1. Write a C++ program to input marks of n students (let's say 5), store them in an array, and display them.
2. Write a program to find the maximum and minimum values in an array of integers with size 10.
3. Write a program that stores temperatures for 7 days and calculates the average temperature.
4. Write a program to store sales of 3 products in 4 regions using a 2D array and find the total sales per product.

	R1	R2	R3	R4
P1	100	120	130	90
P2	80	95	100	110
P3	60	70	75	65

5. Write a C++ program to add two 2x2 matrices.
6. Write a program to find the transpose of a 2x3 matrix.

```
Original Matrix:
1 2 3
4 5 6

Transpose Matrix:
1 4
2 5
3 6
```

7. Given a 3x3 matrix, find the row with the highest sum.

Output

```
Original Matrix:
3 2 1
4 5 6
1 1 1

Row with max sum: 2 (Sum = 15)
```

8. Write a program to check if a Sudoku row has duplicate numbers.

```
Output
Sudoku Grid (8x8):
5 3 8 6 7 2 1 4
6 7 2 1 8 5 3 4
1 8 4 3 5 6 2 7
2 5 3 3 4 1 6 8
4 2 7 8 6 1 5 3
8 6 1 5 3 7 4 2
3 4 5 2 1 8 7 6
7 1 6 4 2 3 8 5
Duplicate number in row 4: 3
Duplicate number in column 4: 3
Duplicate number in column 6: 1
Duplicate number in column 8: 4

❌ Sudoku grid has one or more errors (see above).
```

9. **Write a program to increase the brightness of a 3x3 grayscale image by 20.**

   Each pixel value of the image is represented as an integer between **0 and 255**. After increasing brightness, **ensure that no pixel value exceeds 255** (i.e., cap the values at 255).

```
Output
Original Image Matrix:
230 100 50
200 180 90
60  30  250

Brightness-Increased Image Matrix:
250 120 70
220 200 110
80  50  255
```

10. Implement a program that uses a 2D array to represent a tic-tac-toe board. Allow users to input their moves and update the board.

Output

```
Tic-Tac-Toe Board:
O 2 3
4 X 6
7 8 9
Player X, enter your move (1-9): 2

Tic-Tac-Toe Board:
O X 3
4 X 6
7 8 9
Player O, enter your move (1-9): 4

Tic-Tac-Toe Board:
O X 3
O X 6
7 8 9
Player X, enter your move (1-9): 8

Tic-Tac-Toe Board:
O X 3
O X 6
7 X 9
Player X wins!
```