KALINGA INSTITUTE OF
INDUSTRIAL TECHNOLOGY (KIIT)

Deemed to be University U/S 3 of UGC Act, 1956

Name: Prajukta Dey

Section: CSE 13

Subject: Data Structures and Algorithms Lab

Date: 29.07.2022

1. Write a program in C to check a given string is palindrome or not by using a dynamic array of n characters.

kiit-labs > dsa-lab > lab-02 > C quest1.c > main()

```c
//palindrome
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include <ctype.h>

int main()
{
    char *str;
    int i;
    int size;
    int mid;

    //dynamic memory allocation
    str = malloc(sizeof(char) * 100);

    //enter the string
    printf("Enter a string : ");
    scanf("%[^\n]s",str);

    //calculate the size of the string
    size = strlen(str);

    //to check for palindrome
    for(i = 0; i < size/2 ;i++)
    {
        if(tolower(str[i]) != tolower(str[size - 1 -i])) //tolower function converts verything to lowercase
        {
            printf("Not a palindrome string.\n");
            break;
        }
    }

    //display details
    if(i==size/2)
    {
        printf("Entered string is palindrome.\n");
    }
}
```

Output:

```
PS C:\PRAJUKTA\learning-languages\programming> cd "c:\PRAJUKTA\learning-languages\programming\kiit-labs\dsa-lab\lab-02\" ; if ($?) { gcc quest1.c -o quest1 } ; if
 ($?) { .\quest1 }
Enter a string : MADAM
Entered string is palindrome.
PS C:\PRAJUKTA\learning-languages\programming\kiit-labs\dsa-lab\lab-02>
```

2. Write a program in C to arrange the elements of a dynamic array such that all even numbers are followed by all odd numbers.

kiit-labs > dsa-lab > lab-02 > C quest2.c > main()

```c
/*write a program in C to arrange the elements of a dynamic array such that all
even numbers are followed by all odd numbers*/

#include<stdio.h>
#include<stdlib.h>
int main()
{
    int n, *p;
    printf("Enter the number of elements : ");
    scanf("%d", &n);
    p=(int*)malloc(n*sizeof(int));
    printf("Enter the elements: ");
    for(int i=0;i<n;i++)
    {
        scanf("%d",&p[i]);
    }
    printf("\n");
    int t=0;
    for(int i=0;i<n;i++)
    {
        for(int j=0;j<n;j++)
        {
            if(p[j]%2!=0 && p[i]%2==0)
            {
                t=p[j];
                p[j]=p[i];
                p[i]=t;
            }
        }
    }
    printf("The rearranged array is : ");
    for(int i=0;i<n;i++)
    {
        printf("%d\t",p[i]);
    }
    return 0;
}
```

Output:

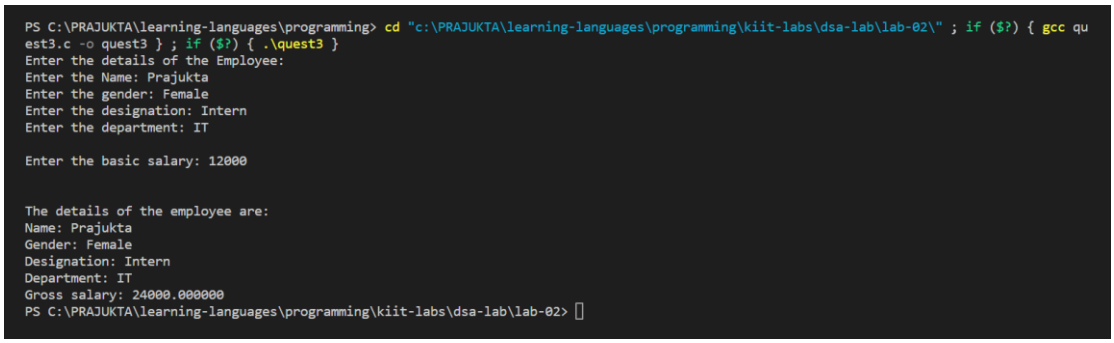
```
PS C:\PRAJUKTA\learning-languages\programming> cd "c:\PRAJUKTA\learning-languages\programming\kiit-labs\dsa-lab\lab-02\" ; if ($?) { gcc qu
est2.c -o quest2 } ; if ($?) { .\quest2 }
Enter the number of elements : 4
Enter the elements: 1 3 2 4

The rearranged array is : 2     4       1       3
PS C:\PRAJUKTA\learning-languages\programming\kiit-labs\dsa-lab\lab-02>
```

3. Write a program in C to store employee's data such as employee name, gender, designation, department, basic pay. Calculate the gross pay of each employees as follows:
Gross pay = basic pay + HR + DA
HR=25% of basic and DA=75% of basic.

kiit-labs > dsa-lab > lab-02 > C quest3.c > Employee

```c
/*Write a program in C to store employee's data such as employee name, gender,
designation, department, basic pay. Calculate the gross pay of each employees as
follows:
Gross pay = basic pay + HR + DA
HR=25% of basic and DA=75% of basic.*/

#include<stdio.h>

struct Employee
{
 char name[50];
 char gender[50];
 char designation[50];
 char department[50];
 int basic;
 float salary;
};

int main()
{
 struct Employee e;
 float da,hra;

 printf("Enter the details of the Employee: \n");
 printf("Enter the Name: ");
 scanf("%s", e.name);
 printf("Enter the gender: ");
 scanf("%s",e.gender);
 printf("Enter the designation: ");
 scanf("%s",e.designation);
 printf("Enter the department: ");
 scanf("%s",e.department);
 printf("\n");
 printf("Enter the basic salary: ");
 scanf("%d",&e.basic);
 printf("\n");

 da=(0.75 * e.basic);
 hra=(0.25 * e.basic);
 e.salary=(e.basic +da + hra);

 printf("\n");
 printf("The details of the employee are: \n");
 printf("Name: %s\n",e.name);
 printf("Gender: %s\n",e.gender);
 printf("Designation: %s\n",e.designation);
 printf("Department: %s\n",e.department);

 printf("Gross salary: %f\n",e.salary);
 return 0;
}
```

Output:


PS C:\PRAJUKTA\learning-languages\programming> cd "c:\PRAJUKTA\learning-languages\programming\kiit-labs\dsa-lab\lab-02\" ; if ($?) { gcc qu
est3.c -o quest3 } ; if ($?) { .\quest3 }
Enter the details of the Employee:
Enter the Name: Prajukta
Enter the gender: Female
Enter the designation: Intern
Enter the department: IT
Enter the basic salary: 12000
The details of the employee are:
Name: Prajukta
Gender: Female
Designation: Intern
Department: IT
Gross salary: 24000.000000
PS C:\PRAJUKTA\learning-languages\programming\kiit-labs\dsa-lab\lab-02>

4. Write a program in C to add two distances (in km-meter) by passing structure to a function.

kiit-labs > dsa-lab > lab-02 > C quest4.c > main()

```c
/*Write a program in C to add two distances (in km-meter) by passing structure to
a function.*/

#include <stdio.h>

struct distance
{
    int km;
    int m;
};

void addDistance(struct distance d1, struct distance d2)
{
    struct distance d3;
    d3.km = d1.km + d2.km;
    d3.m = d1.m + d2.m;

    d3.km = d3.km + d3.m / 1000; //1 km has 1000 m
    d3.m = d3.m % 1000;

    printf("\nTotal distance- km: %d, meter: %d", d3.km, d3.m);
}
```

```c
int main()
{
    struct distance d1, d2;

    printf("Enter first distance in km & m:");
    scanf("%d%d", &d1.km, &d1.m);

    printf("Enter second distance in km & m:");
    scanf("%d%d", &d2.km, &d2.m);

    /*add two distances*/
    addDistance(d1, d2);

    return 0;
}
```

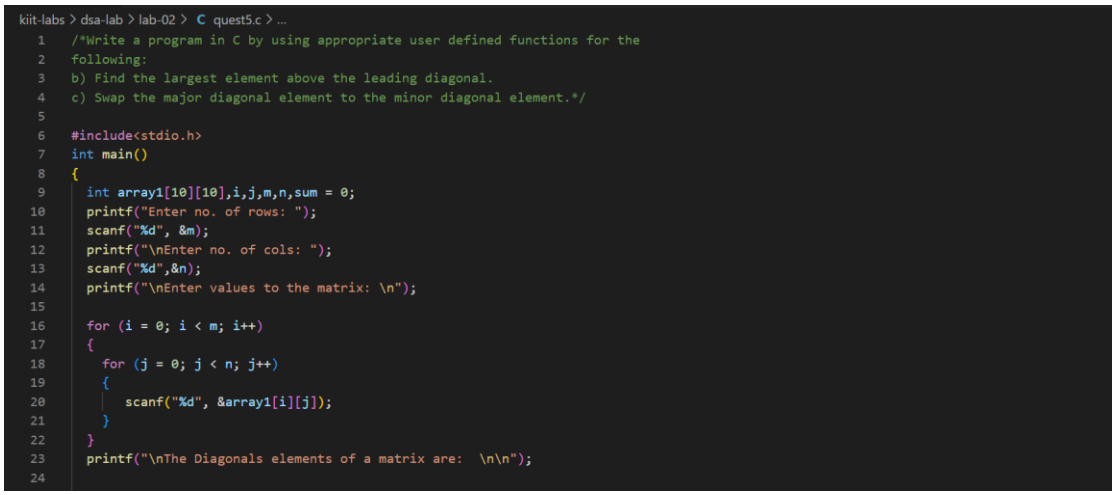
Output:

```
PS C:\PRAJUKTA\learning-languages\programming> cd "c:\PRAJUKTA\learning-languages\programming\kiit-labs\dsa-lab\lab-02\" ; if ($?) { gcc quest4.c -o quest4 } ; if
($?) { .\quest4 }
Enter first distance in km & m:5 500
Enter second distance in km & m:5 500

Total distance- km: 11, meter: 0
PS C:\PRAJUKTA\learning-languages\programming\kiit-labs\dsa-lab\lab-02>
```

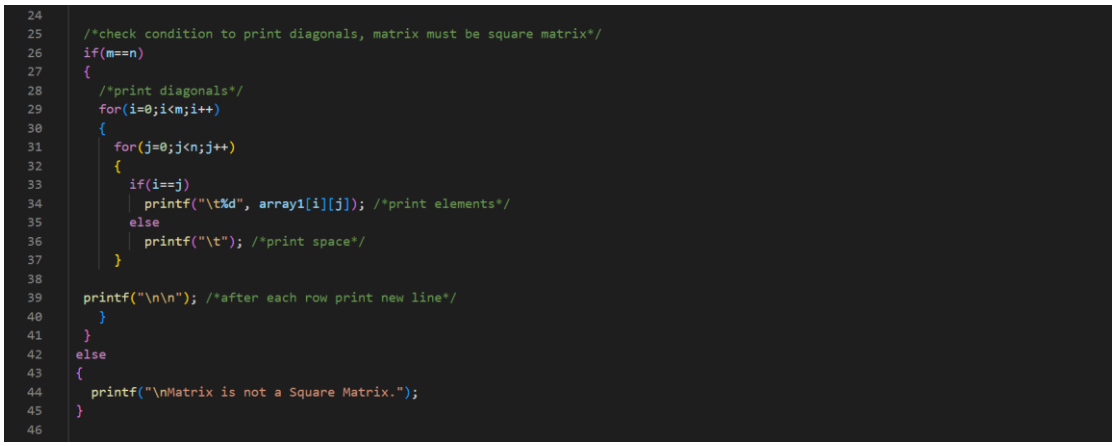
5. Write a program in C by using appropriate user defined functions for the following:

a) Find the largest element on the leading diagonal.

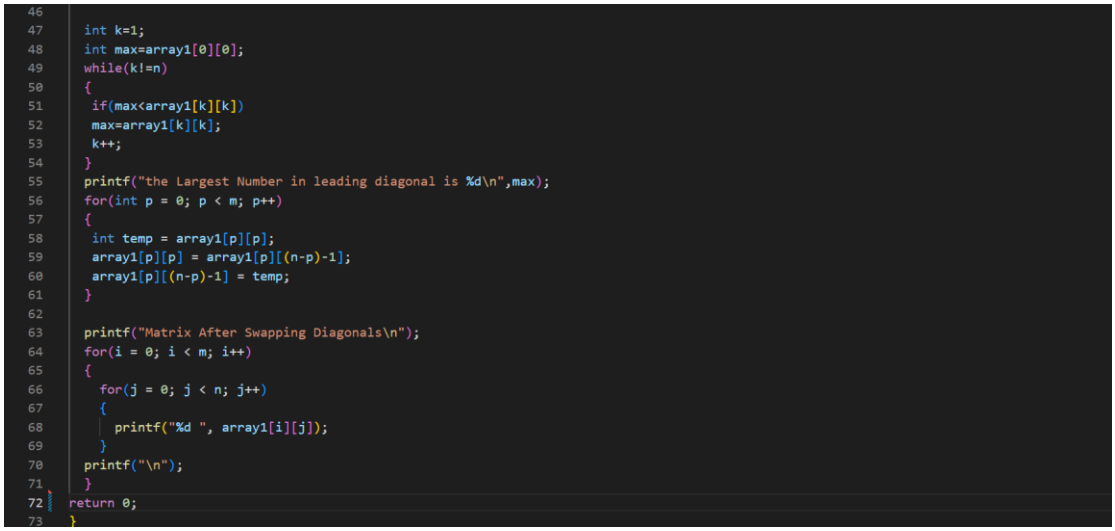
b) Swap the major diagonal element to the minor diagonal element.



```
kiit-labs > dsa-lab > lab-02 > C quest5.c > ...
/*Write a program in C by using appropriate user defined functions for the
following:
b) Find the largest element above the leading diagonal.
c) Swap the major diagonal element to the minor diagonal element.*/

#include<stdio.h>
int main()
{
  int array1[10][10],i,j,m,n,sum = 0;
  printf("Enter no. of rows: ");
  scanf("%d", &m);
  printf("\nEnter no. of cols: ");
  scanf("%d",&n);
  printf("\nEnter values to the matrix: \n");

  for (i = 0; i < m; i++)
  {
    for (j = 0; j < n; j++)
    {
      scanf("%d", &array1[i][j]);
    }
  }
  printf("\nThe Diagonals elements of a matrix are:  \n\n");

```

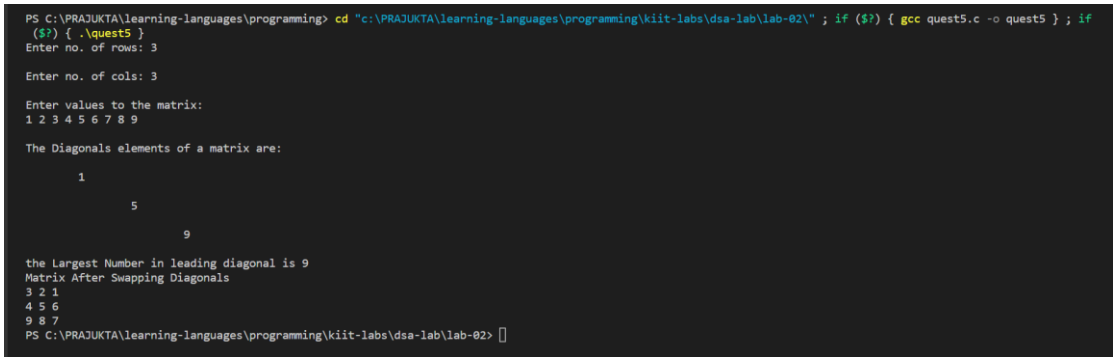


```

  /*check condition to print diagonals, matrix must be square matrix*/
  if(m==n)
  {
    /*print diagonals*/
    for(i=0;i<m;i++)
    {
      for(j=0;j<n;j++)
      {
        if(i==j)
          printf("\t%d", array1[i][j]); /*print elements*/
        else
          printf("\t"); /*print space*/
      }

  printf("\n\n"); /*after each row print new line*/
    }
  }
 else
 {
   printf("\nMatrix is not a Square Matrix.");
 }

```



```

  int k=1;
  int max=array1[0][0];
  while(k!=n)
  {
   if(max<array1[k][k])
   max=array1[k][k];
   k++;
  }
  printf("the Largest Number in leading diagonal is %d\n",max);
  for(int p = 0; p < m; p++)
  {
   int temp = array1[p][p];
   array1[p][p] = array1[p][(n-p)-1];
   array1[p][(n-p)-1] = temp;
  }

  printf("Matrix After Swapping Diagonals\n");
  for(i = 0; i < m; i++)
  {
    for(j = 0; j < n; j++)
    {
      printf("%d ", array1[i][j]);
    }
  printf("\n");
  }
return 0;
}
```

Output:


PS C:\PRAJUKTA\learning-languages\programming> cd "c:\PRAJUKTA\learning-languages\programming\kiit-labs\dsa-lab\lab-02\" ; if ($?) { gcc quest5.c -o quest5 } ; if ($?) { .\quest5 }
Enter no. of rows: 3
Enter no. of cols: 3
Enter values to the matrix:
1 2 3 4 5 6 7 8 9
The Diagonals elements of a matrix are:
1
5
9
the Largest Number in leading diagonal is 9
Matrix After Swapping Diagonals
3 2 1
4 5 6
9 8 7
PS C:\PRAJUKTA\learning-languages\programming\kiit-labs\dsa-lab\lab-02>