EXPERIMENTS (Cycle-4)

1. ***Transpose a 3 x 3 matrix.***
2. ***Sum of two matrices.***
3. ***Multiplication of two m x n matrices.***
4. ***Program to read a two-dimensional array. Which stores marks of 5 students in 3 subjects. Write a program to display the highest marks in each subject.***
5. ***Program to find the trace of a matrix.***
6. ***Program to check whether a matrix is diagonal or not.***
7. ***Program to display lower triangular matrix.***

Expt 18: Transpose a matrix

AIM

To transpose a 3 x 3 matrix.

ALGORITHM

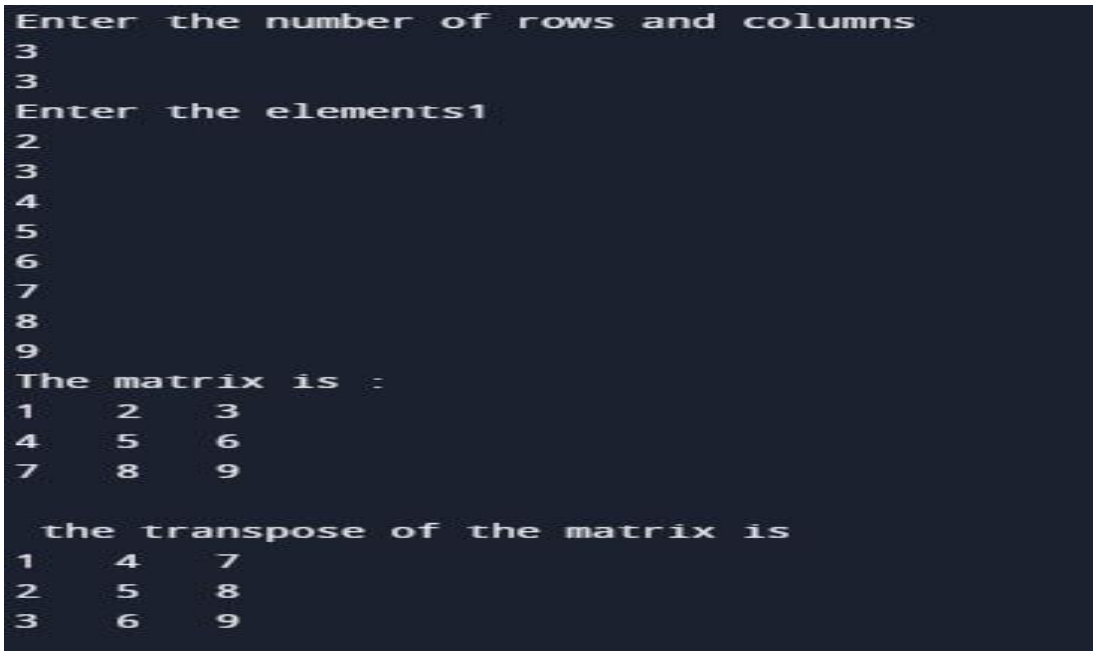
- Declare a 3 x 3 matrix.
- Using the for loops, enter the elements of the matrix.
- Print the matrix.
- Interchange the rows and columns of the matrix to form the transpose ofthe matrix.
- Print the transpose of the matrix.

SOURCE CODE

```
//*************************************************************************
// Program name          : Transpose.c
// Author                : Anantha Krishnan R J
// Date written          : 08/09/2021*
// Date complied         : 08/19/2021*
// Aim of the program : To transpose a 3 x 3 matrix.
//*************************************************************************
//*************************************************************************

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

main()
{
int a[10][10],i,j,m=0,n=0;
printf(" Enter the number of rows and columns\n");
scanf("%d %d",&m,&n);
printf("\n Enter the elements");
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
scanf("%d",&a[i][j]);
}
}
```

```
printf("\n The matrix is : \n");
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
printf("%d \t",a[i][j]);
}
printf("\n");
}
printf("\n the transpose of the matrix is\n");

for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
printf("%d\t",a[j][i]);
}
printf("\n");
}

}
```

OUTPUT :



Expt 19: Sum of two matrices

AIM

To input two m x n matrices and then calculate the sum of their correspondingelements and store it in a third m x n matrix.

ALGORITHM

- Declare three m x n matrices.
- After entering the rows and columns of two matrices,check whether theyhave the same number of rows and columns.If they are not same, show that the addition of matrices is not possible
- Using the for loops ,input the elements of the first two matrices.
- Print the two matrices.
- Then calculate the sum of their corresponding elements and store it in athird m x n matrix.
- Print the third m x n matrix.

SOURCE CODE

```
//*************************************************************************
// Program name          : Sumatrices.c
// Author                : Anantha Krishnan R J
// Date written          : 08/09/2021
// Date complied         : 08/09/2021
// Aim of the program : To input two m x n matrices and then calculate the sumof their
                        corresponding elements and store it in a third
                        m x n matrix.
//*************************************************************************
//*************************************************************************
#include<stdio.h>
#include<process.h>

void main()
{
int a[10][10],b[10][10],c[10][10],i,j,m,n,p,q;

printf("\nenter the rows and columns of first matrix\n"); scanf("%d%d",&m,&n);
printf("\n enter the rows and columns of second matrix\n"); scanf("%d%d",&p,&q);
if(m!=p&&n!=q)

{
```

```
printf("\n addition of matrix is not possible\n");
exit(0);
}
printf("\n enter the elements of first matrix\n");
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
scanf("%d",&a[i][j]);
}
}
printf("\n enter the elements of second matrix\n");
for(i=0;i<p;i++)
{
for(j=0;j<q;j++)
{
scanf("%d",&b[i][j]);
}
}
printf("\n first matrix is\n"); for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
printf("%d\t",a[i][j]);
}
printf("\n");
}
printf("\n second matrix is\n");
for(i=0;i<p;i++)
{
for(j=0;j<q;j++)
{
printf("%d\t",b[i][j]);
}
printf("\n");
}
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
c[i][j]=a[i][j]+b[i][j];

}
}
```

```
printf("\n the sum of two matrix is\n");
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
printf("%d\t",c[i][j]);
}
printf("\n");
}
}
```

OUTPUT

```
enter the rows and columns of first matrix
2
2

 enter the rows and columns of second matrix
2
2

 enter the elements of first matrix
1
2
3
4

 enter the elements of second matrix
1
2
3
4

 first matrix is
1       2
3       4

 second matrix is
1       2
3       4

 the sum of two matrix is
2       4
6       8
```

Expt 20: Multiply two m x n matrices.

AIM

To multiply two m x n matrices.

ALGORITHM

- Declare three m x n matrices.
- Check whether the row of the first matrix and column of the second matrix are the same. If they are not the same ,show that multiplication is not possible.
- Input the elements of the first matrix and second matrix.
- Print the first matrix and the second matrix.
- Find the product of two matrices and store it in the third m x n matrix.
- Print the third m x n matrix**.**

SOURCE CODE

```
//*************************************************************************
// Program name        : Productmatrices.c
// Author              : Anantha Krishnan R J
// Date written        : 08/09/2021
// Date complied       : 08/09/2021
// Aim of the program : To multiply two m x n matrices.
//*************************************************************************
//*************************************************************************

#include<stdio.h>
#include<stdlib.h>

void main()
{
int a[10][10],b[10][10],c[10][10],m,n,j,i,p,q,k,sum;
printf("\n Enter the number of rows and columns of first matrix\n");
scanf("%d%d",&m,&n);
printf("\n Enter the number of rows and columns of second matrix\n");
scanf("%d%d",&p,&q);
if(n!=p)
{
printf("\n Multiplication of two matrix is not possible\n");
exit(0);
}
printf("Enter the elements of first matrix\n"); for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
```

```
{
scanf("%d",&a[i][j]);
}

}
printf("\n Enter the elements of second matrix \n"); for(i=0;i<p;i++)
{
for(j=0;j<q;j++)
{
scanf("%d",&b[i][j]);
}

}
printf("\n First matrix is\n");
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
printf("%d\t",a[i][j]);
}
printf("\n");
}
printf("\n Second matrix is \n");
for(i=0;i<p;i++)
{
for(j=0;j<q;j++)
{
printf("%d\t",b[i][j]);
}
printf("\n");
}


for(i=0;i<m;i++)
{
for(j=0;j<q;j++)
{
c[i][j]=0; sum=0;
for(k=0;k<n;k++)
sum=sum+a[i][k]*b[k][j];
c[i][j]=sum;
}

}

printf("\n The product of two matrix is\n");
for(i=0;i<m;i++)
{
for(j=0;j<q;j++)
{
```

```
printf("%d\t",c[i][j]);
}
printf("\n");
}

}
```

OUTPUT

```
 Enter the number of rows and columns of first matrix
2
2

 Enter the number of rows and columns of second matrix
2
2
Enter the elements of first matrix
1
2
3
4

 Enter the elements of second matrix
2
3
4
5

 First matrix is
1       2
3       4

 Second matrix is
2       3
4       5

 The product of two matrix is
10      13
22      29
```

Expt 21: Reading a 2-D Array of Marks & Printing the Highest Mark.

AIM:

To read a two-dimensional array marks which stores marks of 5 students in 3 subjects and display the highest marks in each subject

ALGORITHM:

1. Read in the matrix elements (using arrays) (3×5).

2. Comparing the marks in each subject using if statement and finding the maximum mark in each subject.

3. Display the highest mark in each subject .

SOURCE CODE

```
//*************************************************************************
// Program name : marks .c
//Author :Anantha Krishnan R J

// Date written : 15/09/21

// Date complied : 15/09/21
//*************************************************************************
//*************************************************************************

#include <stdio.h>
void main()
{
int i,j,mark[3][5],max=0;
for(i=1;i<=3;i++)
{
for(j=1;j<=5;j++)
{
   printf("Enter the marks of student %d in subject %d \n",j,i);
   scanf("%d",&mark[i][j]);
}
printf("\n");
}
for(i=1;i<=3;i++)
{

    max=mark[i][1];
      for(j=1;j<=5;j++)
      {
         if(mark[i][j] > max)
```

```
            max=mark[i][j];
        }
        printf("Highest Mark in Subject %d = %d",i,max );
        printf("\n");

}
}
```

OUTPUT

```
Enter the marks of student 1 in subject 1
20
Enter the marks of student 2 in subject 1
25
Enter the marks of student 3 in subject 1
22
Enter the marks of student 4 in subject 1
24
Enter the marks of student 5 in subject 1
19

Enter the marks of student 1 in subject 2
10
Enter the marks of student 2 in subject 2
23
Enter the marks of student 3 in subject 2
22
Enter the marks of student 4 in subject 2
19
Enter the marks of student 5 in subject 2
17

Enter the marks of student 1 in subject 3
22
Enter the marks of student 2 in subject 3
25
Enter the marks of student 3 in subject 3
23
Enter the marks of student 4 in subject 3
12
Enter the marks of student 5 in subject 3
11

Highest Mark in Subject 1 = 25
Highest Mark in Subject 2 = 23
Highest Mark in Subject 3 = 25
```

Expt 22: Finding Trace of a matrix

AIM

To find trace of a matrix

ALGORITHM

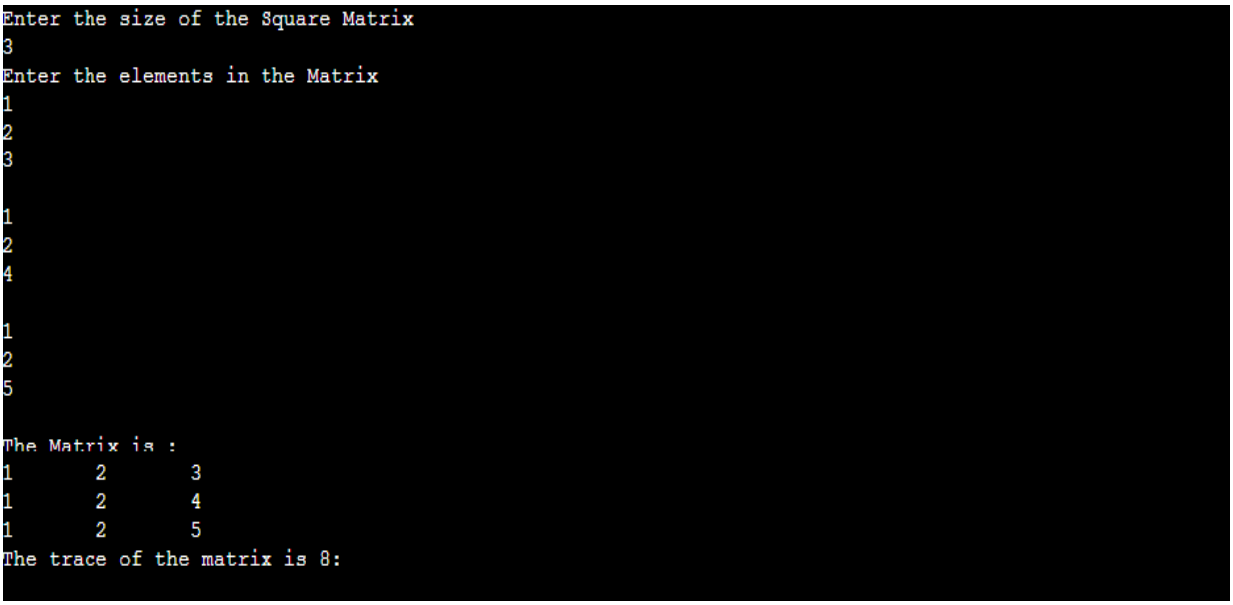
1. Read in the size of the matrix (n).
2. Read in the matrix elements (n*n) .
3. find the sum of diagonal elements.
4. Display the Original matrix.
5. Display the trace of the matrix .

SOURCE CODE

```
//*************************************************************************
// Program name : trace .c
//Author :Anantha Krishnan R J
// Date written : 15/09/21
// Date complied : 15/09/21
//*************************************************************************
//*************************************************************************

#include <stdio.h>
void main()
{
int i,j,n,a[10][10];
printf("Enter the size of the Square Matrix \n");
scanf("%d",&n);
printf("Enter the elements in the Matrix \n");
int trace=0;
for(i=0;i<n;i++)
   {
      for(j=0;j<n;j++)
         scanf("%d",&a[i][j]);
   trace= trace + a[i][i];
   printf("\n");
   }
printf("The Matrix is : \n");
for(i=0;i<n;i++)
   {
```

```
      for(j=0;j<n;j++)
      {
         printf("%d\t",a[i][j]);
      }
   printf("\n");
   }
printf("The trace of the matrix is %d: ",trace);
}
```

OUTPUT



Expt 23: Checking Whether a Matrix is Diagonal or not.

AIM

To find check whether a matrix is diagonal or not.

ALGORITHM

1. Read in the no of rows and columns (r×c).
2. Read in the matrix elements (r×c).
4. Check the elements except the diagonal elements to be zero.
5. Display the output.

SOURCE CODE:

```
//*************************************************************************
// Program name : diagonal.c
 //Author :Anantha Krishnan R J
 // Date written : 15/09/21
 // Date complied : 15/09/21
//*************************************************************************
//*************************************************************************

#include <stdio.h>

#include<stdlib.h>

void main()

{

int i,j,r,c,a[10][10];

printf("Enter the number of rows & coloumns \n");

scanf("%d %d",&r,&c);

if(r==c)

   printf(" This is a square matrix \n");

else

   {
```

```
        printf(" This is not a square matrix matrix \n");
        exit(0);
    }
printf("Enter the elements of the matrix \n");
for(i=0;i<r;i++)
    {
    for(j=0;j<c;j++)
        scanf("%d",&a[i][j]);
    printf("\n");
}
int t=0;
for(i=0;i<r;i++)
    {
        for(j=0;j<c;j++)
            {
                if((i!=j) && (a[i][j]!=0))
                {
                    printf("The matrix is NOT a Diagonal Matrix");
                    exit(0);
                }
            }
     }
printf("The matrix is a Diagonal Matrix");

  }
```

OUTPUT:

```
Enter the number of rows & coloumns
2
2
 This is a square matrix
Enter the elements of the matrix
1
2

3
4

The matrix is NOT a Diagonal Matrix
```

```
Enter the number of rows & coloumns
3
3
 This is a square matrix
Enter the elements of the matrix
1
0
0

0
1
0

0
0
1

The matrix is a Diagonal Matrix
```

Expt 24: Display the lower triangular Matrix.

AIM

To display lower triangular matrix

ALGORITHM

1. Read in the size of the matrix.

2. Read in the matrix elements (using arrays).

3.Display the original matrix.

4. Display the lower triangular matrix using (if statement in the nested loop).

SOURCE CODE

```
//*************************************************************************
// Program name : lowertriangle.c
//Author : Anantha Krishnan R J
// Date written : 15/09/21
// Date complied : 15/09/21
//*************************************************************************
//*************************************************************************

#include <stdio.h>

 void main()

 {

 int i,j,n,a[10][10];

 printf("Enter the size of matrix \n");

 scanf("%d" ,&n);

 printf("Enter the elements of the matrix \n");

 for(i=0;i<n;i++)

    {
```

```
        for(j=0;j<n;j++)
            scanf("%d",&a[i][j]);
    printf("\n");
    }
printf("The Matrix is : \n");
for(i=0;i<n;i++)
    {
    for(j=0;j<n;j++)
        printf("%d\t",a[i][j]);
    printf("\n");
    }

printf("The Lower triangular Matrix is : \n");
for(i=0;i<n;i++)
    {
        for(j=0;j<n;j++)
            {
                if(i>=j)
                printf("%d\t",a[i][j]);
                else
                 printf("0");
            }
    printf("\n");
```

```
    }
}
```

OUTPUT

```
Enter the size of matrix
3
Enter the elements of the matrix
1
2
3

4
5
6

7
8
9

The Matrix is :
1       2       3
4       5       6
7       8       9
The Lower triangular Matrix is :
1       00
4       5       0
7       8       9
```