

Output: ----- Demonstrate various datatypes -----

Name of student:

Tanvi

Address of the student:

Las Vegas

Roll no of student

12

Percentage of the student

90.04

Grade of student

A

Mobile No

1234567890

student name: Tanvi

student address: Las Vegas

student rollno: 12

student percent: 90.04

student grade: A

student mobileno: 1234567890

PRACTICAL - 01

AIM: To study the use of different types of datatypes

SOURCE CODE:

```
#include <stdio.h>
#include <conio.h>
void main()
{
    char name[50];
    char add[50];
    int rollno;
    float percent;
    char grade;
    long int mob;
    clrscr();
    printf("----- Demonstrate various
    datatypes ----- \n");
    printf("Name of the student \n");
    scanf("%s", &name);
    printf("Address of the student \n");
    scanf("%s", &add);
    printf("Roll no of the student \n");
    scanf("%d", &rollno);
    printf("Percentage of student \n");
    scanf("%f", &percent);
}
```

```

printf ("Grade of student In");
scanf ("%s", &grade);
printf ("Mobile no In");
scanf ("%ld", &mob);
printf ("In student name: %s", name);
printf ("In student address: %s", add);
printf ("In student rollno : %d", roll-no);
printf ("In student percent: %f", percent);
printf ("In student grade: %c", grade);
printf ("In student mobileno : %d", mob);
getch();
}

```

Program - 2

Area of circle

SOURCE CODE:

```

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

```

```

float r;
float pi = 3.14;
float area;
clrscr();
printf ("Enter radius In");
scanf ("%f", &r);
area = pi * r * r;
printf ("Area of circle = %f", area);
getch();
}

```

output:

Enter radius

4

Area of circle: ~~50.240002~~

28

Sri
29/11/19

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

PRACTICAL - 2

a) AIM: Write a C program which will show the use of various different types of operators.

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();

```

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();
```

Amrit
06/10/19

output:

enter 1st value : 9

enter 2nd value : 8

Enter 3rd value : 2

value 1 is : 0

value 2 is : 1

value 3 is : 1

value 4 is : 0

value 5 = 1

~~output: The biggest number is 100~~

Ternary operator

```
#include <conio.h>
```

```
#include <stdio.h>
```

```
void main()
```

```
{
```

```
int a=100, b=20, c=50, big;  
clrscr();
```

```
big = a>b || a>c ? a:b;
```

```
printf ("The biggest number is: %.d", big);  
getch();
```

```
}
```

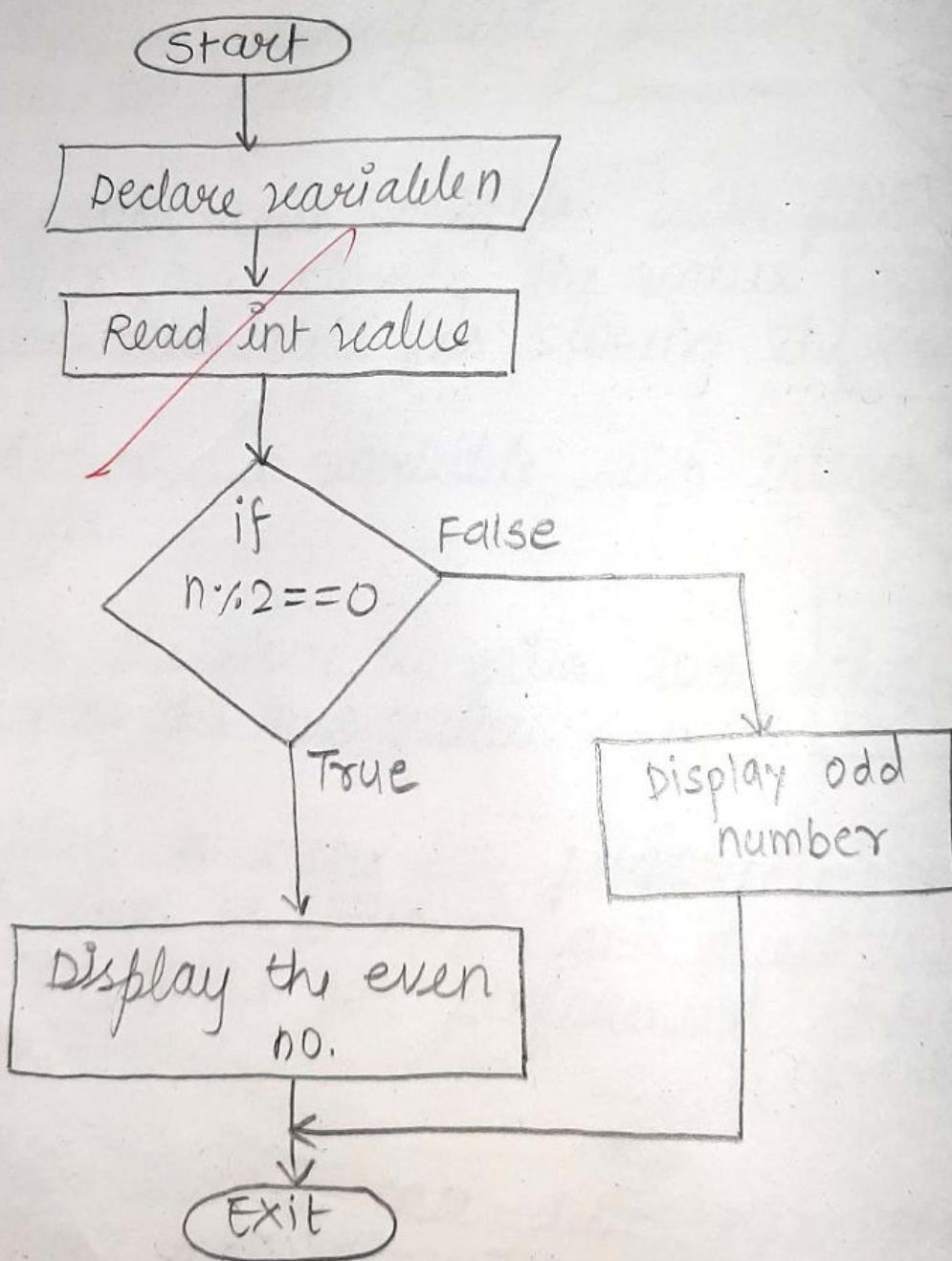
Fri 17/10/2020

Output:

Enter a number : 26
Even number

Enter a number : 57
Odd number

Flowchart:



PRACTICAL - 3

AIM: Decision statements

write a program to find out odd & even numbers:

ALGORITHM:

- step 1 : start
- step 2 : [take input] Read a number from the user
- step 3 : check if $\text{number} \% 2 == 0$ then
print even number.
- step 4 : 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)
        {
            2) printf ("Even number!");
            3)
        }
}
```

```

else
{
    printf(" Odd number : ");
}
getch();
}

```

= write a program to find the entered year is a leap year or not!

ALGORITHM

Step 1: start

Step 2: [Take Input] Read year from the user

Step 3: If $\text{year} \% 4 = 0$ and $\text{year} \% 400 == 0$ or
 $\text{year} \% 4 = 0$ and $\text{year} \% 100 != 0$
 print "Leap year".
 else print NOT A LEAP YEAR

Step 4: EXIT

Source code:

```

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

```

} {

```

int year;
clrscr();

```

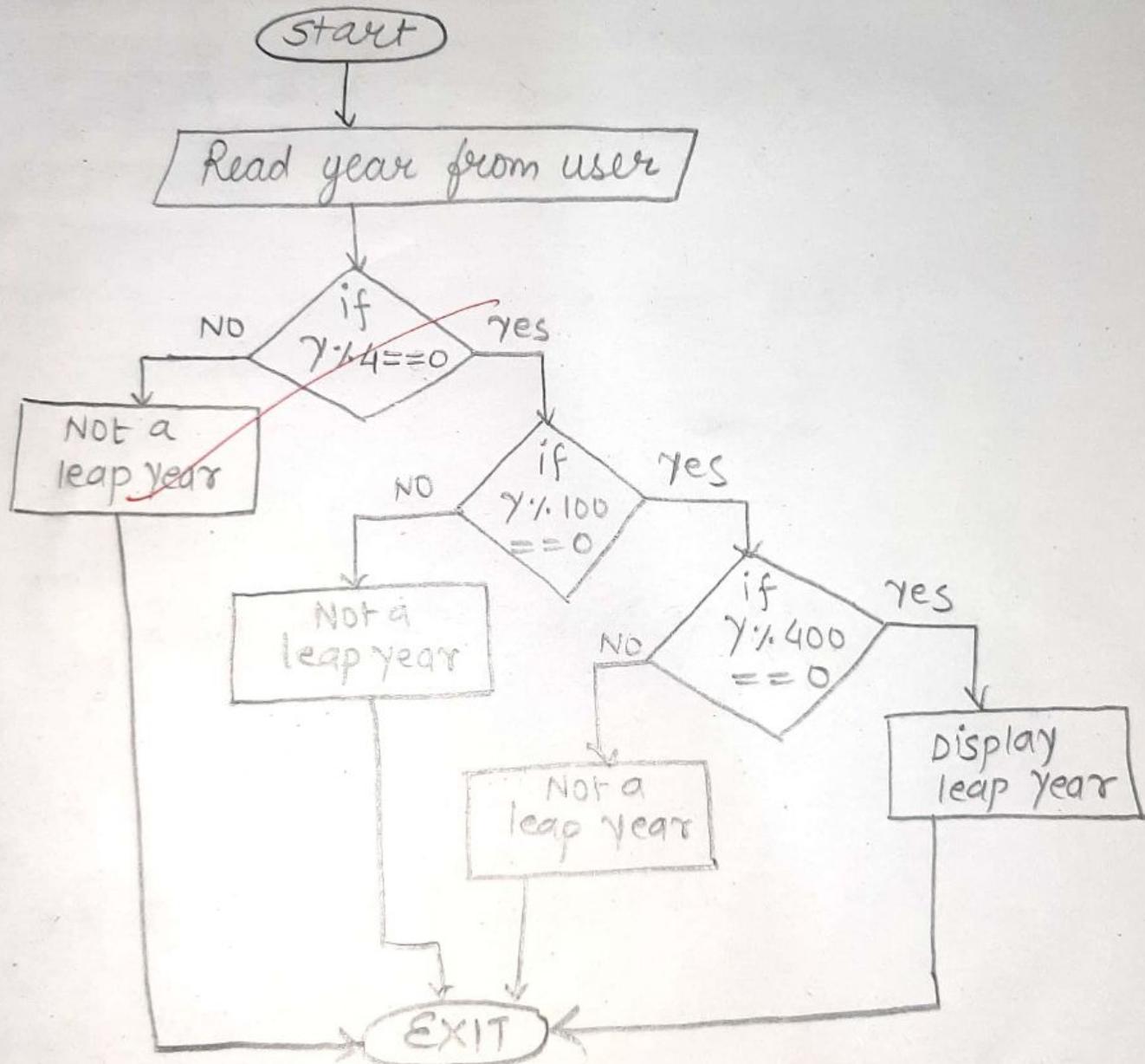
output:

enter a year : 2017
NOT a leap year

34

=
enter a year : 2026
leap year

FLOWCHART:



```

printf("Enter a year: ");
scanf("y-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");
}
getch();
~>

```

8
write a program to find whether the character is vowel or consonant.

ALGORITHM:

Step 1: start

Step 2: [Take Input] Read character 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");
```

→ }

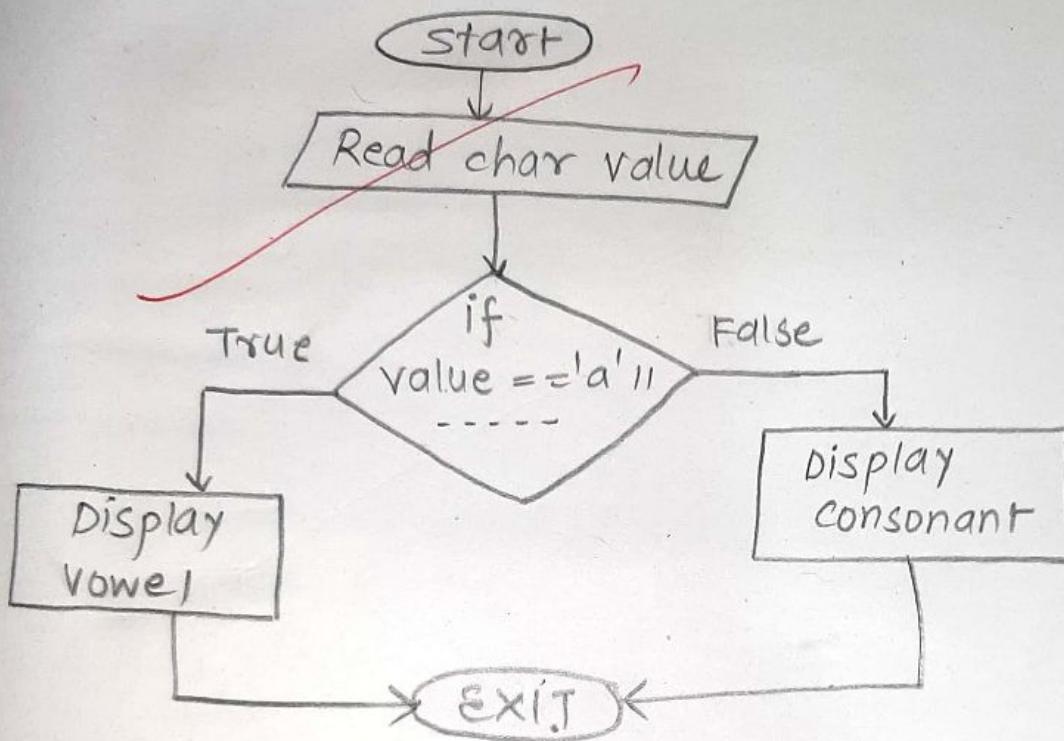
Output:

Enter a alphabet: o
vowel

= Enter a alphabet: x
consonant

36

FLOWCHART:



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

~~Frw:
24/10/11 2020~~

PRACTICAL - 04

AIM: To study use of different loops.

- 1) To print even nos. between 1 to 50 using while loop:

Step 1: start

Step 2: Initialization of variable

Step 3: use while conditional statement with if condition and print the same.

[Iterate the variable]

Step 4: stop

Source code:

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

→ {

```
clrscr(); int i=1;
```

```
clrscr();
```

```
while (i<=50)
```

→ {

```
if (i%2==0)
```

```
→ {
```

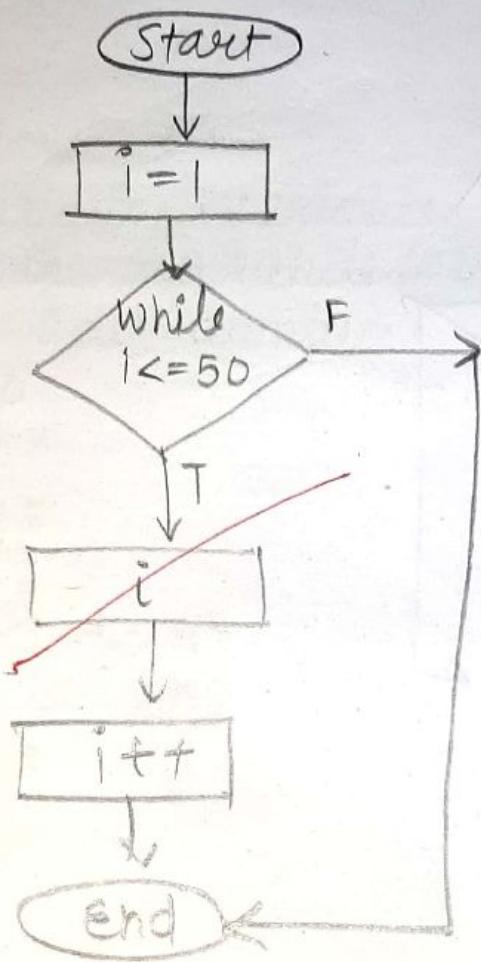
```
printf ("\n%d", i);
```

```
→ }  
i++;
```

→ }

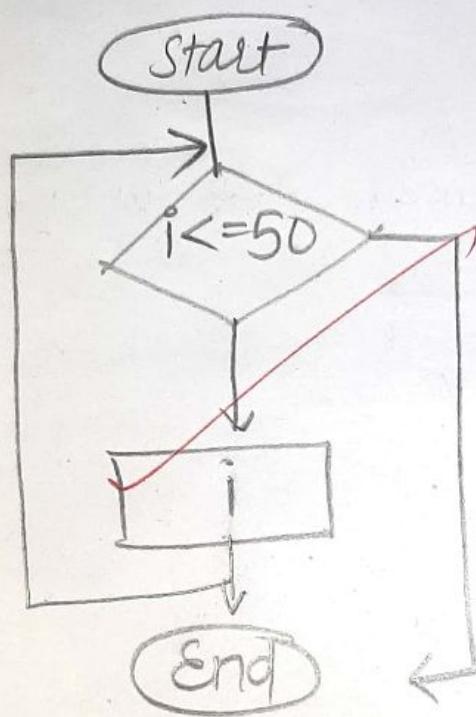
output:

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



Output:

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



\Rightarrow getch();

- 2) Print out odd nos b/w 1 to 50 using do while loop

Step 1: Start

Step 2: Initialize variable

Step 3: Use the do while loop for printing the odd number.

Step 4: Stop

Source code:

```
#include <stdio.h>
#include <conio.h>
```

```
void main()
```

$\Rightarrow \{$

```
int i = 1;
```

```
clrscr();
```

```
do
```

$\Rightarrow \{$

```
printf("%d\n", i);
i = i + 2;
```

$\Rightarrow \}$

```
while (i <= 50);
```

$\Rightarrow \}$

```
getch();
```

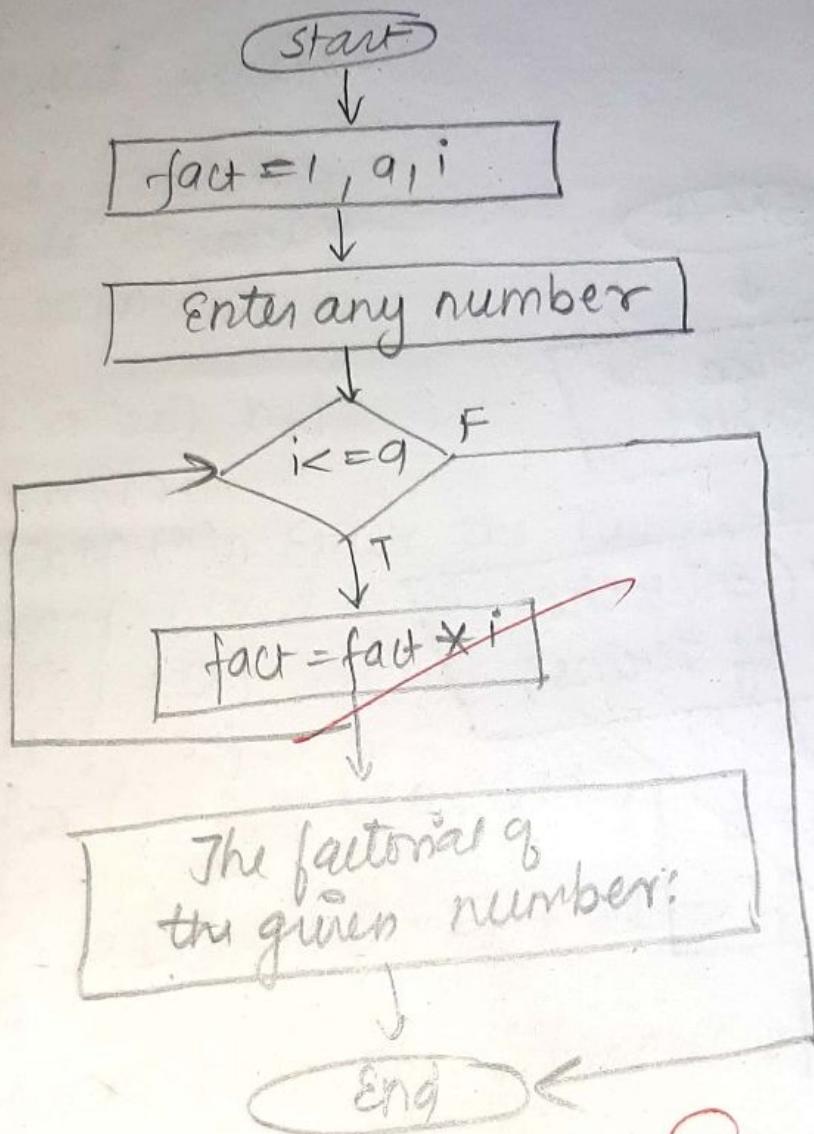
3) To find factorial

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int fact=1, a, i;
    clrscr();
    printf("Enter any number to find factorial : ");
    scanf("%d", &a);
    for(i=1; i<=a; i++)
    {
        fact = fact * i;
    }
    printf("The factorial of the number is : %d");
    getch();
}
```

ALGORITHM:

- Step 1: Start
- Step 2: Initialise the variables.
- Step 3: Display to the user for entering a number.
- Step 4: Accept the number entered from the user.
- Step 5: Use the conditional loop for printing the factorial.
- Step 6: Stop

output:
enter any number to find factorial = 5
The factorial of the number is 120 40

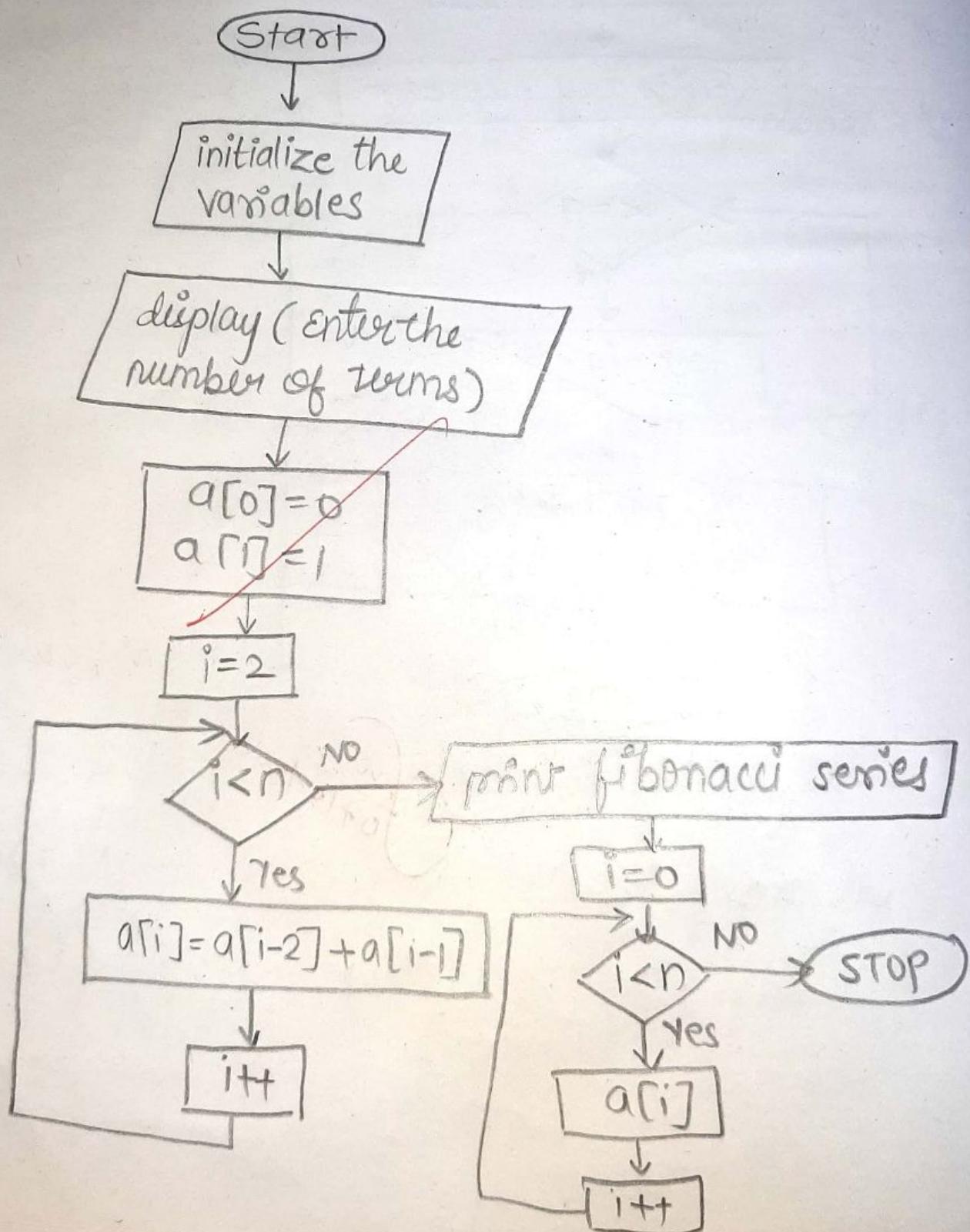


Dr.
07/09/2020

Output :

Enter the number of terms : 4
The fibonacci series upto 4 terms is
0 1 1 2

Flowchart:



PRACTICAL-5

#1: Fibonacci series:

```

#include <conio.h>
#include <stdio.h>
void main()
{
    int a[20], n, i;
    clrscr();
    printf ("\n Enter the number of terms : ");
    scanf ("%d", &n);
    a[0] = 0;
    a[1] = 1;
    for (i=2 ; i<n ; i++)
    {
        a[i] = a[i-2] + a[i-1];
    }
    printf ("\n The fibonacci series upto %d
            term is : %n", n);
    for (i=0; i<n; i++)
    {
        printf ("%d \t", a[i]);
    }
    getch();
}

```

Algorithm:

- step 1: start
- step 2: Initialize the variables
- step 3: Print ("Enter the number till fibonacci series")
- step 4: scan the entered value from the user.
- step 5: use the for conditional loop for the fibonacci series.
i.e. by initialising the 0th & the 1st position.
- step 6: when the condition in for loop is false print or display the appropriate values stored in array to the user.
- step 7: stop.

2: Programs to print inverted half pyramid using * and numbers.

ALGORITHM:

- step 1: start
- step 2: Initialize variables as i, j and rows.
- step 3: display (Enter number of rows) to the user and subsequently scan the value too.
- step 4: using for conditional loop we will initialise value of i as rows and value of i is greater than equal to 1 and decrement

outputs:

* * * * * *

* * * * *

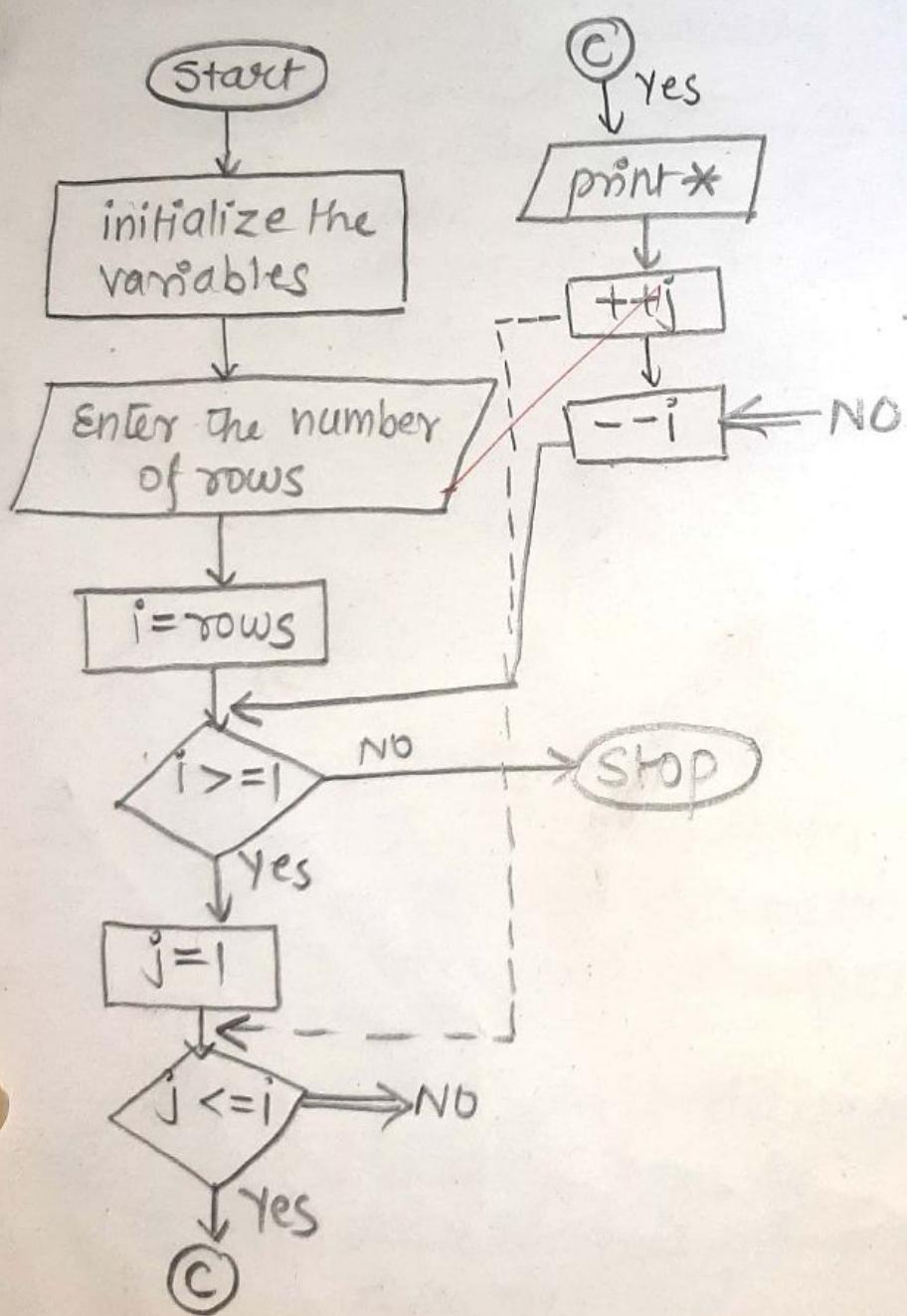
* * * *

* * *

* *

*

Flowchart:



step 5: subsequently use nested for loop that is again a for loop by initializing value of j as 1 and j is less than equal to i and pre-increment j .

step 6: print the start *

step 7: stop.

source code:

```
# include <stdio.h>
# include <conio.h>
void main()
{
    int i, j, rows;
    clrscr();
    printf("Enter number of rows:");
    scanf("%d", &rows);
    for (i = rows; i >= 1; --i)
    {
        for (j = 1; j <= i; ++j)
        {
            printf("*");
        }
        printf("\n");
    }
    getch();
}
```

Output:

Enter elements of 1st matrix:

Enter a11: 2;

Enter a12: 3;

Enter a21: 1;

Enter a22: 4;

Enter elements of 2nd Matrix:

Enter b11: 0;

Enter b12: 1;

Enter b21: 7;

Enter b22: 23;

sum of Matrix:

2 4

8 27

ALGORITHM:

- step 1: start
- step 2: initialise the multi-dimensional array.
- step 3: display to the user to enter to enter values in matrix.
- step 4: use for conditional loop i.e Nested loop to take the elements.
- step 5: according step 4 take (input) the 2nd matrix
- step 6: Add the two matrix using for conditional loop. and print the matrix.
- step 7: stop.

source code:

```
#include <stdio.h>
#include <conio.h>
void main()
{
    float a[2][2], b[2][2], result[2][2];
    clrscr();
    printf ("Enter elements of 1st Matrix : ");
    for (int i=0 ; i<2 ; ++i)
    {
        for (int j=0 ; j<2 ; ++j)
        {
            printf ("Enter b%d%d.%d : ", i+1, j+1);
            scanf ("%f", &b[i][j]);
        }
    }
}
```

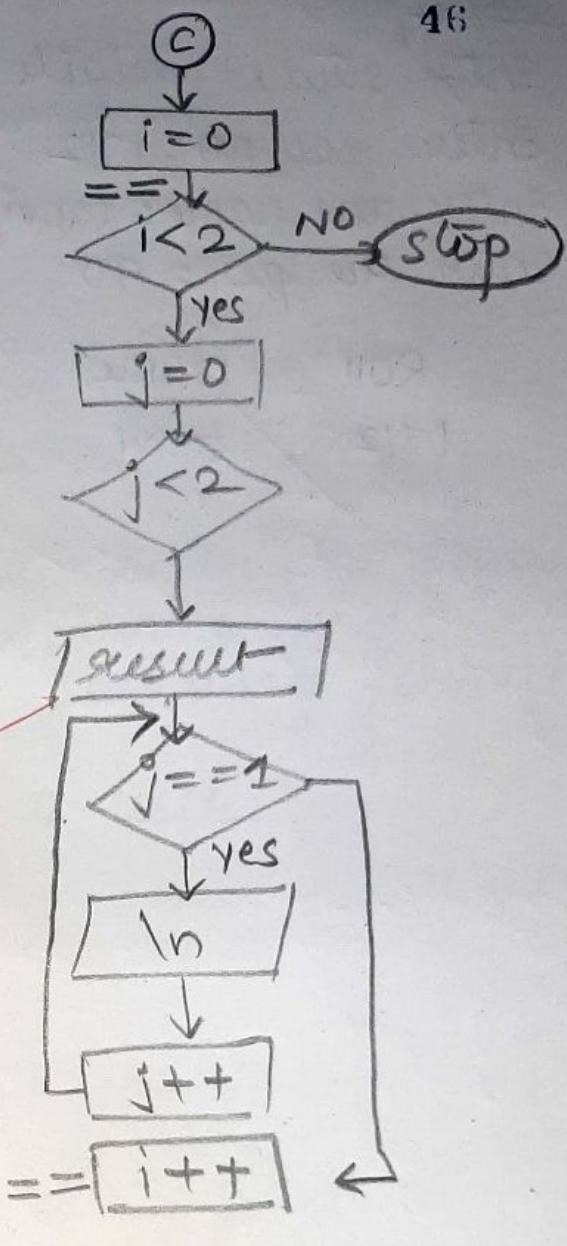
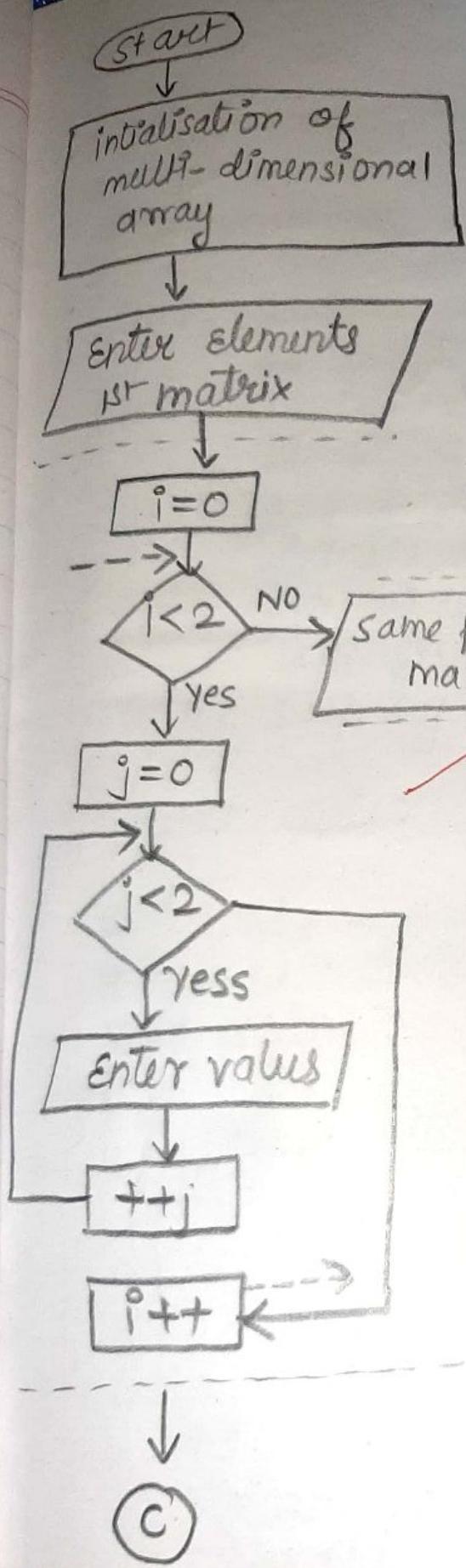
```
for (int i=0; i<2; ++i)
  {
    for (j=0; j<2; ++j)
      result[i][j] = a[i][j] + b[i][j];
  }
```

```
printf("In sum of Matrix: ");
for (int i=0; i<2; ++i)
```

```
  {
    for (int j=0; j<2; ++j)
```

```
    printf("%d\t", result[i][j]);
    if (j==1)
      printf("\n");
  }
```

```
}
```



*Final answer
141212020*

Output:

Enter student details :

Enter roll no : 1712

Enter the name : tanvi

Enter the per : 90

| Roll | name | per |
|------|-------|-----------|
| 1712 | tanvi | 90.000000 |

Practical - 6

Program on structure

step 1: Declare the structure with initialization of variables.

step 2: Call the declared structure with structure object.

step 3: Print to the user to enter the student details as rollno, name and percentage. with following format specifier.

step 4: Display the same to user.

source code:

```
# include <stdio.h>
# include <conio.h>
void main()
{
    struct stud
    {
        char name [20];
        int roll;
        float per;
    };
}
```

```
struct stud s1;
clrscr();
```

```
printf("In Enter student details : ");
printf("In Enter roll no : ");
scanf("%d", &st.roll);
printf("In Enter the name : ");
scanf("%s", &st.name);
printf("Enter the per : ");
scanf("%f", &st.per);
printf("Init ROLL It name It per");
printf("Init %d It %s It %f",
st.roll, st.name, st.per);
getch();
```

#2:

Algorithm:

- step 1: start
- step 2: Declare structure student which will take input as roll number in integer, name in character & percentage in float.
- step 3: Depending upon the number of inputs declare the structure objects.
- step 4: Display to the user to enter roll, Name & Percentage for the 1st user & 2nd user respectively.
- step 5: Display the same by scanning the inputs.

Output: Roll
Name
Percentage

1712

Tanvi

90

1711

Tanmay

91

| Roll | Name | Percentage |
|------|--------|------------|
| 1712 | Tanvi | 90 |
| 1711 | Tanmay | 91 |

source code:

```
#include <stdio.h>
#include <conio.h>
void main()
{
    struct student
    {
        int roll;
        char name[20];
        float per;
    } s1, s2;
    clrscr();
    printf("Enter Roll No Name & Percentage");
    scanf("%d %s %f", &s1.roll, &s1.name, &s1.per);
    scanf("%d %s %f", &s2.roll, &s2.name, &s2.per);
    printf("\n%d %s %f", s1.roll, s1.name, s1.per);
    printf("\n%d %s %f", s2.roll, s2.name, s2.per);
    getch();
}
```

#35

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

 {

```
        struct employee
    → {
```

```
        int id;
        char name [30];
        char add [30];
```

```
    → };
```

```
    struct employee e[50];
```

```
    int size, i;
```

```
    clrscr();
```

```
    printf ("In Enter how many records you  
want to enter: ");
```

~~```
 scanf ("%d", &size);
```~~~~```
    for(i=1; i<=size; i++)
```~~

 → {

```
        printf ("In Enter the ID: ", i);
```

~~```
 scanf ("%d", &e[i].id);
```~~

```
 printf ("In Enter the name: ", i);
```

~~```
        scanf ("%s", e[i].name);
```~~

```
        printf ("In Enter the address: ");
```

~~```
 scanf ("%s", e[i].add);
```~~

```
 printf ("In Employee record is: ");
```

```
 printf ("In ID %s Name %s Address %s",
```

output:

- 1 enter the ID : 1
- 1 enter the name : Tanvi
- 1 enter the address : Canada
- 2 enter the ID : 2
- 2 enter the name : Tanmay
- 2 enter the address : California
- 3 enter the ID : 3
- 3 enter the name : Nikhil
- 3 enter the address : ~~cape town~~

employee record is :

| ID | NAME   | ADDRESS    |
|----|--------|------------|
| 1  | Tanvi  | CANADA     |
| 2  | Tanmay | CALIFORNIA |
| 3  | Nikhil | CAPE TOWN  |

```
for(i=1; i<=size; i++)
 {
 cout<<endl<<i%10<<i%100<<i%1000<<e[i].id<<e[i].name,
 e[i].addr;
 }
 getch();
}
```

## PRACTICAL - 7

#1: call by value :

```
#include <stdio.h>
#include <conio.h>
int sample (int, int)
void main()
{
 int x, y, z;
 clrscr();
 printf("Enter the value of x:");
 scanf ("%d", &x);
 printf(" Enter the value of y:");
 scanf ("%d", &y);
 z = x + y;
 printf("In before function call the number are:");
 printf ("In x=%d It y=%d It z=%d", x, y, z);
 z = sample (x, y);
 printf(" after function call the numbers are:");
 printf ("In x=%d It y=%d It z=%d", x, y, z);
 getch();
}
```

```
int sample (int a, int b)
```

```
{
```

```
 int result;
```

```
 a=10;
```

```
 b=20;
```

```
 result=a+b;
```

output:

enter the value of  $x: 5$

enter the value of  $y: 6$

before function call the numbers are

$$x = 5 \quad y = 6 \quad z = 11$$

inside the function

$$x = 10 \quad y = 20 \quad z = 30$$

after function call the numbers are :

$$x = 5 \quad y = 6 \quad z = 30$$

```

printf("In inside the function");
printf("In x=%d It y=%d It z=%d", a, b, result);
return (result);
}

```

algorithm:

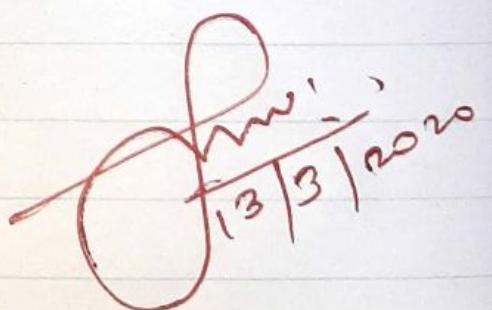
- step 1: start
- step 2: Declare function with integer parameters.
- step 3: Declare variables display the user enter the value of x & y respectively and scan the same.
- step 4: add the values and store in an another variable.
- step 5: Display the number before function call.
- step 6: call the function and display the same
- step 7: Define the declared function and print the same
- step 8: stop.



```

#22 #include <stdio.h>
#include <conio.h>
#include <string.h>
Void main()
{
 char str[50];
 char st[10];
 clrscr();
 printf("\n Enter a string: ");
 gets(str);
 printf("\n Enter substring to find in the
alone string");
 gets(st);
 if (strstr(str,st) == NULL)
 {
 printf("\n STRING NOT FOUND!");
 }
 else
 {
 printf("\n string found");
 }
 getch();
}

```


  
 Jyoti  
 13/3/2020

enter a string : cs is all about programming  
enter substring: au  
string found!

51

Output:

$$p_1 = 65524$$

$$p_2 = 65522$$

$$*p_1 + *p_2 = 15$$

$$p_1 - p_2 = 1$$

$$p_1 + r = 65526$$

$$p_2 -- = 65520$$



## Practical - 8

#1:

```

#include <stdio.h>
#include <conio.h>
void main()
{
 int m=5, n=10, o=0;
 int *p1, *p2, *p3;
 clrscr();
 p1 = &m;
 p2 = &n;
 printf ("p1 = %d\n", p1);
 printf ("p2 = %d\n", p2);
 o = *p1 + *p2;
 printf ("*p1 + *p2 = %d\n", o);
 p3 = p1-p2;
 printf ("p1-p2 = %d\n", p3);
 p1++;
 printf ("p1++ = %d\n", p1);
 p2--;
 printf ("p2-- = %d\n", p2);
 getch();
}

```

g

output: hello

```

#2: #include <stdio.h>
#include <conio.h>
void main()
{
 char str[6] = "hello";
 char *ptr = str;
 clrscr();
 while (*ptr != '\0')
 {
 printf("%c", *ptr);
 ptr++;
 }
 getch();
}

```

Algo:

- Step 1: Start
- Step 2: Create character string (declare)
- Step 3: Assign the same with character pointer
- Step 4: Use the while condition loop till the Null character
- Step 5: Print the same and iterate the same
- Step 6: Stop.

```
#3: #include <stdio.h>
#include <conio.h>
void main()
{
 char s[25], *t;
 int len = 0;
 clrscr();
 printf("Enter string: \n");
 scanf("%s", &s);
 t = s;
 while (*t != '\0')
 {
 len++;
 t++;
 }
 printf("Length is: %d", len);
 getch();
}
```

Step 1: Start

Step 2: Declare character string array, pointer variable & length.

Step 3: Accept the user string from the user

Step 4: Scanf the same value

Step 5: Using while loop count the characters until Null character and display the count

Step 6: Stop

Output: Enter string & Tanvi  
length & 5