

Άσκηση 1: Βασική Εφαρμογή του Counting Sort

Άσκηση: Δώσε μια λίστα αριθμών και χρησιμοποίησε το Counting Sort για να τους ταξινομήσεις σε αύξουσα σειρά. Η λίστα περιλαμβάνει τους αριθμούς: [4, 2, 2, 8, 3, 3, 1].

Λύση:

```
using System;
```

```
class CountingSortExercise1
```

```
{
```

```
    static void Main()
```

```
    {
```

```
        int[] numbers = { 4, 2, 2, 8, 3, 3, 1 };
```

```
        int[] sortedNumbers = CountingSort(numbers);
```

```
        Console.WriteLine("Ταξινομημένοι αριθμοί:");
```

```
        foreach (int num in sortedNumbers)
```

```
        {
```

```
            Console.Write(num + " ");
```

```
        }
```

```
    }
```

```
    static int[] CountingSort(int[] array)
```

```
    {
```

```
        int max = FindMax(array);
```

```
        int[] count = new int[max + 1];
```

```
        // Μετράμε κάθε αριθμό
```

```
        foreach (int num in array)
```

```
        {
```

```
            count[num]++;
```

```
        }
```

```
        int index = 0;
```

```
        // Δημιουργούμε τη νέα ταξινομημένη λίστα
```

```
        for (int i = 0; i < count.Length; i++)
```

```

        {
            while (count[i] > 0)
            {
                array[index++] = i;
                count[i]--;
            }
        }

        return array;
    }

    static int FindMax(int[] array)
    {
        int max = array[0];
        for (int i = 1; i < array.Length; i++)
        {
            if (array[i] > max)
                max = array[i];
        }
        return max;
    }
}

```

Άσκηση 2: Ταξινοδμηση Θετικών Αριθμών

Άσκηση: Δώσε μια λίστα με θετικούς αριθμούς και χρησιμοποίησε το Counting Sort για να τους ταξινομήσεις σε αύξουσα σειρά. Η λίστα περιλαμβάνει τους αριθμούς: [9, 4, 6, 7, 1, 2, 2, 4, 5].

Λύση:

```
using System;
```

```

class CountingSortExercise2
{
    static void Main()
    {
        int[] numbers = { 9, 4, 6, 7, 1, 2, 2, 4, 5 };
        int[] sortedNumbers = CountingSort(numbers);

        Console.WriteLine("Ταξινομημένοι αριθμοί:");
        foreach (int num in sortedNumbers)

```

```

    {
        Console.Write(num + " ");
    }
}

```

```

static int[] CountingSort(int[] array)

```

```

{
    int max = FindMax(array);
    int[] count = new int[max + 1];

    // Μετράμε κάθε αριθμό
    foreach (int num in array)
    {
        count[num]++;
    }

    int index = 0;
    // Δημιουργούμε τη νέα ταξινομημένη λίστα
    for (int i = 0; i < count.Length; i++)
    {
        while (count[i] > 0)
        {
            array[index++] = i;
            count[i]--;
        }
    }

    return array;
}

```

```

static int FindMax(int[] array)

```

```

{
    int max = array[0];
    for (int i = 1; i < array.Length; i++)
    {
        if (array[i] > max)
            max = array[i];
    }
    return max;
}

```

```
}
```

Άσκηση 3: Ταξινομήση Μικρής Κλίμακας με Κύκλους

Άσκηση: Δώσε μια λίστα με αριθμούς που περιλαμβάνει κάποιους αριθμούς πολλές φορές και χρησιμοποίησε το Counting Sort για να τους ταξινομήσεις. Η λίστα περιλαμβάνει τους αριθμούς: [5, 2, 5, 2, 3, 1, 4, 5, 2].

Λύση:

```
using System;
```

```
class CountingSortExercise3
```

```
{
```

```
    static void Main()
```

```
    {
```

```
        int[] numbers = { 5, 2, 5, 2, 3, 1, 4, 5, 2 };
```

```
        int[] sortedNumbers = CountingSort(numbers);
```

```
        Console.WriteLine("Ταξινομημένοι αριθμοί:");
```

```
        foreach (int num in sortedNumbers)
```

```
        {
```

```
            Console.Write(num + " ");
```

```
        }
```

```
    }
```

```
    static int[] CountingSort(int[] array)
```

```
    {
```

```
        int max = FindMax(array);
```

```
        int[] count = new int[max + 1];
```

```
        // Μετράμε κάθε αριθμό
```

```
        foreach (int num in array)
```

```
        {
```

```
            count[num]++;
```

```
        }
```

```
        int index = 0;
```

```
        // Δημιουργούμε τη νέα ταξινομημένη λίστα
```

```
        for (int i = 0; i < count.Length; i++)
```

```
        {
```

```

        while (count[i] > 0)
        {
            array[index++] = i;
            count[i]--;
        }
    }

    return array;
}

static int FindMax(int[] array)
{
    int max = array[0];
    for (int i = 1; i < array.Length; i++)
    {
        if (array[i] > max)
            max = array[i];
    }
    return max;
}
}

```

Άσκηση 4: Δοκιμή με Μεγάλες Τιμές

Άσκηση: Δώσε μια λίστα με μεγάλους θετικούς αριθμούς και χρησιμοποίησε το Counting Sort για να τους ταξινομήσεις. Η λίστα περιλαμβάνει τους αριθμούς: [100, 300, 200, 400, 100, 300].

Λύση:

```

using System;

class CountingSortExercise4
{
    static void Main()
    {
        int[] numbers = { 100, 300, 200, 400, 100, 300 };
        int[] sortedNumbers = CountingSort(numbers);

        Console.WriteLine("Ταξινομημένοι αριθμοί:");
        foreach (int num in sortedNumbers)

```

```

        {
            Console.Write(num + " ");
        }
    }

```

```

static int[] CountingSort(int[] array)

```

```

{
    int max = FindMax(array);
    int[] count = new int[max + 1];

    // Μετράμε κάθε αριθμό
    foreach (int num in array)
    {
        count[num]++;
    }

    int index = 0;
    // Δημιουργούμε τη νέα ταξινομημένη λίστα
    for (int i = 0; i < count.Length; i++)
    {
        while (count[i] > 0)
        {
            array[index++] = i;
            count[i]--;
        }
    }

    return array;
}

```

```

static int FindMax(int[] array)

```

```

{
    int max = array[0];
    for (int i = 1; i < array.Length; i++)
    {
        if (array[i] > max)
            max = array[i];
    }
    return max;
}

```

