

Άσκηση 1: Αναζήτηση αριθμού σε ταξινομημένο πίνακα

Να γράψετε μια μέθοδο σε C# που να εφαρμόζει τον αλγόριθμο δυαδικής αναζήτησης για να βρει τη θέση ενός αριθμού σε έναν ταξινομημένο πίνακα ακεραίων. Αν ο αριθμός βρεθεί, επιστρέφεται τη θέση του, αλλιώς επιστρέφεται -1.

```
using System;
```

```
public class BinarySearchExercise
{
    public static int BinarySearch(int[] arr, int target)
    {
        int low = 0;
        int high = arr.Length - 1;

        while (low <= high)
        {
            int mid = low + (high - low) / 2;

            if (arr[mid] == target)
                return mid;
            else if (arr[mid] < target)
                low = mid + 1;
            else
                high = mid - 1;
        }

        return -1; // Επιστρέφει -1 αν το στοιχείο δε βρεθεί
    }

    public static void Main()
    {
        int[] arr = { 1, 3, 5, 7, 9, 11, 13, 15 };
        int target = 7;
        int result = BinarySearch(arr, target);

        if (result != -1)
            Console.WriteLine($"Ο αριθμός {target} βρίσκεται στη θέση {result} του πίνακα.");
        else
            Console.WriteLine($"Ο αριθμός {target} δε βρέθηκε στον πίνακα.");
    }
}
```

Άσκηση 2: Αναζήτηση σε λίστα αντικειμένων

Έστω ότι έχετε μια λίστα με αντικείμενα τύπου `Person`, η οποία είναι ταξινομημένη κατά το πεδίο `Age`. Να γράψετε μια μέθοδο που να εφαρμόζει τον αλγόριθμο δυαδικής αναζήτησης για να βρει τη θέση ενός αντικειμένου `Person` με συγκεκριμένη ηλικία στη λίστα. Αν το αντικείμενο βρεθεί, επιστρέφεται τη θέση του, αλλιώς επιστρέφεται -1.

```
using System;
using System.Collections.Generic;

public class Person
{
    public string Name { get; set; }
    public int Age { get; set; }
}

public class BinarySearchObjectExercise
{
    public static int BinarySearch(List<Person> people, int targetAge)
    {
        int low = 0;
        int high = people.Count - 1;

        while (low <= high)
        {
            int mid = low + (high - low) / 2;

            if (people[mid].Age == targetAge)
                return mid;
            else if (people[mid].Age < targetAge)
                low = mid + 1;
            else
                high = mid - 1;
        }

        return -1; // Επιστρέφει -1 αν το στοιχείο δε βρεθεί
    }

    public static void Main()
    {
        List<Person> people = new List<Person>
        {
            new Person { Name = "Alice", Age = 25 },
            new Person { Name = "Bob", Age = 30 },
            new Person { Name = "Charlie", Age = 35 },
        }
    }
}
```

```
    new Person { Name = "David", Age = 40 }  
};  
  
int targetAge = 30;  
int result = BinarySearch(people, targetAge);  
  
if (result != -1)  
    Console.WriteLine($"Το άτομο με ηλικία {targetAge} βρίσκεται στη θέση {result} της  
λίστας.");  
else  
    Console.WriteLine($"Δε βρέθηκε άτομο με ηλικία {targetAge} στη λίστα.");  
}  
}
```