

Άσκηση 1: Αναζήτηση Στοιχείου σε Διδιάστατο Πίνακα

Περιγραφή: Γράψτε έναν αλγόριθμο που ψάχνει για ένα συγκεκριμένο στοιχείο σε έναν διδιάστατο πίνακα και επιστρέφει τη θέση του.

Κώδικας:

```
using System;

class Program
{
    static void Main()
    {
        int[,] matrix = {
            { 1, 2, 3 },
            { 4, 5, 6 },
            { 7, 8, 9 }
        };

        int target = 5;
        var result = Search(matrix, target);

        if (result.HasValue)
        {
            Console.WriteLine($"Element {target} found at: ({result.Value.row}, {result.Value.col})");
        }
        else
        {
            Console.WriteLine($"Element {target} not found.");
        }
    }

    static (int row, int col)? Search(int[,] matrix, int target)
    {
        for (int row = 0; row < matrix.GetLength(0); row++)
        {
            for (int col = 0; col < matrix.GetLength(1); col++)
            {
                if (matrix[row, col] == target)
                {
                    return (row, col);
                }
            }
        }
        return null;
    }
}
```

Άσκηση 2: Εύρεση Πρώτου Στοιχείου Μικρότερου από Δεδομένο Αριθμό

Περιγραφή: Γράψτε έναν αλγόριθμο που ψάχνει για το πρώτο στοιχείο σε έναν διδιάστατο πίνακα που είναι μικρότερο από έναν δεδομένο αριθμό και επιστρέφει τη θέση του.

Κώδικας:

```
using System;

class Program
{
    static void Main()
    {
        int[,] matrix = {
            { 10, 20, 30 },
            { 40, 50, 60 },
            { 70, 80, 90 }
        };

        int target = 35;
        var result = FindFirstLessThan(matrix, target);

        if (result.HasValue)
        {
            Console.WriteLine($"First element less than {target} found at: ({result.Value.row}, {result.Value.col})");
        }
        else
        {
            Console.WriteLine($"No element less than {target} found.");
        }
    }

    static (int row, int col)? FindFirstLessThan(int[,] matrix, int target)
    {
        for (int row = 0; row < matrix.GetLength(0); row++)
        {
            for (int col = 0; col < matrix.GetLength(1); col++)
            {
                if (matrix[row, col] < target)
                {
                    return (row, col);
                }
            }
        }
        return null;
    }
}
```

Άσκηση 3: Εύρεση Αθροίσματος Στοιχείων σε Στήλη

Περιγραφή: Γράψτε έναν αλγόριθμο που υπολογίζει το άθροισμα των στοιχείων μιας συγκεκριμένης στήλης σε έναν διδιάστατο πίνακα.

Κώδικας:

```
using System;

class Program
{
    static void Main()
    {
        int[,] matrix = {
            { 1, 2, 3 },
            { 4, 5, 6 },
            { 7, 8, 9 }
        };

        int column = 1; // 2η στήλη (0-βάση)
        int sum = SumColumn(matrix, column);

        Console.WriteLine($"Sum of column {column} is: {sum}");
    }

    static int SumColumn(int[,] matrix, int column)
    {
        int sum = 0;
        for (int row = 0; row < matrix.GetLength(0); row++)
        {
            sum += matrix[row, column];
        }
        return sum;
    }
}
```

Άσκηση 4: Εύρεση Μέγιστου Στοιχείου σε Διδιάστατο Πίνακα

Περιγραφή: Γράψτε έναν αλγόριθμο που βρίσκει το μέγιστο στοιχείο σε έναν διδιάστατο πίνακα και επιστρέφει τη θέση του.

Κώδικας:

```
using System;

class Program
{
```

```

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

    var result = FindMax(matrix);

    Console.WriteLine($"Max element {result.value} found at: ({result.row}, {result.col})");
}

static (int value, int row, int col) FindMax(int[,] matrix)
{
    int maxValue = int.MinValue;
    int maxRow = -1;
    int maxCol = -1;

    for (int row = 0; row < matrix.GetLength(0); row++)
    {
        for (int col = 0; col < matrix.GetLength(1); col++)
        {
            if (matrix[row, col] > maxValue)
            {
                maxValue = matrix[row, col];
                maxRow = row;
                maxCol = col;
            }
        }
    }

    return (maxValue, maxRow, maxCol);
}
}

```

Άσκηση 5: Αναζήτηση Στοιχείου σε Τρισδιάστατο Πίνακα

Περιγραφή: Γράψτε έναν αλγόριθμο που ψάχνει για ένα συγκεκριμένο στοιχείο σε έναν τρισδιάστατο πίνακα και επιστρέφει τη θέση του.

Κώδικας:

```
using System;
```

```

class Program
{
    static void Main()
    {
        int[, ,] matrix = {
            {
                { 1, 2, 3 },
                { 4, 5, 6 }
            },
            {
                { 7, 8, 9 },
                { 10, 11, 12 }
            }
        };

        int target = 11;
        var result = Search3D(matrix, target);

        if (result.HasValue)
        {
            Console.WriteLine($"Element {target} found at: ({result.Value.depth}, {result.Value.row}, {result.Value.col})");
        }
        else
        {
            Console.WriteLine($"Element {target} not found.");
        }
    }

    static (int depth, int row, int col)? Search3D(int[, ,] matrix, int target)
    {
        for (int depth = 0; depth < matrix.GetLength(0); depth++)
        {
            for (int row = 0; row < matrix.GetLength(1); row++)
            {
                for (int col = 0; col < matrix.GetLength(2); col++)
                {
                    if (matrix[depth, row, col] == target)
                    {
                        return (depth, row, col);
                    }
                }
            }
        }
        return null;
    }
}

```

