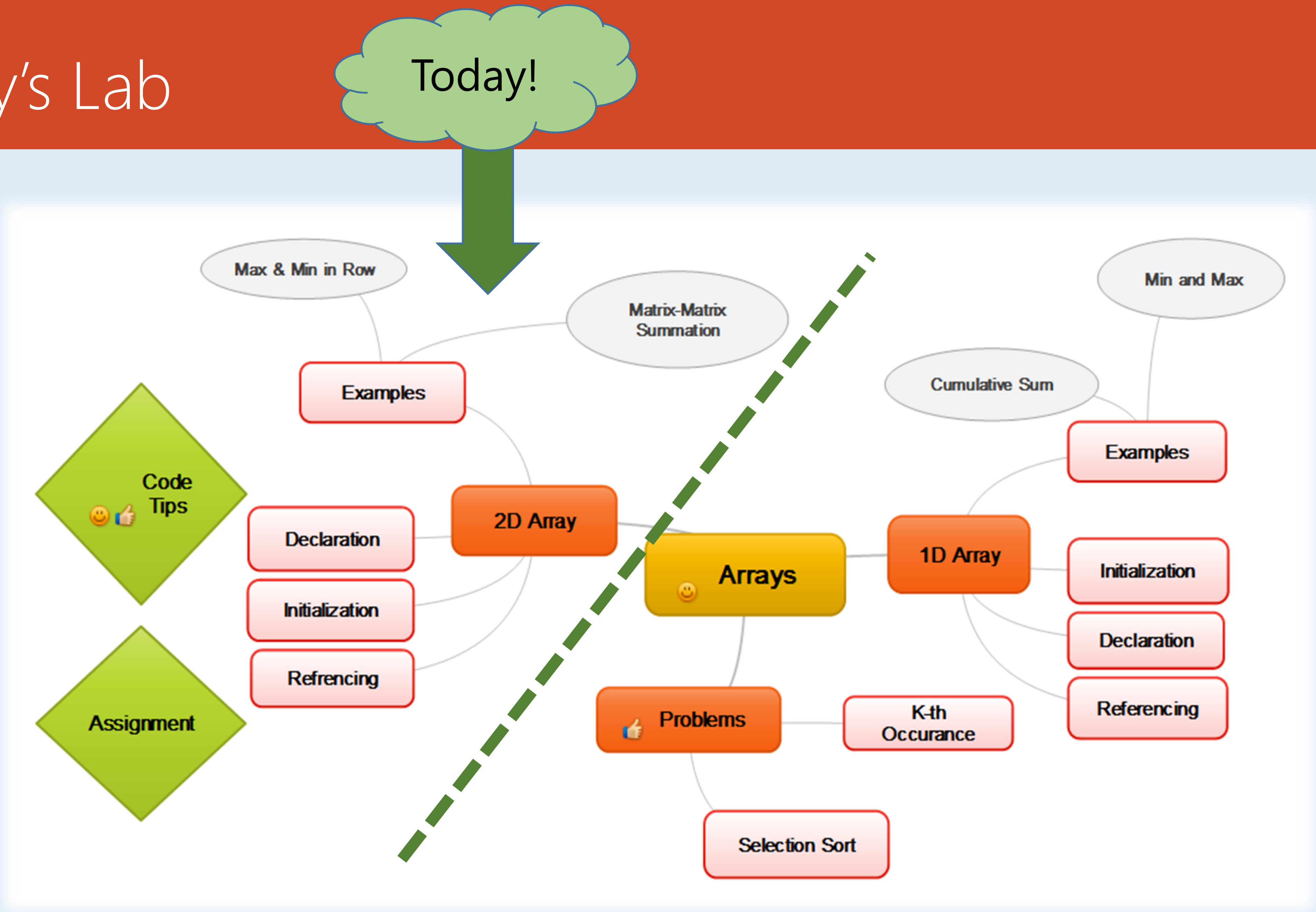
Lab #3

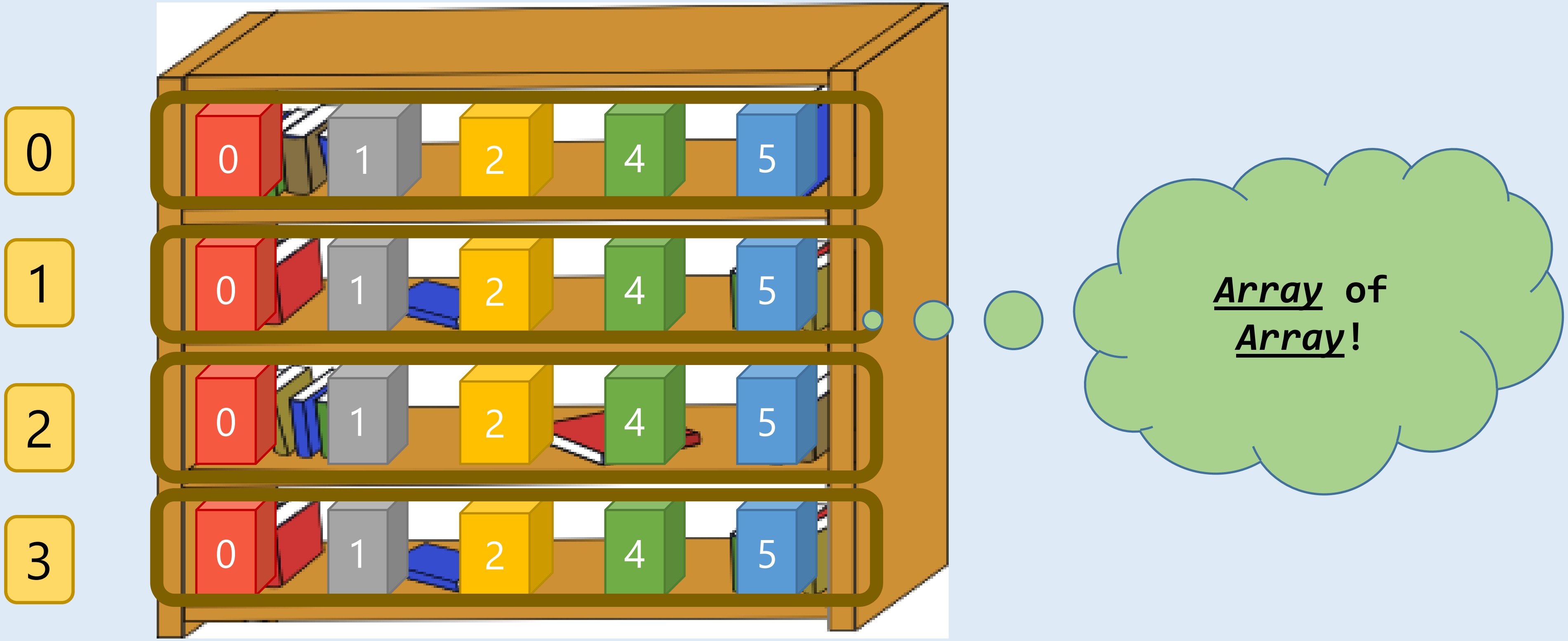
2D Arrays

Structured Programming 2017/2018

Today's Lab


Today!
Max & Min in Row
Matrix-Matrix Summation
Examples
Code Tips
2D Array
Declaration
Initialization
Refrencing
Assignment
Min and Max
Cumulative Sum
Examples
1D Array
Arrays
Initialization
Declaration
Referencing
Problems
K-th Occurance
Selection Sort

2D Arrays



Array **of** ***Array*****!**

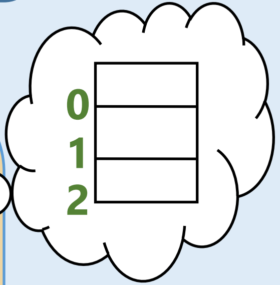
Declaration & Initialization

One Element

```
//declaration
int element;

//declaration &
  initialization
int element = 10;
```

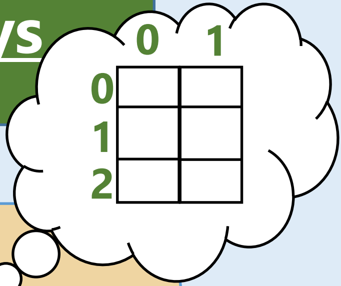
Array of elements

```
//declaration
int elements[3];

//declaration &
  initialization
int elements[3] =
          {1,2,3};
```



```
int elements[] =
   {1,2,3};
```

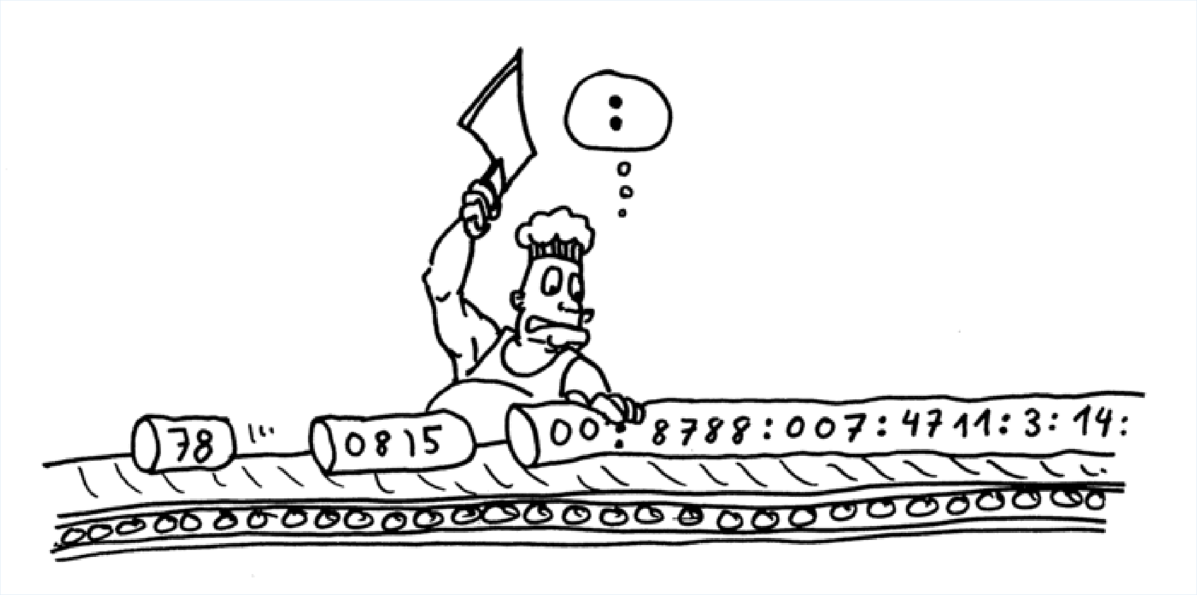
Array of Arrays



```
//declaration
int elements[3][2];

//declaration &
  initializtion
int elements[3][2] =
  {{1,2},{3},{4,5}};
```

```
int elements[][2] =
 {{1,2,3},{},{4}};
```

Exercises!



1. Read and Display 2D array

Write a program that asks the user to enter ***2×3*** matrix and outputs it again in an ***2×3*** matrix form.

Sample Execution:

```
Enter Matrix elements:
1     2     3
4     5     6
```

```
The output of Matrix
in an 2x3 format:
1     2     3
4     5     6
```

```
#include <iostream>
using namespace std;
void main()
{

    //1. declare matrix
    int arr[2][3];

    //2. read arr elements
    cout<< "Enter Matrix elements: \n";
    for(int i = 0 ; i < 2 ; i++)
    {
        for(int j = 0 ; j < 3 ; j++)
        {
            cin>> arr[i][j];
        }
    }
```

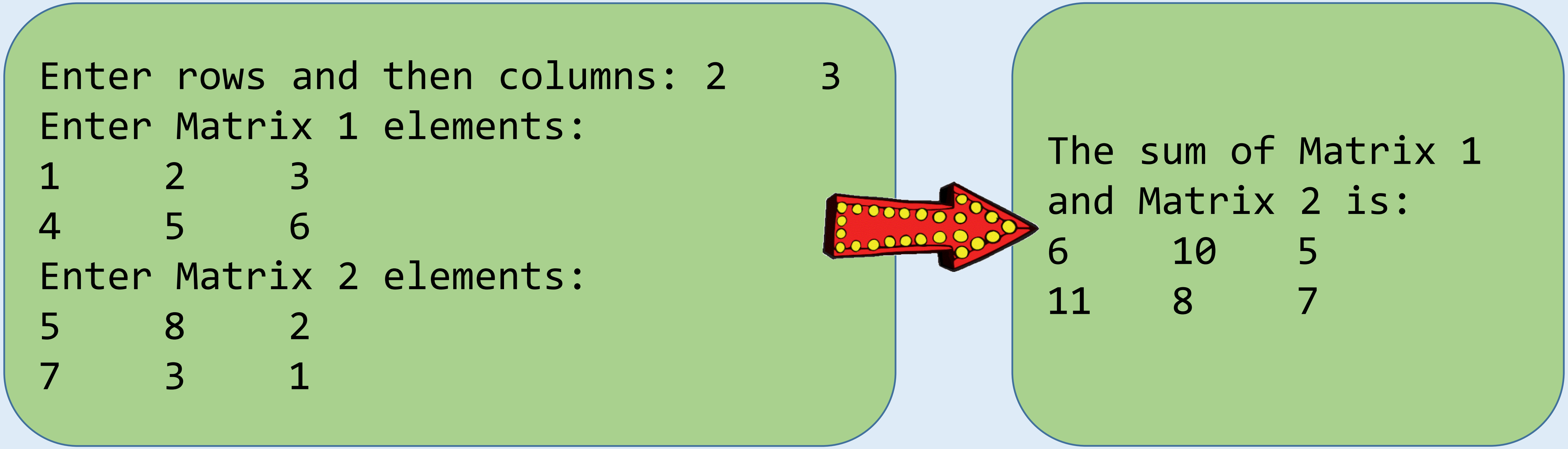
```cpp
    //3. The output of Matrix in an 2x3 format
    cout<<"The output of Matrix in an 2x3 format: \n";

    for(int i = 0 ; i < 2 ; i++)
    {
        for(int j = 0 ; j < 3 ; j++)
        {
            cout<<arr[i][j]<<" ";
        }
        cout<<endl;
    }
} // This bracket is the end of main()
```

Ready… Steady… Code!

1. Matrix Operations - Summation

Write a program that asks the user to enter **two *n×m*** matrices and outputs their **sum** in an ***n×m*** matrix form.

Sample Execution:



```
Enter rows and then columns: 2    3
Enter Matrix 1 elements:
1     2     3
4     5     6
Enter Matrix 2 elements:
5     8     2
7     3     1
```

```
The sum of Matrix 1
and Matrix 2 is:
6     10    5
11    8     7
```

```
#include <iostream>
using namespace std;
void main()
{

    //1. declare matrices
    int matA[10][10], matB[10][10], matC[10][10];
    int row, col;

    //2. read dimensions
    cout << "Please enter row then col: " << endl;
    cin >> row >> col;
```

```
//3. read matA matB
cout<< "Enter Matrix 1 elements: \n";
for(int i = 0 ; i < row ; i++)
{
    for(int j = 0 ; j < col ; j++)
    {
        cin>> matA[i][j];
    }
}

cout<< "Enter Matrix 2 elements: \n";
for(int i = 0 ; i < row ; i++)
{
    for(int j = 0 ; j < col ; j++)
    {
        cin>> matB[i][j];
    }
}
```

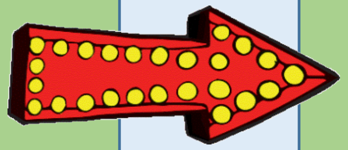
```
    cout<<"The Sum of matrix 1 and matrix 2 is: \n";

    //4. calculate sum of mat A and B the put the res in
    mat c
    for(int i = 0 ; i < row ; i++)
    {
        for(int j = 0 ; j < col ; j++)
        {
            matC[i][j] = matA[i][j] + matB[i][j];
            cout<<matC[i][j]<<" ";
        }
        cout<<endl;
    }
} // This bracket is the end of main()
```

2. Get The Diagonal

Write a program that asks the user to enter ***n×n*** matrix and outputs its **diagonal**.

Sample Execution:

```
Enter rows and columns: 3
Enter Matrix elements:
1     2     3
4     5     6
7     8     9
```



```
The Matrix diagonal
is:
1     5     9
```

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

void main()
{
    //1. declare matrix
    int arr[10][10];
    int n;


    //2. read n
    cout << "Please enter n :" << endl;
    cin >> n;
    //3. read n x n matrix
    cout << "Enter n x n Matrix elements:" << endl;
    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j < n; j++)
            cin >> arr[i][j];
    }
    //4. display the diagonal
    cout << "The Matrix diagonal is:" << endl;

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

        cout << arr[i][i] << '\t';

    }
    cout << endl;

    system("pause");
}
```

3. Get The Sum

Write a program that asks the user to enter ***n×n*** matrix (that contains students grades, each row contains grades of certain student) and outputs the Sum of each row.

Sample Execution:

```
Enter rows and columns: 3
Enter Matrix elements:
10    2    3
4     50   6
70    8    9
```

```
The sum of each row
is:
#row1 :- 15
#row2 :- 60
#row3 :- 87
```

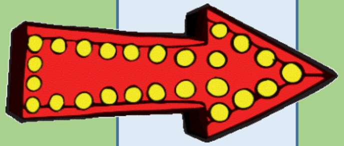
```cpp
#include <iostream>
using namespace std;

void main()
{
    //1. declare matrix
    int arr[10][10];
    int n;


    //2. read n
    cout << "Please enter n :" << endl;
    cin >> n;
    //3. read n x n matrix
    cout << "Enter n x n Matrix elements:" << endl;
    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j < n; j++)
            cin >> arr[i][j];
    }
    //4. display the diagonal
    cout << "The sum foreach row :" << endl;

    for (int i = 0; i < n; i++)
    {
        int sum = 0;
        for (int j = 0; j < n; j++)
        {
            sum += arr[i][j] ;
        }
        cout << "#row "<<i <<" " << sum<< endl;

    }
    cout << endl;

    system("pause");
}
```

4. Get The Maximum

Write a program that asks the user to enter ***n×n*** matrix and outputs the maximum in each row.

Sample Execution:

```
Enter rows and columns: 3
Enter Matrix elements:
10    2    3
4     50   6
70    8    9
```



```
The maximum in each
row is:
#row1 :- 10
#row2 :- 50
#row3 :- 70
```

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

void main()
{
    //1. declare matrix
    int arr[10][10];
    int n;


    //2. read n
    cout << "Please enter n :" << endl;
    cin >> n;
    //3. read n x n matrix
    cout << "Enter n x n Matrix elements:" << endl;
    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j < n; j++)
            cin >> arr[i][j];
    }
    //4. display the diagonal
    cout << "The Maximum in each row :" << endl;

    for (int i = 0; i < n; i++)
    {
        int max = arr[i][0];
        for (int j = 0; j < n; j++)
        {
            if (arr[i][j] > max)
            {
                max = arr[i][j];
            }
        }
        cout << "#row "<<i <<" max is " << max<< endl;

    }
    cout << endl;

    system("pause");
}
```

Home Exercise

4.Matrix Transpose

Write a program that asks the user to enter ***n×m*** matrix and outputs the Transposed Matrix.

Sample Execution:

```
Enter rows and then columns: 3    2
Enter Matrix elements:
1  2
3  4
5  6
```

```
The Transposed
Matrix:
1  3  5
2  4  6
```

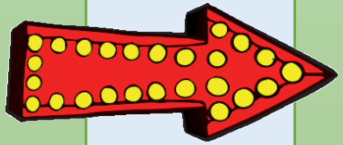
4.Matrix Mutiplication

Write a program that asks the user to enter two matrices ***n×m and m×n*** matrix and outputs their multiplication.

4.Matrix Mutiplication (Cont.)

Sample Execution:

```
Enter rows and columns of firs matrix: 2   3
Enter the elements of first matrix:
3          -2          5
3          0           4

Enter rows and columns of second matrix: 3   2

Enter the elements of second matrix:
2          3
-9         0
0          4
```



```
The output Matrix:
24     29
6      25
```

Thank you!