Ans 1 - Given a 2D matrix with m rows and n columns containing integers, find and print the maximum value present in the array.

.vscode > assignment_2daarray.cpp

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

int main(){
    int n,m;
    cin>>n>>m;
    int arr[n][m];
    for(int i=0;i<n;i++){
        for(int j=0;j<m;j++){
            cin>>arr[i][j];
        }
    }
    int ans=arr[0][0];
    for(int i=0;i<n;i++){
        for(int j=0;j<m;j++){
            if(ans<arr[i][j]){
                ans=arr[i][j];
            }
        }
    }
    cout<<"Mximum value : "<<ans;
    return 0;
}
```

```
3
PS D:\cpprograme\.vscode> cd "d:\cpprograme\.vscode\" ;
 assignment_2daarray } ; if ($?) { .\assignment_2daarray
3
5
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
Mximum value : 15
PS D:\cpprograme\.vscode>
```

Ans 2 - You are given a n*n square matrix, you need to rotate the matrix by 90 degrees in clockwise direction. You need to do it in-place i.e. you are not allowed to make a new matrix and allocate the elements to it. Make the changes in the same matrix and print it.

recursion.cpp U | recursion1.cpp U | assignment_2daarray.cpp U

.vscode > assignment_2daarray.cpp

```cpp
#include<vector>
using namespace std;
void rotateArray(vector<vector<int>> &vec){
    int n=vec.size();
    // for transpose
    for(int i=0;i<n;i++){
        for(int j=0;j<n;j++){
            swap(vec[i][j],vec[i][j]);
        }
    }
    // for reverse
    for(int i=0;i<n;i++){
        reverse(vec[i].begin(),vec[i].end());
    }
    return;
}

int main(){
    int n;
    cin>>n;
    vector<vector<int >> vec(n,vector<int >(n));
    for(int i=0;i<n;i++){
        for(int j=0;j<n;j++){
            cin>>vec[i][j];
        }
    }
    rotateArray(vec);
    for(int i=0;i<n;i++){
        for(int j=0;;j<n;j++){
            cout<<vec[i][j];
        }
    }

    return 0;
```

Ans 3 - Given a m*n integer matrix. If an element of the matrix is 0 then set the complete row and column of that element to 0. Make the changes inplace and print the matrix.

.vscode > assignment_2darray1.cpp

```cpp
#include <iostream>
#include <vector>

using namespace std;

void setZeroes(vector<vector<int>>& matrix) {
    int m = matrix.size();
    int n = matrix[0].size();
    bool first_row_has_zero = false;
    bool first_col_has_zero = false;

    // Check if the first row and column should be set to zero
    for (int j = 0; j < n; j++) {
        if (matrix[0][j] == 0) {
            first_row_has_zero = true;
            break;
        }
    }

    for (int i = 0; i < m; i++) {
        if (matrix[i][0] == 0) {
            first_col_has_zero = true;
            break;
        }
    }

    // Use the first row and column as a flag to indicate whether the rest of t
    for (int i = 1; i < m; i++) {
        for (int j = 1; j < n; j++) {
            if (matrix[i][j] == 0) {
                matrix[i][0] = 0;
                matrix[0][j] = 0;
            }
        }
```

.vscode > assignment_2darray1.cpp

```cpp
        for (int j = 1; j < n; j++) {
            if (matrix[i][j] == 0) {
                matrix[i][0] = 0;
                matrix[0][j] = 0;
            }
        }
    }

    // Set the rows and columns to zero based on the flag in the first row and colum
    for (int i = 1; i < m; i++) {
        for (int j = 1; j < n; j++) {
            if (matrix[i][0] == 0 || matrix[0][j] == 0) {
                matrix[i][j] = 0;
            }
        }
    }

    // Set the first row and column to zero if necessary
    if (first_row_has_zero) {
        for (int j = 0; j < n; j++) {
            matrix[0][j] = 0;
        }
    }

    if (first_col_has_zero) {
        for (int i = 0; i < m; i++) {
            matrix[i][0] = 0;
        }
    }
}

int main() {
    // Example usage
```

.vscode > assignment_2darray1.cpp

```
56          }
57      }
58  }
59
60  int main() {
61      // Example usage
62      vector<vector<int>> matrix = {{1, 1, 1}, {1, 0, 1}, {1, 1, 1}};
63
64      cout << "Original matrix:" << endl;
65      for (int i = 0; i < matrix.size(); i++) {
66          for (int j = 0; j < matrix[i].size(); j++) {
67              cout << matrix[i][j] << " ";
68          }
69          cout << endl;
70      }
71
72      setZeroes(matrix);
73
74      cout << "Modified matrix:" << endl;
75      for (int i = 0; i < matrix.size(); i++) {
76          for (int j = 0; j < matrix[i].size(); j++) {
77              cout << matrix[i][j] << " ";
78          }
79          cout << endl;
80      }
81
82      return 0;
83  }
84
```

```
PS D:\cpprograme\.vscode> cd "d:\cpprograme\.vscode\"
y1 } ; if ($?) { .\assignment_2darray1 }
Original matrix:
1 1 1
1 0 1
1 1 1
Modified matrix:
1 0 1
0 0 0
1 0 1
PS D:\cpprograme\.vscode>
```

Ln 84