



데이터 구조

13주차: 기수정렬(radix sort)

기수 정렬 (Radix Sort)

● 정의

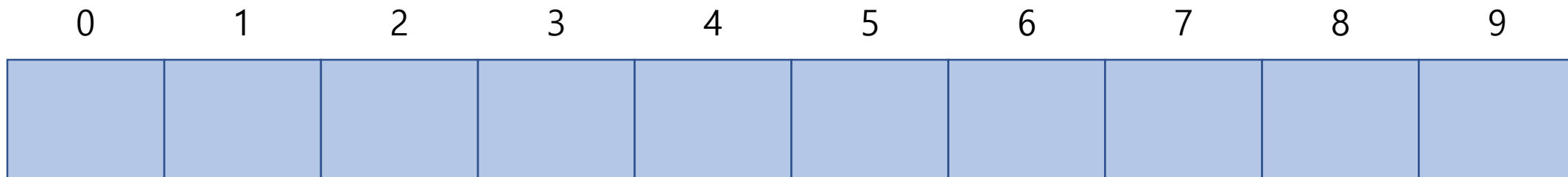
- 자릿수를 기준으로 정렬하는 알고리즘으로 버킷(Bucket)을 활용한다.
- 데이터 간에 비교 연산을 하지 않고, 정렬할 수 있는 안정적인 정렬 알고리즘의 일부

● 특징

- 빠른 속도: 기수 정렬은 데이터들을 비교하지 않고 자릿수를 기준으로 한다는 점에서 빠른 정렬속도를 가지게 된다.
- 안정정렬: 기존에 정렬된 자릿수의 값이 같은 경우, 정렬이 바뀌지 않고 기존의 순서를 유지하게 되므로 원하는 정렬을 구성할 수 있다.
- 추가 메모리: 제자리 정렬 형태가 아니기 때문에 데이터를 보관하기 위해 추가적인 메모리를 필요로 한다.

기수 정렬 (Radix Sort)

● Bucket 이란?



각 자릿수 (0~9) 별로 버킷(Bucket)이 있고, 각 버킷(Bucket)들은 Queue 로 구성된다.
각 버킷들은 Queue로 구현하여 제일 먼저 추가된 데이터가 제일 먼저 출력되도록 한다.

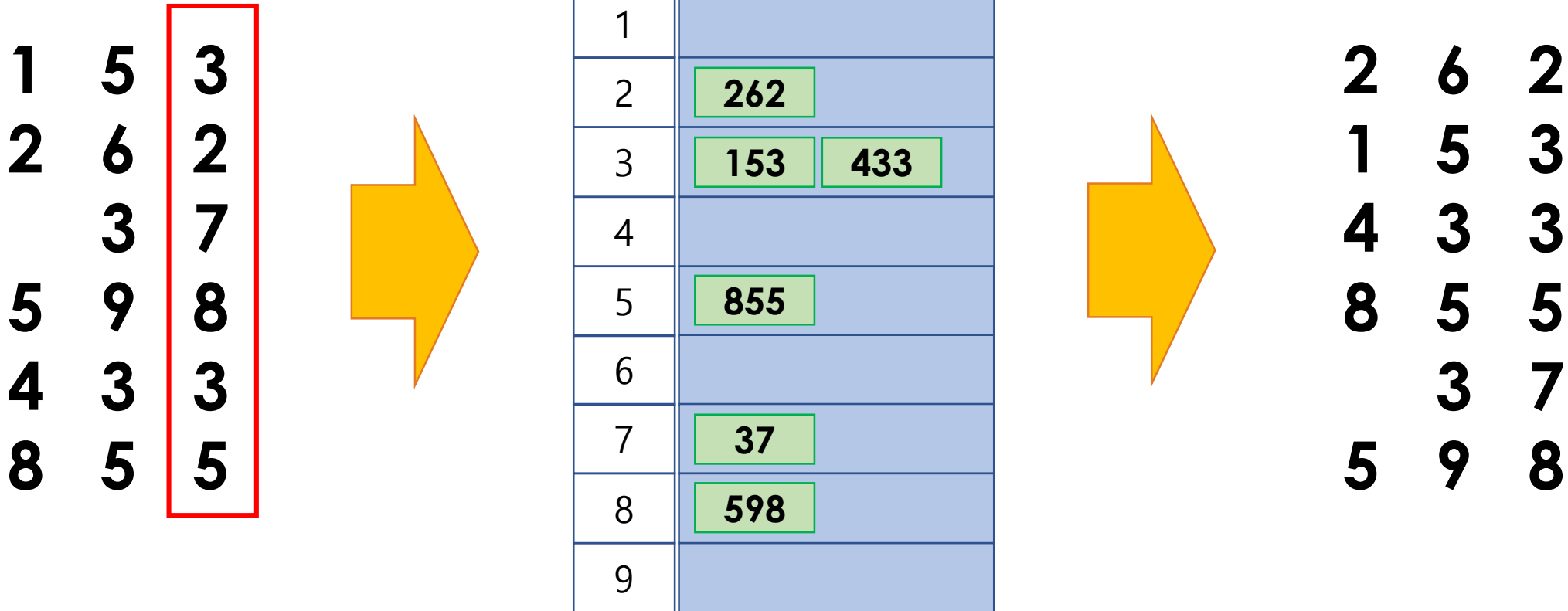
● 기수 정렬의 방식

1. MSD (Most Significant Digit): 가장 큰 자릿수부터 정렬하는 방식
 - 즉, 정수를 기준으로 왼쪽 자릿수 부터 정렬
2. LSD (Least Significant Digit): 가장 작은 자릿수부터 정렬하는 방식
 - 즉, 정수를 기준으로 오른쪽 자릿수 부터 정렬

기수 정렬 (Radix Sort) 예시 – LSD 방식

데이터: [153, 262, 37, 598, 433, 855]

Step 1. 1자릿수를 판단하여 자릿수 별 Bucket에 저장
=> 버킷 순서별로 저장된 데이터를 출력



기수 정렬 (Radix Sort) 예시 – LSD 방식

Step 2. 10자리수를 판단하여 자릿수 별 Bucket에 저장
=> 버킷 순서별로 저장된 데이터를 출력

2 6 2
1 5 3
4 3 3
8 5 5
 3 7
5 9 8



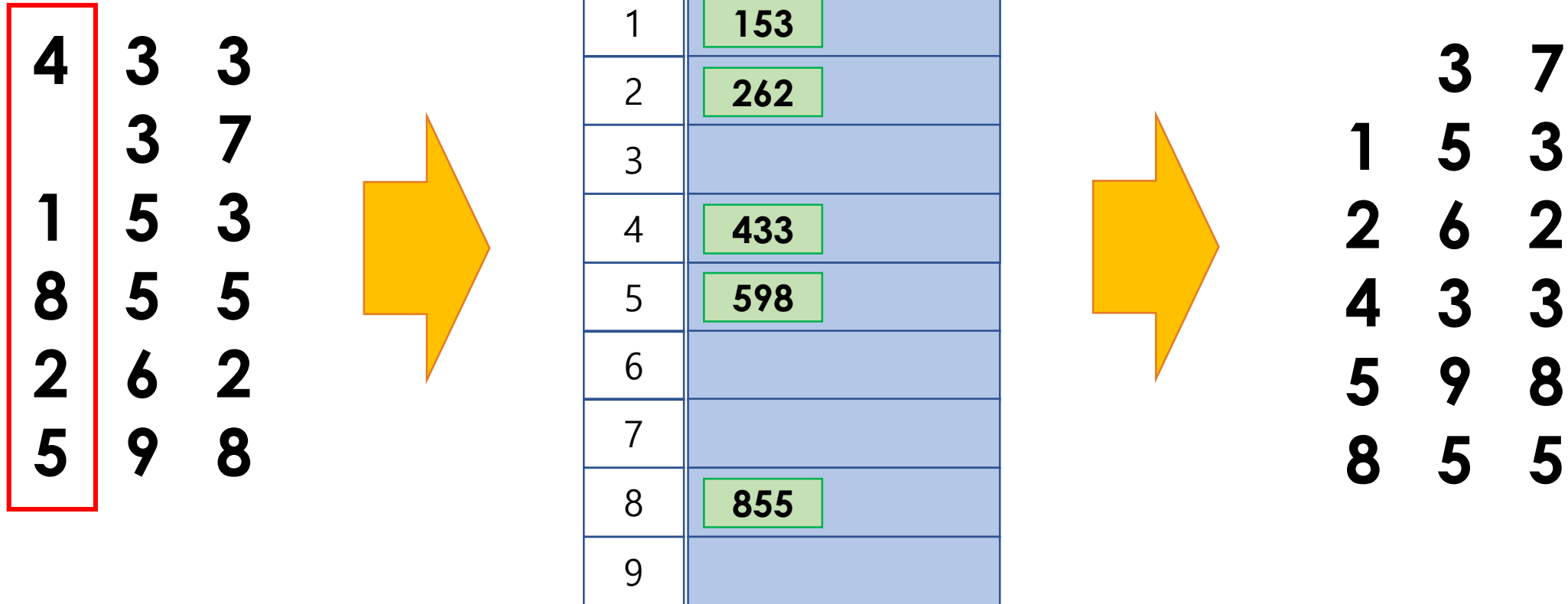
0	
1	
2	
3	433 37
4	
5	153 855
6	262
7	
8	
9	598



4 3 3
 3 7
1 5 3
8 5 5
2 6 2
5 9 8

기수 정렬 (Radix Sort) 예시 – LSD 방식

Step 3. 100자리수를 판단하여 자릿수 별 Bucket에 저장
=> 버킷 순서별로 저장된 데이터를 출력



과제

- ▣ 강의 중 LSD 기수정렬 프로그래밍 코드를 기반으로 MSD 기수정렬 프로그래밍을 완성하세요. (코드 및 실행결과 포함)
- ▣ Big-O 표기법에 대해 설명하세요.
 - 과제 6월 11일 까지 제출
 - 제출방법 : swcberet@naver.com
 - [메일제목] 학번_성명_데이터구조_4차과제
 - [파일명] 학번_성명_데이터구조_4차과제