

# Radix sort





# outline

- Introduction
- LSD radix sort implement
- Example
- Conclusion



# Introduction

- Linear-time sorting method
  - 當排序技巧並非採用 comparision-based skill 時, 則時間複雜度有機會突破  $\Omega(n \log n)$  限制
  - 常見的方法：
    - Radix Sort
    - Bucket Sort
    - Counting Sort
- Radix sort (基數排序法)



# Introduction

- Radix sort
  - 採用 divide and conquer 技巧
  - 可分為
    - LSD Radix sort (Least Significant Digit)
    - MSD Radix sort (Most Significant Digit)

1276531  
Most      Least



# LSD radix sort 1

- 令  $r$  為要排序元素的基底數, 則需準備  $r$  個容器
  - for example:
    - 排列元素為 10 進位數字, 則需準備 0~9 共 10 個容器
    - 排列元素為 8 進位數字, 則需準備 0~7 共 8 個容器
    - 排列元素為 大寫英文字母, 則需準備 A~Z 共 26 個容器



## LSD radix sort 2

- 令  $d$  為 input data 最多的位數個數, 則表示需要做  $d$  回合才能完成 sorting 動作
  - for example:
    - 10, 0, 2, 3 其中最多的位數個數為 2 (10)
    - ABCDE, D, EA, Z 其中最多的位數個數為 5 (ABCDE)
  - 所以, 若今天 input 的鍵值範圍受到限制, 我們可以預先知道需做幾回合結束 sorting



## LSD radix sort 3

- 由低位數往高位數, 依序做各回合分派合併工作, 即可完成排序
  - e.g., 從個位數 > 百位數 (multi-pass)
    - pass1 : 個位數
    - pass2 : 十位數
    - pass3 : 百位數
- 每個 pass 主要分為兩個工作 :
  - 分派 : 依照 Data 之某位數值, 將它丟到對應的容器之中 (1 中有提到總共  $r$  個)
  - 合併 : 全部分派完後, 依序合併所有容器內的 data, 恢復成序列



# LSD radix sort Example

- 用 radix sort 從小到大排序：
- input：

179	208	306	93	859	984	55	9	271	33
-----	-----	-----	----	-----	-----	----	---	-----	----

LSD radix sort



9	33	55	93	179	208	271	306	859	984
---	----	----	----	-----	-----	-----	-----	-----	-----

output





# LSD radix sort Example

1. 基底 base 為 10，所以準備10 個容器

0	1	2	3	4	5	6	7	8	9
---	---	---	---	---	---	---	---	---	---

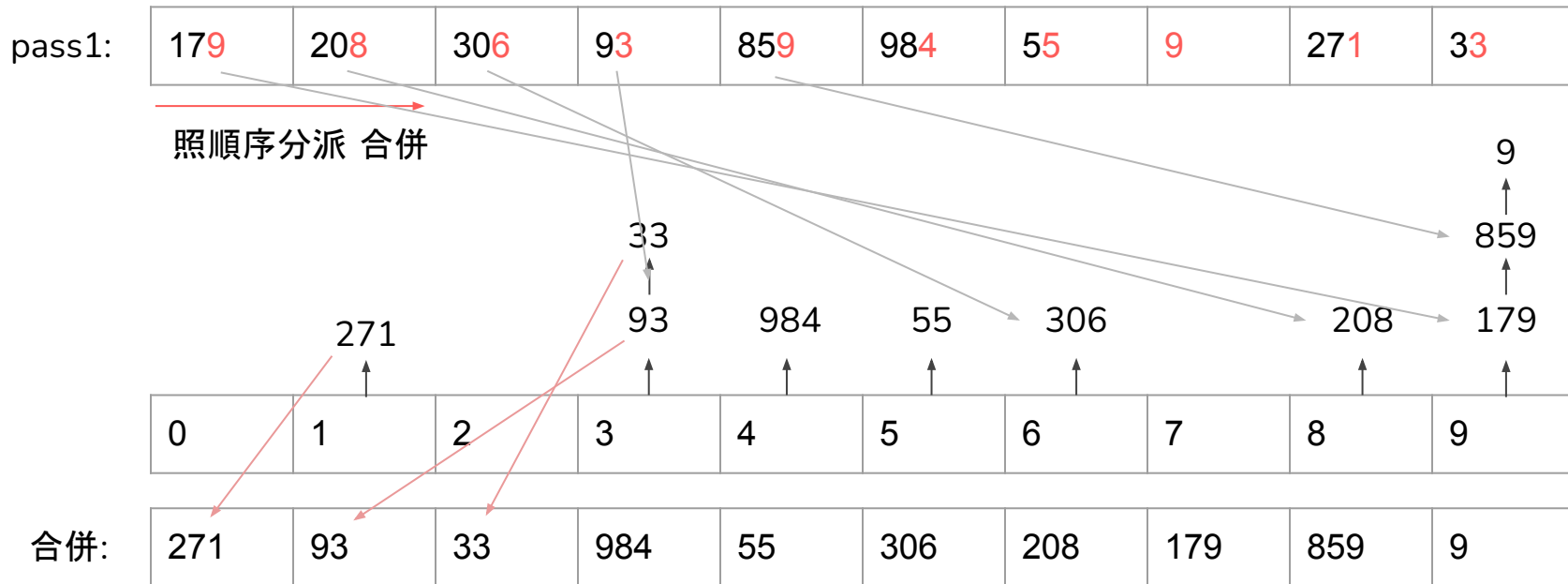
2. 知道 data 中最大為 984, 所以我們可用3個回合完成排序

179	208	306	93	859	984	55	9	271	33
-----	-----	-----	----	-----	-----	----	---	-----	----



# LSD radix sort Example

3. 從最低位數往最高位數, 每回合做分派與合併工作 (個位數)





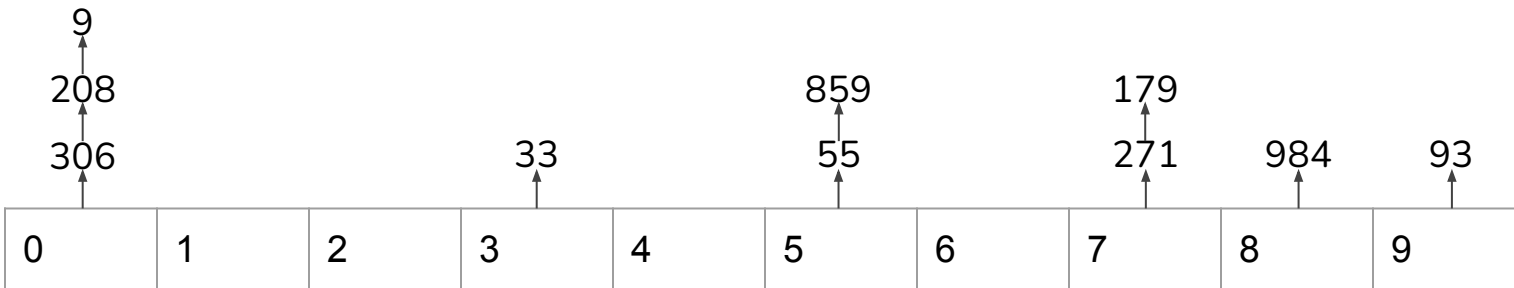
## LSD radix sort Example

3. 從最低位數往最高位數, 每回合做分派與合併工作 (十位數)

pass2:

271	93	33	984	55	306	208	179	859	09
-----	----	----	-----	----	-----	-----	-----	-----	----

照順序分派 合併



合併:

306	208	9	33	55	859	271	179	984	93
-----	-----	---	----	----	-----	-----	-----	-----	----



# LSD radix sort Example

3. 從最低位數往最高位數, 每回合做分派與合併工作 (百位數)

pass3:

306	208	009	033	055	859	271	179	984	093
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

照順序分派 合併



合併:

9	33	55	93	179	208	271	306	859	984
---	----	----	----	-----	-----	-----	-----	-----	-----



# Conclusion

- Time complexity :
  - 每回合：
    - 分派 :  $n$  次
    - 合併 :  $r$  次
  - 總共  $d$  個回合
  - 所以 Time complexity :  $O(d*(n+r))$
  - $r$  基底,  $d$  鍵值範圍, 皆為固定, 可視為常數
- space complexity :  $n*r$
- stable 的 sorting