LAB IX

1. Complete the details of the program given below at appropriate places (see the comment parts of the program). Remove the comment parts in the program you write. After completion of the function **reverse**, the first string is written in reverse order. After completion of the function **check_char**, if flag=1 then the input character is present in the second string; otherwise flag=0.

```
#include<stdio.h>
//Write the prototype of the functions here
int main()
{
int flag=0;
int c;
char p[]="Hellow world";
char *q="MTH409: C program";

//Print the first string here
reverse(p);
//Print the first string here
//Read a character from the keyboard into variable c
flag=check_char(q,c);
if(flag==0)
{
        printf("The character %c is not present in \"%s\"\n",c,q);
}
else
{
                printf("The character %c is present in \"%s\"\n",c,q);
}

return 0;
}
//Write details of function reverse here
//Write details of function check_char here
```

Test data and expected output:

```
First string is :Hellow world
First string after function call is :dlrow wolleH
Enter the search character:9
The character 9 is present in "MTH409: C program"

First string is :Hellow world
First string after function call is :dlrow wolleH
Enter the search character:c
The character c is not present in "MTH409: C program"
```

2. A file **poly.dat** contains the degree of a polynomial and its $(n+1)$-coefficients. Write a program that do the following:

   (a) It accepts the degree n of the polynomial, creates a dynamic array **a** that can hold its $(n+1)$-coefficients and then read the $(n+1)$-coefficients in the array **a**.

(b) It then prints the polynomial in an appropriate from.

For each stage (a) and (b), use appropriate function.

Test data and expected output:
For poly.dat

```
7
2.5
0
0
0
0
-3.0
0
2.0
```

```
The polynomial is :2.50 x^7-3.00 x^2+2.00
```

3. Create a data file **matA.dat** that contains matrix A. In this file, first line contains the row and column dimensions of the matrix and the following lines contain the matrix elements stored row-wise. Write a C program that do the following:

   (a) It reads the row and column dimensions of A. Creates a dynamic 2-D array like variable that can hold the matrix and then reads the matrix A from **matA.dat**

   (b) It then prints out A in the terminal.

   For each stage (a) and (b), use appropriate functions.

Test data and expected output:
For matA.dat

```
2 3
1.0  0  4.0
-2.0 4.0  1.0
```

```
The matrix A is:
1.000  0.000  4.000
-2.000  4.000  1.000
```