

관계 중심의 사고법

# 쉽게 배우는 알고리즘

## 5장. 선택 알고리즘

# 5장. 선택 알고리즘

일을 시작하기 위해 기분이 내킬 때까지 기다리는 따위의 짓을  
하지 않으려면 시험 제도는 좋은 훈련이 된다.

-아놀드 토인비

# 학습목표

- 평균 선형 선택 알고리즘의 원리를 이해한다.
- 평균 선형 선택 알고리즘의 수행 시간 분석을 이해한다.
- 최악의 경우에도 선형 시간 보장하는 선택 알고리즘의 원리를 이해한다.
- 최악의 경우에도 선형 시간 보장하는 선택 알고리즘의 수행 시간 분석을 이해한다.
- 평균 선형 선택 알고리즘과 최악의 경우에도 선형 시간 보장하는 선택 알고리즘의 관계를 이해한다.

# 선택 알고리즘 ( $i$ 번째 작은 수 찾기)

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

# 평균 선형시간 선택 알고리즘

**select** ( $A, p, r, i$ )

▷ 배열  $A[p \dots r]$ 에서  $i$ 번째 작은 원소를 찾는다

{

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

$q \leftarrow \text{partition}(A, p, r)$  ;

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

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

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

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

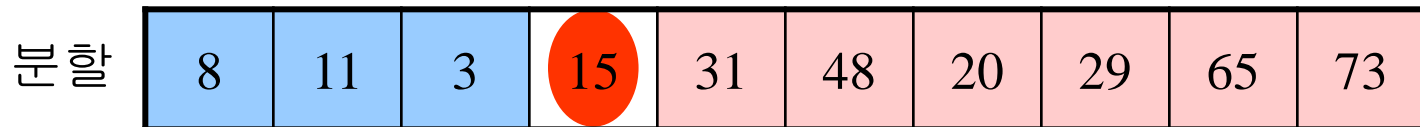
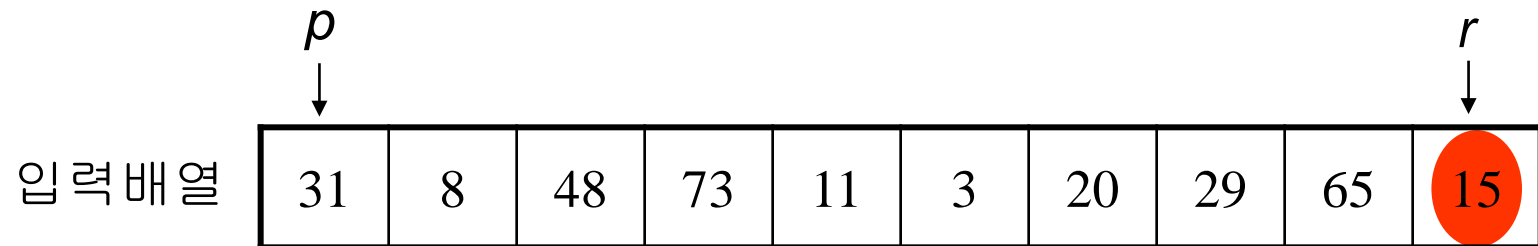
}

✓ 평균 수행 시간:  $\Theta(n)$

✓ 최악의 경우 수행 시간:  $\Theta(n^2)$

# 선택 알고리즘 작동 예 1

## 2번째 작은 원소 찾기

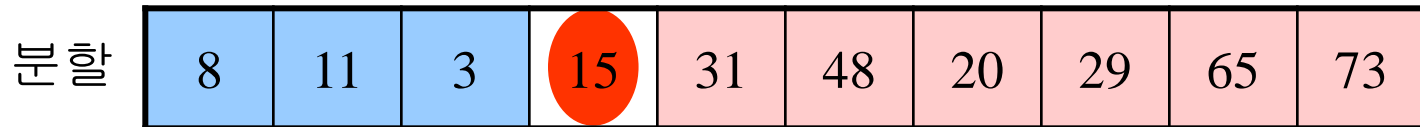
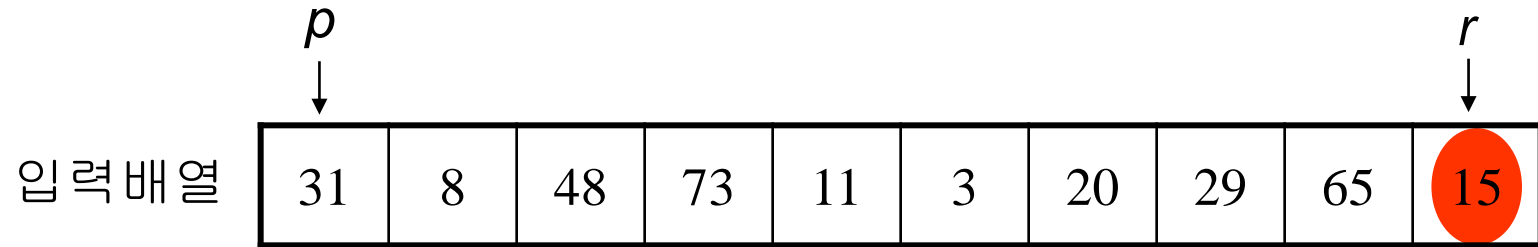


왼쪽 그룹에서 2번째 작은 원소를 찾는다

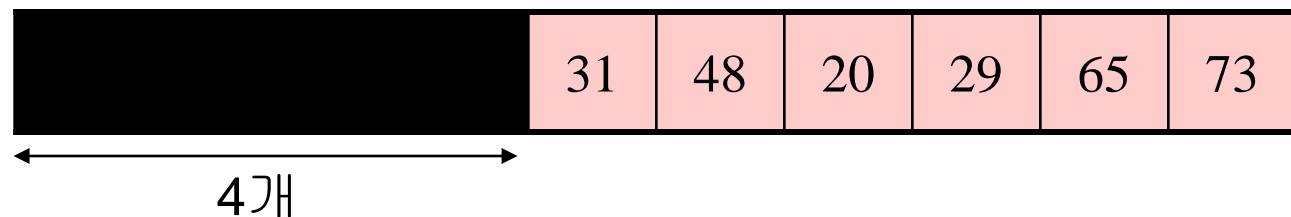


## 선택 알고리즘 작동 예 2

7번째 작은 원소 찾기



오른쪽 그룹에서 3번째 작은 원소를 찾는다



## 평균 수행 시간

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

↑  
분할된 양쪽 중 큰 쪽을 처리하는 비용

↑  
재귀호출을 제외한 오버헤드  
(분할이 대부분)

이것은  $T(n) \leq cn$ 임을 추정 후 증명법으로 증명할 수 있다

$$\therefore T(n) = O(n)$$

$$T(n) = \Omega(n) \text{임은 자명하므로 } T(n) = \Theta(n)$$

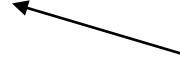


## 최악의 경우 수행 시간

$$T(n) = T(n-1) + \Theta(n)$$



분할이  $0:n-1$ 로 되고 큰 쪽을 처리하는 비용



재귀호출을 제외한 오버헤드  
(분할이 대부분)

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

# 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

**linearSelect** ( $A, p, r, i$ )

▷ 배열  $A[p \dots r]$ 에서  $i$ 번째 작은 원소를 찾는다

{

① 원소의 총 수가 5개 이하이면 원하는 원소를 찾고 알고리즘을 끝낸다.

② 전체 원소들을 5개씩의 원소를 가진  $\lceil n/5 \rceil$ 개의 그룹으로 나눈다.

(원소의 총수가 5의 배수가 아니면 이중 한 그룹은 5개 미만이 된다.)

③ 각 그룹에서 중앙값을 (원소가 5개이면 3번째 원소) 찾는다.

이렇게 찾은 중앙값들을  $m_1, m_2, \dots, m_{\lceil n/5 \rceil}$ 이라 하자.

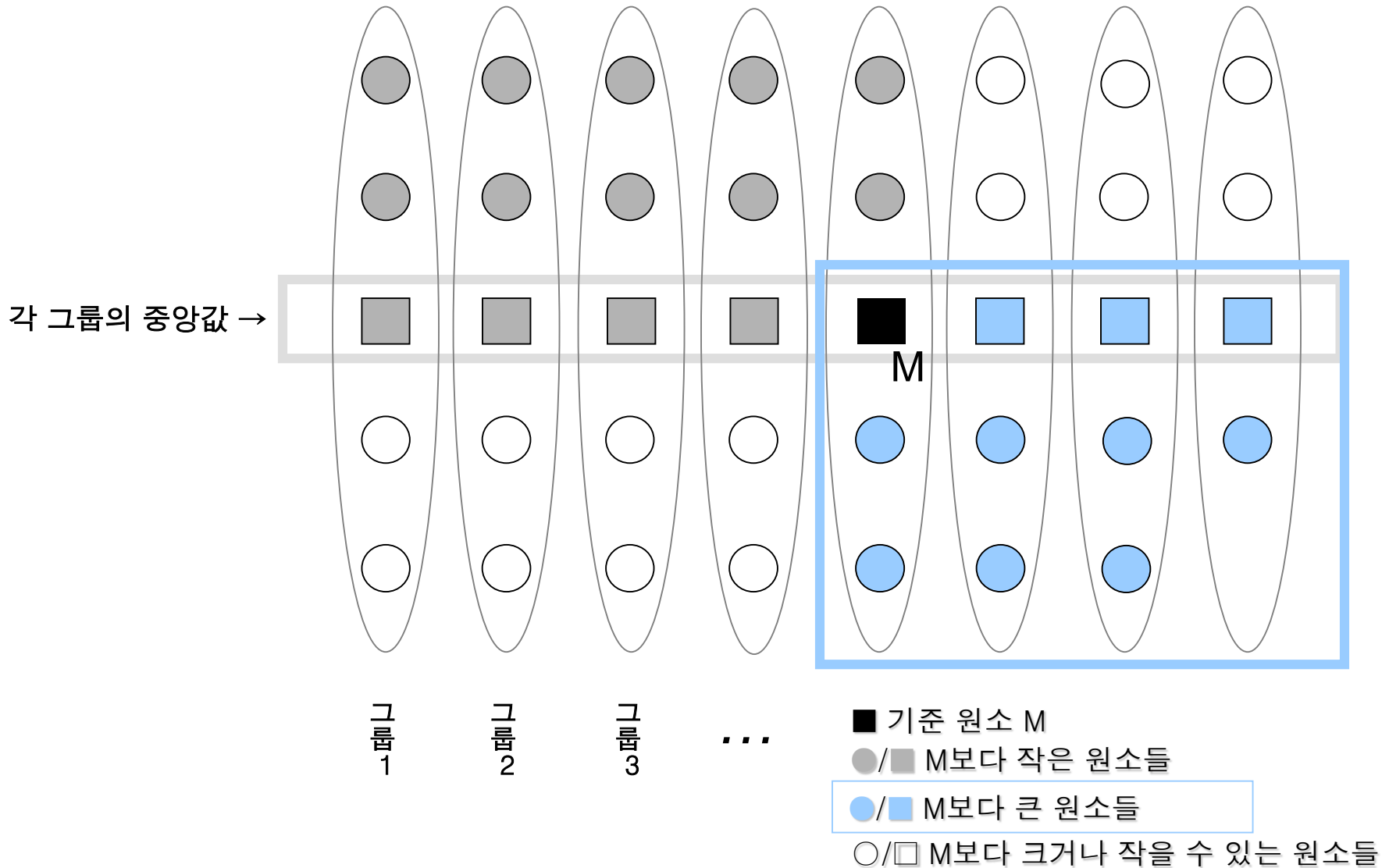
④  $m_1, m_2, \dots, m_{\lceil n/5 \rceil}$ 들의 중앙값  $M$ 을 재귀적으로 구한다. (원소의 총수가 홀수면 중앙값이 하나이므로 문제가 없고, 원소의 총수가 짝수일 경우는 두 중앙값 중 아무거나 임의로 선택한다.) ▷ call **linearSelect**()

⑤  $M$ 을 기준원소로 삼아 전체 원소를 분할한다( $M$ 보다 작거나 같은 것은  $M$ 의 왼쪽에,  $M$ 보다 큰 것은  $M$ 의 오른쪽에 오도록).

⑥ 분할된 두 그룹 중 적합한 쪽을 선택하여 단계 ①~⑥을 재귀적으로 반복한다. ▷ call **linearSelect**()

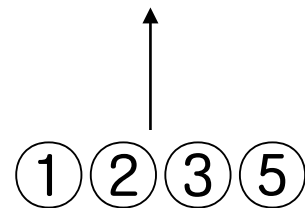
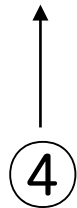
}

# 기준 원소를 중심으로 한 대소 관계



## 최악의 경우 수행 시간

$$T(n) \leq T(\lceil n/5 \rceil) + T(7n/10 + 4) + \Theta(n)$$



이것은  $T(n) \leq cn$ 임을 추정 후 증명법으로 증명할 수 있다

$$\therefore T(n) = O(n)$$

$$T(n) = \Omega(n) \text{임은 자명하므로 } T(n) = \Theta(n)$$



**Thank you**

---