

VERİ YAPILARI

ESMA BALIKÇI

24080410019

1-DİZİDEKİ ELEMANLARIN TOPLAMINI BULMA:

```
int[] dizi = { 2, 5, 6, 8, 11 };
int toplam = 0;
for (int i = 0; i < dizi.Length; i++)
{
    toplam += dizi[i];
}
Console.WriteLine($"Dizideki sayıların toplamı: {toplam}");
```

Döngü n defa çalışır.

Zaman karmaşıklığı: $T(n)=c \cdot n \Rightarrow O(n)$

En iyi durum (T_{best}): $O(n)$

En iyi durumda bile algoritma tüm elemanları toplamak zorunda.

En kötü durum (T_{worst}): $O(n)$

En kötü durumda da aynı: bütün elemanlar üzerinden geçer.

2-MATRİSLERİN ÇARPIMI:

```
int[,] matris1 = { { 2, 5, 1 }, { 3, 2, 4 }, { 1, 3, 5 } };
int[,] matris2 = { { 2, 5, 3 }, { 4, 2, 4 }, { 3, 4, 1 } };
int[,] carpim = new int[3, 3];

for (int i = 0; i < carpim.GetLength(0); i++)
{
    for (int j = 0; j < carpim.GetLength(1); j++)
    {
        carpim[i, j] += 0;
        for (int k = 0; k < 3; k++)
        {
            carpim[i, j] += matris1[i, k] * matris2[k, j];
        }
    }
}

for (int i = 0; i < matris1.GetLength(0); i++)
{
    for (int j = 0; j < matris2.GetLength(1); j++)
    {
        Console.Write(carpim[i, j] + " ");
    }
    Console.WriteLine();
}
```

Her döngü n kez çalışıyor.

Toplam işlem sayısı: $n \times n \times n = n^3$

Zaman Karmaşıklığı: $T(n) = O(n^3)$

En iyi durum (T_{best}): $O(n^3)$

En kötü durum (T_{worst}): $O(n^3)$

3-DİZİDE İKİLİ ARAMA YAPMA:

```
static void Main(string[] args)
{
    int[] dizi = { 3, 7, 10, 15, 19 };
    int sayi = 7;
    int sonuc = ikiliarama(dizi,dizi.Length,sayi);

    if (sonuc != -1)
    {
        Console.WriteLine($"Sayı {sayi} dizinin {sonuc+1}. elemanıdır.");
    }
    else
    {
        Console.WriteLine("Sayı bulunamadı.");
    }
    Console.ReadKey();
}

public static int ikiliarama(int[] A,int n, int sayi)
{
    int sol = 0;
    int sag = n - 1;

    while (sol <= sag)
    {
        int orta=(sol+sag)/2;
        if (A[orta] == sayi)
        {
            return orta;
        }
        else if (sayi < A[orta])
        {
            sag = orta - 1;
        }
        else
        {
            sol = orta + 1;
        }
    }
    return -1;
}
```

En iyi durum (T_{best}): $O(1)$

Sıralı bir dizide aranan eleman ilk başta ortaya çıkabilir.

En kötü durum (T_{worst}): $O(\log n)$

Sayı dizinin başı, sonu veya hiç dizide yok olabilir.

Her seferinde arama aralığı yarıya düşer.