PROGRAMMING & PROBLEM SOLVING

Assignment - 3

1) Develop the logic to check whether the given matrix is an identity matrix or not.

```
#include<stdio.h>
int main (void)
{
    int a[10] [10];
    int i=0, j=0, row=0, col=0;
    printf("Enter the order of the matrix (mxn):\n");
    printf("\n where m= number of rows; and \n");
    printf("\n where n= number of columns\n");
    scanf("%d %d", & rows, & col);
int flag=0;
printf("\n Enter the elements of the matrix\n");
for(i=0; i<row; i++)
{
    for(j=0;j<col;j++)
    {
    scanf("%d", &a[i] [j]);
}
}
for(i=0; i<row; i++)
{
    for (j=0 ; j<col ; j++)
    {
    if (i=j && a[i][j] != 1)
     {
       flag = -1;
       break;
     }
    else if(i!=j && a[i][j] != 0)
    {
      flag = -1;
      break;
    }
    }
   }
  }
if (flag==0)
{
 printf("\n It is a Idendity Matrix ");
}
else
```

```
{
printf("It is not an identity matrix");
}
return 0;
}
```

2) Write a program to check if a matrix is a lower or upper triangle.

```
#include<stdio.h>
 void main()
 {
 int a[10][10], i, j, l=0, k=0, n;
printf("Enter the length :");
scanf("%d", &n);
for(i=0; i<n; i++)
{   for(j=0; j<n; j++)
    {
    scanf("%d", &a[i][j]);
    }
}
for(i=0; i<n; i++)
 {
```

```
for (j=0;j<n;j++)
 {
   if (i>j && a[i][j] ! =0)
   {
     l=1;
   }
  else if (i<j  && a[i][j] ! =0)
  {
    k=1;
  }
 }
}
if (l==k)
 {
 printf("\nNot a triangular matrix");
 }
else
 {
   if (l==0)
   {
   printf("\n" Upper triangular matrix");
   }
else
```

```
{
printf("\n "Lower triangular matrix");
}
}
}
```

3) Explain inter-function communication with proper examples

When a function gets executed in the program, the execution is transferred from calling a function to called function and executes function definition, and finally comes back to the calling function.

In this process, both calling and called functions have to communicate with each other to exchange information. This process is called inter-function communication.

In c the inter function communication as follows

- Downward communication
- Upward communication
- Bi-directional communication

Down ward communication Ex program

```
#include <stdio.h>
# include <conio.h>
void main ()
{
 int num 1, num 2;
 void addition(int, int);
 clrscr ();

 num1 = 10;

 num2 = 20;
printf("\n Before swap : num 1 = %d, num2 = %d",
                num1 , num 2);

addition (num1 , num2);

getch ();
}
void addition (int a, int b)
{
 printf(" sum = %d", a+b);

}
```

Upward communication Ex program

```
#include<stdio.h>
#include <conio.h>
void main ()
{
 int result;
 int addition ();
 clrscr ();
 result = addition ();
printf (" sum = %d", result);
 getch();
}
int addition ()
{
int num1, num 2;
num1 = 10;
num 2 = 20;
return(num1+ num 2);
}
```

Bi- directional communication Ex program

```
#include< stdio.h>
# include <conio.h>
void main ()
{
int num1, num2, result;
int addition (int, int);
clrscr();
num1 = 10;
num2= 20;
result = addition(num1, num2);
printf(" sum =%d", result);
getch();
}
int addition(inta, intb)
{
 return (a+b)
}
```

4) Write a c program to arrange a given string in ascending order.

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

int main()
 {
 int i,j, count;
 char str[25][25], temp[25]:
 puts("How many strings you are going
                          to enter ? :");

Scanf("%d", &count);
puts(" Enter strings one by one :");
for(i=0; i<= count ; i++)
      gets(str[i]);
 for (i=0 ; i<= count ; i++)
      gets (str [i]);
 for (i=0 ; i<= count ; i++)
    for(j=i+1 ;j<= count ;j++)
  {
    if(strcmp(str[i], str[j] >0)
     { strcpy (temp, str[i]);
strcpy (str [i] , str [j]);
strcpy (str [j] , temp);
}
}
printf ("norder of sorted strings :");
for (i=0 ; i<= count ; i++)
  puts (str [i]);
  return 0;
}
```