

Sum : Programs to understand the basic data type and I/O

Program 1:

```
# include <stdio.h>
# include <conio.h>
void main ()
{
    char name [50];
    char add [50];
    int roll_no;
    float percent;
    char grade;
    char mob [10];
    clrscr ();
    print f ("**** demonstration of various
    data types ****");
    print f ("\n Name of the student : ");
    gets (name);
    print f ("\n Address of the student : ");
    scan f ("% s", & add);
    print f ("\n Roll no of student : ");
    scan f ("% d", & roll_no);
    print f ("\n Percentage of student : ");
    scan f ("% . +", & percent);
    print f ("\n grade of student : ");
    scan f ("% s", & grade);
}
```

Output:

24

```
***** demonstration of various data types ****
Name of the student : shamika yadav
Address of the student : Mumbai
Roll no of the student : 1754
Percentage of the student : 701
Grade of the student : A
Mobile no of the student : 8108497301
Student name : shamika yadav
Student address : Mumbai
Student roll no : 1754
Student percent : 701.
Student grade : A
Student mobile no : 8108497301
```

Output:

enter the side: 5
area of a square: 25

```
print f("\n mobile no. :");
scanf (" %d ", & mob);
print f("\n student name: %s ", name);
print f("\n student address: %s ", add);
printf ("\n student roll_no: %d ", roll_no);
print f ("\n student percent: %f ", percent);
print f ("\n student grade: %c ", grade);
print f ("\n student mobile_no: %d ", mob);
getch();
```

Program : 2

Source code

```
#include < stdio.h>
#include < conio.h>
void main ()
{
    int side, area;
    clrscr ();
    print f ("Enter the side \n");
    scanf (" %d ", & side);
    area = side * side;
    print f ("\n area of square %d ", area);
    getch ();
```

*Om
22/10/2020*

Aim: write C programs which will show the use of various different types of operators.

1 Arithmetic operators

Source code:-

```
#include <stdio.h>
#include <conio.h>
void main ()
{
    int num1, num2, add, sub, mul, div;
    clrscr ();
    printf ("Enter 1st number:");
    scanf ("%d", &num1);
    printf ("Enter 2nd number:");
    scanf ("%d", &num2);
    add = num1 + num2;
    printf ("Addition of 2 numbers: %d
            \n", add);
    sub = num1 - num2;
    printf ("Subtraction of 2 numbers: %d
            \n", sub);
    mul = num1 * num2;
    printf ("Multiplication of 2 numbers: %d
            \n", mul);
    div = num1 / num2;
    printf ("Division of 2 numbers: %d", div);
    getch ();
}
```

Output:

```
Enter 1st number: 8
Enter 2nd number: 2
Addition of 2 numbers: 10
Subtraction of 2 numbers: 6
Multiplication of 2 numbers: 16
Division of 2 numbers: 4
```

2 Logical Operators :-

```

#include <stdio.h>
#include <conio.h>
void main()
{
    int x,y,z,value1,value2,value3,value4,value5;
    clrscr();
    printf("Enter 1st value:");
    scanf("%d", &x);
    printf("Enter 2nd value:");
    scanf("%d", &y);
    printf("Enter 3rd value:");
    scanf("%d", &z);
    value1 = (x < y) & (z > y);
    printf("value1 is : %d \n", value1);
    value2 = (x == y) & (z < y);
    printf("value2 is : %d \n", value2);
    value3 = (x < y) & (z == y);
    printf("value3 is : %d \n", value3);
    value4 = !(x == y);
    printf("value4 is : %d \n", value4);
    value5 = (x == y);
    printf("value5 is : %d \n", value5);
    getch();
}

```

3 Ternary operators

```
#include <conio.h>
#include <stdio.h>
void main ()
{
    int a=100, b=20, c=50, big ;
    clrscr ();
    big = a>b? a:c ;
    printf ("The biggest number is : %d", big);
    getch ();
}
```

Output:

The biggest number is 100.

28

Output:

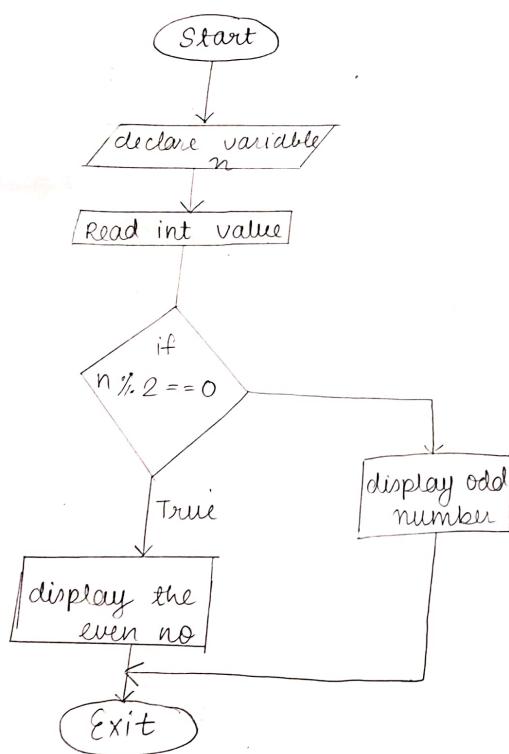
Enter a number : 26

Even number

Enter a number : 53

Odd number

flow chart:



Practical: 03

29

dim: Decision statements

WAP to find out odd and even numbers:

algorithm:

- I Start
- II Take input (Read a number from the user)
- III Check if number % 2 == 0 then print even number
- IV Exit

source code:

```
# include <stdio.h>
# include <conio.h>
void main()
```

```
int n;
clrscr();
printf("enter a number:");
scanf("%d", &n);
if (n % 2 == 0)
{
    printf("Even number!");
}
```

```
else
{
    printf("Odd number!");
}
```

85

```
#include <stdio.h>
int main()
{
    int year;
    printf("Enter a year: ");
    scanf("%d", &year);
    if (year % 4 == 0)
        if (year % 100 != 0)
            if (year % 400 == 0)
                printf("Leap Year");
            else
                printf("Not a leap year");
        else
            printf("Not a leap year");
    else
        printf("Not a leap year");
}
```

WAP to find the entered year is a leap year or not!

Algorithm:

- I Start
- II Take input (Read year from the user)
- III if year $\% 4 = 0$ and year $\% 400 \neq 0$ or
year $\% 4 = 0$ and year $\% 100 \neq 0$ print
NOT A LEAP YEAR.
- IV EXIT

Source code:

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int year;
    clrscr();
    printf("Enter a year: ");
    scanf("%d", &year);
    if (year % 4 == 0)
        if (year % 100 == 0)
            if (year % 400 == 0)
                printf("Leap Year");
            else
                printf("Not a leap year");
        else
            printf("Not a leap year");
    else
        printf("Not a leap year");
}
```

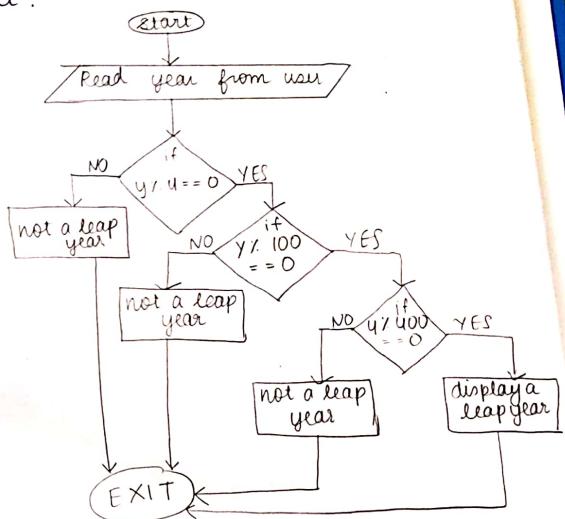
Output :

Enter a year : 2017
Not a leap year

Enter a year: 2020
leap year

30

Flow chart :



```
if (year % 400 == 0)
{
    printf ("leap year!")
}
else
{
    printf ("NOT a leap year")
}
else
{
    printf ("NOT a leap year")
}
else
{
    printf ("NOT a leap year")
}
getch();
```

Q8

WAP to find whether the character is vowel or consonant

Algorithm:

- Step 1: start
- Step 2: Take input (Read characters value from user)
- Step 3: (Check) if value == 'a' || value == 'e' || value == 'i' || value == 'o' || value == 'u' || value == 'A' || value == 'E' || value == 'I' || value == 'O' || value == 'U'
- Step 4: exit

Source code:

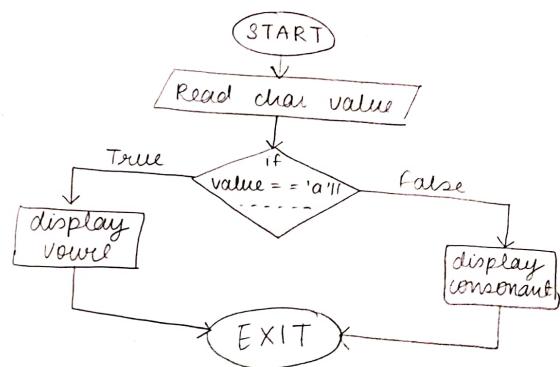
```
#include <stdio.h>
#include <conio.h>
void main()
{
    char a;
    clrscr();
    printf("Enter the alphabet:");
    scanf("%c", &a);
    if (a == 'a' || a == 'e' || a == 'i' || a == 'o' ||
        a == 'u' || a == 'A' || a == 'E' || a == 'I' ||
        a == 'O' || a == 'U')
    {
        printf("vowel");
    }
}
```

Output:

Enter a alphabet : o
vowel
Enter a alphabet : x
consonant

32

Flow chart :



```
else  
{  
    printf("consonant");  
}  
getch();
```

Ex

Practical: 04

Sum : loop statements

- a) WAP to print all even numbers from 1 to 50 using while loop.

source code:

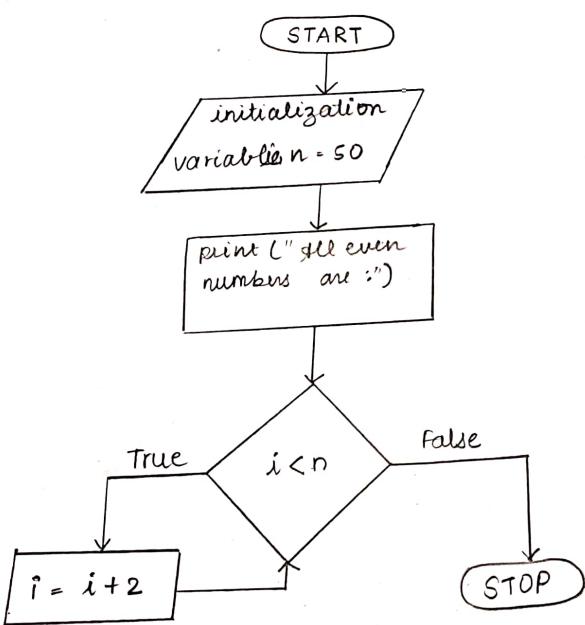
```
# include <stdio.h>
# include <conio.h>
void main ()
{
    int i, n = 50;
    clrscr();
    printf("all even numbers from 1 to 50 are:
           \n", n);
    i = 2;
    while ( i <= n )
    {
        printf("%d \n", i);
        i = i + 2;
    }
    getch();
}
```

Output:

All even numbers from 1 to 50 are:

2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
32
34
36
38
40
42
44
46
48
50

34



35

algorithm :

- I start
- II initialise two variable with static variable where $n = 50$ and $i = 2$
- III Use while loop for printing the even number upto the range 50
- IV adding 2 to current even number will get next even number.
- V display the appropriate output
- VI exit

Q. b) WAP to print odd numbers between 1 to 50 using do while loop

source code:

```
# include <stdio.h>
# include <conio.h>
void main()
{
    int i, n = 50;
    clrscr();
    printf("odd numbers from 1 to 50 are:
           \n", n);
    i = 1;
    do
    {
        if (i % 2 == 1)
        {
            printf("%d \n", i);
            i++;
        }
    } while (i <= n);
```

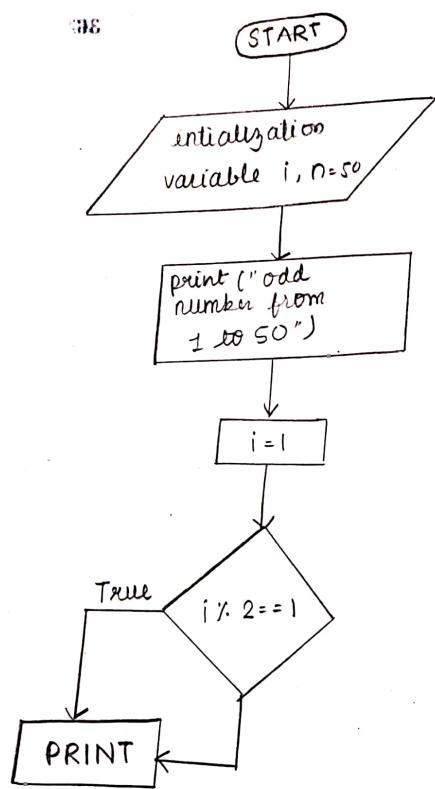
output:

odd numbers from 1 to 50 are:

```
1
3
5
7
9
11
13
15
17
19
21
23
25
27
29
31
33
35
37
39
41
43
45
47
49
```

36

36



37

Algorithm :

- I Start
- II Initialise two variable : $n=50$ $i=1$
- III Use do while loop for taking number from 1 to 50.
- IV Use if condition statement to check whether given number is even or odd.
- V Increment the value of i .
- VI display the appropriate output
- VII STOP.

58

WAP to print sum of all even number between 1 to n using for loop.

source code:

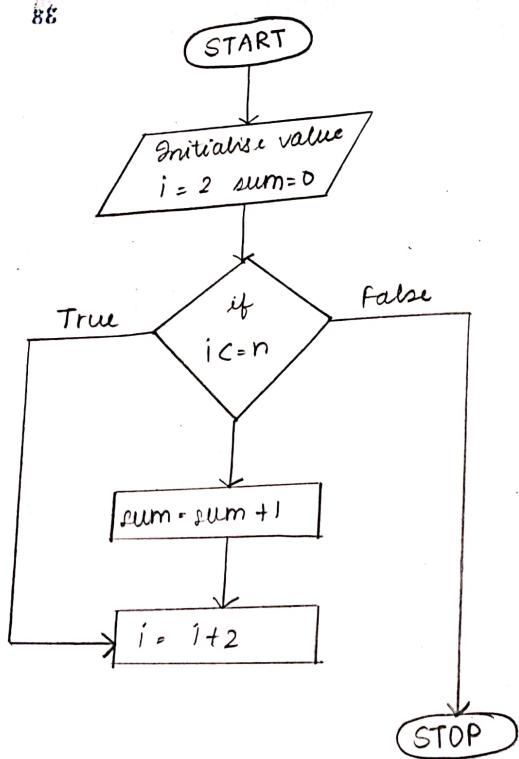
```
# include <stdio.h>
# include <conio.h>
void main()
{
    int, i, n, sum=0;
    clrscr();
    printf ("enter the range :");
    scanf ("%d", &n);
    for (i = 2; i < n; i = i + 2)
    {
        sum = sum + i;
    }
    printf ("sum of all even number
            upto the range are : ", sum);
    getch();
}
```

Output :

Enter the range: 10

sum of all even number upto the range are: 30

38



algorithm:

- I start
- II Initialise the variable and one is dynamic $i = 2, sum = 0; n;$
- III Use for loop for check the condition satisfy the given range
- IV Add current even number.
- V display the appropriate output.
- VI stop

es

Practical : 05

Topic: array statements

1. Aim: WAP to print the input array elements.

Source code:

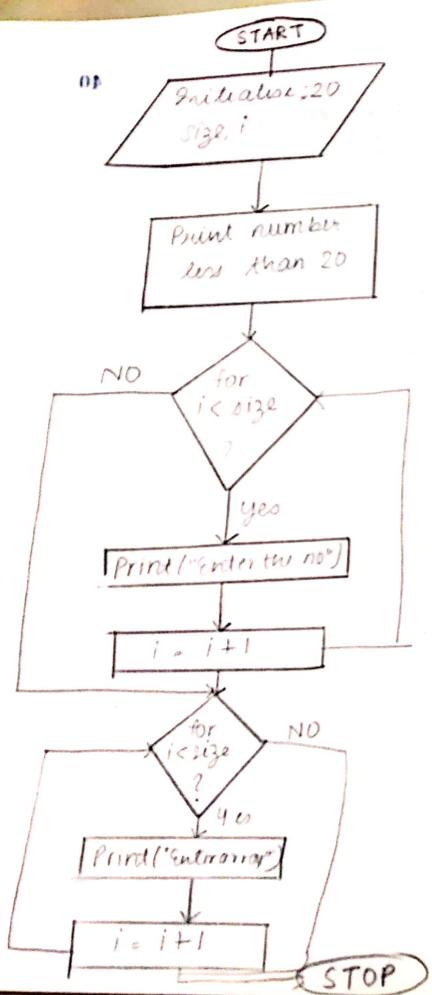
```
#include < stdio.h>
#include < conio.h>
void main()
{
    int a[20];
    int size, i;
    clrscr();
    printf("Enter number less than 20:");
    scanf("%d", &size);
    for (i=0; i< size; i++)
    {
        printf("Enter number less than 20:");
        scanf("%d", &size);
    }
    printf("\n the displayed array \n:");
    for (i=0; i< size; i++)
    {
        printf("Enter a [%d] %d", i);
    }
    getch();
}
```

Output:

```
Enter the number less than 20: 5 40
Enter the a [0] no. element 85
Enter the a [1] no. element 2
Enter the a [2] no. element 1
Enter the a [3] no. element 4
Enter the a [4] no. element 8
```

The displayed array:

```
Enter a [0] 5
Enter a [1] 2
Enter a [2] 4
Enter a [3] 1
Enter a [4] 8
```



41

Algorithm:

- I Start
- II declare an array of user specified size
- III Initialise two variables of integer type i.e size and;
- IV Take range from the user that to be printed which should be less than the specified size of an array.
- V Use nested for conditional loop for printing the elements in arrays according to its indexing.
- VI Print the appropriate output
- VII Stop.

2. dim : To find the sum of elements of the arrays.

Source code:

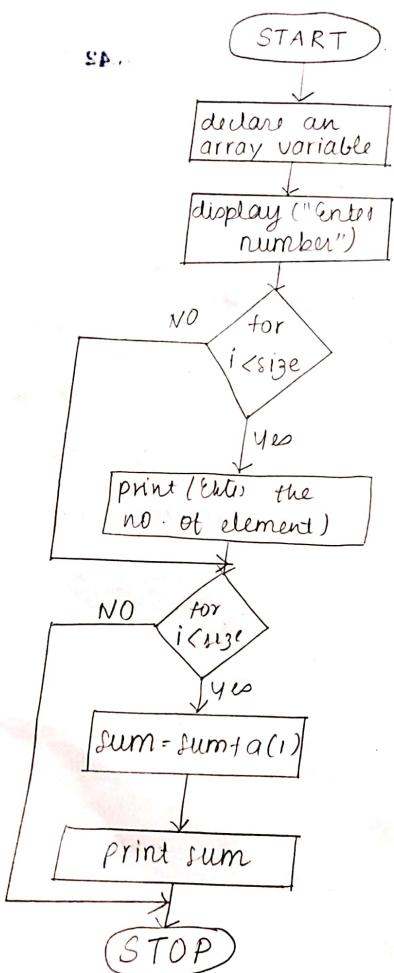
```
#include <conio.h>
#include <stdio.h>
void main()
{
    int a[20], sum=0, size, i;
    clrscr();
    printf("Enter number less than 20:");
    scanf("%d", &size);
    for (i=0; i<size, i++)
    {
        printf("Enter the a[%d] no element", i);
        scanf("%d", &a[i]);
    }
    printf("In the displayed array\n");
    for (i=0; i<size; i++)
    {
        sum = sum + a[i];
    }
    printf("sum of the arrays: [%d]\n", sum);
    getch();
}
```

Output:

Enter the number less than 20: 5
Enter the a [0] no element 2
Enter the a [1] no element 3
Enter the a [2] no element 1
Enter the a [3] no element 2
Enter the a [4] no element 3

42

The displayed array:
sum of the array [11]



algorithm

43

- I Start
- II declare an array type of user specified size.
- III Initialize three variable one of static type and two of dynamic type i.e. sum=a, i, size.
- IV Take range from the user that to be printed & add, which should be less than the specified size of an array.
- V Use nested for conditional loop for printing the elements in array according to its indexing.
- VI Adding the elements of the array.
- VII Print the appropriate output.
- VIII Stop.

3 dim: WAP to find out fibonacci series using array.

source code:

```
# include <stdio.h>
# include <conio.h>
void main()
{
    int a[20], n, i;
    clrscr();
    printf("enter the no. of terms\n");
    scanf("%d", &n);
    a[0] = 0;
    a[1] = 1;
    for (i=2; i<n; i++)
    {
        a[i] = a[i-2] + a[i-1];
    }
    printf("The fibonacci series upto
    %d term is\n", n);
    for (i=0; i<n; i++)
    {
        printf("%d \t", a[i]);
    }
    getch();
}
```

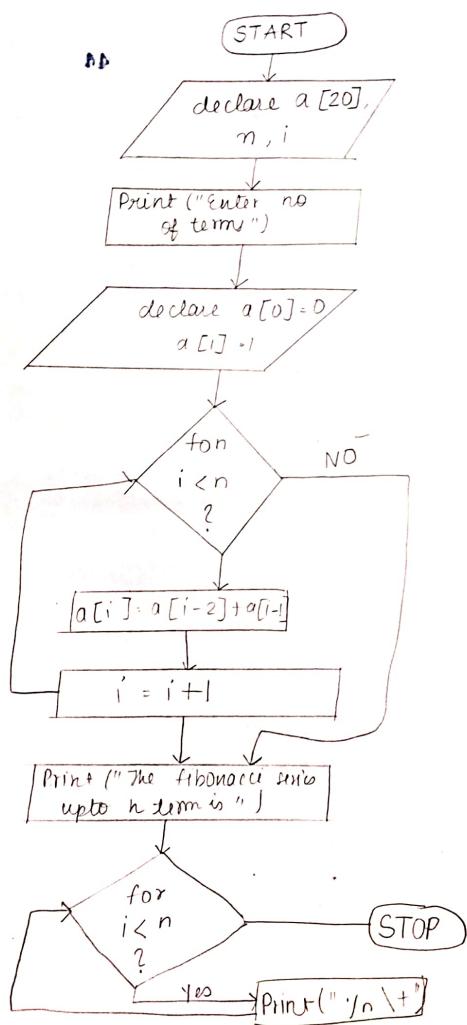
Output:

Enter the no. of terms

The fibonacci series upto 7 terms is

0 1 1 2 3 5 8

44



Algorithm:

- 45
- I Start
 - II declare an array of integer type of specified size by the user.
 - III Initialise two variables of dynamic variable i.e., i, n
 - IV Take the no. of terms from the user what the no. should be printed
 - V Initialise indexing value of $a[0] = a$ and $a[1] = 1$ for printing the fibonacci series.
 - VI Using the for condition loop for looping of numbers.
 - VII Indexing value of present array is equal to previous indexing value
 - VIII Print the fibonacci series upto the term given by the user.
 - IX Use for loop for printing the output in tabular form.
 - X Print the appropriate output- STOP.

4 dim: WAP to represent a multidimensional array in matrix input

Source code:

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int a[20][20];
    int row, col, i, j;
    clrscr();
    printf("Enter no. of rows:");
    scanf("%d", &row);
    printf("Enter no. of columns:");
    scanf("%d", &col);
    for (i = 0; i < row; i++)
    {
        for (j = 0; j < col; j++)
            printf("Enter the a[%d][%d] no element: ", i, j);
        scanf("%d", &a[i][j]);
    }
    printf("The displayed matrix is\n");
    for (i = 0; i < row; i++)
        for (j = 0; j < col; j++)
            printf("%d ", a[i][j]);
}
```

Output:

46

Enter the no. of rows : 2

Enter the no. of columns : 2

Enter the a[0][0] no. element = 1

Enter the a[0][1] no. element = 5

Enter the a[1][0] no. element = 4

Enter the a[1][1] no. element = 2

The displayed matrix is :

1 5

4 2

Algorithm :

```
{
    printf("\t\t%d", a[i][j]);
    printf("\n");
}
getch();
```

Algorithm :

- I start
- II declare multi-dimensional array and row, column, i and j
- III display the enter no of rows
- IV scan the same.
- V similarly for columns.
- VI use the for conditional for accessing the array elements.
- VII use another for loop for displaying the array values
- VIII stop.

Topic - Functions

1 WAP to find factorial of a number using function.

Source code:

```
#include <stdio.h>
#include <conio.h>
void factorial(int);
void main()
{
    int a;
    clrscr();
    printf("enter the number : ");
    scanf("%d", &a);
    factorial(a);
}

void factorial(int n)
{
    int i, fact = 1;
    for (i = 1; i <= n; i++)
    {
        fact = fact * i;
    }
    printf("factorial is : %d", fact);
    getch();
}
```

Output :

enter the number : 8
factorial is : 40320

2) WAP to find sum of digits of entered number.

Source Code:

```
#include <stdio.h>
#include <conio.h>
void abc (int n);
void main()
{
    int n ;
    clrscr ();
    printf ("\n Enter number : ");
    scanf ("%d", &n );
    abc (n );
    getch ();
}
```

void abc (int n)

{

~~int s~~

int r, <= 0;

while (n != 0)

{

r = n % 10;

s = s + r;

n = n / 10;

}

} printf ("\n sum of digits = %d", s);