Böl ve Fethet (Hükmet) Metodu

Divide & Conquer

1) Problemi alt problemlere böl — Böl
2) Alt problemleri özyinelemeli olarak çöz ve hükmet — Hükmet
3) Alt problemlerin çözümlerini birleştir

Merge Sort

1) Bölmek:
2) Hükmetmek: 2 alt problemi özyinelemeli böl
3) Birleştirmek: Doğrusal zamanda, $\Theta(n)$

$$T(n) = 2 \cdot T(n/2) + \Theta(n)$$

- 2: alt problem sayısı
- $n/2$: alt problemlerin boyutu
- $\Theta(n)$: Birleştirme zamanı

Master metod ile: $n \overset{?}{\leftrightarrow} n^{\log_2 2} = n$

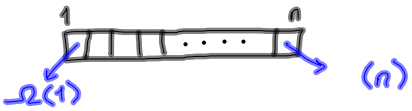
$n = \Theta(n)$

durum 2, $T(n) = \Theta(n \cdot \lg n)$

Problem: İkili Arama (Binary Search)

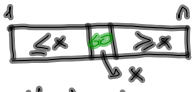
Sıralanmış bir dizide x elemanını bulma

Saf Algoritma:



Böl ve Fethet

1. Böl: Orta elemanı bul x ile karşılaştır.



2. Hükmet: 1 alt dizide özyinelemeli arama yap
3. Birleştir: Yok

Çalışma Zamanı: $T(n) = 1 \cdot T(n/2) + \Theta(1)$

1 alt problem var

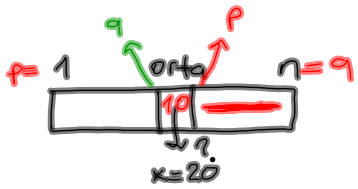
Master metod ile: $1 \overset{?}{\leftrightarrow} n^{\log_2 1} = n^0 = 1$

$1 = \Theta(1)$

durum 2, $T(n) = \Theta(\lg n)$

İkili Arama için sözde kod

1) İteratif:




```
İkili_Arama(A, x)
    p=1
    q=n
    while p≤q
        orta = ⌊(p+q)/2⌋
        if A[orta]=x
            return true
        if A[orta]<x
            p=orta+1
        else
            q=orta-1
    return false
```

2. Rekörsif (recursive)

```
İkili_Arama_Rec(A, x, p, q)
    if p≤q
        orta = ⌊(p+q)/2⌋
        if A[orta]=x
            return true
        if A[orta]<x
            İkili_Arama_Rec(A, x, orta+1, q)
        else
            İkili_Arama_Rec(A, x, p, orta-1)
    else
        return false
```

İlk Çağırım: `İkili_Arama_Rec(A, x, 1, n)`

Problem: Bir sayının kuvvetini hesaplama

x bir sayı ve n pozitif bir sayı için x^n bul.

Saf Algoritma: $x^n = \underbrace{x \cdot x \cdots x}_{n \text{ defa}}$

$T(n) = \Theta(n)$

Böl ve Fethet Alg:

$$x^n = \begin{cases} x^{n/2} \cdot x^{n/2} & \text{eğer } n \text{ çift ise} \\ x^{\frac{n-1}{2}} \cdot x^{\frac{n-1}{2}} \cdot x & \text{eğer } n \text{ tek ise} \end{cases}$$

$x^{16} = x^8 \cdot x^8$

$\downarrow$

$x^4 \cdot x^4$

$\downarrow$

$x^2 \cdot x^2$

$\downarrow$

$x \cdot x$

Çalışma Zamanı

$T(n) = 1 \cdot T(n/2) + \Theta(1$

$T(n) = \Theta(\lg n)$

```
UsAl(x, n)
  if n=1
     return x
  if n çift ise                          → Yanlış
     return UsAl(x, n/2) * UsAl(x, n/2)
  if n tek ise
     return UsAl(x,(n-1)/2) * UsAl(x,(n-1)/2)
```

UsAl(2,7)

$\downarrow$

UsAl(2,3) * UsAl(2,3) * x

$\downarrow$ → UsAl(2,1) * UsAl(2,1)

UsAl(2,1) * UsAl(2,1) * x

$T(n) = 2 \cdot T(n/2) + \Theta(1)$

$1 \leftrightarrow n^{\log_2 2} = n^1$

Doğrusu

```
UsAl(x, n)
    if n=1
        return x
    if n çift ise
        result= UsAl(x, n/2)
        return result * result
    if n tek ise
        result= UsAl(x, (n-1)/2)
        return result * result * x
```

Matrislerde çarpma

Girdi: $A = [a_{ij}]$, $B = [b_{ij}]$.
Çıktı: $C = [c_{ij}] = A \cdot B$.
$i, j = 1, 2, \ldots, n$.

$$\begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & \cdots & c_{nn} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \cdot \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{nn} \end{bmatrix}$$

$$c_{ij} = \sum_{k=1}^{n} a_{ik} \cdot b_{kj}$$

Standart algoritma

```
for i ← 1 to n          (i 1'den n'ye kadar)
    do for j ← 1 to n   (j 1'den n'ye kadar)
        do c_ij ← 0
           for k ← 1 to n
               do c_ij ← c_ij + a_ik · b_kj
```

$T(n) = \Theta(n^3)$

```
for i=1 to n
    ..Θ(1)..
    for j=1 to n
        .....
        j=j*2
```

$T(n) = \Theta(n \lg n)$

$j = 1\ 2\ 4\ 8\ 16\ \ldots\ n$

```
for i=1 to n
    .....
    for j=1 to √n
        ....
        for k=n to 1
            ....
            k=k/2
```

$T(n) = \Theta(n^{3/2} \lg n)$

```
for i=1 to n/2
    ....
    for j=1 to n
        ....
        j=j+2
for k=1 to n
    .....
    .....
```

$T(n) = \frac{n}{2} \cdot \frac{n}{2} + n$

$T(n) = \Theta(n^2)$