EX:NO: 01(a)

MS-WORD EXERCISES

DOCUMENT CREATION AND TEXT MANIPULATION

AIM:

To create a document using MS-WORD and perform text manipulations with scientific notations.

ALGORITHM:

Step 1: Start.

Step 2: Open a new document from the file menu and type a paragraph.

Step 3: Select the bullets and numbering from the home menu and apply to the paragraph.

Step 4: Select the font style and font size from the home menu and apply to the paragraph.

Step 5: Select the page number option from the insert menu and display the page to every page of the document.

Step 6: Select the paragraph and apply the left indent and right indent from home menu.

Step 7: Go to Insert menu and select Symbols to insert the necessary symbols

Step 8: Save the file.

Step 9: Stop.

VAIGAI COLLEGE OF ENGG - Department of CSE

RESULT:

Thus the document creation and performing text manipulations with scientific notations were performed using MS-WORD.

Do your self:

1. Type a leave letter and format this letter
2. Create a Bio data (Apply Bullets and numberings)
3. Type the following formulae using Scientific notation

$$\left[\begin{array}{ccc|c} 3 & 4 & -2 & 15 \\ 2 & -3 & -8 & 11 \\ 5 & -2 & -3 & 16 \end{array}\right].$$

$$y = ax^3 + bx^2 + cx + d.$$

$$\int \tan(\pi x)dx = \frac{1}{\pi}\ln(|\sec(\pi x)|) + C$$

$$\bar{x} = \frac{1}{n}\sum_{j=1}^{n} x_j.$$

$$\lim_{x\to\infty} f(x) = a.$$

$$(x+y)^2 \neq x^2 + y^2.$$

$$|x| = \begin{cases} x & \text{for } x \geq 0 \\ -x & \text{for } x < 0 \end{cases}$$

$$y = a\left(\frac{x}{b} + c\right).$$

4. Create a Greeting Card (Put page border and shadings wherever needed)
5. Create an advertisement for your company

VAIGAI COLLEGE OF ENGG - Department of CSE

EX: NO: 01(b)

TABLE CREATION AND FORMATTING

AIM:

To create, format and convert a table in the MS-WORD.

ALGORITHM:

Step 1: Start

Step 2: Open a blank document in MS-WORD.

Step 3: Click the insert menu and click the table option. Select the number of rows and columns to be created and then click ok.

Step 4: To insert rows, columns and cells in the table right click in the table and click insert option and click rows or columns or cells.

Step 5: To delete rows, columns and cells in the table right click in the table and click delete option and click rows or columns or cells.

Step 6: To merge two cells in the table click the two cells and then right click and select merge option.

Step 7: To split a cell in the table, right click the cell and click split option.

Step 8: To convert text to table click insert menu and click table option and select "convert text to table option" and to convert table to text select "convert table to text" option from the layout menu.

Step 9: Stop.

RESULT:

Thus the creation, formatting and conversion of the table was performed in MS-WORD.

Do your self:

1. Create a your Weekly schedule.(Apply all table formatting methods)
2. Create a table to maintain your details as Name, Roll Number, and a column for marks and under this column provide 5 columns to maintain 5 different marks, total, Average. Finally sort the table by names in descending order.

VAIGAI COLLEGE OF ENGG., Department of CSE

EX: NO: 02

MS-EXCEL

CHART CREATION- LINE, XY, BAR and PIE

AIM:

To create chart-Line, XY, Bar and Pie charts for an employee using MS-EXCEL.

ALGORITHM:

Step 1: Start

Step 2: Open an excel spreadsheet and enter the employee number, name and basic pay.

Step 3: Calculate the Dearness Allowance (DA), House Rent Allowance (HRA) and Provident Fund (PF) from the basic pay using the specific formula.

Step 4: Calculate the gross pay of an employee by adding the basic pay and the other allowances of the employee.

Step 5: Calculate the net pay of an employee from the gross pay and PF of the employee.

Step 6: Then select the fields in the spreadsheet and click the insert menu.

Step 7: Click the type of the chart to insert.

Step 8: Stop.

VAIGAI COLLEGE OF ENGG - Department of CSE

RESULT:

Thus the creation of charts likes Line, XY, Bar and Pie using MS-EXCEL was performed.

EX: NO: 03

FLOW CHART

AIM:

To prepare a flowchart in MS-WORD to find the sum of first 100 natural numbers.

ALGORITHM:

Step 1: Start

Step 2: Open a new document and then click the "insert" menu.

Step 3: Then click the shapes option and then click the different symbols in the flowchart option to draw a flowchart.

Step 4: Stop.

RESULT:

Thus the flowchart to find the sum of 100 natural numbers was drawn in MS-WORD.

Do your self:

1. Draw flow chart for the following
    a. Addition of 2 numbers
    b. Biggest among three numbers
    c. Swapping of two variables without using temporary variable
    d. Quadratic equation solving

VAIGAI COLLEGE OF ENGG - Department of CSE

C EXERCISES

VAIGAI COLLEGE OF ENGG - Department of CSE

PROGRAMS USING SIMPLE STATEMENTS AND EXPRESSIONS

EX: NO: 04 (a)

INTEGER TO FLOAT CONVERSION

AIM:

To write a c program to convert a value of one data type into another data type.

ALGORITHM:

Step 1: Declare the necessary variables a and b as integer and float.
Step 2: Read the input of the integer a.
Step 3: Assign the value of the integer to a float variable and a double variable.
Step 3: Display the value of the float variable and the double variable.

PROGRAM:

```
#include<stdio.h>
main()
{
        float b;
        int a;
        printf("\nEnter an Integer\n");
        scanf("%d",&a);
        b=(float)a;
        printf("\nThe Converted float value is %f",b);
}
```

OUTPUT:

```
Enter an Integer
45
The Converted float value is 45.00000
```

RESULT:

Thus the c program to convert the integer into float was written, entered, executed and the output was verified.

VAIGAI COLLEGE OF ENGG - Department of CSE

PROGRAMS USING SIMPLE STATEMENTS AND EXPRESSIONS

EX: NO: 04 (b)

MULTIPLICATION OF TWO NUMBERS

AIM:

To write a c program to produce the Multiplication result of given two Numbers.

ALGORITHM:

Step 1: Declare the necessary variables a, b and c as integer.
Step 2: Read the input of the integer a and b.
Step 3: Multiply a & b and store the result in c
Step 3: Display the value of c

PROGRAM:

```
#include<stdio.h>
main()
{
        int a,b,c;
        printf("Enter Number 1\n");
        scanf("%d",&a);
        printf("Enter Number 2\n");
        scanf("%d",&b);
        c=a*b;
        printf("\nThe Multiplication Result is %d\n",c);
}
```

OUTPUT:

```
Enter Number 1
34
Enter Number 2
7
The Multiplication Result is 238
```

RESULT:

Thus the c program to produce the Multiplication result of given two Numbers was written, entered, executed and the output was verified.

VAIGAI COLLEGE OF ENGG - Department of CSE

PROGRAMS USING SIMPLE STATEMENTS AND EXPRESSIONS

EX: NO: 04 (c)

AVERAGE OF FIVE MARKS

AIM:

To write a c program to calculate the Average of given five Numbers.

ALGORITHM:

Step 1: Declare five integer variables to get the marks.
Step 2: Read the input of five marks and store them into integer variables.
Step 3: Calculate the sum of five numbers.
Step 4: Divide the sum by 5.
Step 5: Display the Average

PROGRAM:

```
#include<stdio.h>
main()
{
        int m1,m2,m3,m4,m5,tot;
        float avg;
        printf("Enter 5 Marks\n");
        scanf("%d%d%d%d%d",&m1,&m2,&m3,&m4,&m5);
        tot=m1+m2+m3+m4+m5;
        avg=tot/5;
        printf("\nThe Average is %f\n",avg);
}
```

OUTPUT:

```
Enter Number 1
34
Enter Number 2
7
The Multiplication Result is 238
```

RESULT:

Thus the c program to calculate the Average of given five Numbers was written, entered, executed and the output was verified.

VAGAI COLLEGE OF ENGG - Department of CSE

PROGRAMS USING SIMPLE STATEMENTS AND EXPRESSIONS

EX: NO: 04 (d)

EVALUATION OF AN EQUATION

AIM:

To write a c program to evaluate the given Equation.

ALGORITHM:

Step 1: Declare Necessary variables
Step 2: Read the input for Equation terms
Step 3: Calculate the value of Numerator
Step 4: Calculate the value of Denominator
Step 5: Divide the Numerator by Denominator
Step 6: Display the Result

PROGRAM:

```
#include<stdio.h>
main()
{
        int v,g,c,d,dr;
        float r,nr;
        printf("Enter the value of v\n");
        scanf("%d",&v);
        printf("Enter the value of g\n");
        scanf("%d",&g);
        printf("Enter the value of c\n");
        scanf("%d",&c);
        printf("Enter the value of d\n");
        scanf("%d",&d);
        nr=(2*v)+(6.22*c*d);
        dr=g+v;
        r=nr/dr;
        printf("The Evaluated Result is %f\n",r);
}
```

VAIGAI COLLEGE OF ENGG - Department of CSE

OUTPUT:

```
Enter the value of v
2
Enter the value of g
4
Enter the value of c
6
Enter the value of d
8
The Evaluated Result is 50.426666
```

VAIGAI COLLEGE OF ENGG - Department of CSE

RESULT:

Thus the c program to evaluate the given Equation was written, entered, executed and the output was verified.

PROGRAMS USING SIMPLE STATEMENTS AND EXPRESSIONS

EX: NO: 04 (e)

MEASUREMENT CONVERSION

AIM:

To write a c program to convert given millimeter measurement into meter.

ALGORITHM:

Step 1: Declare a variable to get Millimeter

Step 2: Read the input

Step 3: Multiply given input by 1000

Step 4: Display the Result

PROGRAM:

```
#include<stdio.h>
main()
{
        int mm,m;
        printf("Enter the Millimeter\n");
        scanf("%d",&mm);
        m=mm*1000;
        printf("The Converted meter is %d",m);
}
```

OUTPUT:

```
Enter the Millimeter
12
The Converted meter is 12000
```

RESULT:

Thus the c program to convert given millimeter measurement into meter was written, entered, executed and the output was verified.

VAIGAI COLLEGE OF ENGG - Department of CSE

SCIENTIFIC PROBLEM SOLVING USING DECISION MAKING AND LOOPING

EX: NO: 05 (a)

ODD OR EVEN

AIM:

To write a c program to check whether given Number is odd or even.

ALGORITHM:

Step 1: Declare a variable to get a Number

Step 2: Read the input

Step 3: Get the remainder of given number using modulo operator

Step 4: If remainder is 0 prints "Even Number", else print "Odd Number".

PROGRAM:

```
#include<stdio.h>
main()
{
        int a,rem;
        printf("Enter a Number\n");
        scanf("%d",&a);
        rem=a%2;
        if(rem==0)
                printf("The Given Number is Even");
        else
                printf("The Given Number is Odd");
}
```

OUTPUT:

```
Enter a Number
13
The Given Number is Odd
```

RESULT:

Thus the c program to check whether given Number is odd or even was written, entered, executed and the output was verified.

VAIGAI COLLEGE OF ENGG - Department of CSE

SCIENTIFIC PROBLEM SOLVING USING DECISION MAKING AND LOOPING

EX: NO: 05 (b)

BIGGEST OF 3 NUMBERS

AIM:

To write a c program to examine the biggest of given three numbers.

ALGORITHM:

Step 1: Declare three integer variables

Step 2: Read the 3 inputs

Step 3: Compare first two numbers and go to Step4

Step 4: If first number is greater than second number then compare first number with third number else go to step 6

Step 5: If first number is greater than third number print first number as biggest number else print third number as biggest

Step 6: Compare second number with third number

Step 7: If second number is greater than third number print second number as biggest number else print third number as biggest

PROGRAM:

```
#include<stdio.h>
main()
{
        int a,b,c;
        printf("Enter 3 Numbers\n");
        scanf("%d%d%d",&a,&b,&c);
        if(a>b)
        {
                if(a>c)
                {
                        printf("The First Number %d(a) is Biggest\n",a);
                }
        }
        else if(b>c)
```

VAIGAI COLLEGE OF ENGG - Department of CSE

```
        {
                printf("The Second Number %d(b) is Biggest\n",b);
        }
        else
                printf("The Third Number %d(c) is Biggest\n",c);
}
```

OUTPUT:

```
Enter 3 Numbers
5
9
2
The Second Number 89(b) is Biggest
```

VAIGAI COLLEGE OF ENGG - Department of CSE

RESULT:

Thus the c program to examine the biggest of given three numbers was written, entered, executed and the output was verified.

SCIENTIFIC PROBLEM SOLVING USING DECISION MAKING AND LOOPING

EX: NO: 05 (c)

SUM OF 'N' NATURAL NUMBERS

AIM:

To write a c program to find the sum of 'N' natural numbers for given range.

ALGORITHM:

Step 1: Initialize the sum as 0

Step 2: Read the range as input

Step 3: Initialize a counter with 1

Step 4: Overwrite the sum by adding counter value & sum

Step 5: Increment the counter value by 1

Step 6: Repeat the steps 4 & 5 until the counter is less than or equal to range

Step 7: Print the sum

PROGRAM:

```
#include<stdio.h>
main()
{
        int i,n,sum=0;
        printf("Enter the range\n");
        scanf("%d",&n);
        i=1;
        while(i<=n)
        {
                sum=sum+i;
                i++;
        }
        printf("\nThe sum of first %d numbers is %d\n",n,sum);
}
```

VAIGAI COLLEGE OF ENGG - Department of CSE

OUTPUT:

```
Enter the range
16
The sum of first 16 numbers is 136
```

VAIGAI COLLEGE OF ENGG - Department of CSE

RESULT:

Thus the c program to find the sum of 'N' natural numbers for given range was written, entered, executed and the output was verified.

SCIENTIFIC PROBLEM SOLVING USING DECISION MAKING AND LOOPING

EX: NO: 05 (d)

SUM OF DIGITS

AIM:

To write a c program to find the sum of digits for a given number.

ALGORITHM:

Step 1: Declare a integer variable and initialize the sum as 0

Step 2: Read the input

Step 3: Take the remainder of given number while dividing 10

Step 4: Overwrite the sum by adding above remainder with available sum

Step 5: Overwrite the number by divide with 10

Step 6: Repeat the steps 3 to 5 until the number is greater than 0

Step 7: Print the sum

PROGRAM:

```
#include<stdio.h>
main()
{
        int n,i,sum=0;
        printf("Enter a Number\n");
        scanf("%d",&n);
        do
        {
                i=n%10;
                sum=sum+i;
                n=n/10;
        }while(n>0);
        printf("The Sum of digit is %d\n",sum);
}
```

VAIGAI COLLEGE OF ENGG - Department of CSE

OUTPUT:

```
Enter a Number
5891
The Sum of digit is 23
```

VAIGAI COLLEGE OF ENGG - Department of CSE

RESULT:

Thus the c program to find the sum of digits for a given number was written, entered, executed and the output was verified.

SCIENTIFIC PROBLEM SOLVING USING DECISION MAKING AND LOOPING

EX: NO: 05 (e)

EVALUATION OF SINE SERIES

AIM:

To write a c program to evaluate the sine series.

ALGORITHM:

Step 1: Get the input for number of terms

Step 2: Get the input for value of x

Step 3: Initialize a counter with 1

Step 4: Initialize a sign counter with positive value of 1

Step 5: Initialize the sum as 0

Step 6: Calculate the power of x assigned to counter

Step 7: Multiply the above result with sign counter and store as Numerator

Step 8: Calculate the factorial value of counter value as follows

a. Initialize a loop counter to1
b. Initialize the product to 1
c. Obtain the new product by multiplying counter value with old product
d. Increment the loop counter by 1
e. Repeat the steps c & d until the loop counter is less than or equal to counter value
f. Store the product as denominator

Step 9: Divide Numerator by Denominator

Step 10: Obtain the sum by adding available sum with above division result

Step 11: Multiply the sign counter with -1

VAIGAI COLLEGE OF ENGG - Department of CSE

Step 12: Increment the counter value by 2

Step 13: Repeat the steps 6 to 12 until counter is less than or equal to range

Step 14: Print the sum

PROGRAM:

```
#include<stdio.h>
#include<math.h>
int factorial(int n)
{
        int i,sum=1;
        for(i=1;i<=n;i++)
                sum=sum*i;
        return sum;
}
main()
{
        int i,n,j,dr;
        float res=0.0,x,nr;
        printf("\nEnter the Value of x\n");
        scanf("%f",&x);
        printf("\nEnter the total no of terms\n");
        scanf("%d",&n);
        j=1;
        for(i=1;i<n*2;i+=2)
        {
                nr=pow(x,i)*j;
                dr=factorial(i);
                res+=(nr/dr);
                j=-j;
        }
        printf("The Result of sine series is : %f\n",res);
}
```

VAIGAI COLLEGE OF ENGG - Department of CSE

OUTPUT:

```
Enter the Value of x
0.21
Enter the total no of terms
5
The Result of sine series is : 0.208460
```

VAIGAI COLLEGE OF ENGG - Department of CSE

RESULT:

Thus the c program to find the sum of digits for a given number was written, entered, executed and the output was verified.

SCIENTIFIC PROBLEM SOLVING USING DECISION MAKING AND LOOPING

EX: NO: 05 (f)

ARITHMETIC CALCULATOR

AIM:

To write a menu driven c program to implement an Arithmetic Calculator.

ALGORITHM:

Step 1: Get the two numbers

Step 2: Enter the choice

Step 3: Pass the choice into switch case

Step 4: In case 1, add the two numbers and print the result

Step 5: In case 2, subtract the two numbers and print the result

Step 6: In case 3, multiply the two numbers and print the result

Step 7: In case 4, divide the two numbers and print the result

PROGRAM:

```
#include<stdio.h>
main()
{
        int a,b,ch,c;
        printf("\nEnter the Number 1:\n");
        scanf("%d",&a);
        printf("\nEnter the Number 2:\n");
        scanf("%d",&b);
        printf("\n1.Add\n2.Subtract\n3.Multiply\n4.Divide\n");
        printf("\nEnter the Choice:\n");
        scanf("%d",&ch);
        switch(ch)
        {
                case 1:
```

VAIGAI COLLEGE OF ENGG - Department of CSE

```
                c=a+b;
                printf("\n %d + %d = %d\n",a,b,c);
                break;
        case 2:
                c=a-b;
                printf("\n %d - %d = %d\n",a,b,c);
                break;
        case 3:
                c=a*b;
                printf("\n %d * %d = %d\n",a,b,c);
                break;
        case 4:
                c=a/b;
                printf("\n %d / %d = %d\n",a,b,c);
                break;
    }
}
```

OUTPUT:

```
Enter the Number 1:
15

Enter the Number 2:
56
1.Add
2.Subtract
3.Multiply
4.Divide
Enter the Choice:
2
 15 - 56 = -41
```

RESULT:

Thus the menu driven c program to implement an Arithmetic Calculator was written, entered, executed and the output was verified.

VAIGAI COLLEGE OF ENGG - Department of CSE

SCIENTIFIC PROBLEM SOLVING USING DECISION MAKING AND LOOPING

EX: NO: 05 (g)

NUMBER CHECKING

AIM:

To write a menu driven c program to check whether the given number is Palindrome, Armstrong and Prime.

ALGORITHM:

Step 1: Get a number from the user

Step 2: Enter the choice

Step 3: Pass the choice into switch case

Step 4: In case 1,

a. Copy the given number into a variable
b. Initialize a counter to 1 and sum to 0
c. Extract the remainder of given number while dividing 10
d. Multiply the sum by 10
e. Overwrite the sum by adding above remainder with available sum
f. Overwrite the number by divide with 10
g. Repeat the steps a to f until the number is greater than 0
h. Compare the sum and copy of the number
i. If they are equal print as “Palindrome” else print “Not Palindrome”

Step 5: In case 2,

a. Copy the given number into a variable
b. Initialize a counter to 1 and sum to 0
c. Extract the remainder of given number while dividing 10
d. Calculate the value of remainder by assigning power 3

VAIGAI COLLEGE OF ENGG - Department of CSE

e. Overwrite the sum by adding above result with available sum
f. Overwrite the number by divide with 10
g. Repeat the steps a to e until the number is greater than 0
h. Compare the sum and copy of the number
i. If they are equal print as “Armstrong” else print “Not Armstrong”

Step 6: In case 3,

a. Initialize a flag value with 5
b. Initialize a counter to 2
c. Extract the remainder of given number by dividing with counter value
d. If the remainder is 0 changes the flag value to 0 and go to step g else go to next step.
e. Increment the counter value by 1
f. Repeat the steps a to e until counter is less than or equal to square root of the given number
g. Check the flag value
h. If flag value is 0 then print as “Prime Number” else print as “Not Prime”

PROGRAM:

```
#include<stdio.h>
#include<math.h>
main()
{
        int a,i,sum=0,n,ch,m;
        printf("\nEnter a Number\n");
        scanf("%d",&a);
        printf("\n1.Palindrome\n2.Armstrong\n3.Prime\n");
        printf("\nEnter the Choice:\n");
        scanf("%d",&ch);
        switch(ch)
        {
                case 1:
```

VAIGAI COLLEGE OF ENGG - Department of CSE

```
        n=a;
        while(a>0)
        {
                i=a%10;
                sum=(sum*10)+i;
                a=a/10;
        }
        if(n==sum)
                printf("Given Number is Palindrome\n");
        else
                printf("Given Number is Not Palindrome\n");
        break;
case 2:
        n=a;
        do
        {
                i=a%10;
                sum=sum+(i*i*i);
                a=a/10;
        }while(a>0);
        if(n==sum)
                printf("Given Number is Armstrong\n");
        else
                printf("Given Number is Not Armstrong\n");

        break;
case 3:
        m=5;
        n=sqrt(a);
        for(i=2;i<=n;i++)
        {
                if(a%i==0)
                {
                        m=0;
                        break;
                }
        }
        if(m==0)
                printf("Given Number is Prime\n");
        else
```

VAIGAI COLLEGE OF ENGG - Department of CSE

```
                    printf("Given Number is Not Prime\n");
                break;

        }
}
```

OUTPUT:

```
Enter a Number
121
1.Palindrome
2.Armstrong
3.Prime

Enter the Choice:
1
Given Number is Palindrome
```

VAIGAI COLLEGE OF ENGG - Department of CSE

RESULT:

Thus the menu driven c program to check whether the given number is Palindrome, Armstrong and Prime was written, entered, executed and the output was verified.

SIMPLE PROGRAMMING FOR ONE DIMENSIONAL AND TWO DIMENSIONAL ARRAYS

EX: NO: 06 (a)

SUM OF ARRAY ELEMENTS

AIM:

To write a c program to find the sum of given array elements.

ALGORITHM:

Step 1: Declare an array with necessary size

Step 2: Get the value for total number of elements

Step 3: Initialize an index value to 0

Step 4: Read the input

Step 5: Increment the index value by 1

Step 6: Repeat steps 4 & 5 until counter less than total no. of elements

Step 7: Initialize an index value to 0 and sum to 0

Step 8: Obtain the sum by adding current index array value with available Sum

Step 9: Increment the index value by 1

Step 10: Repeat steps 8 & 9 until index value less than total no. of elements

Step 11: Print the sum

PROGRAM:

```
#include<stdio.h>
main()
{
        int i,n,a[10],sum=0;
        printf("Enter total no. of Elements\n");
        scanf("%d",&n);
```

VAIGAI COLLEGE OF ENGG - Department of CSE

```
        printf("Enter Array elements one by one\n");
        for(i=0;i<n;i++)
                scanf("%d",&a[i]);
        for(i=0;i<n;i++)
                sum=sum+a[i];
        printf("The Sum of Array Elements is %d\n",sum);
}
```

OUTPUT:

```
Enter total no. of Elements
8
Enter Array elements one by one
15
69
32
10
45
66
32
11
The Sum of Array Elements is 280
```

VAIGAI COLLEGE OF ENGG - Department of CSE

RESULT:

Thus the menu driven c program to find the sum of given array elements was written, entered, executed and the output was verified.

SIMPLE PROGRAMMING FOR ONE DIMENSIONAL AND TWO DIMENSIONAL ARRAYS

EX: NO: 06 (b)

DISPLAY EVEN NUMBERS OF AN ARRAY

AIM:

To write a c program to print the even numbers of given array elements.

ALGORITHM:

Step 1: Declare an array with necessary size

Step 2: Get the value for total number of elements

Step 3: Initialize an index value to 0

Step 4: Read the input

Step 5: Increment the index value by 1

Step 6: Repeat steps 4 & 5 until counter less than total no. of elements

Step 7: Initialize an index value to 0

Step 8: Extract the remainder by dividing array index value with 2

Step 9: If the remainder is 0 print the value

Step 10: Increment the index value by 1

Step 11: Repeat steps 8 to 10 until index value less than total no. of elements

PROGRAM:

```
#include<stdio.h>
main()
{
        int i,n,a[10];
        printf("Enter total no. of Elements\n");
        scanf("%d",&n);
        printf("Enter Array elements one by one\n");
```

VAIGAI COLLEGE OF ENGG - Department of CSE

```
        for(i=0;i<n;i++)
                scanf("%d",&a[i]);
        printf("The even numbers of given array:\n");
        for(i=0;i<n;i++)
        {
                if(a[i]%2==0)
                        printf("%d\n",a[i]);
        }
}
```

OUTPUT:

```
Enter total no. of Elements
6
Enter Array elements one by one
98
11
35
61
22
14
The even numbers of given array:
98
22
14
```

VAIGAI COLLEGE OF ENGG - Department of CSE

RESULT:

Thus the menu driven c program to print the even numbers of given array elements was written, entered, executed and the output was verified.

SIMPLE PROGRAMMING FOR ONE DIMENSIONAL AND TWO DIMENSIONAL ARRAYS

EX: NO: 06 (c)

MULTIPLICATION OF 2*2 MATRIXES

AIM:

To write a c program to perform 2*2 matrixes multiplication.

ALGORITHM:

Step 1: Start

Step 2: Declare the two dimensional integer arrays a[2][2], b[2][2] and c[2][2] and declare the variables k, I and j as integers.

Step 3: Read the input for the matrixes A and B.

Step 4: Print the matrixes A and B.

Step 5: Multiply the matrixes A and B and print the result in a matrix C.

Step 6: Stop

OUTPUT:

```
Enter the value of the first matrix:
    2    3
    3    4
Enter the value of the second matrix:
    3    4
    4    5
Product of the two matrices is
   18   23
   25   32
```

VAIGAI COLLEGE OF ENGG - Department of CSE

RESULT:

Thus the c program to perform 2*2 matrixes multiplication was written, entered, executed and the output was verified.

SOLVING PROBLEMS USING STRING FUNCTIONS

EX: NO: 07 (a)

STRING PALINDROME CHECKING

AIM:

To write a c program to check whether the given string is palindrome or not

ALGORITHM:

Step 1: Create a character array with necessary size
Step 2: Read the String
Step 3: Copy the String into another character array
Step 4: Get reverse string of input by using strrev function
Step 5: Compare the above result with copied string
Step 6: If two string s are same print "Palindrome" else print "Not Palindrome"

PROGRAM:

```
#include<stdio.h>
#include<string.h>
main()
{
        char s[20],s1[20];
        printf("Enter a String\n");
        scanf("%s",s);
        strcpy(s1,s);
        if(strcmp(s,s1)==0)
                printf("The Given String is Palindrome\n");
        else
                printf("The Given String is Not Palindrome\n");
}
```

OUTPUT:

```
Enter a String
madam
The Given String is Palindrome
```

RESULT:

Thus the c program to check whether the given string is palindrome or not was written, entered, executed and the output was verified.

VAIGAI COLLEGE OF ENGG - Department of CSE

SOLVING PROBLEMS USING STRING FUNCTIONS

EX: NO: 07 (b)

STRING CONCATENATION

AIM:

To write a c program to find the length of given two strings and concatenate them

ALGORITHM:

Step 1: Create two character arrays with necessary size

Step 2: Read the Strings

Step 3: Calculate the string lengths using strlen function

Step 4: Print the string lengths

Step 5: Join the two strings using strcat function

Step 6: Print the concatenated string

PROGRAM:

```
#include<stdio.h>
#include<string.h>
main()
{
        char s[20],s1[20];
        printf("Enter a String1\n");
        scanf("%s",s);
        printf("Enter a String2\n");
        scanf("%s",s1);
        strcat(s,s1);
        printf("The Concatenated String is %s\n",s);

}
```

VAIGAI COLLEGE OF ENGG - Department of CSE

OUTPUT:

```
Enter a String1
hai
Enter a String2
hello
The Concatenated String is haihello
```

VAIGAI COLLEGE OF ENGG - Department of CSE

RESULT:

Thus the c program to find the length of given two strings and concatenate them was written, entered, executed and the output was verified.

PROGRAMS WITH USER DEFINED FUNCTIONS

EX: NO: 08 (a)

FUNCTIONS WITHOUT ARGUMENTS & RETURN TYPE

AIM:

To write a c program to check whether the given year is leap or not using functions.

ALGORITHM:

Step 1: Create a function isleap()

Step 2: Inside the function

a. Read the year as input
b. Extract the remainder from division operation of year by 4
c. If remainder is 0 print "Given year is Leap year" else print "Given year is not a Leap year"

Step 3: Inside the main function call the isleap() function

PROGRAM:

```
#include<stdio.h>
void isleap()
{
        int yr;
        printf("Enter a Year\n");
        scanf("%d",&yr);
        if(yr%4==0)
                printf("Given Year is Leap year");
        else
                printf("Given Year is Not a Leap year");
}
main()
{
        isleap();
}
```

VAIGAI COLLEGE OF ENGG - Department of CSE

OUTPUT:

```
Enter a Year
1965
Given Year is Not a Leap year
```

VAIGAI COLLEGE OF ENGG - Department of CSE

RESULT:

Thus the c program to check whether the given year is leap or not using functions was written, entered, executed and the output was verified.

PROGRAMS WITH USER DEFINED FUNCTIONS

EX: NO: 08 (b)

FUNCTIONS WITHOUT ARGUMENTS & WITH RETURN TYPE

AIM:

To write a c program to calculate the area of triangle using functions.

ALGORITHM:

Step 1: Create a function area()

Step 2: Inside the function

a. Read the 3 sides of triangle
b. Calculate the sum of 3 sides
c. Divide the sum by 2 and store it into s
d. Subtract the sides from sand store them into variables
e. Multiply s with above 3 results
f. Take the square root of above result
g. Return the above result as area

Step 3: Inside the main function call the function area()

Step 4: Print the area by obtaining the return value of area()

PROGRAM:

```
#include<stdio.h>
#include<math.h>
float area()
{
        int a,b,c;
        float s,ar;
        printf("Enter 3 Sides\n");
        scanf("%d%d%d",&a,&b,&c);
        s=(a+b+c)/2;
        ar=sqrt(s*(s-a)*(s-b)*(s-c));
        return ar;
}
```

VAIGAI COLLEGE OF ENGG - Department of CSE

```
main()
{
        float a;
        a=area();
        printf("The Area of Triangle is %f\n",a);
}
```

OUTPUT:

```
Enter 3 Sides
12
8
7
The Area of Triangle is 19.748418
```

VAIGAI COLLEGE OF ENGG - Department of CSE

RESULT:

Thus the c program to calculate the area of triangle using functions was written, entered, executed and the output was verified.

PROGRAMS WITH USER DEFINED FUNCTIONS

EX: NO: 08 (c)

FUNCTIONS WITH ARGUMENTS & WITHOUT RETURN TYPE

AIM:

To write a c program to sort the given array of elements using functions.

ALGORITHM:

Step 1: Create a function sort()

Step 2: Inside the function

a. Initialize a index to 0
b. Initialize the sub index to counter + 1
c. Compare the two numbers which are available in array index value and array sub index value
d. If the first number is greater than second number swap them
e. Increment the sub index by 1
f. Repeat the steps c to e until sub index less than total number of elements
g. Increment the index by 1
h. Repeat the steps b to g until sub index less than total number of elements
i. Print the array elements

Step 3: Inside the main function

a. Create an integer array with necessary size
b. Get the total number of elements
c. Read the array elements one by one
d. Call the sort() function by passing array and no. of elements as arguments

PROGRAM:

```
#include<stdio.h>
void sorting(int a[],int n)
```

VAIGAI COLLEGE OF ENGG - Department of CSE

```
{
        int i,j,t;
        for(i=0;i<n-1;i++)
        {
                for(j=i+1;j<n;j++)
                {
                        if(a[i]>a[j])
                        {
                                t=a[i];
                                a[i]=a[j];
                                a[j]=t;
                        }
                }
        }
        printf("Array Elemets before sorting\n");
        for(i=0;i<n;i++)
                printf("%d\t",a[i]);
}
main()
{
        int i,a[10],n;
        printf("Enter total no. of elements\n");
        scanf("%d",&n);
        printf("Enter Array Elements one by one\n");
        for(i=0;i<n;i++)
                scanf("%d",&a[i]);
        printf("Array Elemets before sorting\n");
        for(i=0;i<n;i++)
                printf("%d\t",a[i]);
        printf("\n");
        sorting(a,n);
}
```

OUTPUT:

```
Enter total no. of elements
6
Enter Array Elements one by one
21
2
```

VAIGAI COLLEGE OF ENGG - Department of CSE

```
9
45
30
11
Array Elemets before sorting
21      2       9       45      30      11
Array Elemets before sorting
2       9       11      21      30      45
```

VAIGAI COLLEGE OF ENGG - Department of CSE

RESULT:

Thus the c program to sort the given array of elements using functions was written, entered, executed and the output was verified.

PROGRAMS WITH USER DEFINED FUNCTIONS

EX: NO: 08 (d)

FUNCTIONS WITH ARGUMENTS & RETURN TYPE

AIM:

To write a c program to find the smallest element of given array of elements using functions.

ALGORITHM:

Step 1: Create a function small()

Step 2: Inside the function

a. Store the 0^{th} index value into base
b. Initialize a index to 1
c. Compare the array index value with base
d. If the array index value is smaller than base store the array index value into base
e. Increment the index by 1
f. Repeat the steps c & e until index reaches total no. of elements
g. Return the base value

Step 3: Inside the main function

a. Create an integer array with necessary size
b. Get the total number of elements
c. Read the array elements one by one
d. Call the small() function by passing array and no. of elements as arguments

PROGRAM:

```
#include<stdio.h>
int small(int a[],int n)
{
        int s,i;
        s=a[0];
        for(i=0;i<n;i++)
```

VAIGAI COLLEGE OF ENGG - Department of CSE

```
        {
                if(a[i]<s)
                        s=a[i];
        }
        return s;
}
main()
{
        int i,a[10],n,s;
        printf("Enter total no. of elements\n");
        scanf("%d",&n);
        printf("Enter Array Elements one by one\n");
        for(i=0;i<n;i++)
                scanf("%d",&a[i]);
        printf("Array Elemets:\n");
        for(i=0;i<n;i++)
                printf("%d\t",a[i]);
        printf("\n");
        s=small(a,n);
        printf("The Smallest element of given array is %d",s);
}
```

OUTPUT:

```
Enter total no. of elements
5
Enter Array Elements one by one
1
98
2
66
0
Array Elemets:
1       98      2       66      0
The Smallest element of given array is 0
```

RESULT:

Thus the c program to find the smallest element of given array of elements using functions was written, entered, executed and the output was verified.

VAIGAI COLLEGE OF ENGG - Department of CSE

PROGRAM USING RECURSIVE FUNCTION

EX: NO: 09 (a)

FACTORIAL OF A NUMBER

AIM:

To write a c program to calculate the factorial of a given number

ALGORITHM:

Step 1: Get the number

Step 2: Call the function fact by passing number as an argument

Step 3: Inside the fact()

a. If the received value is 0 or 1than return 1
b. If the received value is not equal to 0 or 1 Multiply the value with return value of fact by passing value -1 as an argument
c. Return the above result

Step 4: Print the result by receiving the return value of fact()

PROGRAM:

```
#include<stdio.h>
int factorial(int n)
{
        if(n==0 || n==1)
                return 1;
        else
                return n*factorial(n-1);
}
main()
{
        int n;
        printf("\nEnter a Number\n");
        scanf("%d",&n);
        printf("\nThe factorial of %d is %d\n",n,factorial(n));
```

VAIGAI COLLEGE OF ENGG Department of CSE

```
}
```

OUTPUT:

```
Enter a Number
6
The factorial of 6 is 720
```

VAIGAI COLLEGE OF ENGG - Department of CSE

RESULT:

Thus the c program to calculate the factorial of a given number was written, entered, executed and the output was verified.

PROGRAM USING RECURSIVE FUNCTION

EX: NO: 09 (b)

SUM OF DIGITS

AIM:

To write a c program to find the sum of digits of a given number

ALGORITHM:

Step 1: Get the number

Step 2: Call the function sum by passing number as an argument

Step 3: Inside the sum()

a. If the received value is less than 10return that value
b. If the received value is greater than 10
c. Extract the remainder of above value by dividing 10
d. Add the remainder with return value of sum by passing value/10 as an argument

Step 4: Print the result by receiving the return value of sum()

PROGRAM:

```
#include<stdio.h>
int sum(int n,int s)
{
        if(n<10)
                return n;
        else
                return (n%10)+sum(n/10,s);

}
main()
{
        int n,s=0;
        printf("\nEnter a Number\n");
```

VAIGAI COLLEGE OF ENGG Department of CSE

```
        scanf("%d",&n);
        s=sum(n,s);
        printf("\nThe sum of digits %d is %d\n",n,sum(n,s));
}
```

OUTPUT:

```
Enter a Number
46612
The sum of digits 46612 is 19
```

VAIGAI COLLEGE OF ENGG - Department of CSE

RESULT:

Thus the c program to find the sum of digits of a given number was written, entered, executed and the output was verified

PROGRAM USING STRUCTURES AND UNIONS

EX:NO: 10 (a)

STUDENT RECORD

AIM:

To write a c program to maintain the student record using structures.

ALGORITHM:

Step 1: Create a structure student with roll no, name, dept and 3 marks as fields

Step 2: Create a structure variable

Step 3: Read the input for student details

Step 4: Calculate the average of student by using 3 marks

Step 5: Print the structure elements using structure variable

PROGRAM:

```
#include<stdio.h>
struct student
{
        int rno,m1,m2,m3;
        float avg;
        char name[20],dept[10];
};
main()
{
        struct student s;
        printf("Enter the Student Details:\n");
        printf("Enter the Stuent roll no:\n");
        scanf("%d",&s.rno);
        printf("Enter the Stuent Name:\n");
        scanf("%s",&s.name);
        printf("Enter the Stuent Dept:\n");
        scanf("%s",&s.dept);
        printf("Enter the 3 marks:\n");
        scanf("%d%d%d",&s.m1,&s.m2,&s.m3);
```

VAIGAI COLLEGE OF ENGG - Department of CSE

```
        s.avg=(s.m1+s.m2+s.m3)/3;
        printf("The Student Average is :%f\n",s.avg);
}
```

OUTPUT:

```
Enter the Student Details:
Enter the Stuent roll no:
12
Enter the Stuent Name:
Kumar
Enter the Stuent Dept:
CSE
Enter the Stuent marks:
40
18
90
The Student Average is :49.000000
```

VAIGAI COLLEGE OF ENGG - Department of CSE

RESULT:

Thus the c program to maintain the student record using structures was written, entered, executed and the output was verified.

PROGRAM USING STRUCTURES AND UNIONS

EX:NO: 10 (b)

ARRAY OF STRUCTURES

AIM:

To write a c program to maintain various number of students record using array of structures.

ALGORITHM:

Step 1: Create a structure student with roll no, name, dept and 3 marks as fields

Step 2: Create a structure variable with necessary size

Step 3: Read the total number of students

Step 3: Read the structure details for all students

Step 4: Calculate the average of students by using 3 marks

Step 5: Write a function to print the student details if roll no. is given

Step 6: Inside the function

a. Initialize the index to 0
b. Compare the roll no. with structure index roll no.
c. If they are same print the student details
d. Increment the index by 1
e. Repeat the steps b to d until index reaches the total no. of students

PROGRAM:

```
#include<stdio.h>
struct student
{
        int rno,m1,m2,m3;
        float avg;
        char name[20],dept[10];
};
void find_student(int a,struct student s[],int n)
```

VAIGAI COLLEGE OF ENGG Department of CSE

```
{
        int i;
        printf("The Student Detail of %d\n",a);
        for(i=0;i<n;i++)
        {
                if(s[i].rno==a)
                {


        printf("%s\t%s\t%d\t%d\t%d\t%f\n",s[i].name,s[i].dept,s[i].m1,s[i].m
                                                                2,s[i].m3,s[i].avg);
                        break;
                }
        }
}
main()
{
        int i,n,rno;
        struct student s[10];
        printf("Enter total no. of Students\n");
        scanf("%d",&n);
        for(i=0;i<n;i++)
        {

                printf("Enter the Student %d Details:\n",(i+1));
                printf("Enter the roll no:\n");
                scanf("%d",&s[i].rno);
                printf("Enter the Name:\n");
                scanf("%s",&s[i].name);
                printf("Enter the  Dept:\n");
                scanf("%s",&s[i].dept);
                printf("Enter the 3 marks:\n");
                scanf("%d%d%d",&s[i].m1,&s[i].m2,&s[i].m3);
                s[i].avg=(s[i].m1+s[i].m2+s[i].m3)/3;
        }
        printf("Enter the rollno to find:\n");
        scanf("%d",&rno);
        find_student(rno,s,n);
}
```

VAIGAI COLLEGE OF ENGG - Department of CSE

OUTPUT:

```
Enter total no. of Students
2
Enter the Student 1 Details:
Enter the roll no:
12
Enter the Name:
Kumar
Enter the  Dept:
cse
Enter the 3 marks:
45
67
88
Enter the Student 2 Details:
Enter the roll no:
13
Enter the Name:
Prabhu
Enter the  Dept:
cse
Enter the 3 marks:
77
89
67
Enter the rollno to find:
13
The Student Detail of 13
Prabhu  cse    77     89     67     77.000000
```

VAIGAI COLLEGE OF ENGG - Department of CSE

RESULT:

Thus the c program to maintain various number of students record using array of structures was written, entered, executed and the output was verified.

PROGRAM USING STRUCTURES AND UNIONS

EX: NO: 10 (c)

PROGRAM FOR SIZE OF UNION

AIM:

To write a c program to store the book information using union.

ALGORITHM:

Step 1: Create an union book

Step 2: Declare one integer and a character array inside the union for book name and book price

Step 3: Get the book name and price

Step 4: Print the book name and price

Step 5: Get the Book name alone

Step 6: Print the Book name

Step 7: Get the Book Price

Step 8: Print the Book price

PROGRAM:

```
#include<stdio.h>
union book
{
        int price;
        char bname[20];
};

main()
{
        union book b;
        printf("Enter the Book Details:\n");
```

VAIGAI COLLEGE OF ENGG - Department of CSE

```
        printf("Enter the Book Name:\n");
        scanf("%s",&b.bname);
        printf("Enter the Book Price:\n");
        scanf("%d",&b.price);
        printf("BOOK DETAILS:\n");
        printf("%s\t%d\n",b.bname,b.price);
        printf("Enter the Book Name:\n");
        scanf("%s",b.bname);
        printf("Book Name=%s\n",b.bname);
}
```

OUTPUT:

```
Enter the Book Details:
Enter the Book Name:
English
Enter the Book Price:
150
BOOK DETAILS:
û      150
Enter the Book Name:
English
Book Name=English
```

VAIGAI COLLEGE OF ENGG - Department of CSE

RESULT:

Thus the c program to store the book information using union was written, entered, executed and the output was verified.