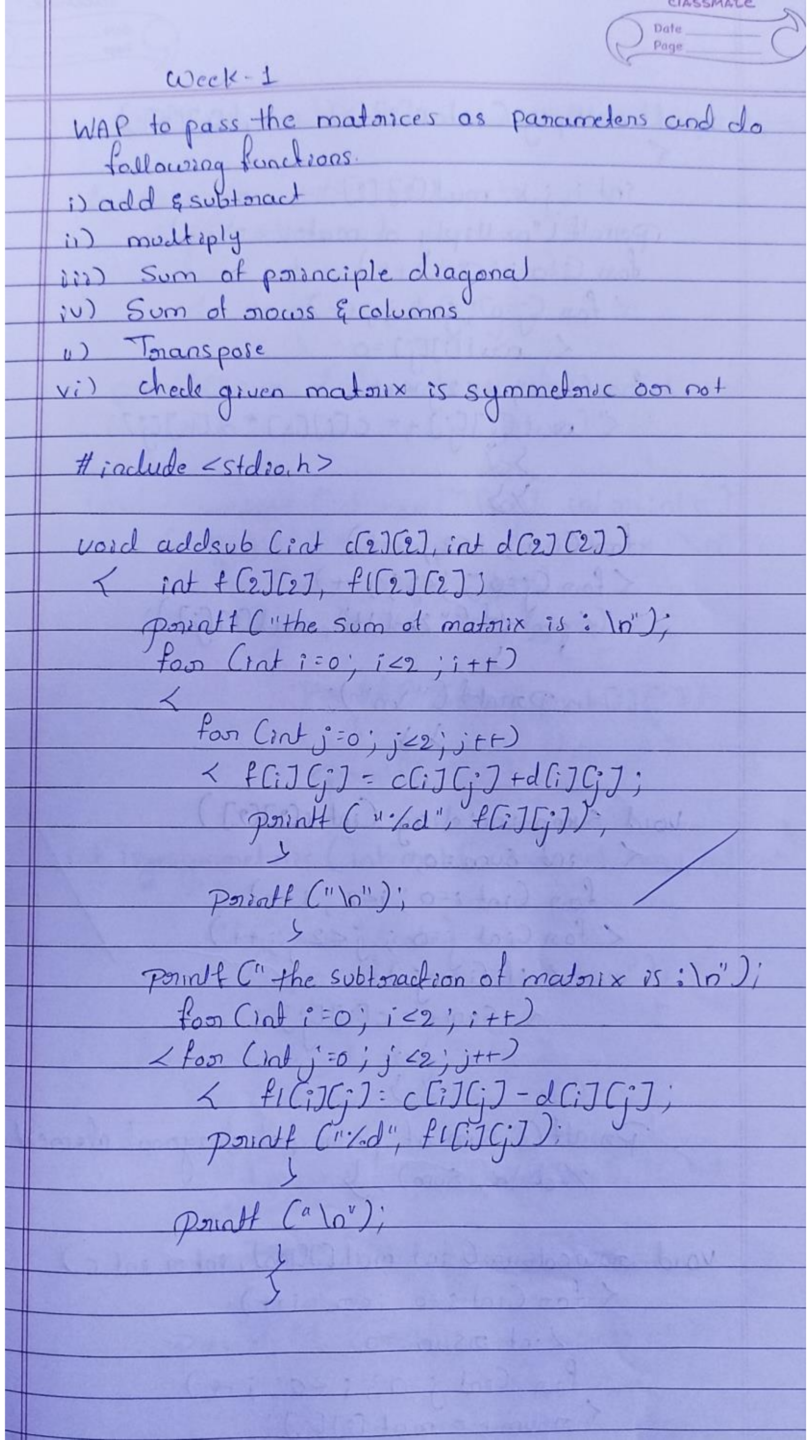
Week 1

Q)Write a C program to do the following by passing matrix as parameter:

1) Matrix addition and subtraction.
2) Matrix multiplication.
3) Sum of principle and non principle diagonal of matrix.
4) Sum of rows and columns.
5) Print the transpose
6) Check if a given matrix is symmetric or not.



Date

Page

Week - 1

WAP to pass the matrices as parameters and do following functions.

i) add & subtract

ii) multiply

iii) Sum of principle diagonal

iv) Sum of rows & columns

v) Transpose

vi) check given matrix is symmetric or not

```
#include <stdio.h>

void addsub (int c[2][2], int d[2][2])
 {  int f[2][2], f1[2][2])
    printf ("the sum of matrix is : \n");
    for (int i=0; i<2 ; i++)
    {
       for (int j=0; j<2; j++)
       { f[i][j] = c[i][j] +d[i][j];
         printf ("%d", f[i][j]);
        }
       printf ("\n");
        }
    printf (" the subtraction of matrix is :\n");
     for (int i=0; i<2 ; i++)
    { for (int j=0; j<2; j++)
       { f1[i][j]= c[i][j] - d[i][j];
       printf ("%d", f1[i][j]);
        }
     printf ("\n");
        }
        }
```

```
void multiply (int c[2][2], int d[2][2])
 {
   int i, j, k, mul [2] [2] ;
  printf ("multiply of matrix :\n"),
   for (i=0 ; i<2 ; i++)
   { for (j=0 ; j<2 ; j++)
      { mul[i][j]=0 ;
   for (k=0 ; k<2 ; k++)
    { mul [i][j] += c[i][k] * d[k][j];
        }
        }}
   for (i=0 ; i<2 ; j++)
   {for (j=0 ; j<2 ; j++)
     {printf ("%d \t", mul[i][j]);
        }
       printf ("\n");
             }}

 void sumprincidiag (int c[2][2])
   {  int sum=0;
      for (int i=0 ; i<2 ; i++)
    { for (int j=0 ; j<2 ; j++)
       { if (i==j)
         { sum+ = c[i][j] ;
             }
             }}
   Printf ("sum of principal diagonal elements
      %d \n", sum) ; }

void rowcolsum( int mat[][100], int r, int c)
    { for (int i=0 ; i<r ; i++)
       {int rsum =0;
       for (int j=0 ; i <c ; j++)
     { rsum += mat[i][j]
          }
```

```
printf("sum of elements in row %d :%d\n", i+1, r);
      }
 for (int j=0 ; j<c ; j++)
    { int csum =0;
      for (int i=0 ; i<r ; i++)
      {  csum += mat [i][j] ;
            }
    printf("sum of elements in column %d : %d\n",
                          j+1, c);
                          }}
 void transpose (int mat[][100], int r, int c)
    {  printf ("transpose of matrix :\n");
       for (int j=0 ; j<c ; j++)
       { for (int i=0 ; i<r ; j++)
         {
              Print f ("%d\t", mat[i][j]);
               }
               Printf ("\n") ;
                   }
int issymmetric (int mat [][20], int rows, int cols )
    {
       if (rows != cols)
         {printf ("not symmetric");
              }
   for (int i=0 ; i<rows ; i++)
    { for (int j=0 ; j<cols ; j++)
     {
       if (matrix [i][j] != matrix[j][i])
      {
        Printf ("not symmetric");
           }}}
  Printf ("matrix is symmetric");
        }.
```

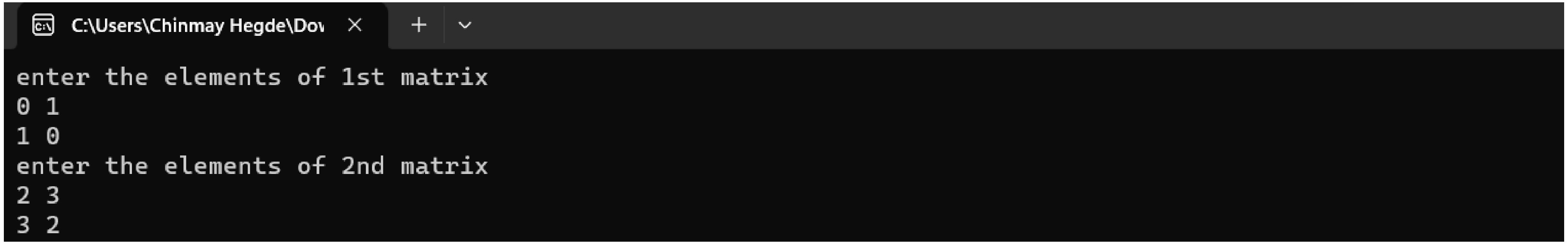
output

```
   enter elements of 1st matrix
      0  1
      1  0
   enter the elements of 2nd matrix
      2  3
      3  2
1.add &sub  2.multiply  3.sum of principle diagonal
   4.row column sum  5.Transpose  6.Symmetric
   check  7.exit

1
the sum of matrix is:
     2 4
     4  2
the subtraction of matrix is:
     -2  -2
     -2  -2

2
multiply of matrix =
     3    2
     2    3
3
sum of principal diagonal element is 0.

4

Sum of elements in row 1  : 1
Sum of elements in row 2  : 1
Sum of elements in column1 : 1
Sum of elements in column2 : 1
```

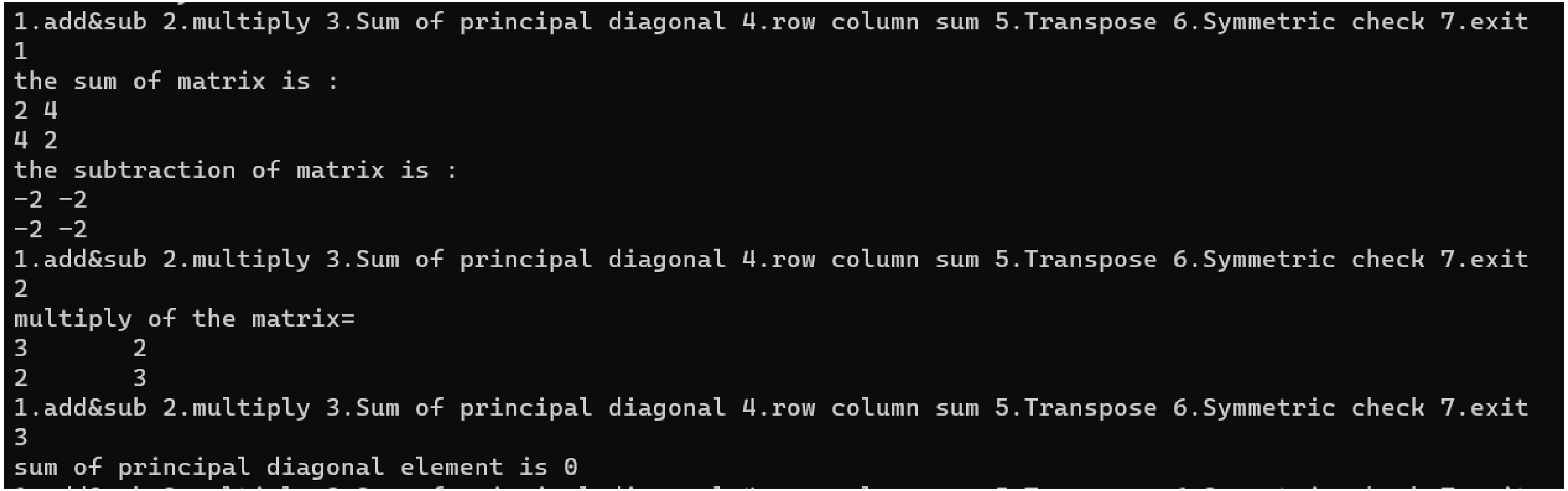
```
5
0  1
1  0
6
matrix is symmetric.
```

15/6/2023

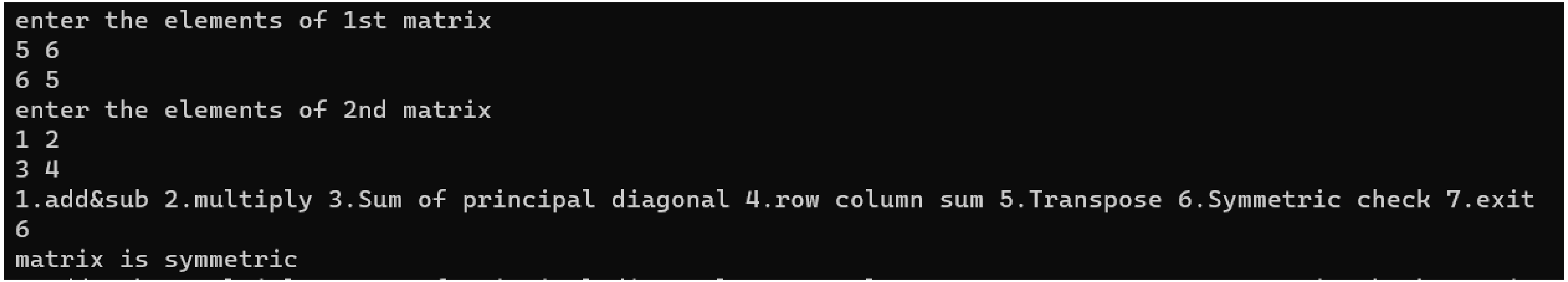
Output:



```
enter the elements of 1st matrix
0 1
1 0
enter the elements of 2nd matrix
2 3
3 2
```



```
1.add&sub 2.multiply 3.Sum of principal diagonal 4.row column sum 5.Transpose 6.Symmetric check 7.exit
1
the sum of matrix is :
2 4
4 2
the subtraction of matrix is :
-2 -2
-2 -2
1.add&sub 2.multiply 3.Sum of principal diagonal 4.row column sum 5.Transpose 6.Symmetric check 7.exit
2
multiply of the matrix=
3       2
2       3
1.add&sub 2.multiply 3.Sum of principal diagonal 4.row column sum 5.Transpose 6.Symmetric check 7.exit
3
sum of principal diagonal element is 0
```



```
enter the elements of 1st matrix
5 6
6 5
enter the elements of 2nd matrix
1 2
3 4
1.add&sub 2.multiply 3.Sum of principal diagonal 4.row column sum 5.Transpose 6.Symmetric check 7.exit
6
matrix is symmetric
```