C++ Assignments | 2D Arrays - 2 | Week 6

Write a program to print the elements of both the diagonals in a square matrix.

Input 1:

```
1 2 3
4 5 6
7 8 9
```

Output 1:

```
1 3
5
7 9
```

Write a program to rotate the matrix by 90 degrees anti-clockwise.

Input 1:

```
1 2 3
4 5 6
7 8 9
```

Output 1:

```
3 6 9
2 5 8
1 4 7
```

Write a program to print the matrix in wave form.

Input :

```
1 2 3
4 5 6
7 8 9
```

Output : 7 4 1 2 5 8 9 6 3

Given a positive integer n, generate a n x n matrix filled with elements from 1 to n2 in spiral order.

Input 1: n = 3

Output 1: [[1,2,3],[8,9,4],[7,6,5]]

Input 2: n = 1

Output 2: [[1]]

Q5. Predict the output :

```
int main(){
int a[][2] = {{1,2},{3,4}};
int i, j;
for (i = 0; i < 2; i++)
for (j = 0; j < 2; j++)
cout << a[i][j];
return 0;
}
```

ANSWER:-

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

void printDiagonals(int matrix[][3], int size) {
    cout << "Primary diagonal: ";
    for (int i = 0; i < size; i++) {
        cout << matrix[i][i] << " ";
    }
    cout << endl;

    cout << "Secondary diagonal: ";
    for (int i = 0; i < size; i++) {
        cout << matrix[i][size - i - 1] << " ";
    }
    cout << endl;
}

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

    printDiagonals(matrix, 3);

    return 0;
}
```

OUTPUT:-

```
Primary diagonal: 1 5 9
Secondary diagonal: 3 5 7
```

2.

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

void rotate90AntiClockwise(int matrix[][3], int size) {
    for (int i = 0; i < size / 2; i++) {
        for (int j = i; j < size - i - 1; j++) {
            int temp = matrix[i][j];
            matrix[i][j] = matrix[j][size - 1 - i];
            matrix[j][size - 1 - i] = matrix[size - 1 - i][size - 1 - j];
            matrix[size - 1 - i][size - 1 - j] = matrix[size - 1 - j][i];
            matrix[size - 1 - j][i] = temp;
        }
    }
}

void printMatrix(int matrix[][3], int size) {
    for (int i = 0; i < size; i++) {
        for (int j = 0; j < size; j++) {
            cout << matrix[i][j] << " ";
        }
        cout << endl;
    }
}

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

    rotate90AntiClockwise(matrix, 3);
    printMatrix(matrix, 3);

    return 0;
}
```

OUTPUT:-

```
3 6 9
2 5 8
1 4 7
```

3.

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

void printWaveForm(int matrix[][3], int size) {
    for (int j = 0; j < size; j++) {
        if (j % 2 == 0) {
            for (int i = size - 1; i >= 0; i--) {
                cout << matrix[i][j] << " ";
            }
        } else {
            for (int i = 0; i < size; i++) {
                cout << matrix[i][j] << " ";
            }
        }
    }
    cout << endl;
}

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

    printWaveForm(matrix, 3);

    return 0;
}
```

OUTPUT:-

```
7 4 1 2 5 8 9 6 3
```

4.

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

vector<vector<int>> generateMatrix(int n) {
    vector<vector<int>> matrix(n, vector<int>(n));
    int value = 1;
    int top = 0, bottom = n - 1, left = 0, right = n - 1;

    while (value <= n * n) {
        for (int i = left; i <= right; i++) {
            matrix[top][i] = value++;
        }
        top++;
        for (int i = top; i <= bottom; i++) {
            matrix[i][right] = value++;
        }
        right--;
        for (int i = right; i >= left; i--) {
            matrix[bottom][i] = value++;
        }
        bottom--;
        for (int i = bottom; i >= top; i--) {
            matrix[i][left] = value++;
        }
        left++;
    }

    return matrix;
}

void printMatrix(const vector<vector<int>>& matrix) {
    for (const auto& row : matrix) {
        for (int val : row) {
            cout << val << " ";
        }
        cout << endl;
    }
}

int main() {
    int n1 = 3;
    vector<vector<int>> matrix1 = generateMatrix(n1);
    cout << "Spiral order matrix for n = " << n1 << ":" << endl;
    printMatrix(matrix1);
    int n2 = 1;
    vector<vector<int>> matrix2 = generateMatrix(n2);
    cout << "Spiral order matrix for n = " << n2 << ":" << endl;
    printMatrix(matrix2);

    return 0;
}
```

OUTPUT:-

```
Spiral order matrix for n = 3:
1 2 3
8 9 4
7 6 5

Spiral order matrix for n = 1:
1
```

5.

OUTPUT IS :-

```
int main() {
    int a[][2] = {{1, 2}, {3, 4}};
    int i, j;
    for (i = 0; i < 2; i++)
        for (j = 0; j < 2; j++)
            cout << a[i][j];
    return 0;
}
```

1234