
2. Remove all the break statements from Ex-1 (with switch-case construct) and try to execute the program with few inputs. Observe the difference.
3. Program to input number of week's day (1-7) and translate to its equivalent name of the day of the week (e.g. 1 to Sunday, 2 to Monday)
4. 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.

5. 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

6. Calculate an amount of a telephone bill for the following criteria. (Without Loop)

Calls	charge per call (Rs.)
1-150	0
151-250	.9
251-400	1.2
401 onwards	1.5

7. Calculate amount of an electricity bill for the following criteria. (Without Loop)

Units	charge per unit (Rs.)
First, 1-100 up to	0
Next, 101-200 up to	1.5
Next, 201-400 up to	2.5
401 onwards	3.5

8. Calculate discount in rupees for the following criterion. (Without Loop)

Cost price	discount
>=800	25%
500-800	20%
<500	no discount (0%)

- **Lab 6. Control Structures: 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 Fibonacci number.

4. Find whether given number is prime or not.
5. Convert the decimal number into binary to decimal. Ex:
   $1101 = 1*2^3 + 1*2^2 + 0*2^1 + 1*2^0 = 13$

6. Reverse a given number

7. Find the sum of n terms of the sin series $\sin(x) = x - x^3 + x^5 - x^7$
8. Check whether a given integer no. is palindrome or not.
9. Find LCM and HCF of two numbers.
10. 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 33 + 73 + 13 = 371.

- **Lab 7. Control Structures: Nested Loops**
1. Print different patterns using nested loops. 1 23

```
4    5    6
7    8    9         10
11   12   13        14  15
```

```
1
2    2
3    3    3
4    4    4    4
5    5    5    5     5   5

5    5    5    5     5
4    4    4    4
3    3    3
2    2
1
```

4. Generate the multiplication table for n numbers up to k terms (nested loops).

    Hint :   1 2 3 4 5 …. K

                     2 4 6 8 10 … 2*k

                  ……………….

                 ……………..

           n............................. n*K

5. Print the Fibonacci numbers that fall in given range.

6. Print the prime numbers that fall in given range.

7. Print that N^{th} the prime number.

8. Print that N^{th} the Armstrong number.

- **1-D Array**
1. Find whether given number is available in an array or not.
2. Find the largest and smallest element in an array.
3. Find the sum of odd index numbers in an array.
4. Print the subarray that lies between the two indexes.
5. Print the ASCII code of character array.
6. Find the number of positive numbers, negative numbers, odd numbers, even numbers and number of 0 of an array.

7. Reverse an array with an auxiliary array.
8. Check whether an array is sorted or not.
9. Arrange the elements of an array in ascending order by simple sorting method. (Selection sort/bubble sort)
10. Take an array of 10 elements. Split it into middle and store the elements in two different arrays. E.g.-

initial array :

58	24	13	15	63	9	8	81	1	78

After splitting:

58	24	13	15	63
9	8	81	1	78

- **2-D Arrays**
1. Write a program to take 3 student marks of 5 subjects. Print the total marks of each student and average marks of each subject.
2. Searching for an element in the matrix and counting the number of occurrences of that element.
3. Multiplication of two matrices.
4. Check if the given matrix is magic square or not.
5. Find whether a given matrix is symmetric or not.
   Hint: $A = A^T$
6. Find the trace and norm of a given square matrix.
   Hint: Trace = sum of principal diagonal elements
   Norm = sort (sum of squares of the individual elements of an array)

- **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. Find the last occurrence of a particular character.
4. Concatenate/length/copy two strings using the library function.
5. Count the number of words in a sentence.
6. Reverse a string.
7. Find the string length of a string without using the predefined function.
8. Find the sub-string of a given string.
9. Check if the given string is a palindrome or not.

- **Functions**
1. Find the factorial of a number using a function.

(Ex: 5! =5*4*3*2*1. Use a function Fact to evaluate factorial & print the result).

2. Find the maximum of a given set of numbers using functions. (Use
   a function Max and return the result to the main function)
3. Find GCD of two numbers recursively.
   (Ex: GCD of 9, 24 is 3)
4. Check whether the given number is prime or not. Using this function generate first n prime numbers using the above function.
5. Write a function to generate the nth Fibonacci term using recursion. Print first N Fibonacci terms using this function.
   Hint: (Fibonacci series is 0, 1, 1, 2, 3, 5, 8,)
6. Check if the given string is a palindrome or not, using string handling function.
7. Write a function **Sort** for sorting a list of names which will use a function **compare** to compare two names. (Selection /bubble Sort may be used).

• **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 sting or not. If present, return the pointer of the first occurrence.

• **Structures**

1. Define a structure personal that would contain person name, date of joining and salary. Using this structure write a program to read this information for one person from the keyboard and print the same on the screen.
2. Create an array of student structure to store the roll no., name and marks in 3 subjects. Input the details of N students into the array and display roll no., name, and total marks of each student in decreasing order of total marks.
3. Create an array of employee structure to store emp-no, name, basic salary and hra. Input the details of N employees and display emp-no, name, basic, hra and net salary. Display the details of all employees whose net salary is more than the average net salary of all employees.
4. Create a structure named Date having day, month and year as its elements. Store the current date in the structure. Now add 45 days to the current date and display the final date.

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.: $19 = 1^2 + 9^2 = 82 => 8^2 + 2^2 = 68 => 6^2 + 8^2 = 100 => 1^2 = 1$}
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.

```
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
1         4         6         4         1
```

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

```
a   b   c   d   …..z
z   y   x   w   …. a
```

 35. Write a program to find the time of the day when
a) Hour and Minute hand are exactly opposite to each other.
b) Hour and Minute hand are overlapping each other.