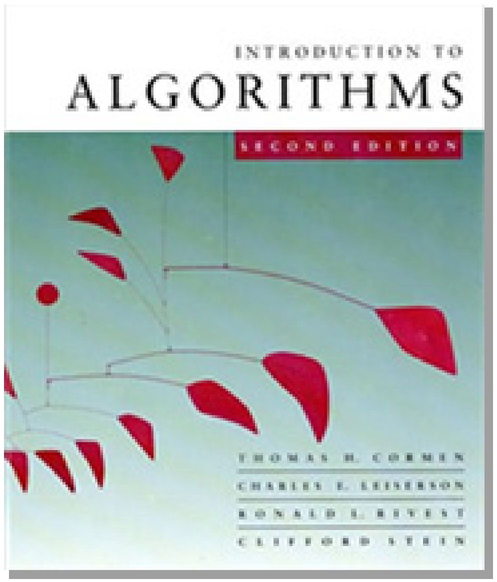
# *Algoritmalara Giriş*

## 6.046J/18.401J

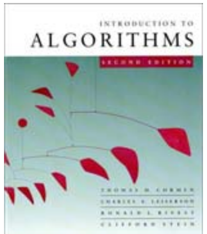



## DERS 6

### Sıra İstatistikleri

- Rastgele böl ve fethet
- Beklenen sürenin çözümlemesi
- En kötü durum doğrusal-süre sıra istatistikleri
- Çözümleme

**Prof. Erik Demaine**




# Sıra istatistikleri

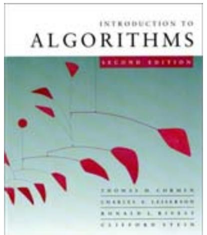
$n$ elemanın $i$'ninci küçük değerini seçin ( $i$ ***ranklē*** eleman).

- $i = 1$: ***minimum***; (en az)
- $i = n$: ***maximum***; (en çok)
- $i = \lfloor (n+1)/2 \rfloor$ veya $\lceil (n+1)/2 \rceil$: ***median***. (ortanca)

***Saf algoritma***: $i$'ninci elemanı sırala ve dizinle.

En kötü koşma süresi $= \Theta(n \lg n) + \Theta(1)$
$= \Theta(n \lg n)$,

birleştirme veya yığin sıralaması kullan (*çabuk sıralamayı değil*).




# Rastgele böl-ve-fethet algoritması

RAND-SELECT$(A, p, q, i)$ ▷ $A[p\,.\,q]$'nın ***i***'ninci en küçüğü

**if** $p = q$ **then return** $A[p]$

$r \leftarrow$ RAND-PARTITION$(A, p, q)$ (Rastgele bölüntü)

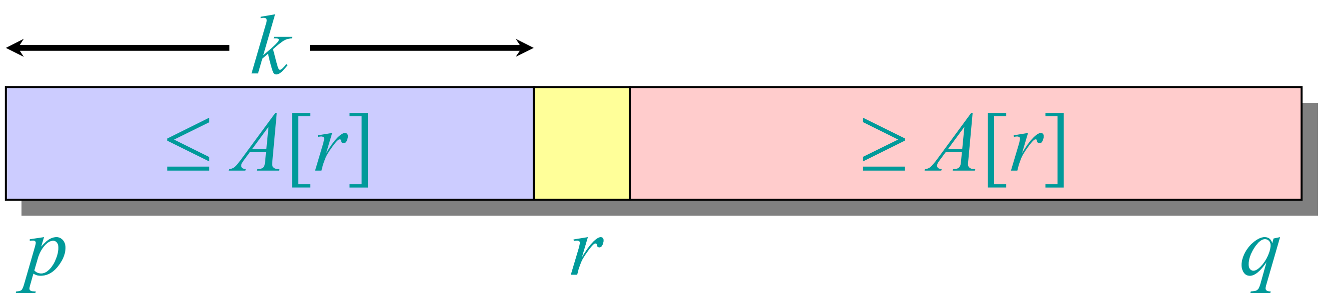
$k \leftarrow r - p + 1$ ▷ $k = \mathrm{rank}(A[r])$ (rütbeli)

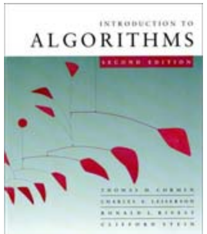
**if** $i = k$ **then return** $A[r]$

**if** $i < k$

**then return** RAND-SELECT$(A, p, r - 1, i)$

**else return** RAND-SELECT$(A, r + 1, q, i - k)$






# Örnek

$i = 7$' nci en küçük olarak seçin:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 6 | 10 | 13 | 5 | 8 | 3 | 2 | 11 |

$i = 7$

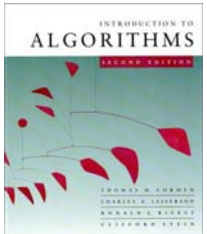
***pivot*** ***(esas eleman)***

Partition (Bölüntü):

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 2 | 5 | 3 | 6 | 8 | 13 | 10 | 11 |

$k = 4$

$7 - 4 = 3$ 'üncü küçüğü özyinelemeyle seçin.




# Çözümlemede sezgi (öngörü)

(Bugünkü çözümlemelerin hepsinde tüm elemanların farklı olduğu varsayılıyor.)

**Şanslı durum:**

$$T(n) = T(9n/10) + \Theta(n) \qquad n^{\log_{10/9} 1} = n^0 = 1$$

$$= \Theta(n) \qquad \text{DURUM 3}$$

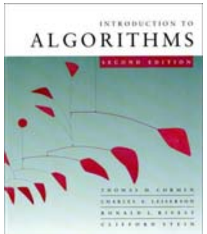
**Şanssız durum:**

$$T(n) = T(n-1) + \Theta(n) \qquad \text{aritmetik seri}$$

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

***Sıralamadan daha kötü!***




# Beklenen süre çözümlemesi

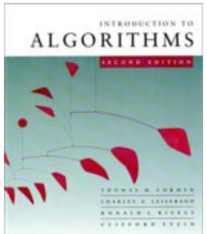
Çözümleme rastgele çabuk sıralamanın benzeri ama bazı farkları var.

$T(n)$ = Rastgele-seçim koşma süresinin rastgele değişkeni olsun ( $n$ boyutlu bir girişte), ve rastgele sayılar birbirinden bağımsız olsun.

$k = 0, 1, \ldots, n-1$ için ***göstergesel rastgele değişkeni*** tanımlayın.

$$X_k = \begin{cases} 1 & \text{eğer BÖLÜNTÜ } k : n-k-1 \text{ bölmeli ise,} \\ 0 & \text{diğer durumlarda.} \end{cases}$$

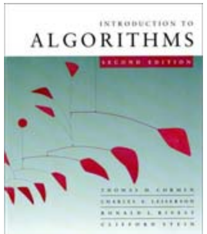



# Çözümleme (devam)

Bir üst sınır elde etmek için, $i'$ ninci elemanın her zaman bölüntünün büyük bölgesinde olduğunu varsayın:

$$T(n) = \begin{cases} T(\max\{0, n-1\}) + \Theta(n), & 0 : n-1 \text{ bölünmesi,} \\ T(\max\{1, n-2\}) + \Theta(n), & 1 : n-2 \text{ bölünmesi,} \\ \quad\vdots & \\ T(\max\{n-1, 0\}) + \Theta(n)\ , & n-1 : 0 \text{ bölünmesi,} \end{cases}$$

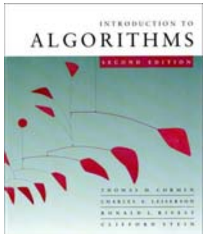
$$= \sum_{k=0}^{n-1} X_k \big(T(\max\{k, n-k-1\}) + \Theta(n)\big).$$




# Beklenenin hesaplanması

$$E[T(n)] = E\left[\sum_{k=0}^{n-1} X_k \big(T(\max\{k, n-k-1\}) + \Theta(n)\big)\right]$$

Her iki taraftaki beklenenleri bulun.



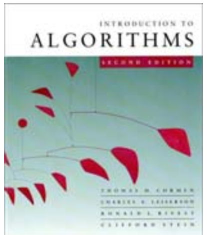

# Beklenenin hesaplanması

$$E[T(n)] = E\left[\sum_{k=0}^{n-1} X_k\big(T(\max\{k, n-k-1\}) + \Theta(n)\big)\right]$$

$$= \sum_{k=0}^{n-1} E\big[X_k\big(T(\max\{k, n-k-1\}) + \Theta(n)\big)\big]$$

Beklenenin doğrusallığı.




# Beklenenin hesaplanması

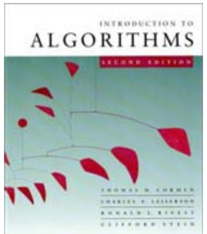
$$E[T(n)] = E\left[\sum_{k=0}^{n-1} X_k\big(T(\max\{k, n-k-1\}) + \Theta(n)\big)\right]$$

$$= \sum_{k=0}^{n-1} E\left[X_k\big(T(\max\{k, n-k-1\}) + \Theta(n)\big)\right]$$

$$= \sum_{k=0}^{n-1} E[X_k] \cdot E[T(\max\{k, n-k-1\}) + \Theta(n)]$$

$X_k$' nın diğer rastgele seçimlerden bağımsızlığı.

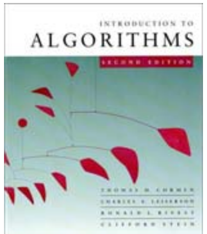



# Beklenenin hesaplanması

$$
\begin{aligned}
E[T(n)] &= E\left[\sum_{k=0}^{n-1} X_k \big(T(\max\{k, n-k-1\}) + \Theta(n)\big)\right] \\
&= \sum_{k=0}^{n-1} E\big[X_k \big(T(\max\{k, n-k-1\}) + \Theta(n)\big)\big] \\
&= \sum_{k=0}^{n-1} E[X_k] \cdot E[T(\max\{k, n-k-1\}) + \Theta(n)] \\
&= \frac{1}{n}\sum_{k=0}^{n-1} E[T(\max\{k, n-k-1\})] + \frac{1}{n}\sum_{k=0}^{n-1} \Theta(n)
\end{aligned}
$$

Beklenenin doğrusallığı; $E[X_k] = 1/n$ .




# Beklenenin hesaplanması

$$E[T(n)] = E\left[\sum_{k=0}^{n-1} X_k \big(T(\max\{k, n-k-1\}) + \Theta(n)\big)\right]$$

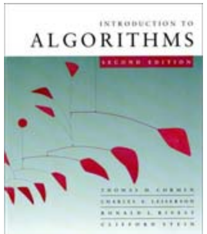
$$= \sum_{k=0}^{n-1} E\big[X_k \big(T(\max\{k, n-k-1\}) + \Theta(n)\big)\big]$$

$$= \sum_{k=0}^{n-1} E[X_k] \cdot E[T(\max\{k, n-k-1\}) + \Theta(n)]$$

$$= \frac{1}{n}\sum_{k=0}^{n-1} E[T(\max\{k, n-k-1\})] + \frac{1}{n}\sum_{k=0}^{n-1}\Theta(n)$$

$$\le \frac{2}{n}\sum_{k=\lfloor n/2 \rfloor}^{n-1} E[T(k)] + \Theta(n)$$

Üstteki terimler iki kez görünüyor.




# Karmaşık yineleme

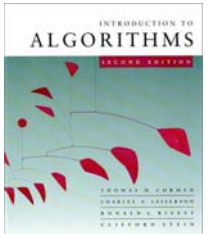
(Ama çabuk sıralamanınki kadar karmaşık değil.)

$$E[T(n)] = \frac{2}{n} \sum_{k=\lfloor n/2 \rfloor}^{n-1} E[T(k)] + \Theta(n)$$

**Kanıtla:** $E[T(n)] \le cn$ sabiti için $c > 0$ .

- $c$ sabiti öyle büyük seçilebilir ki, $E[T(n)] \le cn$ tüm taban durumlarında geçerli olur.

**Veri:** $$\sum_{k=\lfloor n/2 \rfloor}^{n-1} k \le \frac{3}{8} n^2$$ (alıştırma).

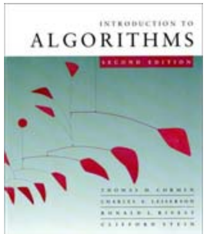



# Yerine koyma metodu

$$E[T(n)] \le \frac{2}{n} \sum_{k=\lfloor n/2 \rfloor}^{n-1} ck + \Theta(n)$$

Tümevarım hipotezini yerleştirin.



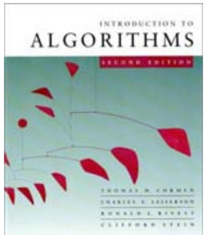

# Yerine koyma metodu

$$E[T(n)] \leq \frac{2}{n} \sum_{k=\lfloor n/2 \rfloor}^{n-1} ck + \Theta(n)$$

$$\leq \frac{2c}{n}\left(\frac{3}{8}n^2\right) + \Theta(n)$$

Veriyi kullanın.




# Yerine koyma metodu

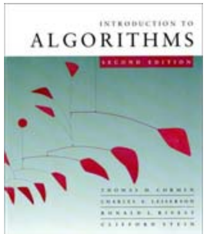
$$E[T(n)] \leq \frac{2}{n} \sum_{k=\lfloor n/2 \rfloor}^{n-1} ck + \Theta(n)$$

$$\leq \frac{2c}{n}\left(\frac{3}{8}n^2\right) + \Theta(n)$$

$$= cn - \left(\frac{cn}{4} - \Theta(n)\right)$$

***istenen – kalan*** şeklinde gösterin.

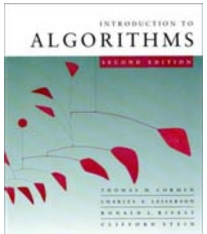



# Yerine koyma metodu

$$
\begin{aligned}
E[T(n)] &\leq \frac{2}{n} \sum_{k=\lfloor n/2 \rfloor}^{n-1} ck + \Theta(n) \\
&\leq \frac{2c}{n}\left(\frac{3}{8}n^2\right) + \Theta(n) \\
&= cn - \left(\frac{cn}{4} - \Theta(n)\right) \\
&\leq cn\,,
\end{aligned}
$$

$c$ yeterince büyük seçilirse
$cn/4$, $\Theta(n)$' nin üstünde olur.



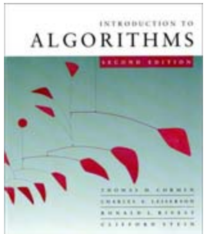

# Rastgele sıra istatistik seçiminin özeti

- Hızlı çalışır: doğrusal beklenen süre.
- Pratikte mükemmel bir algoritma.
- Ama, en kötü durumu ***çok*** kötü: $\Theta(n^2)$.

***Q.*** En kötü durumda doğrusal zamanda çalışan bir algoritma var mıdır?

***A.*** Evet, Blum, Floyd, Pratt, Rivest ve Tarjan [1973] sayesinde vardır.

**FİKİR:** İyi bir pivotu yinelemeyle üretmek.




# En kötü durum doğrusal-zaman sıra istatistikleri

SEÇ $(i, n)$

1. $n$ elemanı 5' li gruplara bölün. Her 5' li grubun ortancasını ezbere bulun.
2. $\lfloor n/5 \rfloor$ gruplarının ortancası olacak $x$' i yinelemeli SEÇME ile pivot olarak belirleyin.
3. Pivot $x$ etrafında bölüntü yapın. $k = \text{rank}(x)$.

   **if** $i = k$ **then return** $x$ **( eğer / öyleyse çıkar)**

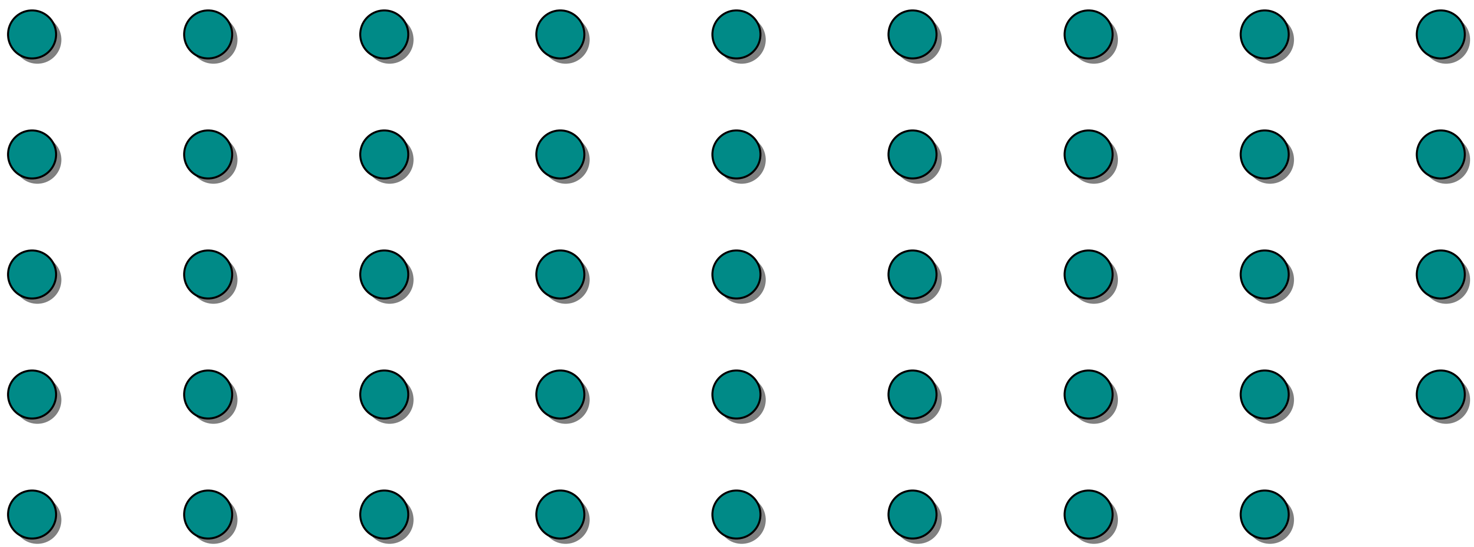
   **elseif** $i < k$ **( diğer durumlarda)**

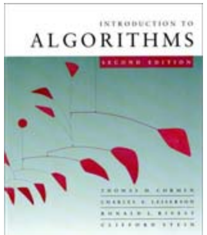
   **then** $i$' ninci en küçük elemanı alt bölgede yinelemeyle SEÇİN.

   **else** $(i–k)$' nıncı en küçük elemanı üst bölgede yinelemeyle SEÇİN.

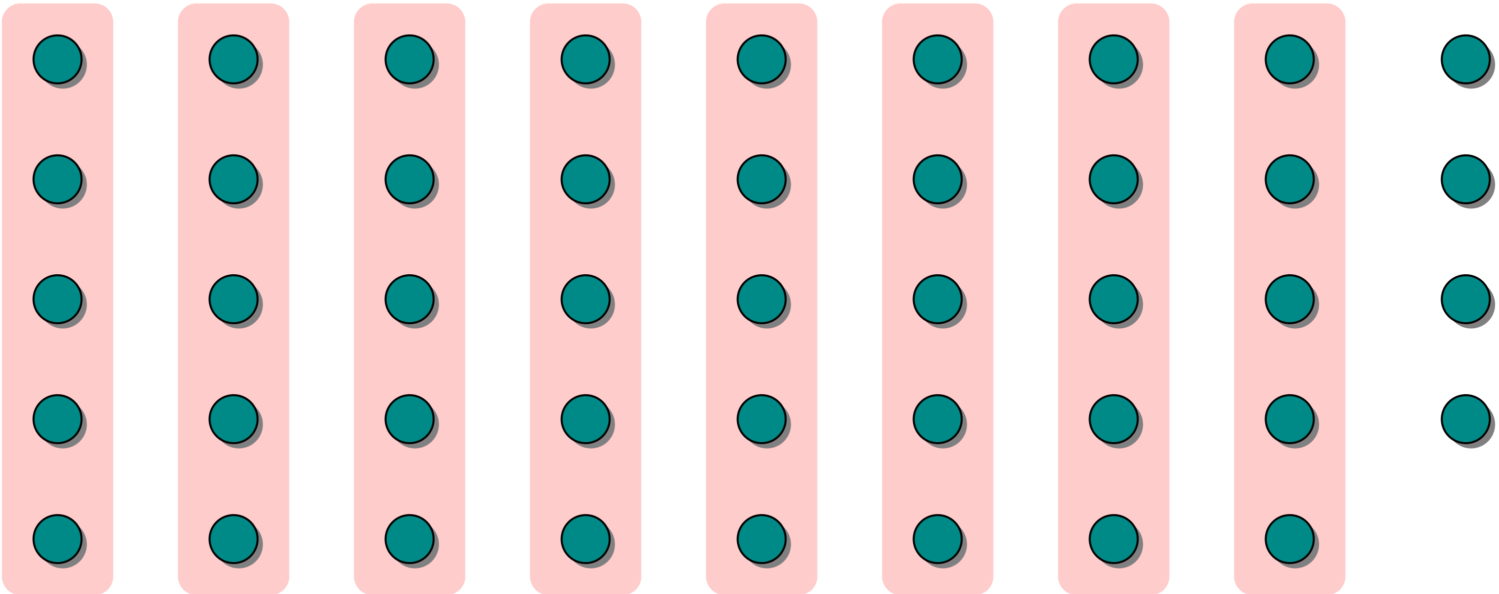
RASTGELE-SEÇİMİN aynısı

# Pivot seçimi





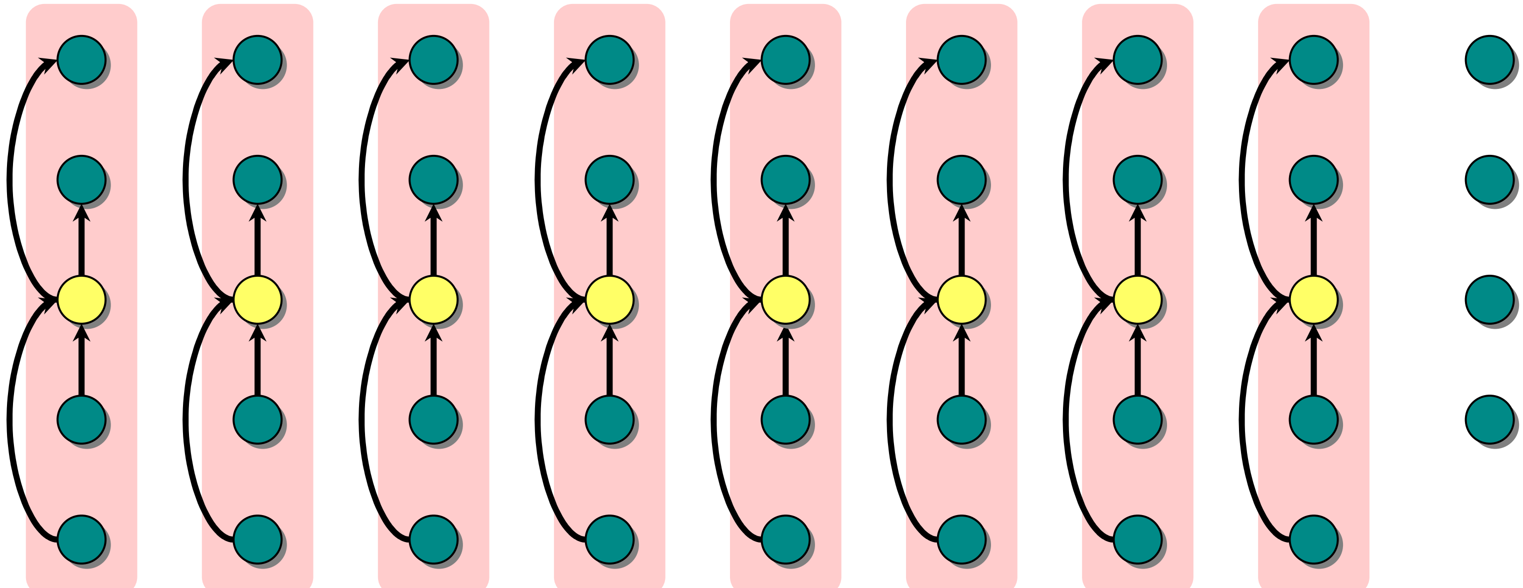

# Pivot seçimi



1. $n$ elemanı 5' li gruplara bölün.

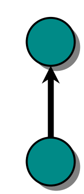
  

# Pivot seçimi

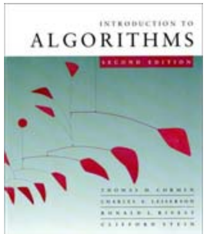


1. *n* elemanlarını 5' li gruplara bölün. 5-elemanlı grupların ortancasını ezbere bulun.

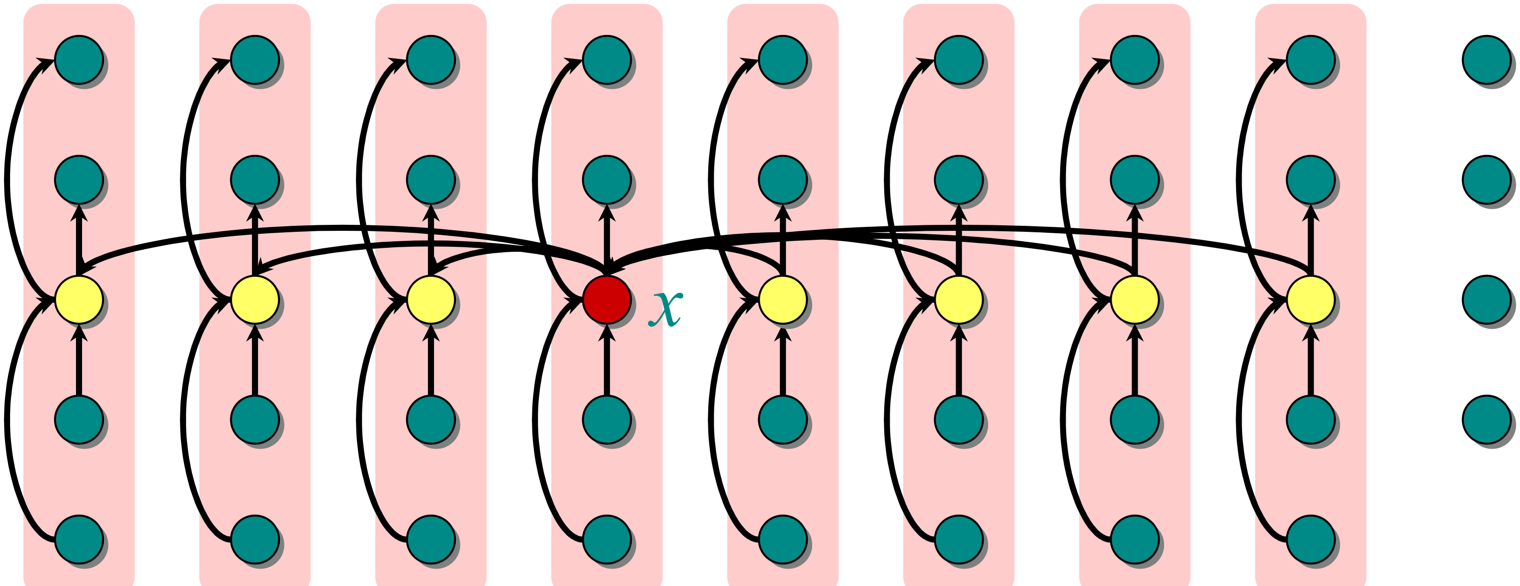
*daha az*



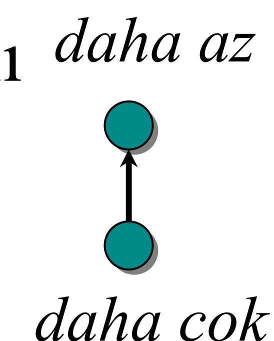
*daha fazla*



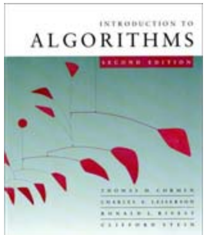

# Pivot seçimi



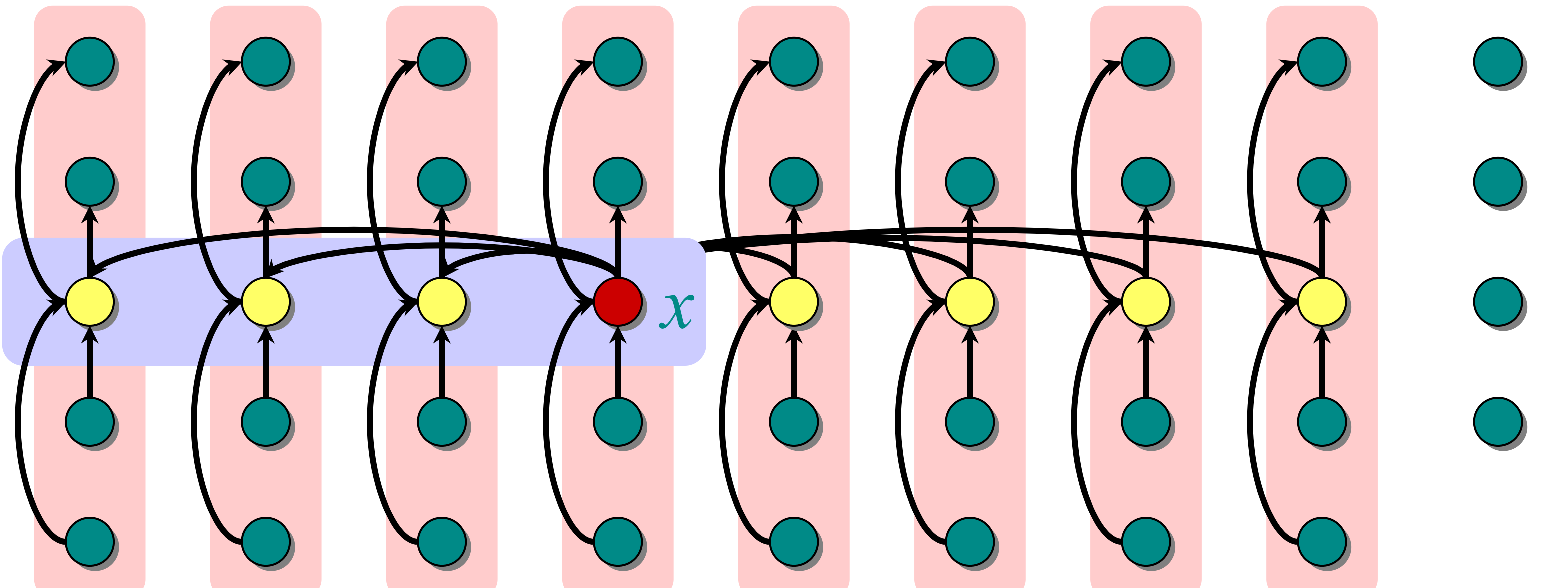

1. $n$ elemanlarını 5' li gruplara bölün. 5 elemanlı grupların ortancalarını ezbere bulun.
2. $\lfloor n/5 \rfloor$ gruplarının ortancası olacak $\boldsymbol{x}$' i, yinelemeli SEÇME ile pivot olarak belirleyin.



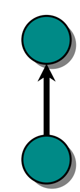

  

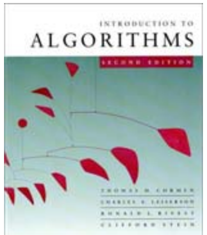


# Çözümleme

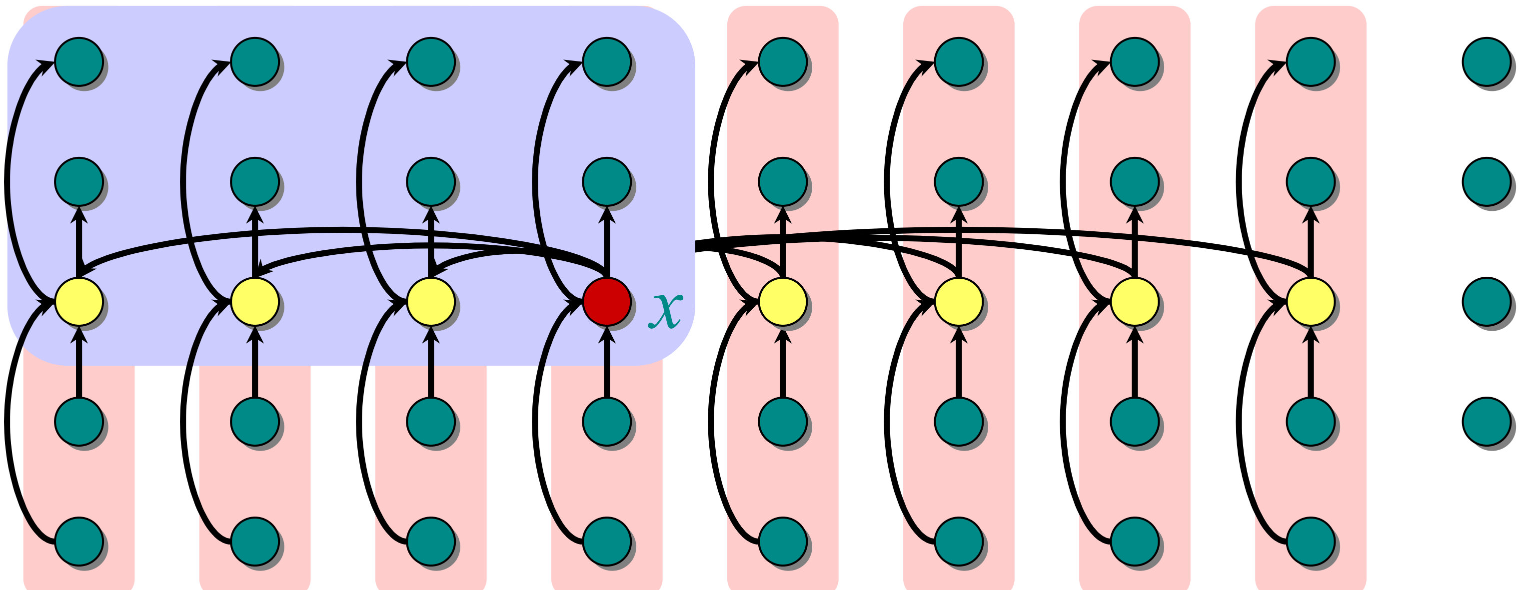



Grup ortancalarının en az yarısı $\leq x$, bu da en az $\lfloor \lfloor n/5 \rfloor /2 \rfloor = \lfloor n/10 \rfloor$ grup ortancası eder.





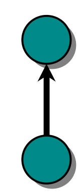

# Çözümleme (Tüm elemanları farklı varsay.)



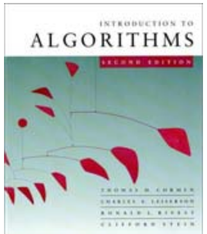

Grup ortancalarının en az yarısı $\leq x$, bu da en az $\lfloor \lfloor n/5 \rfloor /2 \rfloor = \lfloor n/10 \rfloor$ grup ortancası eder.

- Bu nedenle, en az $3\lfloor n/10 \rfloor$ eleman $\leq x$.

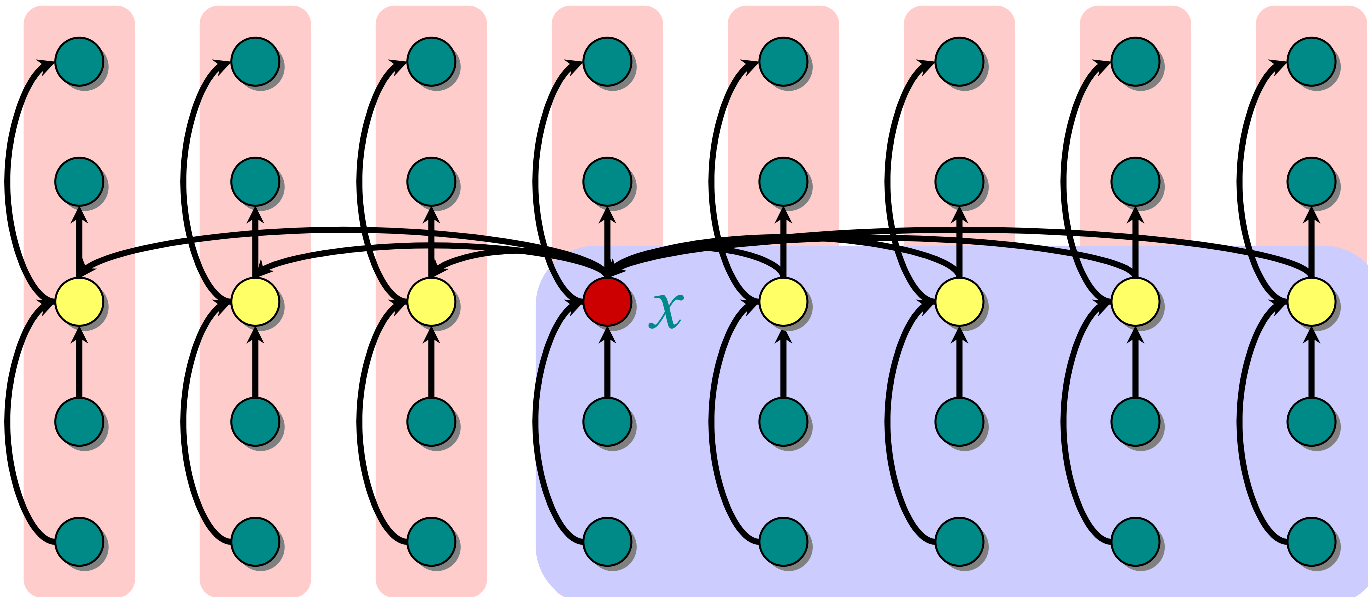
*daha az*



*daha çok*

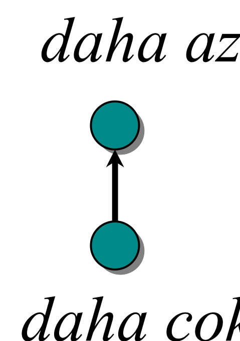


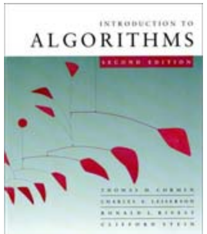
# Çözümleme (Tüm elemanları farklı varsay.)




Grup ortancalarının en az yarısı $\leq x$, bu da en az $\lfloor \lfloor n/5 \rfloor /2 \rfloor = \lfloor n/10 \rfloor$ grup ortancası eder.

- Bu nedenle, en az $3\lfloor n/10 \rfloor$ eleman $\leq x$.
- Benzer şekilde, en az $3\lfloor n/10 \rfloor$ eleman $\geq x$.

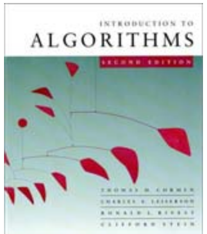





# Önemsiz basitleştirme

- $n \geq 50$ için, $3\lfloor n/10 \rfloor \geq n/4$ olur.
- Bu nedenle, $n \geq 50$ için Adım 4'teki SEÇİM özyinelemeli olarak $\leq 3n/4$ eleman kapsamında yapılır.
- Böylece, koşma süresinin yinelemesinde Adım 4'ün en kötü durumda $T(3n/4)$ zamanı alacağı farz edilebilir.
- $n < 50$ için en kötü sürenin $T(n) = \Theta(1)$ olduğunu biliyoruz.




# Yinelemeyi geliştirmek

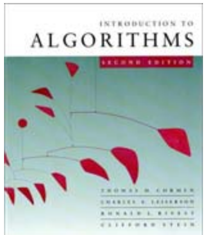
$T(n)$ SELECT $(i, n)$ (SEÇİN)

$\Theta(n)$ { 1. $n$ elemanı 5' li gruplara ayırın. 5-elemanlı grupların ortancasını ezberden bulun.

$(n/5)$ { 2. $\lfloor n/5 \rfloor$ gruplarının ortancası olacak $x$' i, yinelemeli SEÇME ile pivot olarak belirleyin.

$\Theta(n)$ 3. Pivot $x$ etrafında bölüntü yapın. $k = \text{rank}(x)$ olsun.

$T(3n/4)$ { 4. **if** $i = k$ **then return** $x$
**elseif** $i < k$
 **then** $i$' ninci en küçük elemanı alt bölgede yinelemeli olarak SEÇİN.
 **else** $(i–k)$'nıncı en küçük elemanı üst bölgede yinelemeli olarak SEÇİN.




# Yinelemeyi çözmek

$$T(n) = T\left(\frac{1}{5}n\right) + T\left(\frac{3}{4}n\right) + \Theta(n)$$

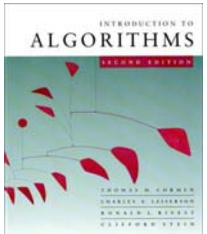
**Yerine koyma:**

$T(n) \le cn$

$$\begin{aligned} T(n) &\le \frac{1}{5}cn + \frac{3}{4}cn + \Theta(n) \\ &= \frac{19}{20}cn + \Theta(n) \\ &= cn - \left(\frac{1}{20}cn - \Theta(n)\right) \\ &\le cn \quad , \end{aligned}$$

$c$, hem $\Theta(n)$' i hem de başlangıç koşullarını gözeterek yeterince büyük seçilirse...




# Sonuçlar

- Yinelemenin her düzeyindeki iş sabit bir kesir (19/20) oranında küçüldüğünden, düzeylerdeki iş bir geometrik seri gibidir ve kökteki doğrusal iş ön plana çıkar.
- Pratikte bu algoritma yavaş çalışır, çünkü $n$' nin önündeki sabit büyüktür.
- Rastgele algoritma çok daha pratiktir.

**Alıştırma:** *Neden 3' lü gruplara bölmüyoruz?*