

**PROGRAM** – WRITE A PROGRAM TO OVERLOAD +, -, X, != FOR CLASS MATRIX USING TEMPLATE CLASS

**//BHAVNA VERMA-171210019-16/04/2019**

**//MENU DRIVEN PROGRAM USING TEMPLATE CLASS**

**//FOR CLASS MATRIX TO OVERLOAD OPERATORS +,-,X,!=**

```
#include<iostream>
```

```
#include<conio.h>
```

```
using namespace std;
```

```
template<class T>
```

```
class MATRIX
```

```
{
```

```
    int rows,columns;
```

```
    T m[5][5];
```

```
    public:
```

```
    int check(MATRIX);
```

```
    void input();
```

```
    void display();
```

```
    MATRIX<T> operator+(MATRIX<T>);
```

```
    MATRIX<T> operator-(MATRIX<T>);
```

```
    MATRIX<T> operator*(MATRIX<T>);
```

```
    bool operator!=(MATRIX<T>);
```

```
};
```

```
template<class T>
```

```
int MATRIX<T>::check(MATRIX<T> A)
```

```
{
```

```
    if(rows==A.rows && columns==A.columns)
```

```
        return 0;
```

```
    return -1;
```

```
}
```

```
template<class T>
```

```
void MATRIX<T>::input()
```

```
{
```

```
    cout<<"\nEnter the no of rows ";
```

```

cin>>rows;

cout<<"\nEnter the no of columns ";

cin>>columns;

for(int i=0;i<rows;i++)
{
    cout<<"\nrow-"<<i+1;

    for(int j=0;j<columns;j++)
    {
        cout<<"Enter element "<<j+1<<" : ";

        cin>>m[i][j];

    }

}

}

template<class T>

void MATRIX<T>::display()

{

    for(int i=0;i<rows;i++)
    {

        for(int j=0;j<columns;j++)

            cout<<m[i][j]<<"\t";

        cout<<endl;

    }

}

template<class T>

MATRIX<T> MATRIX<T>::operator+(MATRIX<T> A)

{

    MATRIX B;

    B.rows=rows; B.columns=columns;

    for(int i=0;i<rows;i++)

        for(int j=0;j<columns;j++)

            B.m[i][j]=m[i][j]+A.m[i][j];

    return B;

}

template<class T>

```

```
MATRIX<T> MATRIX<T>::operator-(MATRIX<T> A)
```

```
{
```

```
    MATRIX B;
```

```
    B.rows=rows; B.columns=columns;
```

```
    for(int i=0;i<rows;i++)
```

```
        for(int j=0;j<columns;j++)
```

```
            B.m[i][j]=m[i][j]-A.m[i][j];
```

```
    return B;
```

```
}
```

```
template<class T>
```

```
MATRIX<T> MATRIX<T>::operator*(MATRIX<T> A)
```

```
{
```

```
    MATRIX B;
```

```
    B.rows=rows;
```

```
    B.columns=A.columns;
```

```
    for(int i=0;i<rows;i++)
```

```
    {
```

```
        for(int j=0;j<A.columns;j++)
```

```
        {
```

```
            B.m[i][j]=0;
```

```
            for(int k=0;k<columns;k++)
```

```
            {
```

```
                B.m[i][j]+=(m[i][k]*A.m[k][j]);
```

```
            }
```

```
        }
```

```
    }
```

```
    return B;
```

```
}
```

```
template<class T>
```

```
bool MATRIX<T>::operator!=(MATRIX<T> temp)
```

```
{
```

```
    int flag=0;
```

```
    for(int i=0;i<this->rows;i++)
```

```
    {
```

```

        for(int j=0;j<this->columns;j++)
        {
            if(this->m[i][j]!=temp.m[i][j])
            {
                flag=1;
                break;
            }
        }
    }
    if(flag==1)
        return 1;
    else
        return 0;
}

int main()
{
    int ch;
    MATRIX<int> m1,m2,m3;
    do
    {
        cout<<"\n-----MENU-----\n";
        cout<<"1.INPUT MATRIX"<<endl;
        cout<<"2.ADD MATRIX"<<endl;
        cout<<"3.SUBTRACT MATRIX"<<endl;
        cout<<"4.MULTIPLY"<<endl;
        cout<<"5.EQUAL OR NOT"<<endl;
        cout<<"6.DISPLAY"<<endl;
        cout<<"7.EXIT"<<endl;
        cout<<"ENTER CHOICE (1/2/3/4/5)"<<endl;
        cin>>ch;
        switch(ch)
        {
            case 1: cout<<"For the first matrix \n";
                    m1.input();

```

```

        cout<<"For the second matrix \n";
        m2.input();
        break;
case 2: if(m1.check(m2))
        cout<<"Addition not possible\n";
        else
        {
                cout<<"Addition\n";
                m3=m1+m2;
                m3.display();
        }
        break;
case 3: if(m1.check(m2))
        cout<<"Subtraction not possible\n";
        else
        {
                cout<<"Subtraction\n";
                m3=m1-m2;
                m3.display();
        }
        break;
case 4: cout<<"Multiplication\n";
        m3=m1*m2;
        m3.display();
        break;
case 5: cout<<"EQUAL OR NOT ?\n";
        if(m1!=m2)
        cout<<"NOT EQUAL\n";
        else
        cout<<"EQUAL\n";
        break;
case 6: cout<<"First matrix \n";
        m1.display();
        cout<<"Second matrix \n";

```

```
        m2.display();  
        break;  
    case 7: exit(0);  
    default: cout<<"Please enter a valid choice\n";  
}  
    cout<<"DO YOU WANT TO CONTINUE IF YES THEN PRESS 1 ELSE ANY  
KEY"<<endl;  
    cin>>ch;  
    }while(ch==1);  
    return 0;  
}
```

## OUTPUT-

C:\Users\VERMA\Desktop\oop\T3.exe

```
-----MENU-----
1.INPUT MATRIX
2.ADD MATRIX
3.SUBTRACT MATRIX
4.MULTIPLY
5.EQUAL OR NOT
6.DISPLAY
7.EXIT
ENTER CHOICE (1/2/3/4/5)
1
For the first matrix

Enter the no of rows 2

Enter the no of columns 2

row-1Enter element 1 : 1
Enter element 2 : 2

row-2Enter element 1 : 3
Enter element 2 : 4
For the second matrix

Enter the no of rows 2

Enter the no of columns 2

row-1Enter element 1 : 1
Enter element 2 : 2

row-2Enter element 1 : 3
Enter element 2 : 4
DO YOU WANT TO CONTINUE IF YES THEN PRESS 1 ELSE ANY KEY
1

-----MENU-----
1.INPUT MATRIX
2.ADD MATRIX
3.SUBTRACT MATRIX
4.MULTIPLY
5.EQUAL OR NOT
6.DISPLAY
7.EXIT
ENTER CHOICE (1/2/3/4/5)
```

C:\Users\VERMA\Desktop\oop\T3.exe

```
2
Addition
2      4
6      8
DO YOU WANT TO CONTINUE IF YES THEN PRESS 1 ELSE ANY KEY
1
```

-----MENU-----

```
1.INPUT MATRIX
2.ADD MATRIX
3.SUBTRACT MATRIX
4.MULTIPLY
5.EQUAL OR NOT
6.DISPLAY
7.EXIT
ENTER CHOICE (1/2/3/4/5)
```

```
3
Subtraction
0      0
0      0
DO YOU WANT TO CONTINUE IF YES THEN PRESS 1 ELSE ANY KEY
1
```

-----MENU-----

```
1.INPUT MATRIX
2.ADD MATRIX
3.SUBTRACT MATRIX
4.MULTIPLY
5.EQUAL OR NOT
6.DISPLAY
7.EXIT
ENTER CHOICE (1/2/3/4/5)
```

```
4
Multiplication
7      10
15     22
DO YOU WANT TO CONTINUE IF YES THEN PRESS 1 ELSE ANY KEY
1
```

-----MENU-----

```
1.INPUT MATRIX
2.ADD MATRIX
3.SUBTRACT MATRIX
4.MULTIPLY
```



```
5.EQUAL OR NOT
6.DISPLAY
7.EXIT
ENTER CHOICE (1/2/3/4/5)
5
EQUAL OR NOT ?
EQUAL
DO YOU WANT TO CONTINUE IF YES THEN PRESS 1 ELSE ANY KEY
1

-----MENU-----
1.INPUT MATRIX
2.ADD MATRIX
3.SUBTRACT MATRIX
4.MULTIPLY
5.EQUAL OR NOT
6.DISPLAY
7.EXIT
ENTER CHOICE (1/2/3/4/5)
6
First matrix
1      2
3      4
Second matrix
1      2
3      4
DO YOU WANT TO CONTINUE IF YES THEN PRESS 1 ELSE ANY KEY
```