

고급 정렬 알고리즘

백석대학교 강윤희

고급 정렬 알고리즘

- 평균적으로 $\Theta(n \log n)$ 의 시간이 소요되는 정렬 알고리즘들
 - 병합정렬
 - 퀵정렬
 - 힙정렬

병합정렬

mergeSort(A[], p, r)

▷ A[p ... r]을 정렬한다.

{

if (p < r) then {

q ← $\lfloor (p + r) / 2 \rfloor$; ----- ①

mergeSort(A, p, q); ----- ②

mergeSort(A, q+1, r); ----- ③

merge(A, p, q, r); ----- ④

}

}

▷ p, q의 중간 지점 계산

▷ 전반부 정렬

▷ 후반부 정렬

▷ 병합

merge(A[], p, q, r)

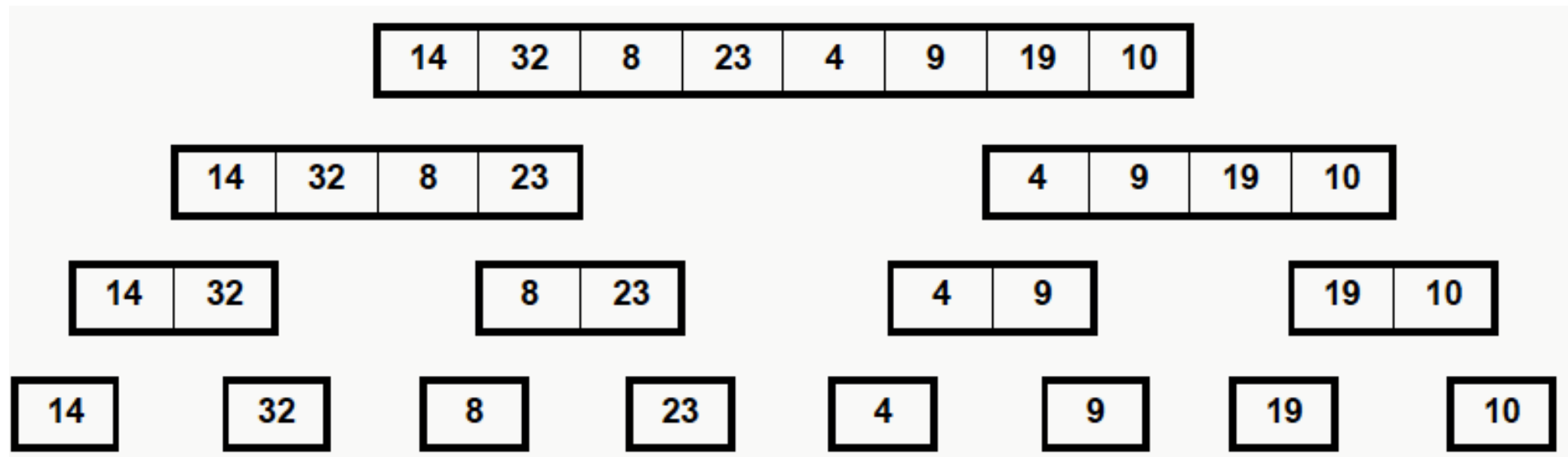
{

정렬되어 있는 두 배열 A[p ... q]와 A[q+1 ... r]을 합쳐
정렬된 하나의 배열 A[p ... r]을 만든다.

}

병합정렬

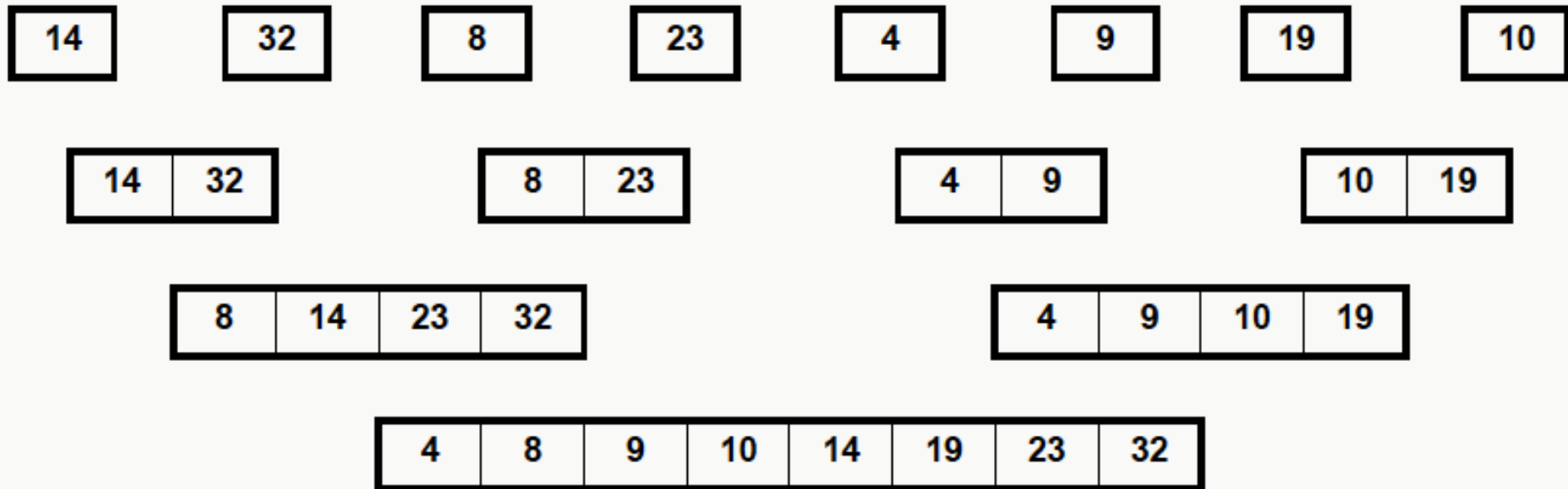
- 나누기 단계



나누기 수준(level) = $\log N$

병합정렬

- 병합 단계



병합정렬의 작동 예

정렬할 배열이 주어짐

| | | | | | | | | | |
|----|---|----|----|---|----|----|----|----|----|
| 31 | 3 | 65 | 73 | 8 | 11 | 20 | 29 | 48 | 15 |
|----|---|----|----|---|----|----|----|----|----|

배열을 반반으로 나눈다

| | | | | | | | | | |
|----|---|----|----|---|----|----|----|----|----|
| 31 | 3 | 65 | 73 | 8 | 11 | 20 | 29 | 48 | 15 |
|----|---|----|----|---|----|----|----|----|----|

 — ①

각각 독립적으로 정렬한다

| | | | | | | | | | |
|---|---|----|----|----|----|----|----|----|----|
| 3 | 8 | 31 | 65 | 73 | 11 | 15 | 20 | 29 | 48 |
|---|---|----|----|----|----|----|----|----|----|

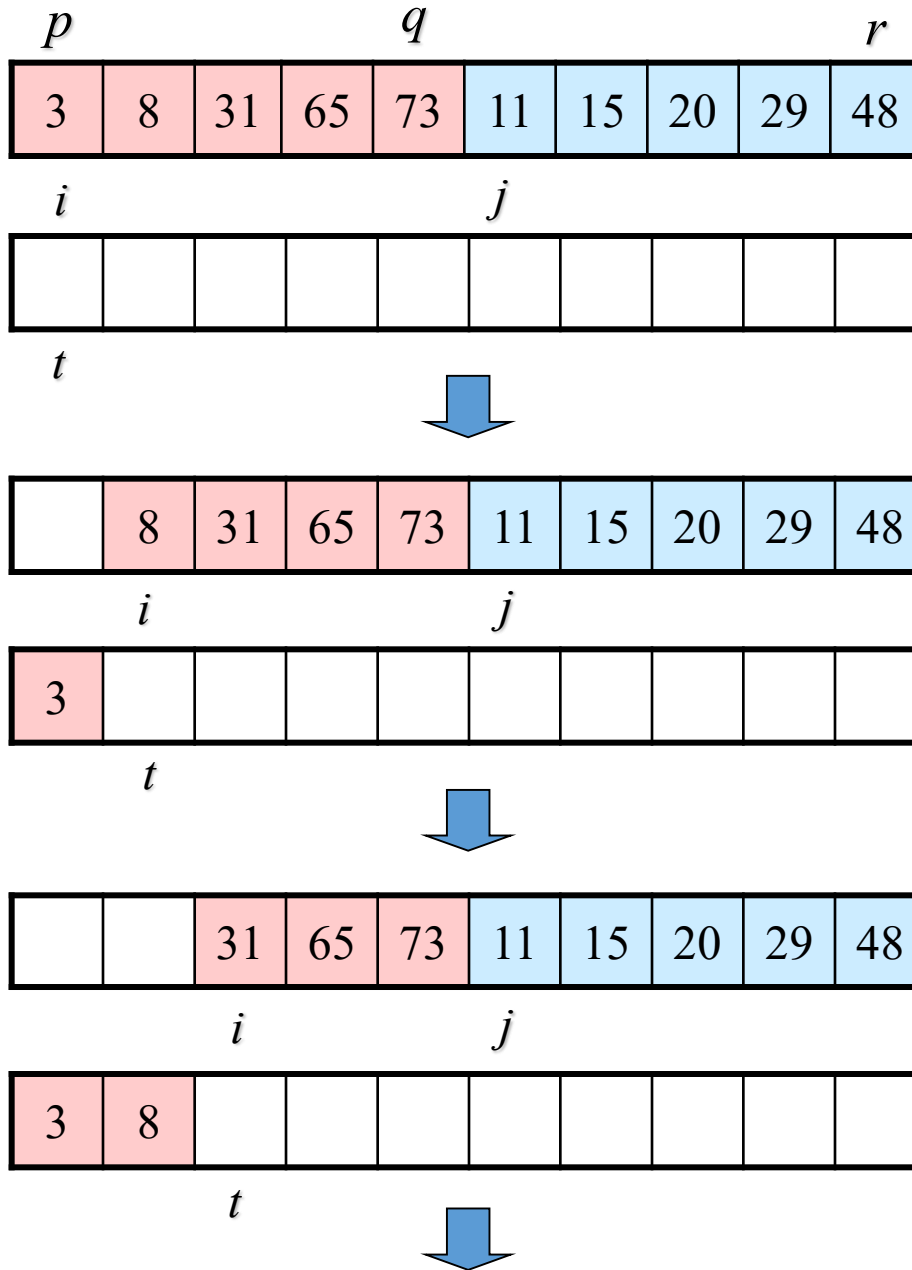
 — ② ③

병합한다 (정렬완료)

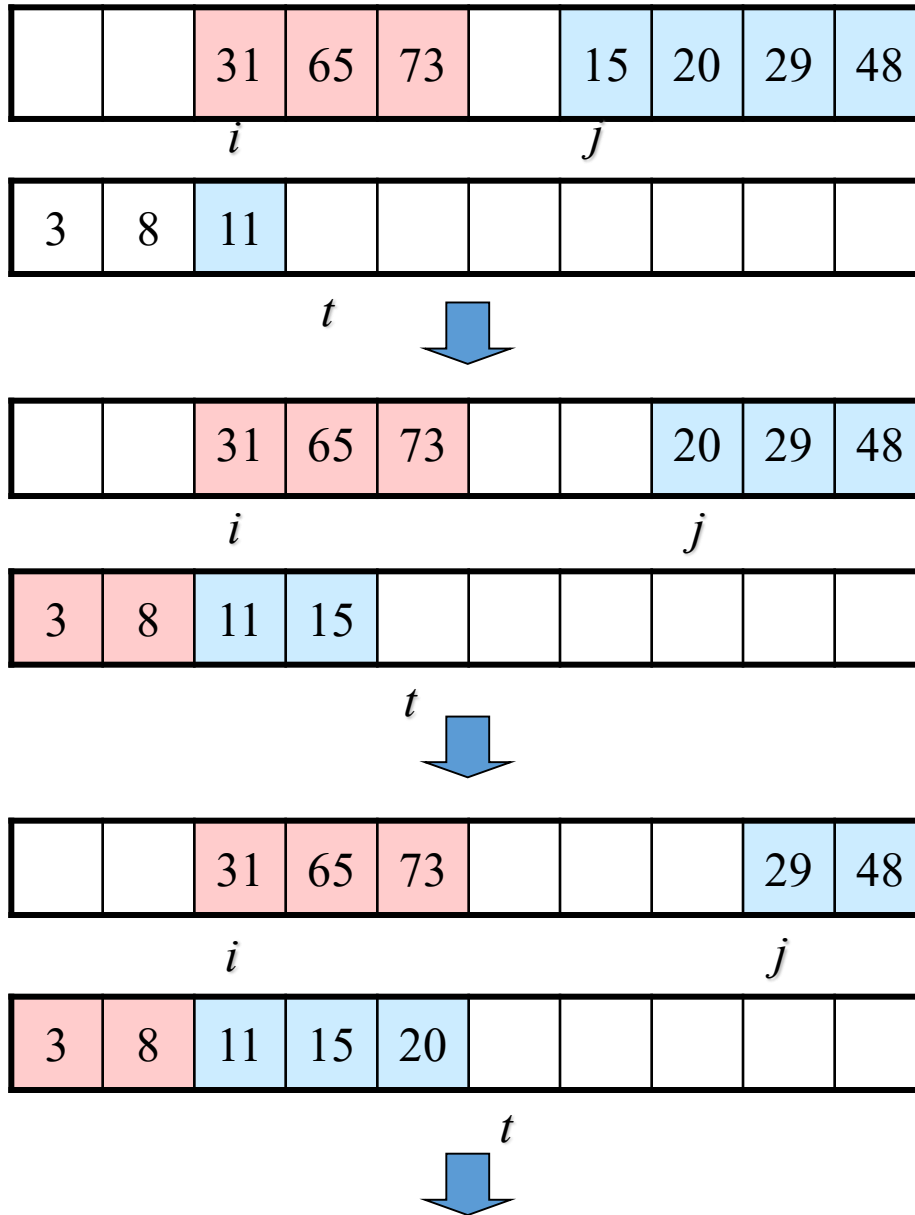
| | | | | | | | | | |
|---|---|----|----|----|----|----|----|----|----|
| 3 | 8 | 11 | 15 | 20 | 29 | 31 | 48 | 65 | 73 |
|---|---|----|----|----|----|----|----|----|----|

 — ④

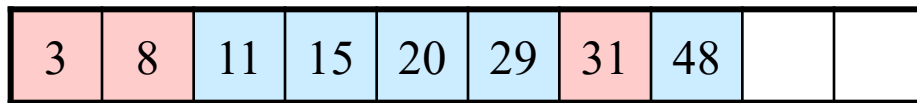
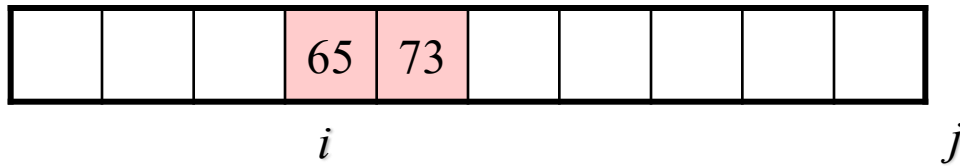
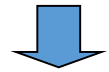
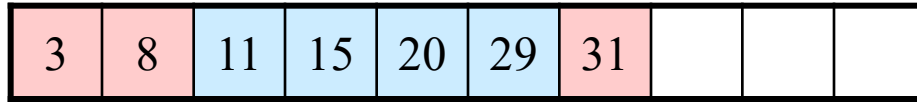
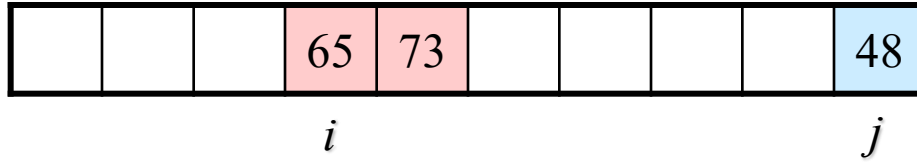
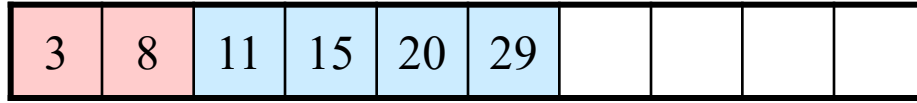
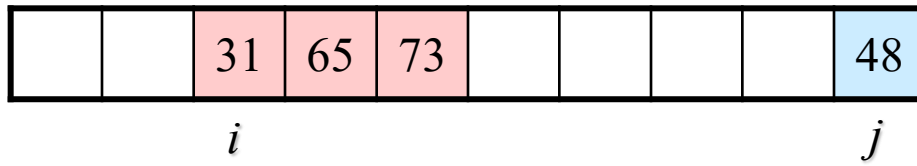
병합(merge)의 작동 예

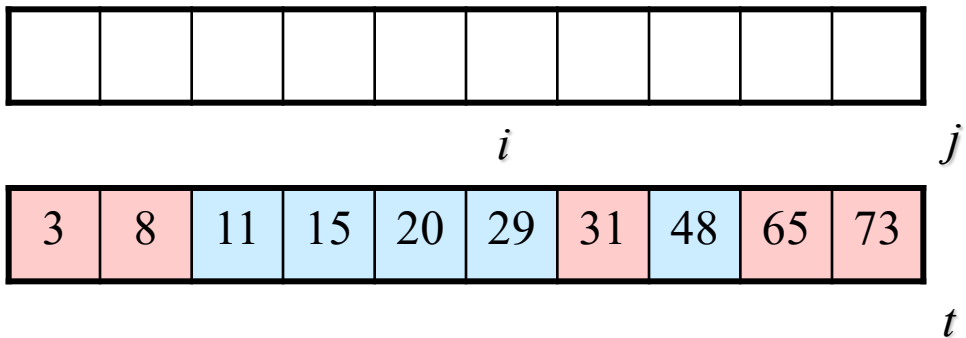


병합(merge)의 작동 예



병합(merge)의 작동 예





병합(merge)의 작동 예