

Algorithm Design & Problem Solving: Insertion Sort & Merge Sort



Contents



1

Insertion sort

2

Sorting algorithm demo

3

Merge sort



factorialgcd
logarithms
BigO
fibonacci
complexity
iteration
pseudocode
insertion-sort
arrays
Tower of Hanoi
recursion
selection-sort
bubble-sort

Insertion Sort



❖ How can this be sorted?

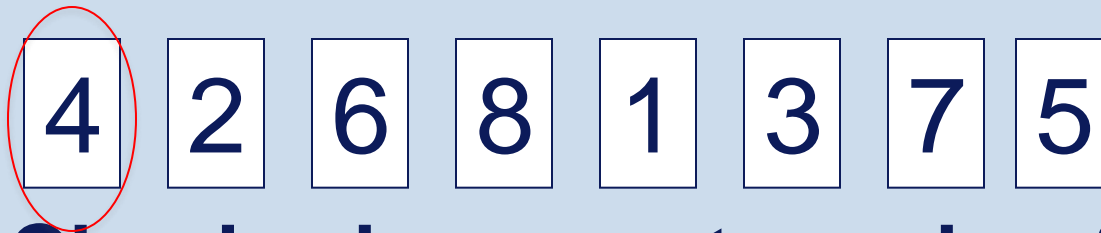
4	2	6	8	1	3	7	5
---	---	---	---	---	---	---	---

Insertion Sort



❖ What are the steps?

1. First number is regarded as sorted



1. Check where next number fits in with the numbers on left (sorted array), put it into that position



2. Revisit 2 until the entire array is sorted

Insertion Sort



❖ How efficient is it?

$$O(n^2)$$

❖ What does that mean?

Insertion Sort



INSERTION SORT

Best

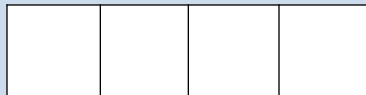
$\Omega(n)$

Average

$\Theta(n^2)$

Worst

$O(n^2)$



Array

Insertion Sort: The algorithm



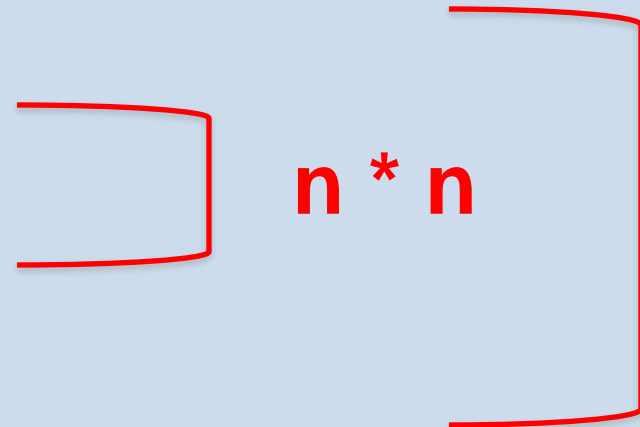
❖ If the performance of insertion sort is $O(n^2)$, this insinuates that there are two loops:

Loop 1

Loop 2

End Loop 2

End Loop 1



Insertion Sort: The algorithm



❖ What does Loop 2 do?

❖ What does Loop 1 do?

Loop 1

Loop 2

End Loop 2

End Loop 1

Insertion Sort: The algorithm



❖ What does Loop 2 do (inner loop)?

- Finds correct position of number



Repeats until correct position found

Insertion Sort: The algorithm



❖ What does Loop 1 do?

- Consider a new element to be put into the left hand side of the array



Repeats this n times

Insertion Sort: The algorithm



❖ Can we refine this description?

Repeat Loop 1 .. Times

Consider next highest index as part of left side of array, call it current

Repeat Loop ... times

Find position of current number

Make swaps needed

End Loop 2

Put current number in correct position

End Loop 1

Now write the pseudo code for this algorithm

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



Temp

Insertion Sort: The algorithm



Input: An array **A** storing **N** items

Output: **A** sorted in ascending order



Algorithm Insertion_Sort (A, N):

For $i = 1, i < N$ do {

$\text{current} = A[i]$

$j = i$

 Consider next element

 while $A[j-1] > \text{current}$

$A[j] = A[j-1]$

$j = j-1$

 Find correct position of current

 End while

$A[j] = \text{current}$

 Put current in correct position

End for

Algorithm Insertion_Sort (A, N):



```
For i = 1, i < N do {  
    current = A[i]  
    j = i  
    while A[j-1] > current AND ??  
        A[j] = A[j-1]  
        j = j-1  
    End while  
    A[j] = current  
End for
```


Sorting Algorithms



❖ **Sorting demo**

Toptal

<https://visualgo.net/en/sorting?slide=7>

Merge Sort



❖ Use the merge sort to sort these numbers

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

Merge Sort (Divide & Conquer)



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

Merge Sort (Divide & Conquer)



4 2 6 8 1 3 7 5

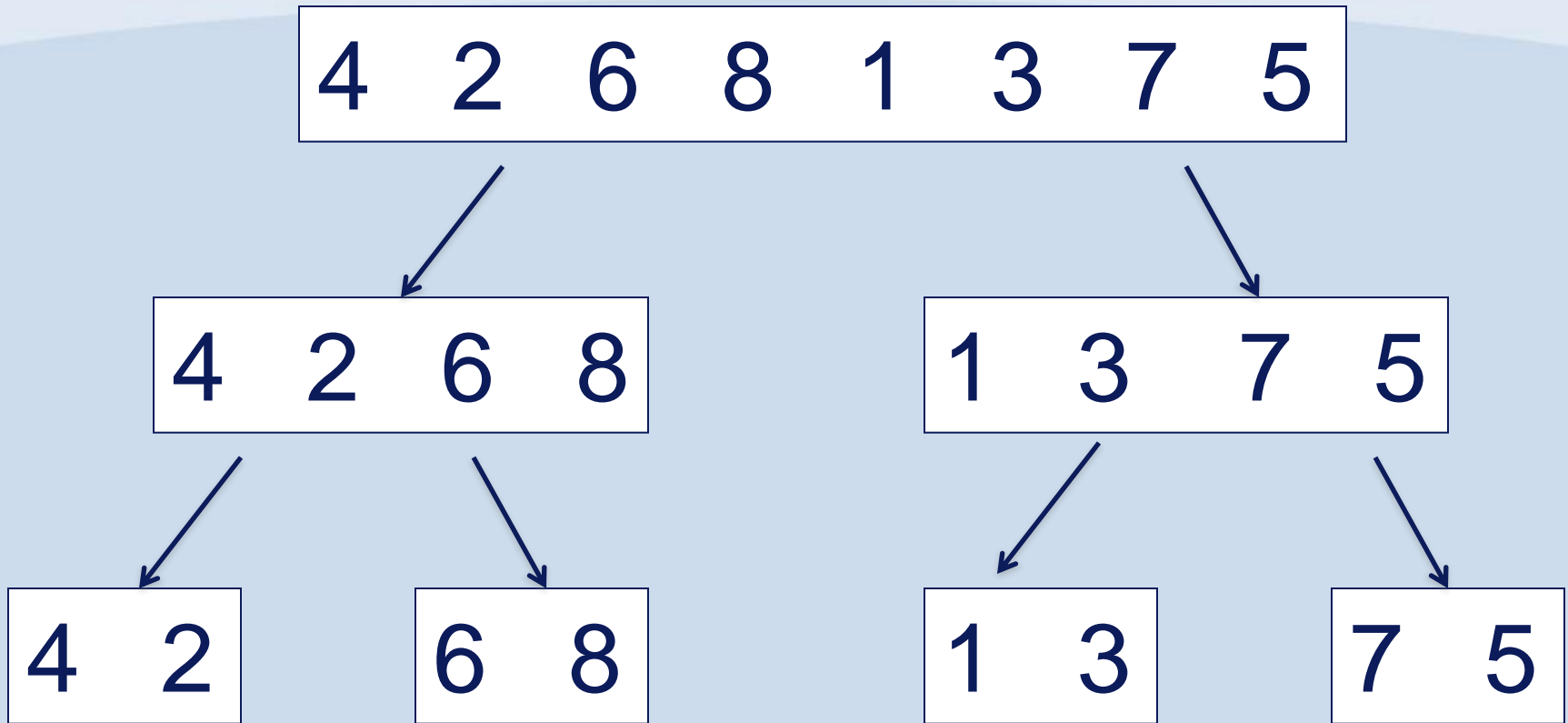


4 2 6 8

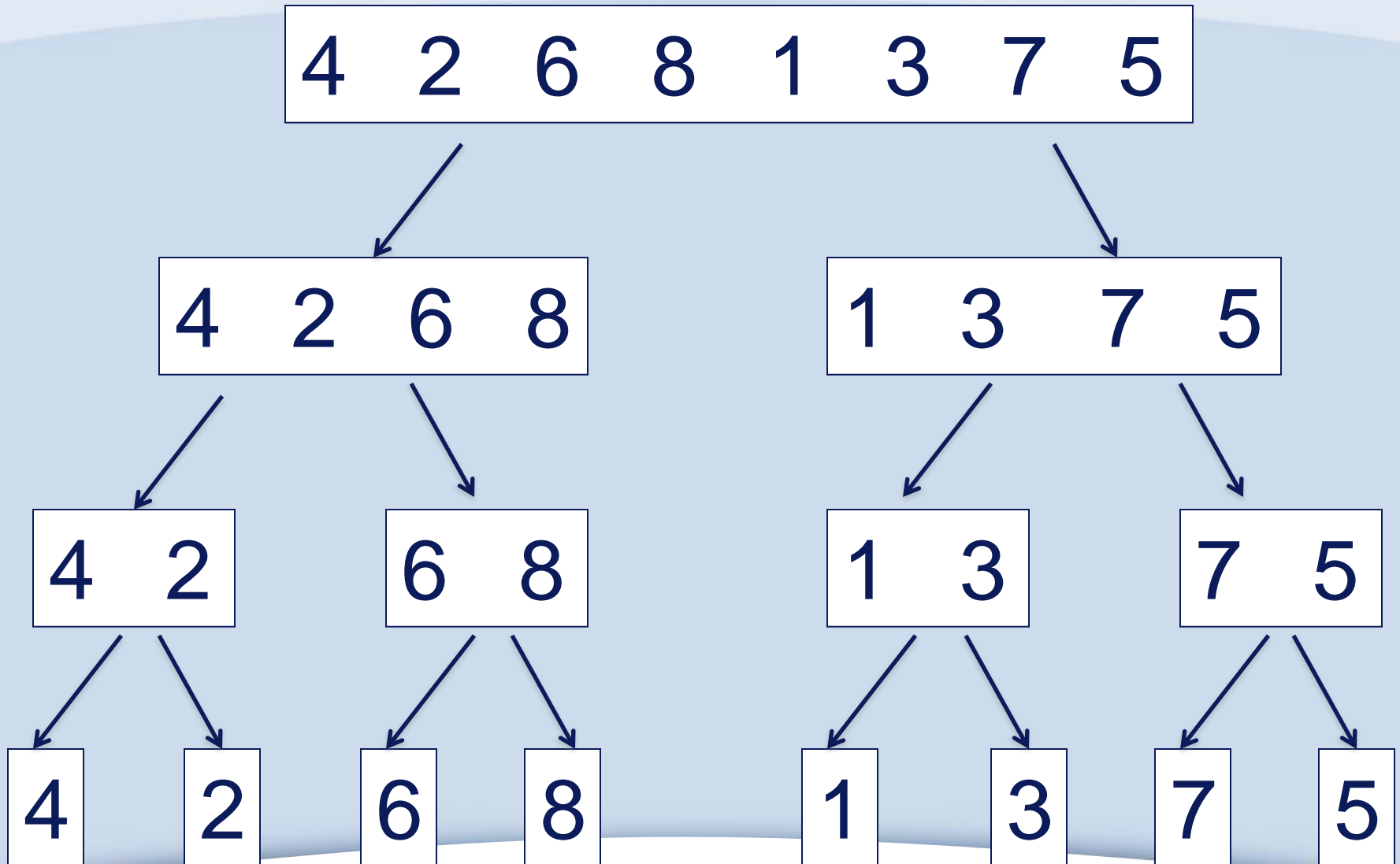


1 3 7 5

Merge Sort (Divide & Conquer)



Merge Sort (Divide & Conquer)

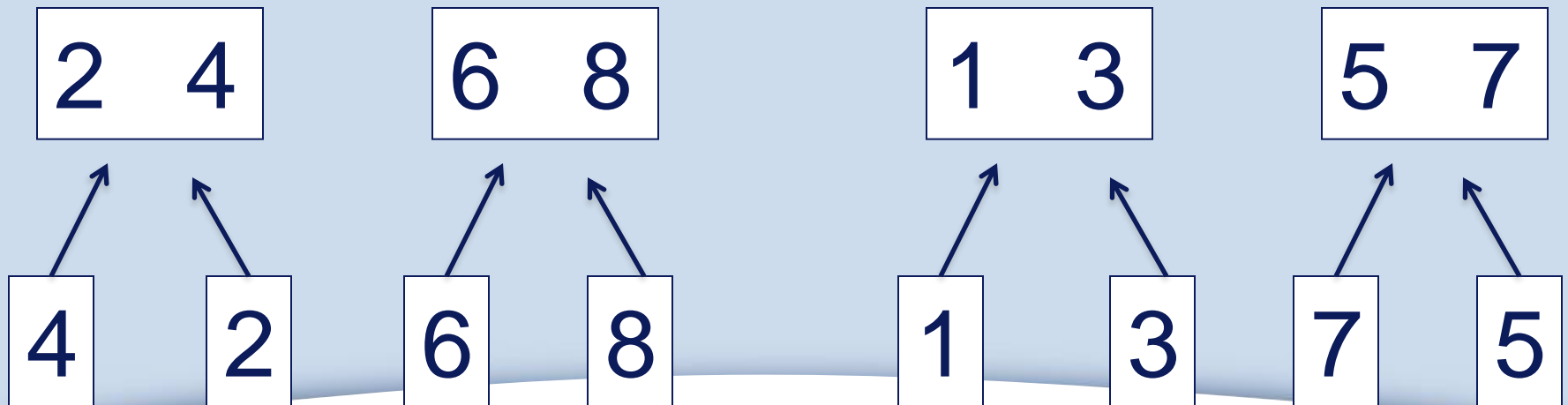


Merge Sort (Combine)

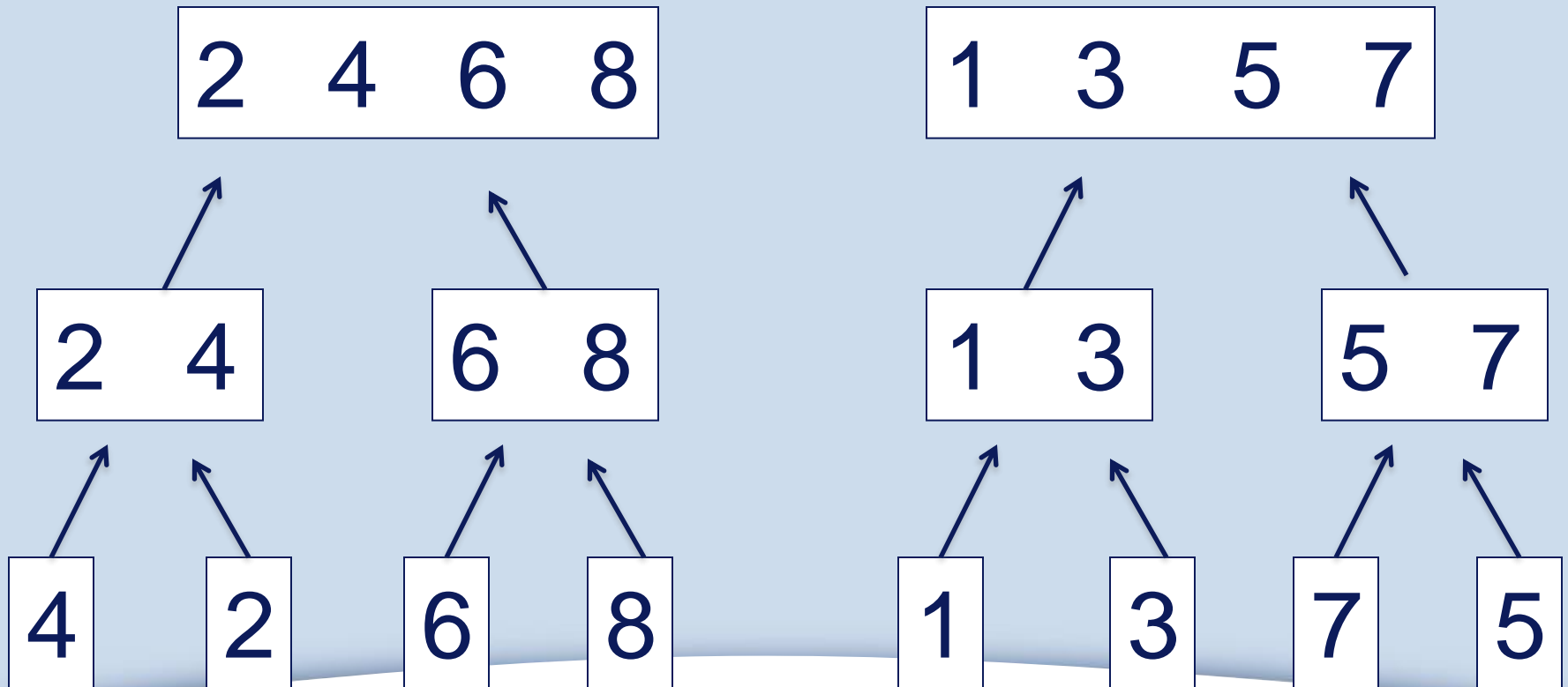


4	2	6	8	1	3	7	5
---	---	---	---	---	---	---	---

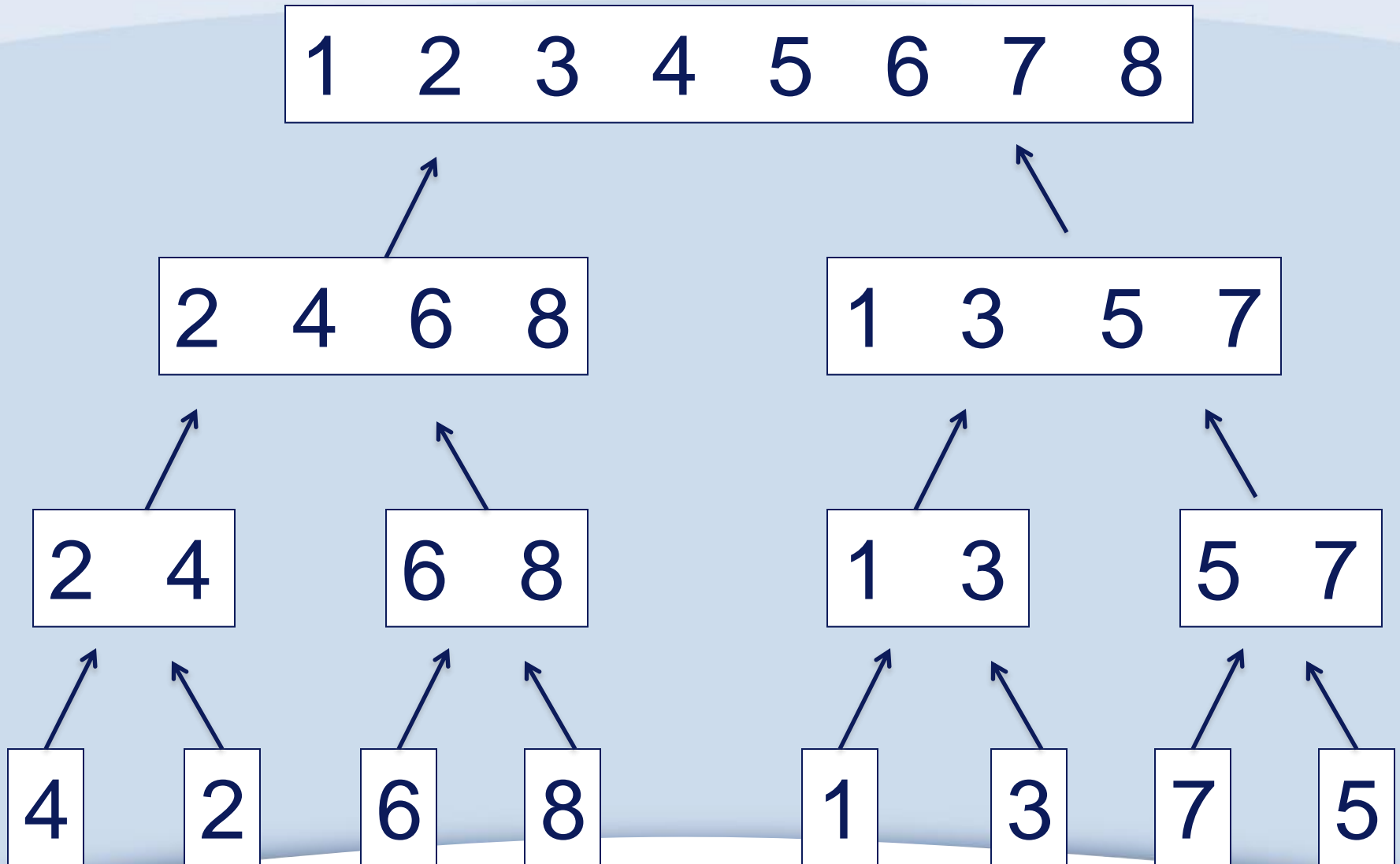
Merge Sort (Combine)



Merge Sort (Combine)



Merge Sort (Combine)



Thank You !

