

EN KÖTÜ ZAMAN VE EN İYİ ZAMAN

j=n, i=n, k=n olsun

1. Dizi Toplamı

	Birim Zaman	Frekans	Toplam
static int DiziToplami(int[] dizi){			
int toplam = 0;	1	1	1
for(int i = 0; i < dizi.Length; i++){	1,1,1	1,(n+1),n	2n+2
toplam += dizi[i];	1	n	n
return toplam;	1	1	1

$$T(N) = 3n+3$$

$$T_{\text{Worst}}(N) = O(n)$$

$$T_{\text{Best}}(N) = \Theta(n)$$

2. Matris Çarpım

	Birim Zaman	Frekans	Toplam
static int[,] MatrisCarpim(int[,] matris1, int[,] matris2){			
int satir = matris1.GetLength(0);	1	1	1
int sutun = matris2.GetLength(1);	1	1	1
int[,] sonuc_matris = new int[satir,sutun];	1	1	1
for(int i = 0; i < satir; i++){	1,1,1	1,(n+1),n	2n+2
for(int j = 0; j < sutun; j++){	1,1,1	n(1,(n+1),n)	2n ² +2n
int toplam = 0;	1	n (n)	n ²
for (int k=0; k<matris2.GetLength(0);k++){	1,1,1	n(n(1,(n+1),n))	2n ³ +2n ²
toplam += matris1[i, k] * matris2[k, j];}	1	n(n(n))	n ³
sonuc_matris[i, j] = toplam;	1	n(n)	n ²
}			
return sonuc_matris;}	1	1	1

$$T(N) = 3n^3+6n^2+4n +6$$

$$T_{\text{Worst}}(N) = O(n^3)$$

$$T_{\text{Best}}(N) = \Theta(n^3)$$

3. İkili Arama

	Birim Zaman	Frekans	Toplam
<code>static int ikiliArama(int[] dizi, int sayi)</code>			
<code>int uzunluk = dizi.Length;</code>	1	1	1
<code>int sag = uzunluk - 1;</code>	1	1	1
<code>while (sol <= sag){</code>	1	$\log_2 n + 1$	$\log_2 n + 1$
<code>int ort = (sol + sag) / 2;</code>	1	$\log_2 n$	$\log_2 n$
<code>if (dizi[ort] == sayi){</code>	1	$\log_2 n$	$\log_2 n$
<code>return ort;</code> ;	1	1	1
<code>else if (sayi < dizi[ort]){</code>	1	$\log_2 n - 1$	$\log_2 n - 1$
<code>sag = ort - 1;</code> ;	1	$\log_2 n - 1$	$\log_2 n - 1$
<code>else{ sol = ort + 1;</code>	1	$\log_2 n - 1$	$\log_2 n - 1$
<code>}</code>			
<code>return -1;</code> ;	1	1	1

$$T(N) = 6\log_2 n + 2$$

$$T_{\text{Worst}}(N) = O(\log_2 n)$$

	Birim Zaman	Frekans	Toplam
<code>static int ikiliArama(int[] dizi, int sayi)</code>			
<code>int uzunluk = dizi.Length;</code>	1	1	1
<code>int sag = uzunluk - 1;</code>	1	1	1
<code>while (sol <= sag){</code>	1	$\log_2 n + 1$	$\log_2 n + 1$
<code>int ort = (sol + sag) / 2;</code>	1	$\log_2 n$	$\log_2 n$
<code>if (dizi[ort] == sayi){</code>	1	$\log_2 n$	$\log_2 n$
<code>return ort;</code> ;	1	1	1
<code>else if (sayi < dizi[ort]){</code>			
<code>sag = ort - 1;</code> ;			
<code>else{ sol = ort + 1;</code>			
<code>}</code>			
<code>return -1;</code> ;			

$$T(N) = 3\log_2 n + 4$$

$$T_{\text{Best}}(N) = 3\log_2 n + 4$$

$$T_{\text{Best}}(N) = \Theta(\log_2 n)$$