CLASSWORK APPLICATIONS FOR MULTI-DIMENSIONAL ARRAYS

Q1. Using multi-dimensional arrays, write a program to display the following matrix.

1	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
0	**2**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**3**	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**4**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**5**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**6**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**7**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**	**8**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**9**

```
#include <iostream>
using namespace std;

int main(){
    int m[9][9]={0};
    for(int i=0; i<9; i++){
        for(int j=0; j<9; j++){
            if(i==j){m[i][j]=i+1;
            } else m[i][j]=0;
        }}
    for(int i=0; i<9; i++){
        for(int j=0; j<9; j++){
            cout<<m[i][j];
        } cout<<endl;
        }
}
```

Q2. Using multi-dimensional arrays, write a program to display the following matrix.

1	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
0	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
0	**0**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
0	**0**	**0**	**1**	**1**	**1**	**1**	**1**	**1**
0	**0**	**0**	**0**	**1**	**1**	**1**	**1**	**1**
0	**0**	**0**	**0**	**0**	**1**	**1**	**1**	**1**
0	**0**	**0**	**0**	**0**	**0**	**1**	**1**	**1**
0	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**1**
0	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**

```cpp
#include <iostream>
using namespace std;

int main(){
    int m[9][9]={0};
    for(int i=0; i<9; i++){
        for(int j=0; j<9; j++){
            if(i<=j){m[i][j]=1;
            } else m[i][j]=0;
        }}
    for(int i=0; i<9; i++){
        for(int j=0; j<9; j++){
            cout<<m[i][j];
        } cout<<endl;
        }
}
```

Q3. Using multi-dimensional arrays, write a program to find; $5 * M$ Where.

$$M= \begin{bmatrix} 2 & 4 & 8 \\ 6 & 5 & 7 \\ 3 & 2 & 1 \end{bmatrix}$$

```cpp
#include <iostream>
using namespace std;

int main(){
    int m[3][3]={{2,4,8},{6,5,7},{3,2,1}};

    for(int i=0; i<3; i++){
        for(int j=0; j<3; j++){
            cout<<5*m[i][j]<<" ";
        }cout<<endl;
    }
}
```

Q4. Using multi-dimensional arrays, write a program to find $M^T + N$.
Where

M=

1	2	5	6
8	9	0	1

N=

1	2
3	4
5	6
7	8

```
#include <iostream>
using namespace std;

int main(){
    int m[2][4]={{1,2,5,6},{8,9,0,1}}, mt[4][2],
        n[4][2]={{1,2},{3,4},{5,6},{7,8}};

    cout<<"Transpose of m : "<<endl;

    for(int i=0; i<4; i++){
        for(int j=0; j<2; j++){
            mt[i][j]=m[j][i];
            cout<<mt[i][j]<<" ";
        }   cout<<endl;
    }

    cout<<"Result of summation : "<<endl;

    for(int i=0; i<4; i++){
        for(int j=0; j<2; j++){
            cout<<mt[i][j]+n[i][j]<<" ";
        }cout<<endl;
    }
}
```

Q5. Using multi-dimensional arrays, write a program to find $2 * M - N^T$.
Where

N=

1	2	5	6
8	9	0	1

M=

1	2
3	4
5	6
7	8

```cpp
#include <iostream>
using namespace std;

int main(){
    int n[2][4]={{1,2,5,6},{8,9,0,1}}, nt[4][2],
        m[4][2]={{1,2},{3,4},{5,6},{7,8}};

    cout<<"Transpose of n : "<<endl;

    for(int i=0; i<4; i++){
        for(int j=0; j<2; j++){
            nt[i][j]=n[j][i];
            cout<<nt[i][j]<<" ";
        }   cout<<endl;
    }

    cout<<"Result of calculation : "<<endl;

    for(int i=0; i<4; i++){
        for(int j=0; j<2; j++){
            cout<<2*m[i][j]-nt[i][j]<<" ";
        }cout<<endl;
    }
}
```