

Indian Institute of Technology Madras
ZANZIBAR CAMPUS

Z5007 · PROGRAMMING AND DATA STRUCTURES

Counting Sort and Radix Sort: Sorting Without Comparing

Week 7 · Monday 16 November 2026 · CLRS Ch 8; Canning, Broder & Lafore Ch 7

Dr. Innocent Nyalala | MTech Semester 1 | IIT Madras Zanzibar

Today's Plan

IIT Madras Zanzibar

1. (0–6 min) The big idea: sort by counting, not comparing
2. (6–20 min) Counting sort: count, running total, place from the right
3. (20–34 min) Radix sort: buckets, one digit at a time
4. (34–40 min) Choosing the right sort
5. (40–48 min) Worksheets: one counting sort, one radix sort, by hand

Learning Outcomes

IIT Madras Zanzibar

By the end of this session, you will be able to:

Explain	why counting and radix sort are not limited by n log n, and what they cost
Perform	counting sort by hand: count, running total, place from the right
Perform	radix sort by hand: buckets for each digit, collect, repeat
Choose	the right sort for a given range of keys

Key Terms for Today

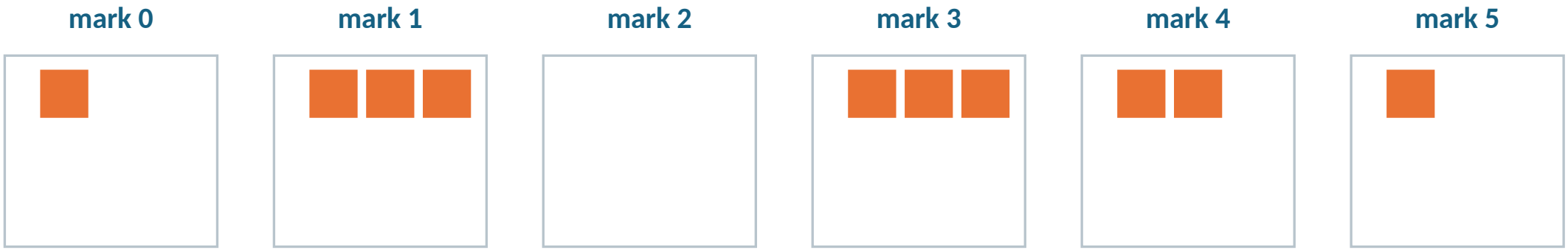
IIT Madras Zanzibar

Term	Meaning	Example
Key	the value we sort by	a mark from 0 to 5
k	the largest key; keys are whole numbers 0 to k	marks out of 5: k = 5
Count array	count[v] = how many items have key v	count[3] = 3
Running total	count[v] = how many items have key ≤ v	tells us where v ends
Bucket	a list collecting the items that share a digit	bucket 7: numbers ending in 7
Stable	equal keys stay in their original order	Asha before Juma

Sort Ten Marks Without Comparing Any Two

IIT Madras Zanzibar





Read the columns:



? **Think:** did we compare any two marks?

Why the n log n Limit Does Not Apply

IIT Madras Zanzibar

1 **The limit**

n! possible orders; each comparison halves them; so at least $\log_2(n!) \approx n \log n$ comparisons

2 **The loophole**

never compare: use the key itself as an address, a column number

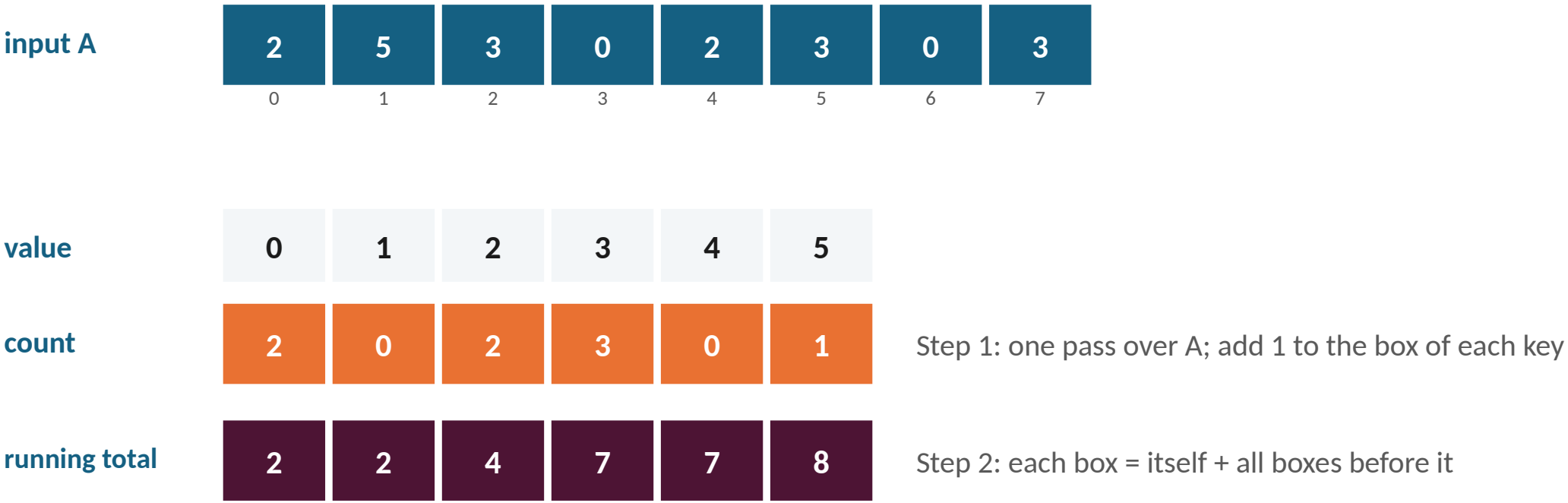
3 **The price**

keys must be whole numbers 0 to k, with k small. Cost: about n + k steps

? **Think:** does the n log n limit apply to merge sort? To counting sort?

Counting Sort, Steps 1 and 2: Count, Then Running Total

IIT Madras Zanzibar

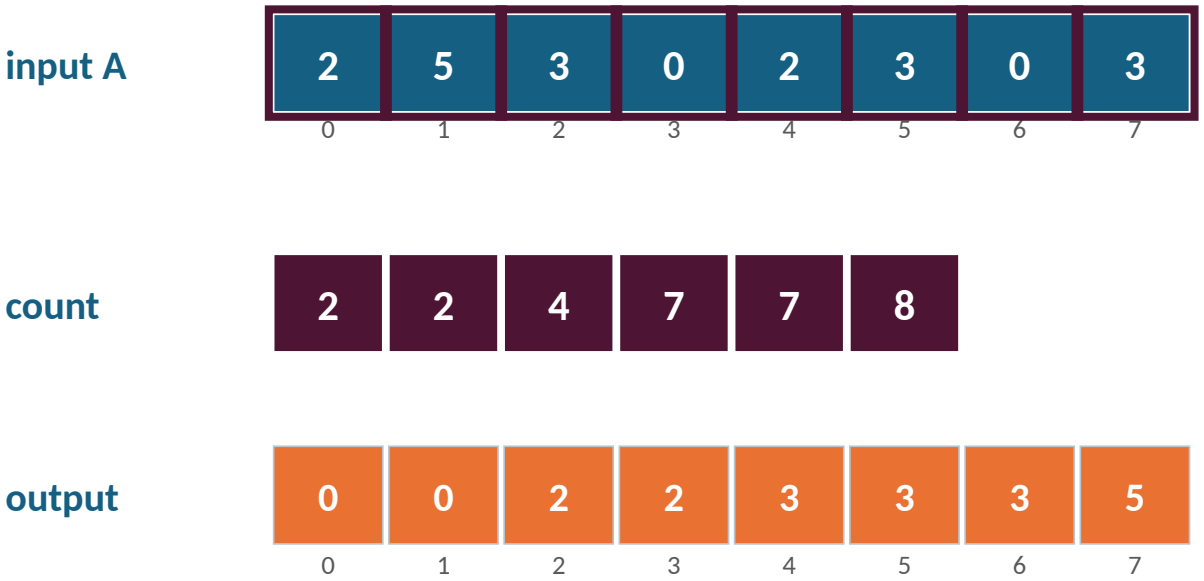


Step 1: one pass over A; add 1 to the box of each key

Step 2: each box = itself + all boxes before it

? **Think:** what does count[5] equal after the running total, and what does it mean?

Counting Sort, Step 3: Place from Right to Left



3: count[3] becomes 6, so slot 6

0: count[0] becomes 1, so slot 1

3: count[3] becomes 5, so slot 5

2: count[2] becomes 3, so slot 3

0: count[0] becomes 0, so slot 0

3: count[3] becomes 4, so slot 4

5: count[5] becomes 7, so slot 7

2: count[2] becomes 2, so slot 2

? **Think:** why do we subtract 1 BEFORE placing the item?

The counting_sort Code, Line by Line

IIT Madras Zanzibar

```
 1  def counting_sort(a, k):
 2      count = [0] * (k + 1)
 3      for x in a:
 4          count[x] += 1
 5      for v in range(1, k + 1):
 6          count[v] += count[v - 1]
 7      out = [0] * len(a)
 8      for x in reversed(a):
 9          count[x] -= 1
10          out[count[x]] = x
11      return out
```

Line by line

Line 2 one box per key 0..k, all 0

Line 3–4 step 1: add 1 to the box of each key

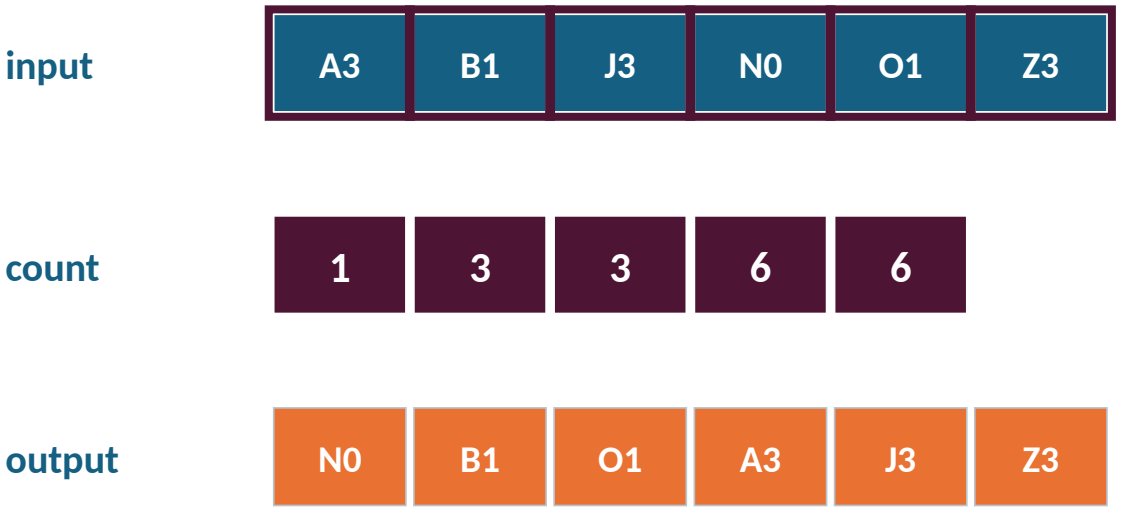
Line 5–6 step 2: running total

Line 7 result list, same length as a

Line 8 step 3: walk a from the LAST item

Line 9–10 subtract 1; put x in that slot

Example 2: Records, and Why Stability Matters



Z3 goes to slot 5

O1 goes to slot 2

N0 goes to slot 0

J3 goes to slot 4

B1 goes to slot 1

A3 goes to slot 3

? **Think:** what goes wrong if we walk through the input from LEFT to RIGHT?

What Counting Sort Costs, and Where It Breaks

IIT Madras Zanzibar

1

Time and memory

O(n + k) time; count array and output: O(n + k) memory; stable if placed right to left

2

When k is large

keys up to 10^9 need a billion boxes: counting sort is the wrong tool

3

Shift keys

keys 100 to 109: subtract 100. Negative keys: add |smallest|

4

Not for

decimals or text directly: only whole numbers can be addresses

?

Think: 10 million exam scores from 0 to 100: which sort, and why?

In-Class Exercise (4 minutes, in pairs)

IIT Madras Zanzibar

IN-CLASS TASK: count, total, place

Sort A with k = 4. Fill in count, running total, then place from the LAST item. Paper worksheet 1.

A

4	1	3	4	3	0	1	3
0	1	2	3	4	5	6	7

count

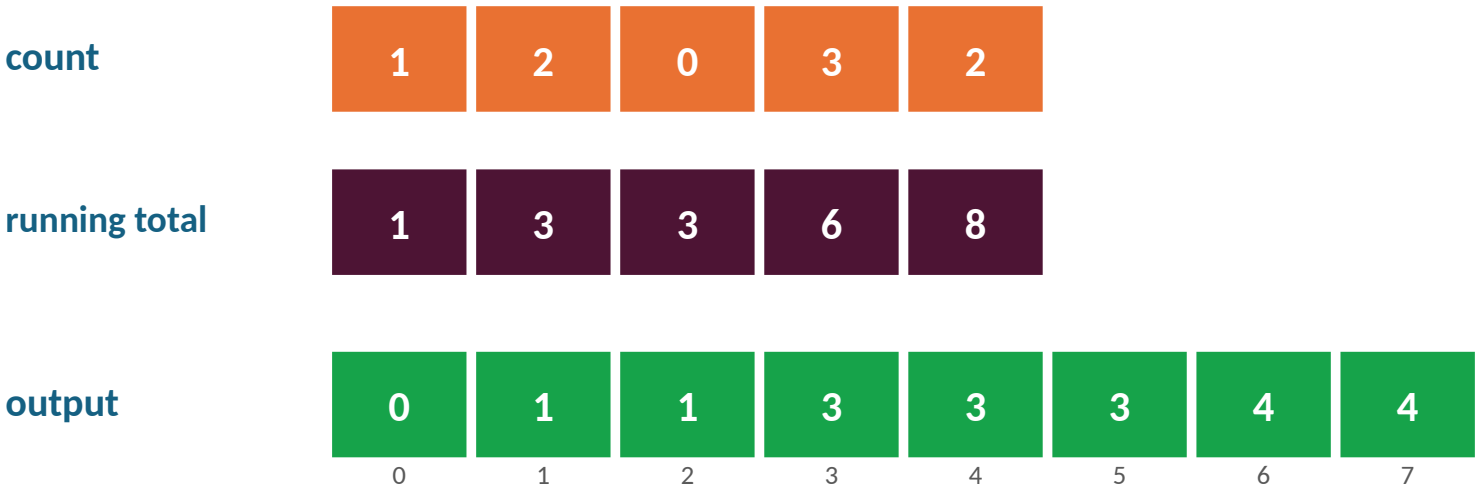
running total

output

(a) How many comparisons did you make?

(b) In what order do the three 3s come out?

Solution: Count, Total, Place



1. From the right: 3 to slot 5, 1 to slot 2, 0 to slot 0, 3 to slot 4, 4 to slot 7, 3 to slot 3, 1 to slot 1, 4 to slot 6.
2. (a) zero comparisons. (b) the 3s keep their input order: 3a, 3b, 3c.

Radix Sort: One Digit at a Time

IIT Madras Zanzibar

1 Last digit

sort all numbers by the ones digit, stably

2 Next digit

sort again by the tens digit, stably

3 Keep going

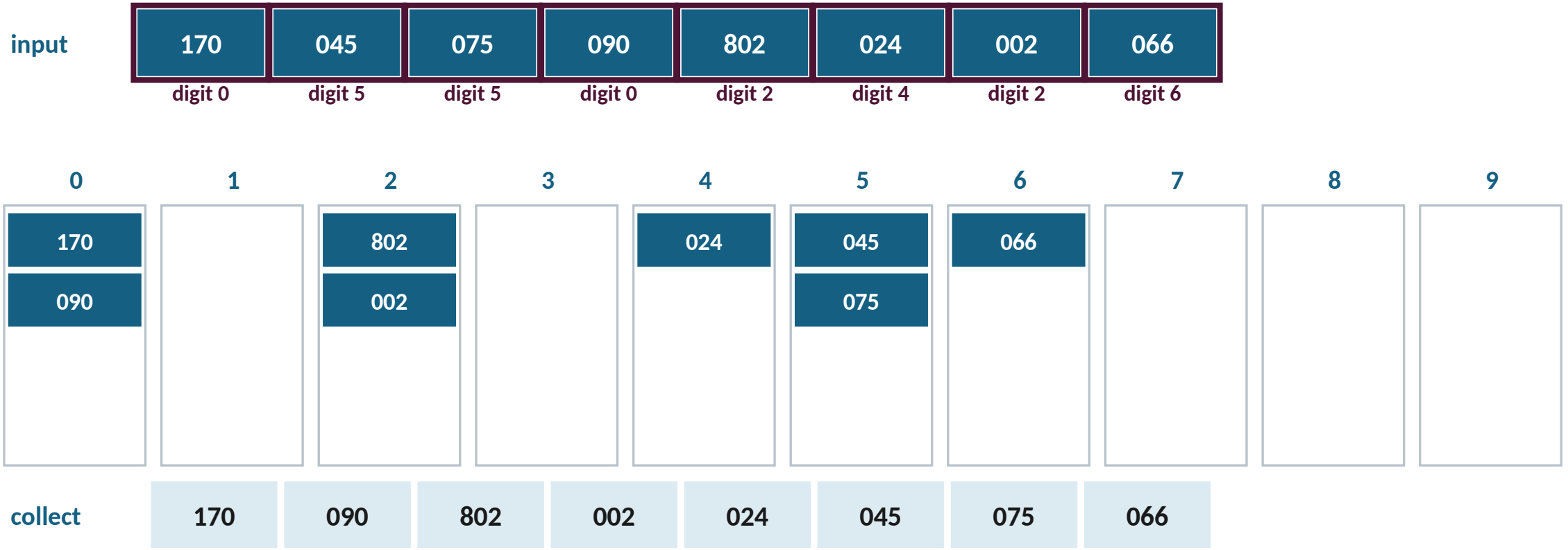
one pass per digit, up to the most significant

x	ones	tens	hundreds
802	2	0	8
45	5	4	0
170	0	7	1

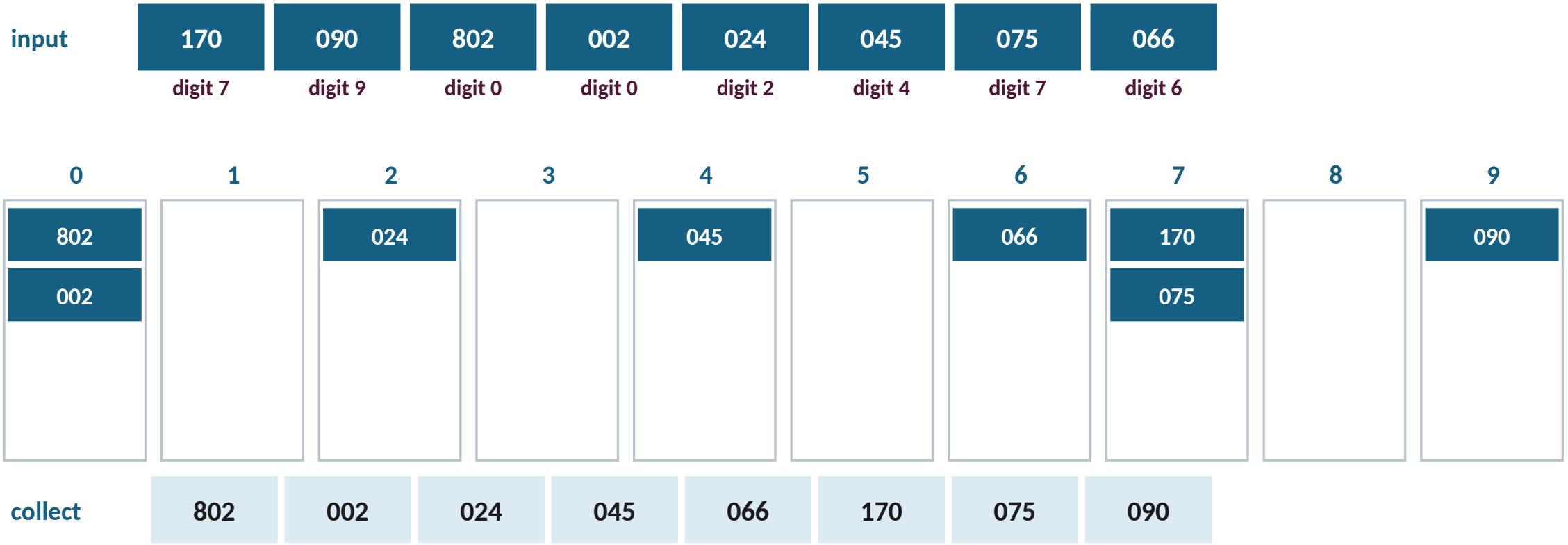
```
digit = (x // exp) % 10
# exp = 1, 10, 100, ...
```

? **Think:** what are the tens digit and the hundreds digit of 45?

Radix Sort, Pass 1: the Ones Digit

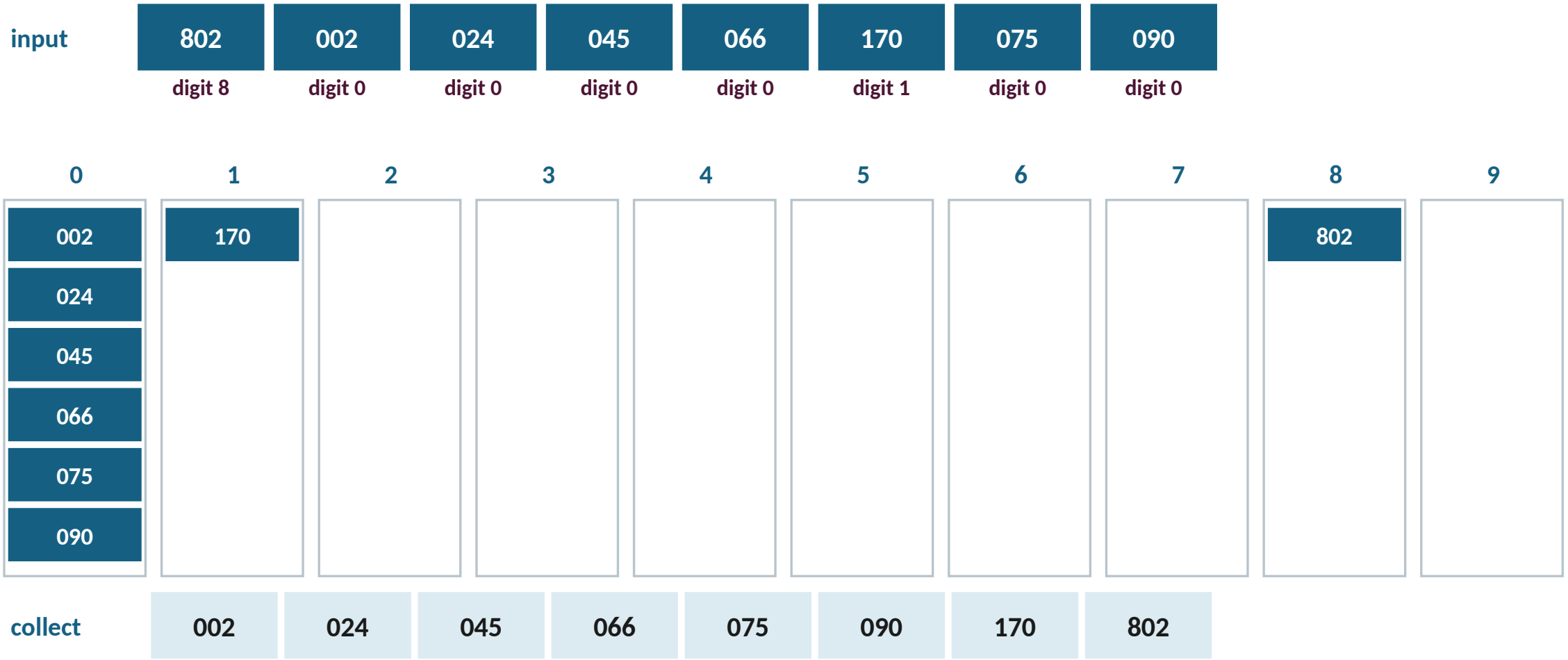


Radix Sort, Pass 2: the Tens Digit



? **Think:** bucket 7 holds 170 and 075. Which goes first, and why?

Radix Sort, Pass 3: the Hundreds Digit



The Same Idea for Words: One Letter at a Time

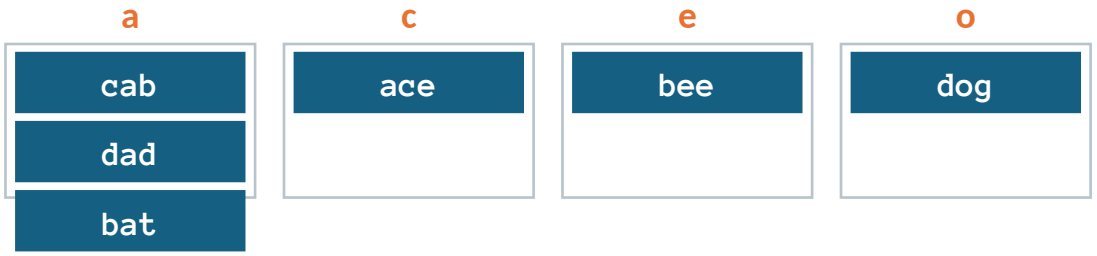
IIT Madras Zanzibar

Pass 1: last letter



collect: cab dad bee ace dog bat

Pass 2: middle letter



collect: cab dad bat ace bee dog

Pass 3: first letter



collect: ace bat bee cab dad dog

The radix_sort Code, Line by Line

```
1  def radix_sort(a):
2      exp, biggest = 1, max(a)
3      while biggest // exp > 0:
4          buckets = [[] for _ in range(10)]
5          for x in a:
6              buckets[(x // exp) % 10].append(x)
7          a = [x for b in buckets for x in b]
8          exp *= 10
9      return a
```

Line by line

Line 2 exp: 1 = ones, 10 = tens…; biggest number

Line 3 while the biggest still has this digit

Line 4 ten empty buckets, 0 to 9

Line 5–6 append each x to the END of its bucket

Line 7 collect buckets 0..9, front to back

Line 8 move to the next digit

Choosing a Sort

IIT Madras Zanzibar

	Merge sort	Quicksort	Counting sort	Radix sort
works by	comparing	comparing	counting keys	digit by digit
time	O(n log n)	O(n log n) avg	O(n + k)	O(d (n + k))
extra memory	O(n)	O(log n)	O(n + k)	O(n + k)
stable?	yes	no	yes	yes
keys	any comparable	any comparable	whole 0..k, k small	fixed length digits or letters

? **Think:** 5 million nine-digit ID numbers: counting sort or radix sort? Why?

Hands-On: Sort Yourselves with Radix Sort

IIT Madras Zanzibar

64	27	91	38	52	15	70	83
46	29	61	94	35	58	17	

The activity (6 minutes)

1. Take a card. Pass 1: walk up in list order and put it at the END of the pile for your ONES digit (piles 0 to 9).
2. Collect the piles: pile 0 first, then 1, and so on; each pile from bottom to top.
3. Pass 2: repeat with the TENS digit, in the new order. Then collect: the cards are sorted.

In-Class Exercise (6 minutes, in pairs)

IIT Madras Zanzibar

IN-CLASS TASK: three passes, in buckets

Sort these with radix sort: 3 passes. Write each number in its bucket; paper worksheet 2.

329	457	657	839	436	720	355

0	1	2	3	4	5	6	7	8	9

Pass 1: ones. Pass 2: tens. Pass 3: hundreds. Collect after each pass.

Solution: The List After Each Pass

start	329	457	657	839	436	720	355
after pass 1 (ones)	720	355	436	457	657	329	839
after pass 2 (tens)	720	329	436	839	355	457	657
after pass 3 (hundreds)	329	355	436	457	657	720	839

Check: in pass 2, 355, 457 and 657 share tens digit 5 and keep the order from pass 1.

Summary and Exam Checkpoints

IIT Madras Zanzibar

Counting sort

count, running total, place from the right; O(n + k); stable

Radix sort

one stable pass per digit, last digit first; O(d (n + k))

Limit

both need whole-number keys; counting needs small k

Choose

small range: counting; long fixed-length keys: radix

Next

Wed 18 Nov: Binary trees and traversals

Thu 19 Nov: Binary search trees; Lab

Quiz 1: results and feedback this week

References and Further Reading

[1] Cormen, T. H., Leiserson, C. E., Rivest, R. L. & Stein, C. (2022). Introduction to Algorithms, 4th ed. MIT Press. Sections 8.1–8.3.

[2] Canning, J., Broder, A. & Lafore, R. (2022). Data Structures & Algorithms in Python. Addison-Wesley. Chapter 7.

[3] Sedgewick, R. & Wayne, K. (2011). Algorithms, 4th ed. Addison-Wesley. Section 5.1 (key-indexed counting, LSD radix sort).