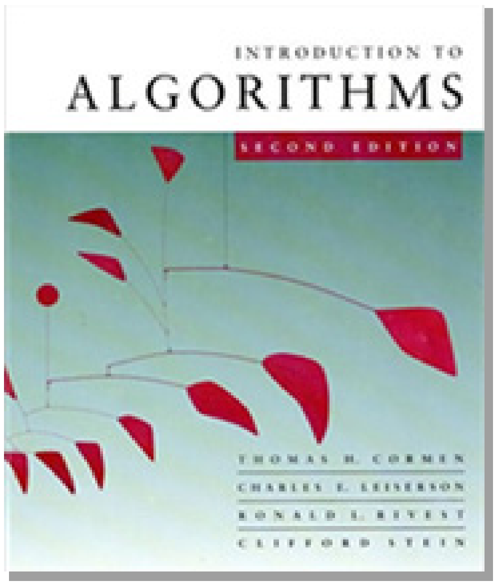
Algoritmalara Giriş

6.046J/18.401J



DERS 2

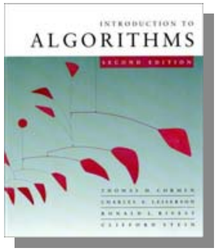
Asimptotik Simgelem

- *O*-, Ω-, ve Θ-simgelemi

Yinelemeler

- Yerine koyma metodu
- Yineleme döngüleri
- Özyineleme ağacı
- Ana Metot (Master metod)

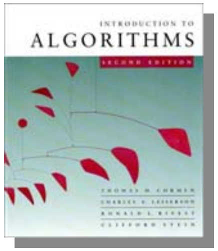
Prof. Erik Demaine



Asimptotik simgelem

O-simgelemi (üst sınırlar):

Tüm $n \geq n_0$ değerleri için sabitler $c > 0$, $n_0 > 0$ ise
$0 \leq f(n) \leq cg(n)$ durumunda
$f(n) = O(g(n))$ yazabiliriz.

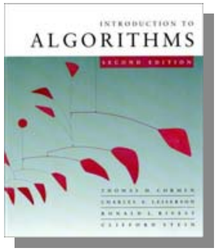


Asimptotik simgelem

O-simgelemi (üst sınırlar):

Tüm n≥n_0 değerleri için sabitler c>0, n_0>0 ise
0≤f(n)≤cg(n) durumunda
f(n)=O(g(n)) yazabiliriz.

ÖRNEK: $2n^2 = O(n^3)$ $\quad (c = 1,\ n_0 = 2)$



Asimptotik simgelem

O-simgelemi (üst sınırlar):

Tüm $n \geq n\_0$ değerleri için sabitler $c>0$, $n\_0>0$ ise
$0 \leq f(n) \leq cg(n)$ durumunda
$f(n)=O(g(n))$ yazabiliriz.

ÖRNEK: $2n^2 = O(n^3) \qquad (c = 1, n_0 = 2)$

fonksiyonlar, değerler değil

Asimptotik simgelem

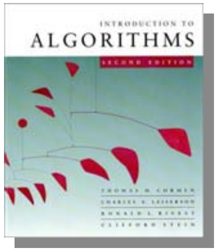
O-simgelemi (üst sınırlar):

Tüm n ≥ n_0 değerleri için sabitler c> 0, n_0 > 0 ise
0 ≤ f(n) ≤ cg(n) durumunda
f(n)=O(g(n)) yazabiliriz.

ÖRNEK: $2n^2 = O(n^3)$ $\quad (c = 1, n_0 = 2)$

fonksiyonlar, değerler değil

komik, "tek yönlü" eşitlik

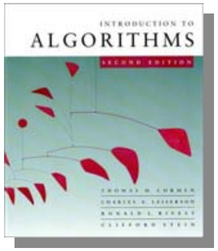


O-simgeleminin tanımı

$O(g(n)) = \{ f(n) :$ tüm $n \geq$ n_0 değerlerinde

$c > 0$, n_0 > 0 ise,

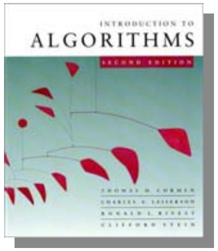
$0 \leq f(n) \leq cg(n) \}$



O-simgeleminin tanımı

$O(g(n)) = \{ f(n) :$ tüm n ≥ n_0 değerlerinde
c>0, n_0 >0 ise,
0≤f(n)≤cg(n) }

Örnek: $2n^2 \in O(n^3)$

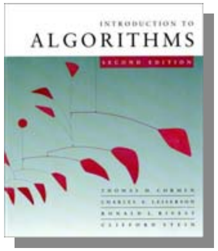


O-simgeleminin tanımı

$O(g(n)) = \{ f(n) :$ tüm n ≥ n_0 değerlerinde
c>0, n_0 >0 ise,
0≤f(n)≤cg(n) }

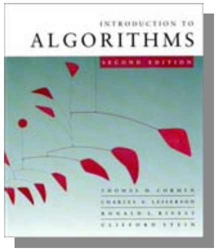
ÖRNEK: $2n^2 \in O(n^3)$

(*Mantıksallar:* $\lambda n.2n^2 \in O(\lambda n.n^3)$; ne olup bittiğini anladığımız sürece özensiz olmak yararlı olabilir.)



Makro ornatımı (substitution)

Uzlaşım (Convention): Bir formülün içindeki bir set, o setin içindeki anonim bir fonksiyonu temsil eder.



Makro ornatımı (substitution)

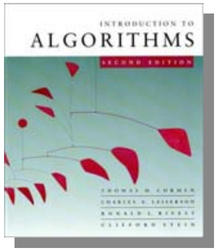
Uzlaşım: Bir formülün içindeki bir set, o setin içindeki anonim bir fonksiyonu temsil eder.

$$f(n) = n^3 + O(n^2),$$

bazı h(n) Є O(n2) için

$$f(n) = n^3 + h(n)$$

anlamına gelir.



Makro ornatımı (substitution)

Uzlaşım: Bir formülün içindeki bir set, o setin içindeki anonim bir fonksiyonu temsil eder.

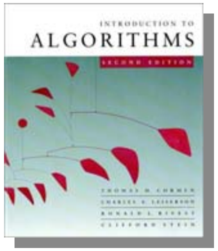
ÖRNEK: $n^2 + O(n) = O(n^2)$

şu anlama da gelir;

her $f(n) \in O(n)$için:

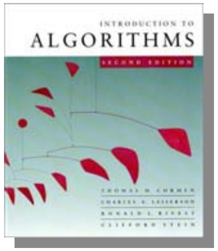
$n^2 + f(n) = h(n),$

bazı $h(n) \in O(n^2)$ olunca.



Ω-simgelemi (alt sınırlar)

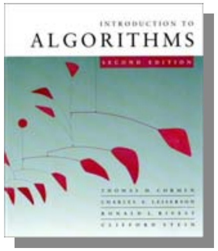
O-simgelemi bir *üst-sınır* simgelemidir. $f(n)$ en az $O(n^2)$'dir demenin bir anlamı yoktur.



Ω-simgelemi (alt sınırlar)

O-simgelemi bir üst-sınır simgelemidir. $f(n)$ en az $O(n^2)$'dir demenin bir anlamı yoktur.

$\Omega(g(n)) = \{ f(n) :$ tüm n ≥ n_0 değerlerinde
c > 0, n_0 > 0 ise,
0 ≤ cg(n) ≤ f(n) }

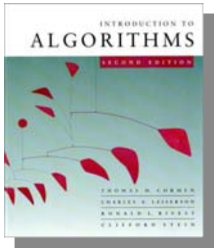


Ω-simgelemi (alt sınırlar)

O-simgelemi bir üst-sınır simgelemidir. $f(n)$ en az $O(n^2)$'dir demenin bir anlamı yoktur.

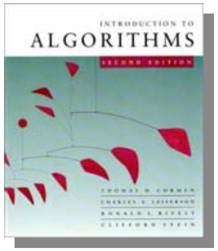
$\Omega(g(n)) = \{ f(n) :$ tüm n ≥ n_0 değerlerinde
c > 0, n_0 > 0 ise,
0 ≤ cg(n) ≤ f(n) }

ÖRNEK: $\sqrt{n} = \Omega(\lg n) \quad (c = 1,\ n_0 = 16)$



Θ-simgelemi (sıkı sınırlar)

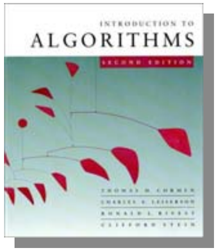
$$\Theta(g(n)) = O(g(n)) \cap \Omega(g(n))$$



Θ-simgelemi (sıkı sınırlar)

$$\Theta(g(n)) = \mathrm{O}(g(n)) \cap \Omega(g(n))$$

ÖRNEK: $\frac{1}{2}n^2 - 2n = \Theta(n^2)$



o-simgelemi ve ω-simgelemi

O-simgelemi ve Ω-simgelemi $\leq$ ve $\geq$ gibidir.
o-simgelemi ve ω-simgelemi $<$ ve $>$ gibidir..

$o(g(n)) = \{ f(n) :$ tüm n≥n_0 değerlerinde,
c>0 sabiti için n_0 sabiti varsa
0≤ f(n)≤ cg(n) }

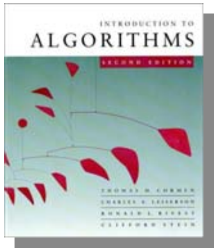
ÖRNEK: $2n^2 = o(n^3)$ $(n_0 = 2/c)$

o-simgelemi ve ω-simgelemi

O-simgelemi ve Ω-simgelemi $\leq$ ve $\geq$ gibidir.
o-simgelemi ve ω-simgelemi $<$ ve $>$ gibidir.

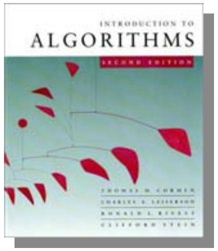
$\omega(g(n)) = \{ f(n) :$ tüm n ≥ n_0 değerlerinde,
c > 0 sabiti için n_0 sabiti varsa
0 ≤ f(n) ≤ cg(n) }

ÖRNEK: $\sqrt{n} = \omega(\lg n)$ $(n_0 = 1 + 1/c)$



Yinelemelerin çözümü

- ***Ders 1***' deki birleştirme sıralaması çözümlemesi bir yinelemeyi çözmemizi gerektirmişti.
- Yinelemeler entegral, türev, v.s. denklemlerinin çözümlerine benzer.
  - Bazı numaralar öğrenin.
- ***Ders 3***: Yinelemelerin "böl-ve-fethet" algoritmalarına uygulanması.



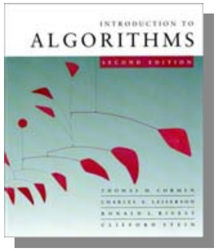
Yerine koyma metodu (yöntemi)

En genel yöntem:

1. Çözümün şeklini tahmin edin.

2. Tümevarım ile doğrulayın.

3. Sabitleri çözün.



Yerine koyma metodu (yöntemi)

En genel yöntem:

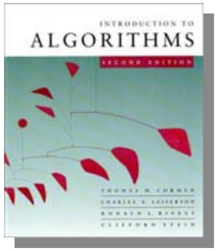
1. Çözümün şeklini tahmin edin.

2. Tümevarım ile doğrulayın.

3. Sabitleri çözün.

ÖRNEK: $T(n) = 4T(n/2) + n$

- [$T(1) = \Theta(1)$ olduğunu varsayın.]
- $O(n^3)$'ü tahmin edin. (O ve Ω ayrı ayrı kanıtlayın.)
- $k < n$ için $T(k) \le ck^3$ olduğunu varsayın.
- $T(n) \le cn^3$'ü tümevarımla kanıtlayın.

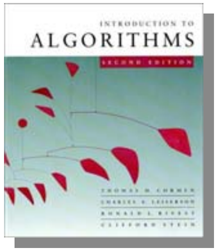


Yerine koyma örneği

$$\begin{aligned}
T(n) &= 4T(n/2) + n \\
&\leq 4c(n/2)^3 + n \\
&= (c/2)n^3 + n \\
&= cn^3 - ((c/2)n^3 - n) \leftarrow \textit{istenen} - \textit{kalan} \\
&\leq cn^3 \leftarrow \textit{istenen}
\end{aligned}$$

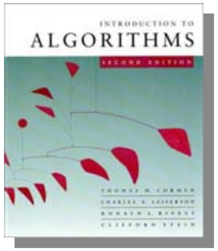
ne zaman ki $(c/2)n^3 - n \geq 0$, örneğin,
eğer $c \geq 2$ ve $n \geq 1$.

kalan



Örnek (devamı)

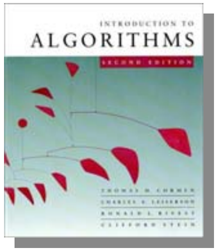
- Başlangıç koşullarını da ele almalı, yani, tümevarımı taban şıklarına (base cases) dayandırmalıyız.
- ***Taban:*** $T(n) = \Theta(1)$ tüm $n < n_0$ için, ki n_0 uygun bir sabittir.
- $1 \le n < n_0$ için, elimizde "$\Theta(1)$" $\le cn^3$, olur; yeterince büyük bir c değeri seçersek.



Örnek (devamı)

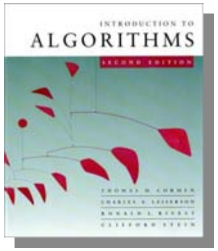
- Başlangıç koşullarını da ele almalı, yani, tümevarımı taban şıklarına (base cases) dayandırmalıyız.
- ***Taban:*** $T(n) = \Theta(1)$ tüm $n < n_0$ için, ki n_0 uygun bir sabittir.
- $1 \le n < n_0$, için, elimizde "$\Theta(1)$" $\le cn^3$, olur; yeterince büyük bir c değeri seçersek.

Bu, sıkı bir sınır değildir !



Daha sıkı bir üst sınır?

$T(n) = O(n^2)$ olduğunu kanıtlayacağız.

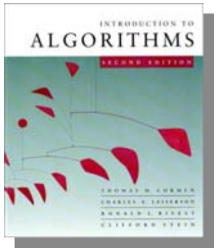


Daha sıkı bir üst sınır?

$T(n) = O(n^2)$ olduğunu kanıtlayacağız.

Varsayın ki $T(k) \leq ck^2$, $k < n$ için olsun :

$$\begin{aligned} T(n) &= 4T(n/2) + n \\ &\leq 4c(n/2)^2 + n \\ &= cn^2 + n \\ &= O(n^2) \end{aligned}$$



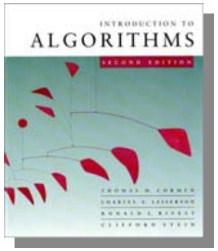
Daha sıkı bir üst sınır?

$T(n) = O(n^2)$ olduğunu kanıtlayacağız.

Varsayın ki $T(k) \le ck^2$; $k < n$: için

$$\begin{aligned} T(n) &= 4T(n/2) + n \\ &\le 4c(n/2)^2 + n \\ &= cn^2 + n \\ &= \cancel{O(n^2)} \end{aligned}$$

Yanlış! I.H.(tümevarım hipotezini) kanıtlamalıyız.



Daha sıkı bir üst sınır?

$T(n) = O(n^2)$ olduğunu kanıtlayacağız.

Varsayın ki $T(k) \le ck^2$; $k < n$: için

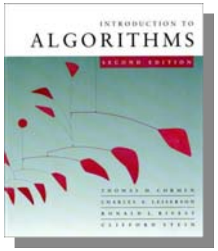
$$\begin{aligned} T(n) &= 4T(n/2) + n \\ &\le 4c(n/2)^2 + n \\ &= cn^2 + n \\ &= \cancel{O(n^2)} \end{aligned}$$

Yanlış! I.H.(tümevarım hipotezini) kanıtlamalıyız.

$$\begin{aligned} &= cn^2 - (-n) \\ &\le cn^2 \end{aligned}$$

[istenen –kalan]

seçeneksiz durum $c > 0$. Kaybettik!



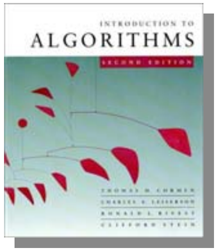
Daha sıkı bir üst sınır!

FİKİR: Varsayım hipotezini güçlendirin.

- Düşük-düzeyli bir terimi ***çıkartın***.

Varsayım hipotezi: $T(k) \leq c_1 k^2 - c_2 k$; $k < n$ için.

(Inductive hypothesis)



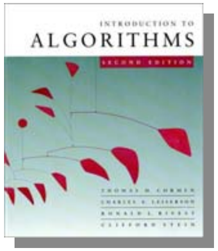
Daha sıkı bir üst sınır!

FİKİR: Varsayım hipotezini güçlendirin.

- Düşük-düzeyli bir terimi ***çıkartın***.

Varsayım hipotezi: $T(k) \le c_1 k^2 - c_2 k$; $k < n$ için.

$$\begin{aligned} T(n) &= 4T(n/2) + n \\ &= 4(c_1(n/2)^2 - c_2(n/2)) + n \\ &= c_1 n^2 - 2c_2 n + n \\ &= c_1 n^2 - c_2 n - (c_2 n - n) \\ &\le c_1 n^2 - c_2 n \text{ eğer } c_2 \ge 1. \end{aligned}$$



Daha sıkı bir üst sınır!

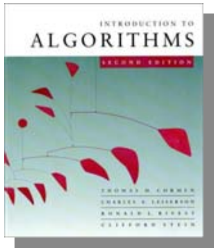
FİKİR: Varsayım hipotezini güçlendirin.

- Düşük-düzeyli bir terimi ***çıkartın***.

Varsayım hipotezi: $T(k) \le c_1 k^2 - c_2 k$; $k < n$ için.

$$\begin{aligned} T(n) &= 4T(n/2) + n \\ &= 4(c_1(n/2)^2 - c_2(n/2)) + n \\ &= c_1 n^2 - 2c_2 n + n \\ &= c_1 n^2 - c_2 n - (c_2 n - n) \\ &\le c_1 n^2 - c_2 n \text{ eğer } c_2 \ge 1. \end{aligned}$$

c_1'i başlangıç koşullarını karşılayacak kadar büyük seçin.



Özyineleme-ağacı metodu

- Özyineleme-ağacı, bir algoritmadaki özyineleme uygulamasının maliyetini (zamanı) modeller.
- Özyineleme-ağacı metodu, elipsleri (…) kullanan diğer yöntemler gibi, güvenilir olmayabilir.
- Öte yandan özyineleme-ağacı metodu "öngörü" olgusunu geliştirir.
- Özyineleme-ağacı metodu "yerine koyma metodu" için gerekli tahminlerinde yararlıdır .

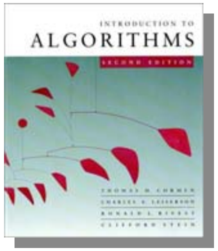
Özyineleme-ağacı örneği

$T(n) = T(n/4) + T(n/2) + n^2$: çözün

Özyineleme-ağacı örneği

$T(n) = T(n/4) + T(n/2) + n^2$: çözün

$$T(n)$$

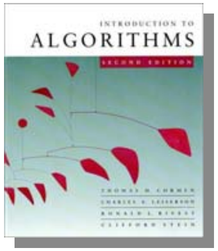


Özyineleme-ağacı örneği

$T(n) = T(n/4) + T(n/2) + n^2$: çözün

$$n^2$$

$$T(n/4) \qquad\qquad T(n/2)$$



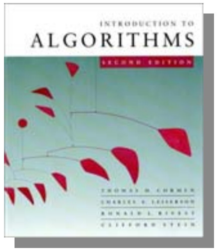
Özyineleme-ağacı örneği

$T(n) = T(n/4) + T(n/2) + n^2$: çözün

n^2

$(n/4)^2$ $(n/2)^2$

$T(n/16)$ $T(n/8)$ $T(n/8)$ $T(n/4)$



Özyineleme-ağacı örneği

$T(n) = T(n/4) + T(n/2) + n^2$:

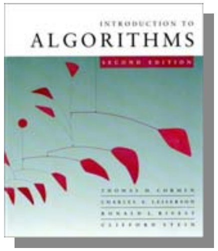
$$n^2$$

$$(n/4)^2 \qquad\qquad (n/2)^2$$

$$(n/16)^2 \quad (n/8)^2 \quad (n/8)^2 \quad (n/4)^2$$

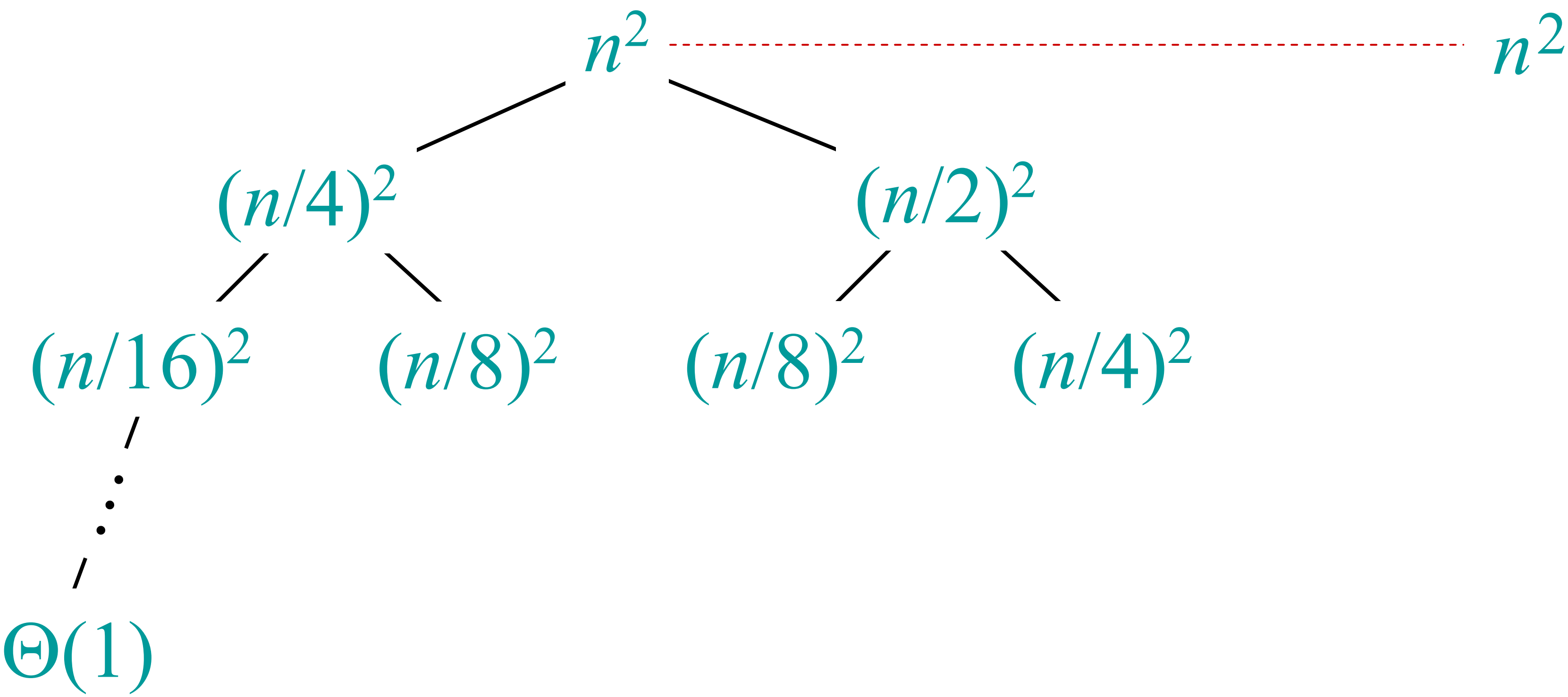
$$\vdots$$

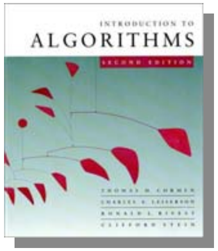
$$\Theta(1)$$



Özyineleme-ağacı örneği

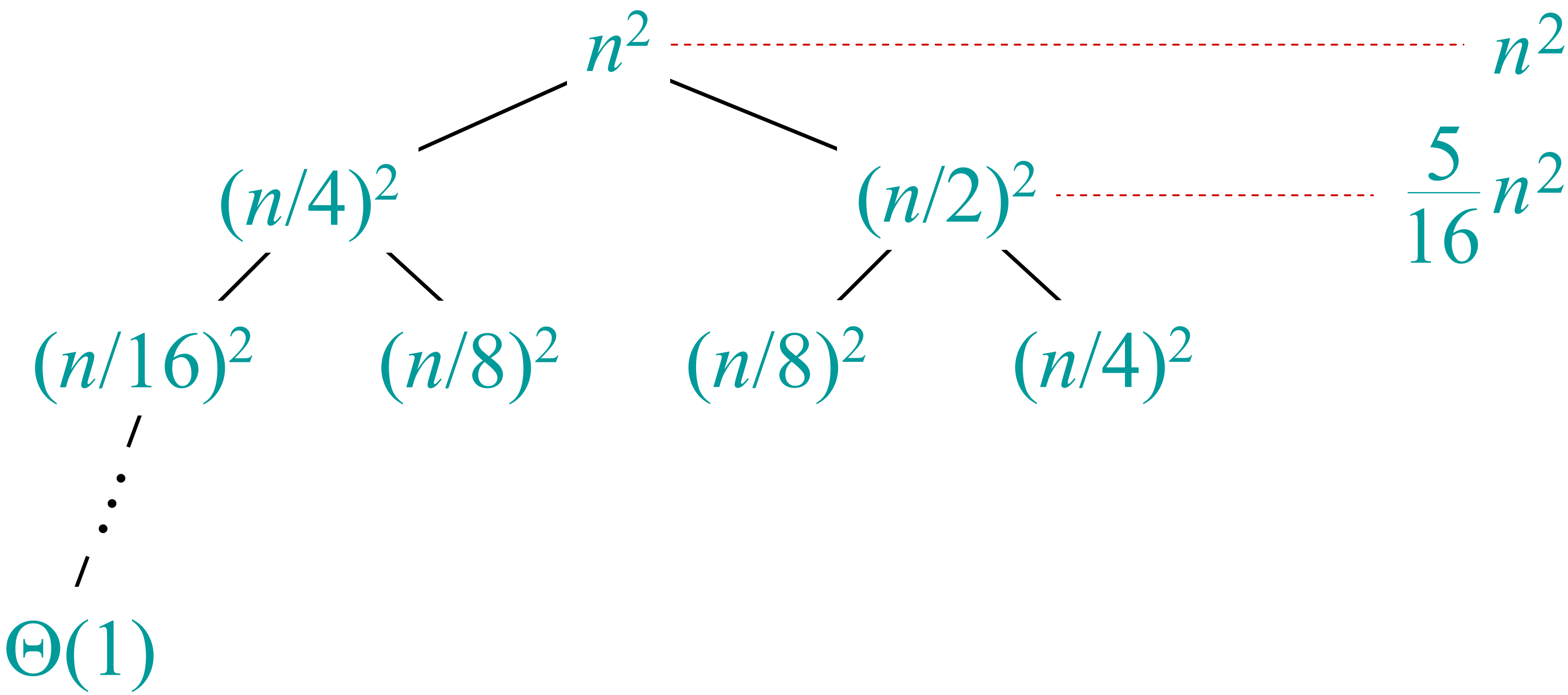
$T(n) = T(n/4) + T(n/2) + n^2$:

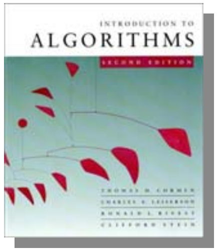




Özyineleme-ağacı örneği

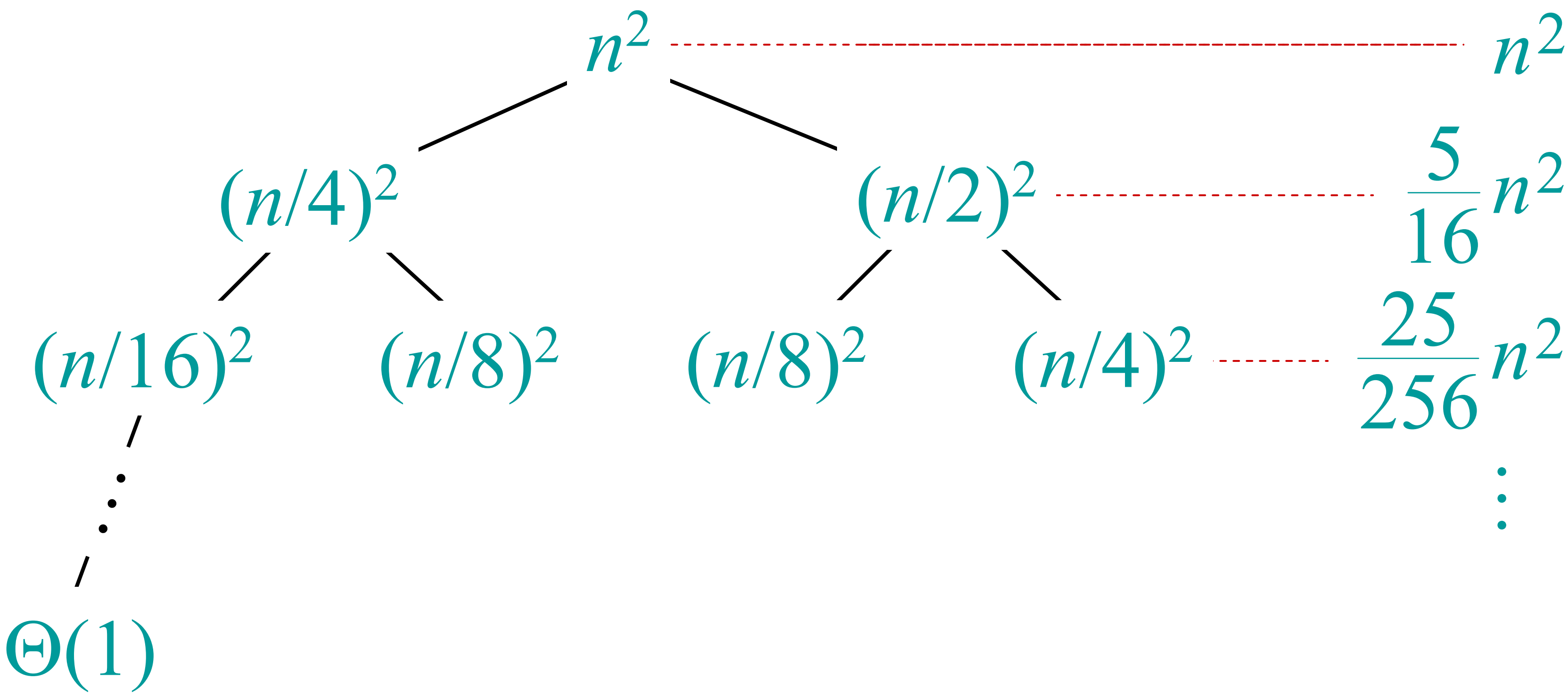
$T(n) = T(n/4) + T(n/2) + n^2$:





Özyineleme-ağacı örneği

$T(n) = T(n/4) + T(n/2) + n^2$:



Özyineleme-ağacı örneği

$T(n) = T(n/4) + T(n/2) + n^2$:

$$n^2 \quad \cdots\cdots \quad n^2$$

$$(n/4)^2 \qquad (n/2)^2 \quad \cdots\cdots \quad \frac{5}{16}n^2$$

$$(n/16)^2 \quad (n/8)^2 \quad (n/8)^2 \quad (n/4)^2 \quad \cdots\cdots \quad \frac{25}{256}n^2$$

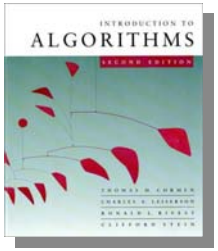
$$\vdots \qquad\qquad \vdots$$

$$\Theta(1)$$

$$\text{Total} = n^2\left(1 + \frac{5}{16} + \left(\frac{5}{16}\right)^2 + \left(\frac{5}{16}\right)^3 + \cdots\right)$$

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

Geometrik seri



Ana Metod (The Master Method)

Ana method aşağıda belirtilen yapıdaki yinelemelere uygulanır:

$$T(n) = a\,T(n/b) + f(n)\ ,$$

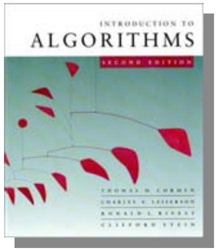
burada $a \geq 1$, $b > 1$, ve f asimptotik olarak pozitiftir.

Üç yaygın uygulama

$f(n)$'i $n^{\log_b a}$ ile karşılaştırın:

1. $f(n) = O(n^{\log_b a - \varepsilon})$ $\varepsilon > 0$ sabiti durumunda
   - $f(n)$ polinomsal olarak $n^{\log_b a}$ göre daha yavaş büyür (n^{ε} faktörü oranında).

   Çözüm: $T(n) = \Theta(n^{\log_b a})$.



Üç yaygın uygulama

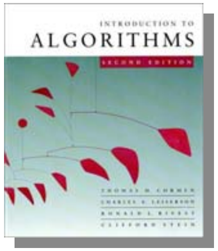
$f(n)$'i $n^{\log_b a}$ ile karşılaştırın:

1. $f(n) = O(n^{\log_b a - \varepsilon})$ $\varepsilon > 0$ sabiti durumunda;
   - $f(n)$ polinomsal olarak $n^{\log_b a}$ göre daha yavaş büyür(n^{ε} faktörü oranında).

   Çözüm: $T(n) = \Theta(n^{\log_b a})$.

2. $f(n) = \Theta(n^{\log_b a} \lg^k n)$ $k \geq 0$ sabiti durumunda;
   - $f(n)$ ve $n^{\log_b a}$ benzer oranlarda büyürler.

   Çözüm: $T(n) = \Theta(n^{\log_b a} \lg^{k+1} n)$.



Üç yaygın uygulama

$f(n)$'i $n^{\log_b a}$ ile karşılaştırın:

3. $f(n) = \Omega(n^{\log_b a + \varepsilon})$ $\varepsilon > 0$ sabiti durumunda;
   - $f(n)$ polinomsal olarak $n^{\log_b a}$ 'ye göre daha hızlı büyür (n^{ε} faktörü oranında),

   ve $f(n)$, ***düzenlilik koşulunu*** $af(n/b) \le cf(n)$ durumunda, $c < 1$ olmak kaydıyla karşılar.

   Çözüm: $T(n) = \Theta(f(n))$.

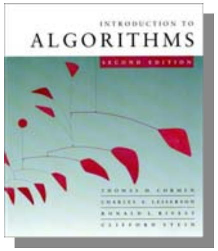
Örnekler

Örnek. $T(n) = 4T(n/2) + n$

$a = 4,\ b = 2 \Rightarrow n^{\log_b a} = n^2;\ f(n) = n.$

Durum 1: $f(n) = O(n^{2-\varepsilon})$ $\varepsilon = 1$ için.

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



Örnekler

Ör. $T(n) = 4T(n/2) + n$

$a = 4,\ b = 2 \Rightarrow n^{\log_b a} = n^2;\ f(n) = n.$

Durum 1: $f(n) = O(n^{2-\varepsilon})$ $\varepsilon = 1$ için.

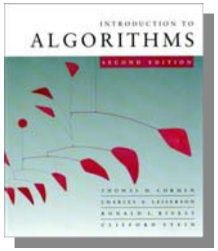
$\therefore\ T(n) = \Theta(n^2).$

Ör. $T(n) = 4T(n/2) + n^2$

$a = 4,\ b = 2 \Rightarrow n^{\log_b a} = n^2;\ f(n) = n^2.$

Durum 2: $f(n) = \Theta(n^2 \lg^0 n)$, yani, $k = 0$.

$\therefore\ T(n) = \Theta(n^2 \lg n).$



Örnekler

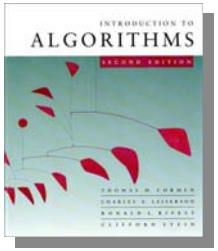
Ör. $T(n) = 4T(n/2) + n^3$

$a = 4$, $b = 2 \Rightarrow n^{\log_b a} = n^2$; $f(n) = n^3$.

DURUM 3: $f(n) = \Omega(n^{2+\varepsilon})$ $\varepsilon = 1$ için

ve $4(n/2)^3 \le cn^3$ (düz. koş.) $c = 1/2$ için.

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



Örnekler

Ör. $T(n) = 4T(n/2) + n^3$

$a = 4,\ b = 2 \Rightarrow n^{\log_b a} = n^2;\ f(n) = n^3.$

Durum 3: $f(n) = \Omega(n^{2+\varepsilon})$ $\varepsilon = 1$ için

ve $4(n/2)^3 \le cn^3$ (düz. koş.) $c = 1/2$ için

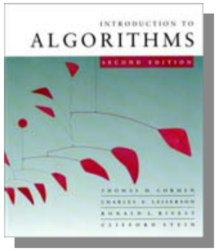
$\therefore\ T(n) = \Theta(n^3).$

Ör. $T(n) = 4T(n/2) + n^2/\lg n$

$a = 4,\ b = 2 \Rightarrow n^{\log_b a} = n^2;\ f(n) = n^2/\lg n.$

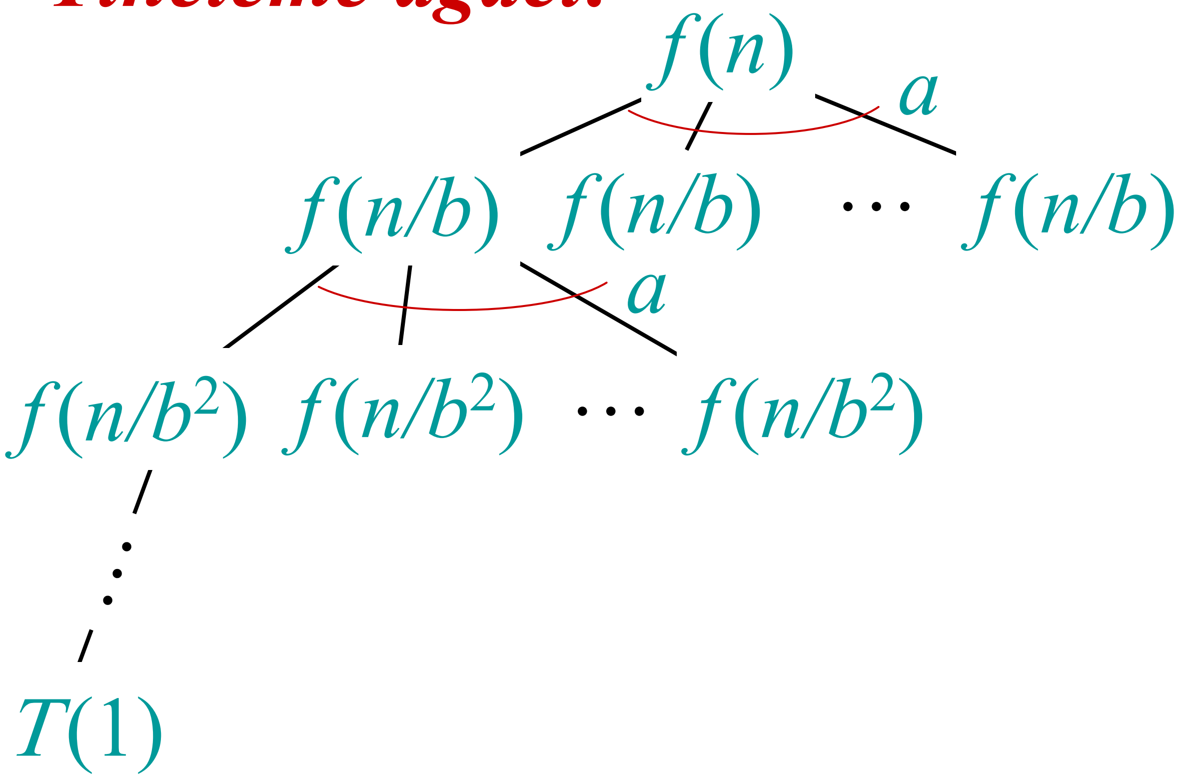
Ana metod geçerli değil. Özellikle,

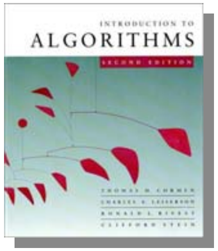
$\varepsilon > 0$ olan sabitler için $n^{\varepsilon} = \omega(\lg n)$ elde edilir.



Master teoremdeki düşünce

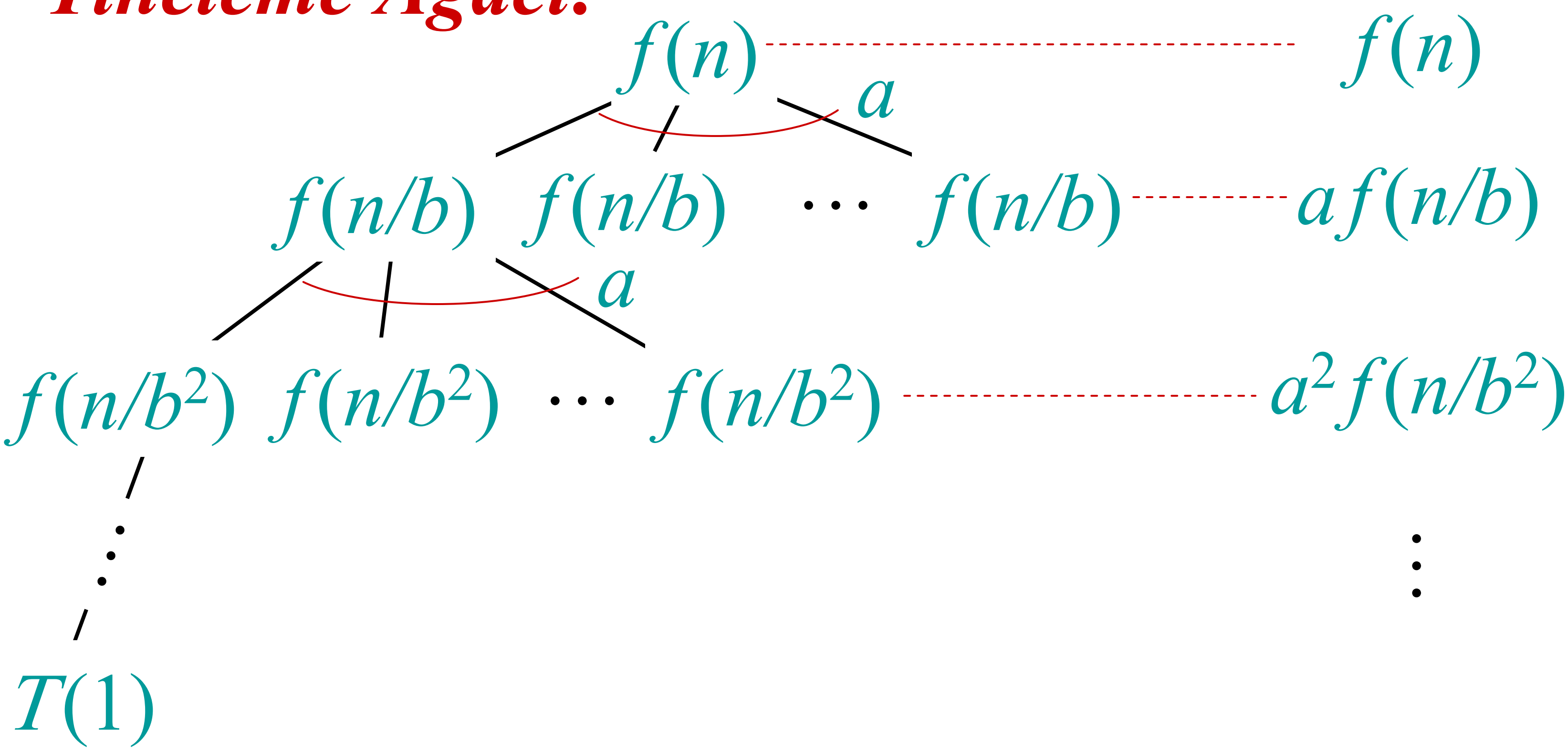
Yineleme ağacı:

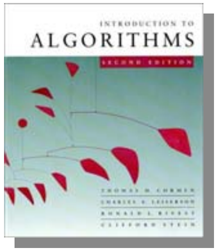




Master teoremdeki düşünce

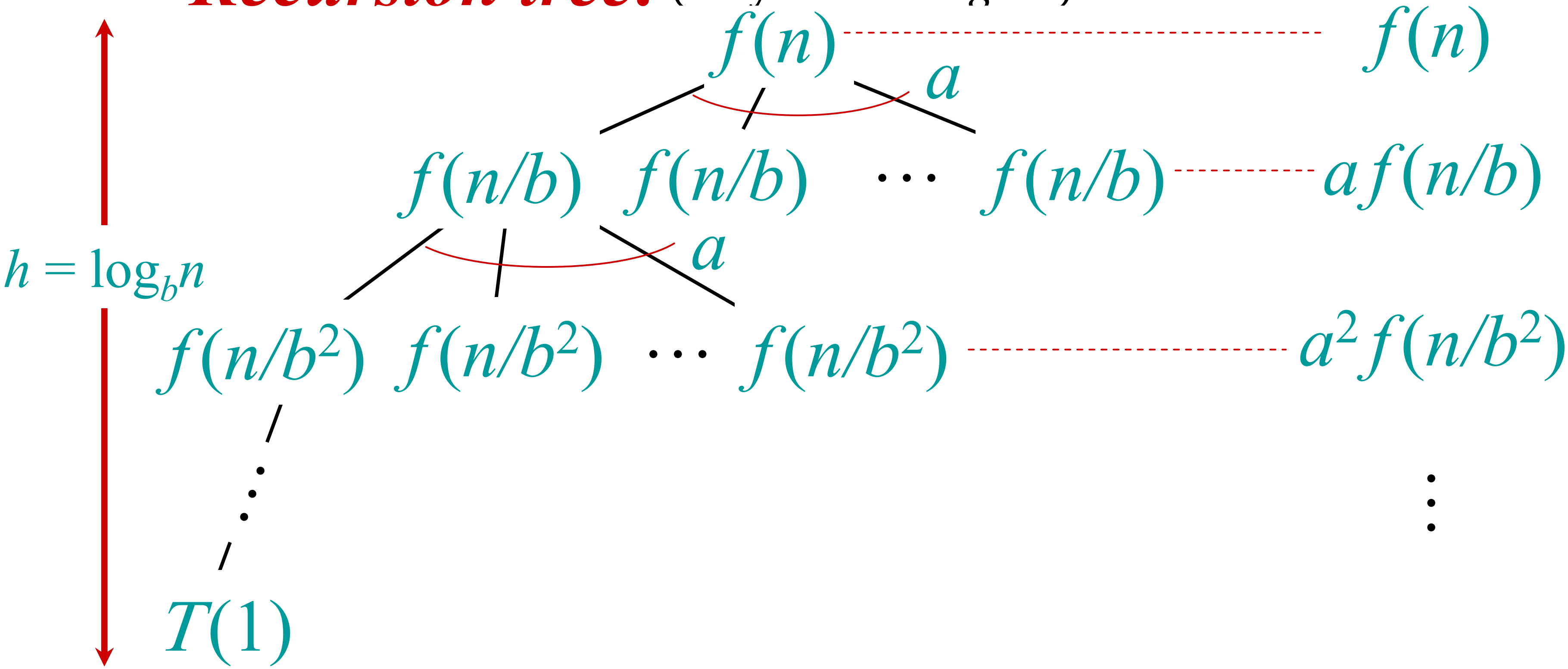
Yineleme Ağacı:

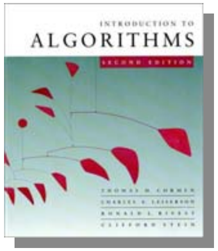




Master teoremdeki düşünce

Recursion tree: (Özyineleme Ağacı)





Master teoremdeki düşünce

Recursion tree: (Özyineleme Ağacı)

$f(n)$ .... $f(n)$

a

$f(n/b) \quad f(n/b) \quad \cdots \quad f(n/b)$ .... $a\,f(n/b)$

a

$h = \log_b n$

$f(n/b^2) \quad f(n/b^2) \quad \cdots \quad f(n/b^2)$ .... $a^2 f(n/b^2)$

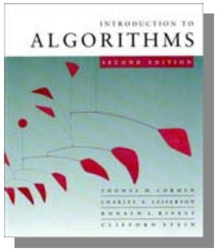
$\vdots$ $\qquad\qquad\qquad\qquad$ $\vdots$

$T(1)$ .... $n^{\log_b a}\, T(1)$

$\#\text{leaves} = a^h$

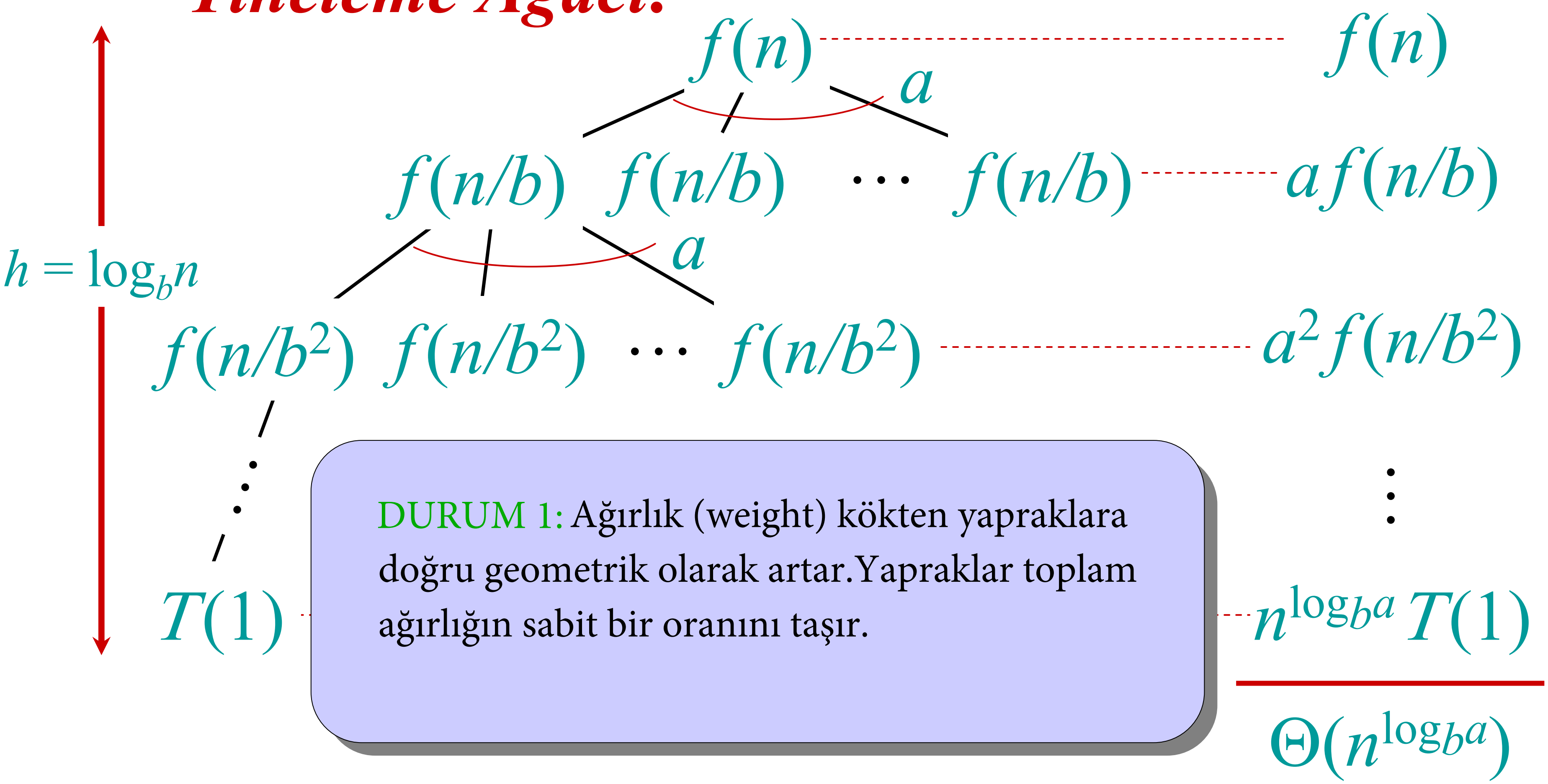
yaprak sayısı $= a^{\log_b n}$

$= n^{\log_b a}$



Master teoremdeki düşünce

Yineleme Ağacı:



DURUM 1: Ağırlık (weight) kökten yapraklara doğru geometrik olarak artar. Yapraklar toplam ağırlığın sabit bir oranını taşır.

Master teoremdeki düşünce

Yineleme Ağacı:

$f(n)$ .... $f(n)$

a

$f(n/b) \quad f(n/b) \quad \cdots \quad f(n/b)$ .... $a\,f(n/b)$

a

$h = \log_b n$

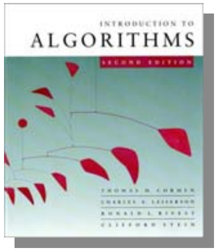
$f(n/b^2) \quad f(n/b^2) \quad \cdots \quad f(n/b^2)$ .... $a^2 f(n/b^2)$

$\vdots$

$T(1)$ .... $n^{\log_b a}\, T(1)$

DURUM 2: (k=0) Ağırlık her log_b n düzeyinde yaklaşık olarak aynıdır.

$\Theta(n^{\log_b a} \lg n)$



Master teoremdeki düşünce

Yineleme ağacı:

$h = \log_b n$

$f(n)$ .... $f(n)$

a

$f(n/b) \quad f(n/b) \quad \cdots \quad f(n/b)$ .... $a\,f(n/b)$

a

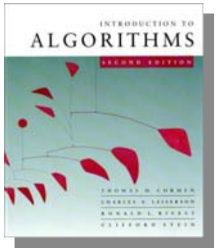
$f(n/b^2) \quad f(n/b^2) \quad \cdots \quad f(n/b^2)$ .... $a^2 f(n/b^2)$

$\vdots$

$T(1)$ .... $n^{\log_b a}\, T(1)$

$\Theta(f(n))$

DURUM 3: Ağırlık kökten yapraklara doğru geometrik olarak azalır. Kök toplam ağırlığın sabit bir oranını taşır.



Appendix/EK: Geometrik seriler

$$1 + x + x^2 + \cdots + x^n = \frac{1 - x^{n+1}}{1 - x} \quad ; x \neq 1 \text{ için}$$

$$1 + x + x^2 + \cdots = \frac{1}{1 - x} \quad ; |x| < 1 \text{ için}$$