Ch.05 선택(Selection)

학습목표

- ✓ 기본 정렬 알고리즘을 이해한다.
- ✓ 정렬을 귀납적 관점에서 볼 수 있도록 한다.
- ✓ 2~3장에서 배운 기법을 사용해 각 정렬의 수행 시간을 분석할 수 있도록 한다.
- ✓ 비교 정렬의 한계를 이해하고, 선형 시간 정렬이 가능한 조건과 선형 시간 정렬 알고리즘을 이해한다.

알고리즘 공부의 묘미

그림 5-1 **알고리즘 공부의 묘미**

선택 알고리즘 (*i*번째 작은 수 찾기)

- **배열 A[*p* ... *r*]에서 *i*번째 작은 원소를 찾는다**
- **두가지 알고리즘을 배운다**
  - 평균적으로 선형시간이 소요되는 알고리즘
  - 최악의 경우에도 선형시간이 소요되는 알고리즘

평균 선형시간 선택 알고리즘

알고리즘 5-1 | **평균 선형 시간 선택**

select(A, p, r, i): ▷ 배열 A[p ... r]에서 i번째 작은 원소를 찾는다

    if ($p = r$) **then return** A[p] ▷ 원소가 하나뿐인 경우. i는 반드시 1

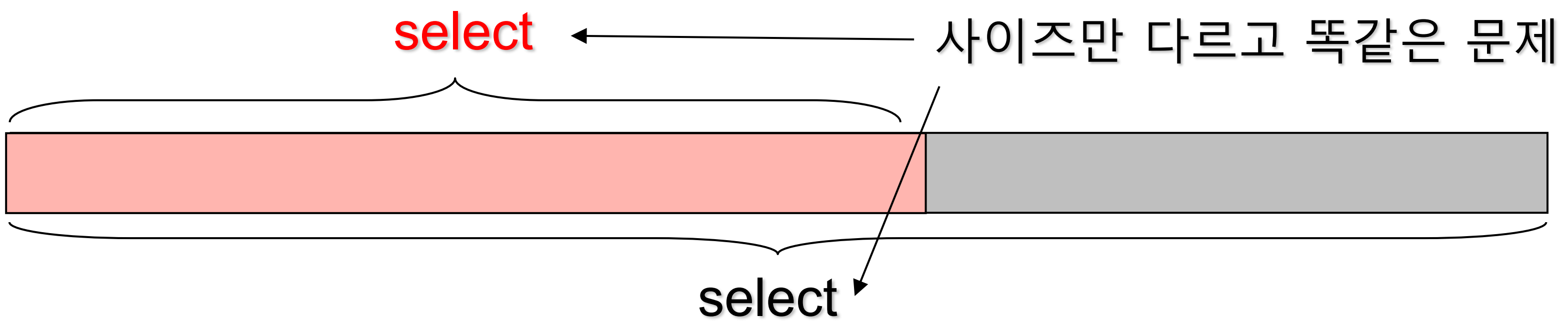
    $q \leftarrow$ partition(A, p, r) ▷ [알고리즘 4-12]의 partition()과 동일

    $k \leftarrow q$-p+1 ▷ k : 기준 원소가 전체에서 k 번째 작은 원소임을 의미

    if ($i < k$) **return** select(A, p, q-1, i) ▷ 왼쪽 그룹으로 범위를 좁힘

    else if ($i = k$) **then return** A[q] ▷ 기준 원소가 바로 찾는 원소임

    else return select(A, p, q-1, i) ▷ 오른쪽 그룹으로 범위를 좁힘



✓평균 수행 시간: $\Theta(n)$
✓최악의 경우 수행 시간: $\Theta(n^2)$

선택 알고리즘 작동 예 1

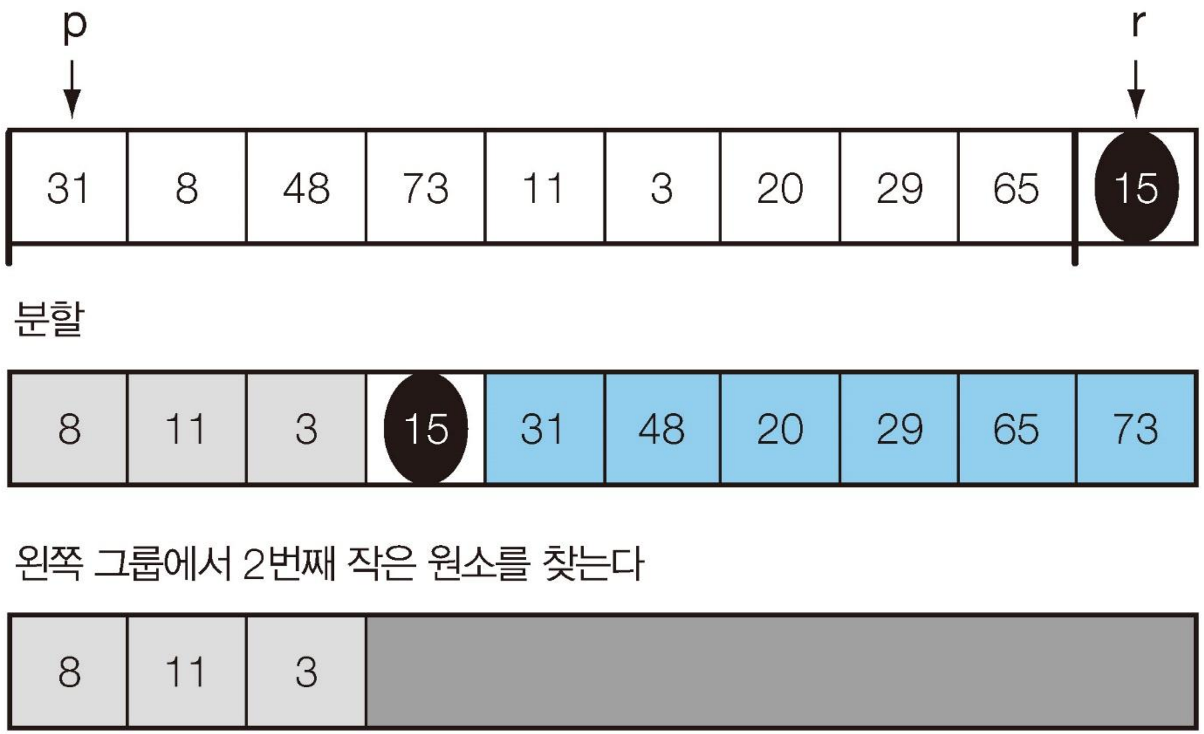


그림 5-2 **선택 알고리즘의 작동 예 1 : 2번째 작은 원소 찾기**

선택 알고리즘 작동 예 2

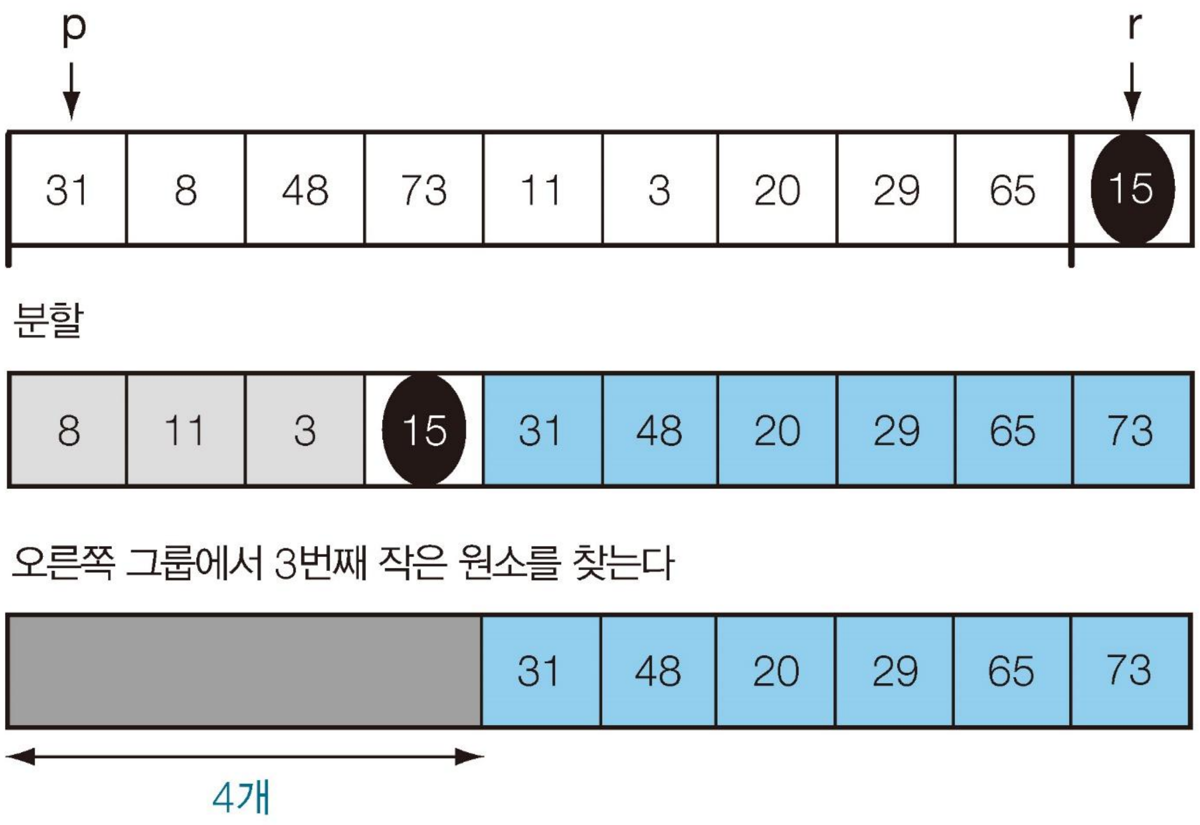


그림 5-3 선택 알고리즘의 작동 예 2 : 7번째 작은 원소 찾기

선택 알고리즘 수행 시간

■ 평균 수행 시간

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

분할된 양쪽 중 큰 쪽을 처리하는 비용
평균보다 조금 크지만 이래도 $O(n)$이 증명되면 okay

재귀호출을 제외한 오버헤드
(분할이 대부분)

이것은 $T(n) \le cn$임을 추정 후 증명법으로 증명할 수 있다(다음 슬라이드).
그러므로 $T(n) = O(n)$
$T(n) = \Omega(n)$임은 명백하므로 $T(n) = \Theta(n)$

선택 알고리즘 수행 시간

$$T(n) \le \max[T(k-1),\ T(n-k)] + \Theta(n)$$

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

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

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

$\lfloor n/2 \rfloor \le k < n$인 모든 k에 대해 $T(k) \le ck$라 가정하면(귀납적 가정) 다음과 같다. ← $\lfloor n/2 \rfloor$ 대신 n_0(경계치)을 사용해도 됨

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

$$= \frac{2}{n}\left(\sum_{k=1}^{n-1} ck - \sum_{k=1}^{\lfloor n/2 \rfloor - 1} ck\right) + \Theta(n)$$

$$= \frac{2}{n}\left(\frac{c(n-1)n}{2} - \frac{c(\lfloor n/2 \rfloor - 1)\lfloor n/2 \rfloor}{2}\right) + \Theta(n)$$

$$\le \frac{2c}{n}\left(\frac{(n-1)n}{2} - \frac{(n/2-2)(n/2-1)}{2}\right) + \Theta(n)$$

$$= c(n-1) - \frac{c}{n}\left(\frac{n^2}{4} - \frac{3n}{2} + 2\right) + \Theta(n)$$

$$= cn + \left(-\frac{cn}{4} + \frac{c}{2} - \frac{2c}{n} + \Theta(n)\right)$$

← 상수 c를 충분히 크게 잡으면 $-\frac{cn}{4}$이 $\Theta(n)$을 압도해 $-\frac{cn}{4} + \frac{c}{2} - \frac{2c}{n} + \Theta(n)$이 음수가 되도록 할 수 있어 성립함

$$\le cn$$

선택 알고리즘 수행 시간

■ 최악의 경우 수행 시간

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

재귀호출을 제외한 오버헤드
(분할이 대부분)

분할이 0 : n-1로 되고 큰 쪽을 처리하는 비용

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

최악의 경우 선형시간 선택 알고리즘

- **앞에서 배운 선택 알고리즘에서**
  - 수행 시간은 분할의 균형에 영향을 받는다

- **이번 알고리즘은**
  - 최악의 경우 분할의 균형이 어느 정도 보장되도록 함으로써 수행 시간이 $\Theta(n)$이 되도록 한다
  - 분할의 균형을 유지하기 위한 오버헤드가 지나치게 크면 안된다

최악의 경우 선형시간 선택 알고리즘

알고리즘 5-2 **최악의 경우 선형 시간 선택**

linearSelect(A, p, r, i): ▷ $A[p...r]$에서 i번째 작은 원소를 찾는다.

1 원소의 총수가 5개 이하이면 i번째 원소를 찾고 알고리즘을 끝낸다.

2 전체 원소를 5개씩의 원소를 가진 $\lceil \frac{n}{5} \rceil$개의 그룹으로 나눈다.
(원소의 총수가 5의 배수가 아니면 이 중 한 그룹은 5개 미만이 된다.)

3 각 그룹에서 중앙값(원소가 5개이면 3번째 원소)을 찾는다.
이렇게 찾은 중앙값들을 $m_1, m_2, \cdots, m_{\lceil n/5 \rceil}$이라 하자.

4 $m_1, m_2, \cdots, m_{\lceil n/5 \rceil}$들의 중앙값 M을 재귀적으로 구한다.
원소의 총수가 홀수이면 중앙값이 하나이므로 문제가 없고,
원소의 총수가 짝수이면 두 중앙값 중 임의로 선택한다. ▷ linearSelect() 호출

5 M을 기준 원소로 삼아 전체 원소를 분할한다(M보다 작거나 같은 것은 M의 왼쪽에,
M보다 큰 것은 M의 오른쪽에 배치). ▷ partition() 호출

6 분할된 두 그룹 중 적합한 쪽을 선택해 단계 1~6을 재귀적으로 반복한다.
▷ linearSelect() 호출

기준 원소를 중심으로 한 대소 관계

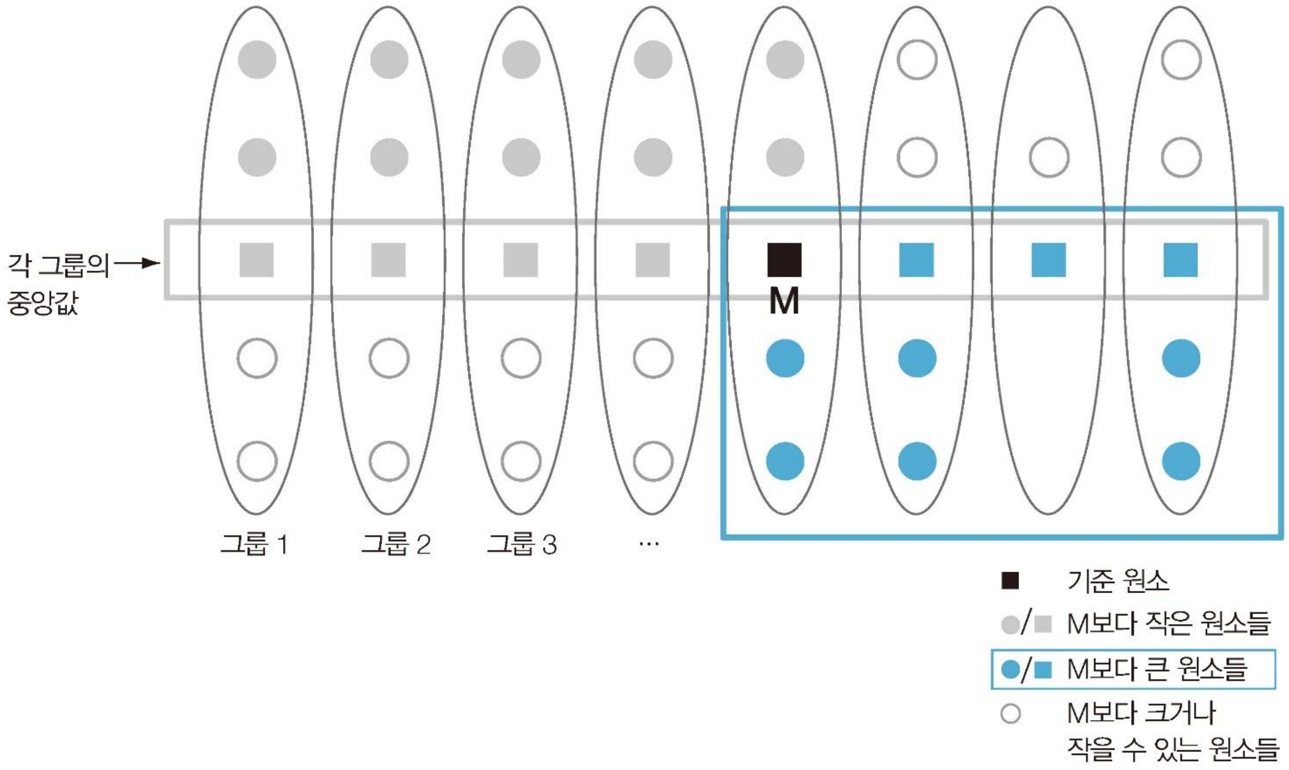


그림 5-4 기준 원소를 중심으로 분할된 상황을 표현한 예

최악의 경우 선형시간 선택 알고리즘

기준 원소를 잘 선택하는 오버헤드

$$T(n) \le \boxed{T(\lceil n/5 \rceil)} + T(7n/10 + 2) + \boxed{\Theta(n)}$$

4 6 1 2 3 5

이것은 $T(n) \le cn$임을 추정 후 증명법으로 증명할 수 있다 (다음 슬라이드)

그러므로 $T(n) = O(n)$

$T(n) = \Omega(n)$임은 명백하므로 $T(n) = \Theta(n)$

최악의 경우 선형시간 선택 알고리즘

$$T(n) \le T(\left\lceil \frac{n}{5} \right\rceil) + T(\frac{7n}{10} + 2) + \Theta(n)$$

$$T(n) \le T(\left\lceil \frac{n}{5} \right\rceil) + T(\frac{7n}{10} + 2) + \Theta(n)$$

$$\le T(\frac{n}{5} + 1) + T(\frac{7n}{10} + 2) + \Theta(n)$$

$$T(n) \le T(\left\lceil \frac{n}{5} \right\rceil) + T(\frac{7n}{10} + 2) + \Theta(n)$$

$$\le T(\frac{n}{5} + 1) + T(\frac{7n}{10} + 2) + \Theta(n)$$

$n_0 \le k < n$인 모든 k에 대해서 $T(k) \le ck$라고 가정하면 다음과 같다(n_0는 경계치).

❶ $\le c(\frac{n}{5} + 1) + c(\frac{7n}{10} + 2) + \Theta(n)$ ← $\frac{n}{5} + 1 < n \;\&\; \frac{7n}{10} + 2 < n$ $\rightarrow 7 \le n$

$= c(\frac{9n}{10} + 3) + \Theta(n)$

$= cn - \frac{cn}{10} + 3c + \Theta(n)$ ← $-\frac{cn}{10}$이 $\Theta(n)$을 압도할 수 있도록 하는 c가 존재하기 때문에 성립함

$\le cn$