

БИНАРНО ПРЕТРАЖИВАЊЕ СОРТИРАНОГ НИЗА

Бинарно претраживање сортираног низа састоји се у одсецању значајног дела простора за претрагу како би се добило на ефикасности тј. оптимизацији.

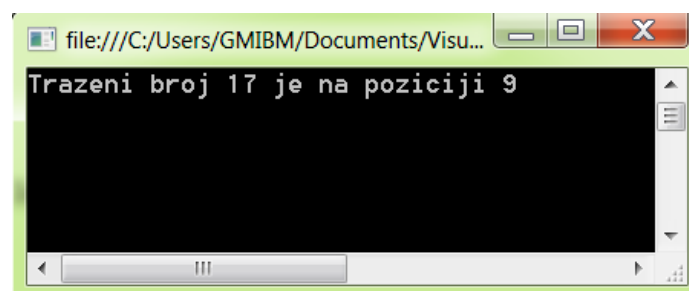
Идеја је да се дефинише ПРЕЛОМНА ТАЧКА – средина низа.

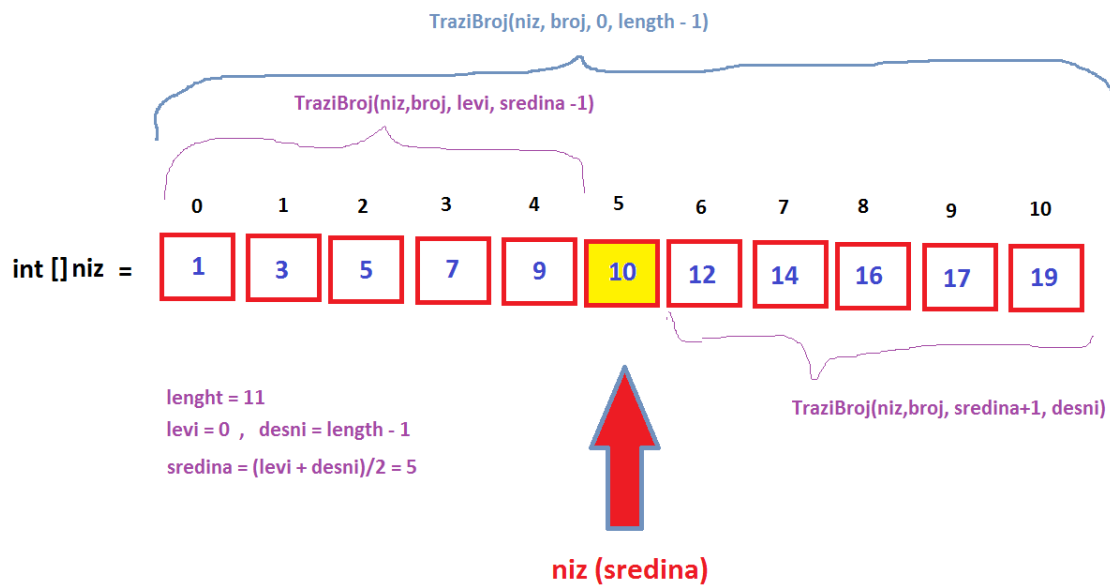
```
using System;

class Program
{
    static void Main(string[] args)
    {
        int[] niz = { 1, 3, 5, 7, 9, 10, 12, 14, 16, 17, 19 };
        int x = TraziBroj(niz, 17);
        Console.WriteLine("Trazeni broj {0} je na poziciji {1}", 17, x);
        Console.ReadKey();
    }

    public static int TraziBroj(int[] niz, int broj)
    {
        return TraziBroj(niz, broj, 0, niz.Length - 1);
    }

    public static int TraziBroj(int[] niz, int broj, int levi, int desni) {
        if (levi > desni) return -1;
        int sredina = (levi + desni) / 2;
        if (broj < niz[sredina]) return TraziBroj(niz, broj, levi, sredina - 1);
        else if (broj > niz[sredina]) return TraziBroj(niz, broj, sredina+1, desni);
        else return sredina;
    }
}
```





У програмском језику C# дат је рекурзивни метод за бинарно претраживање сортираног целобројног низа. Да би код био комплетиран потребно је допунити девети ред (означен линијом) помоћу понуђеног одговора.

```

1. public static int TraziBroj(int[] niz, int broj) {
2.     return TraziBroj(niz, broj, 0, niz.Length - 1);
3. }
4. public static int TraziBroj(int[] niz, int broj, int levi, int desni) {
5.     if(levi > desni) return -1; // broj nije nadjen u nizu
6.     int sredina = (levi + desni) / 2;
7.     if(broj < niz[sredina]) return TraziBroj(niz, broj, levi, sredina-1);
8.     else if(broj > niz[sredina]) return _____;
9.     else return sredina;
10. }
```

Заокружити број испред очекиваног одговора:

1. TraziBroj(niz, broj, sredina + 1, levi)
2. TraziBroj(niz, broj, sredina - 1, levi)
3. TraziBroj(niz, broj, desni, sredina + 1)
4. TraziBroj(niz, broj, sredina + 1, desni)