CS302

Exercise Session 3

Intro to MPI: Parallel Merge Sort

Divide & Conquer

Efficient algorithmic approach

Divide

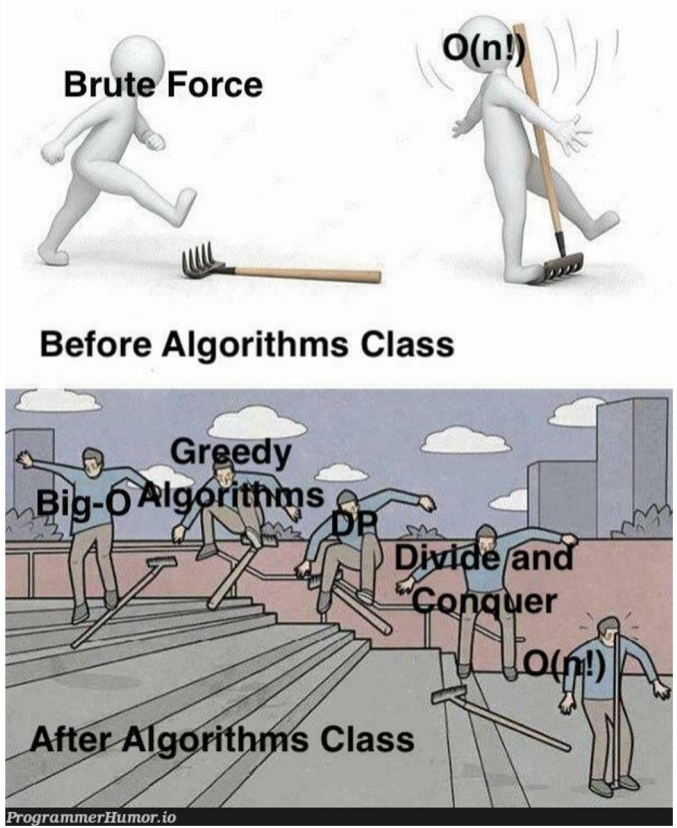
- the problem into smaller subproblems

Conquer

- the subproblems (solve them recursively)

Combine

- the subproblems for the final solution



Application: Merge Sort

Divide

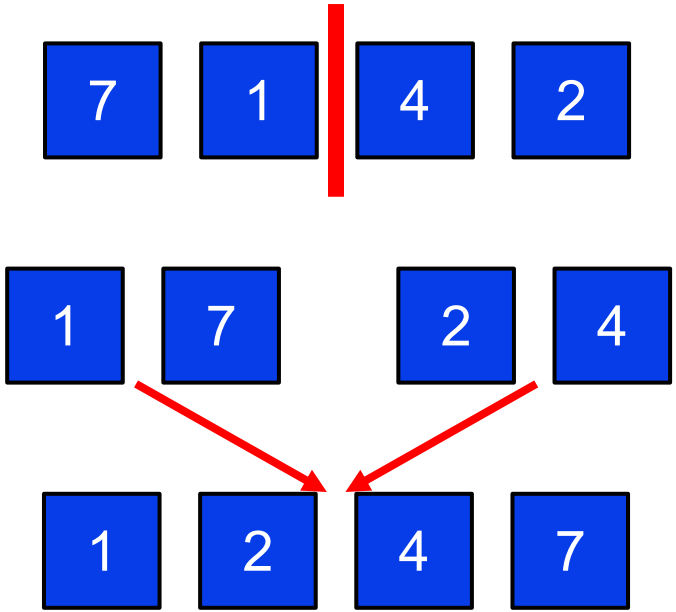
- Cut array in half

Conquer

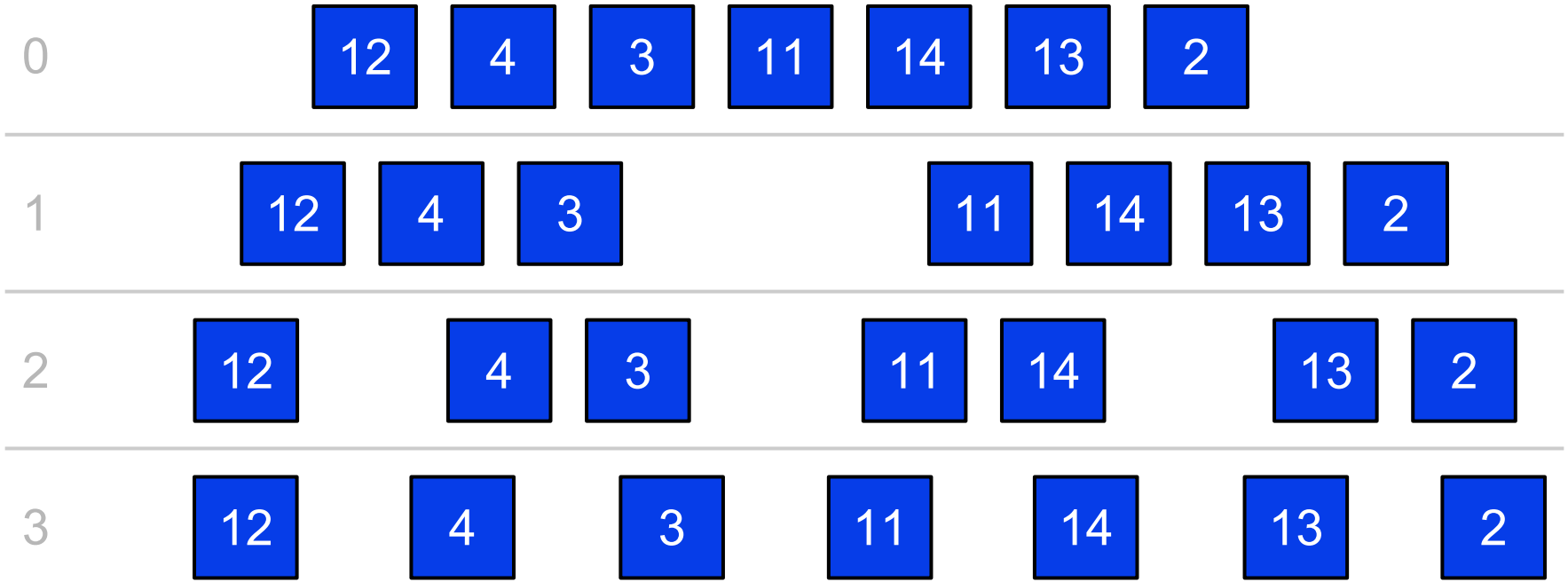
- Sort each half recursively

Combine

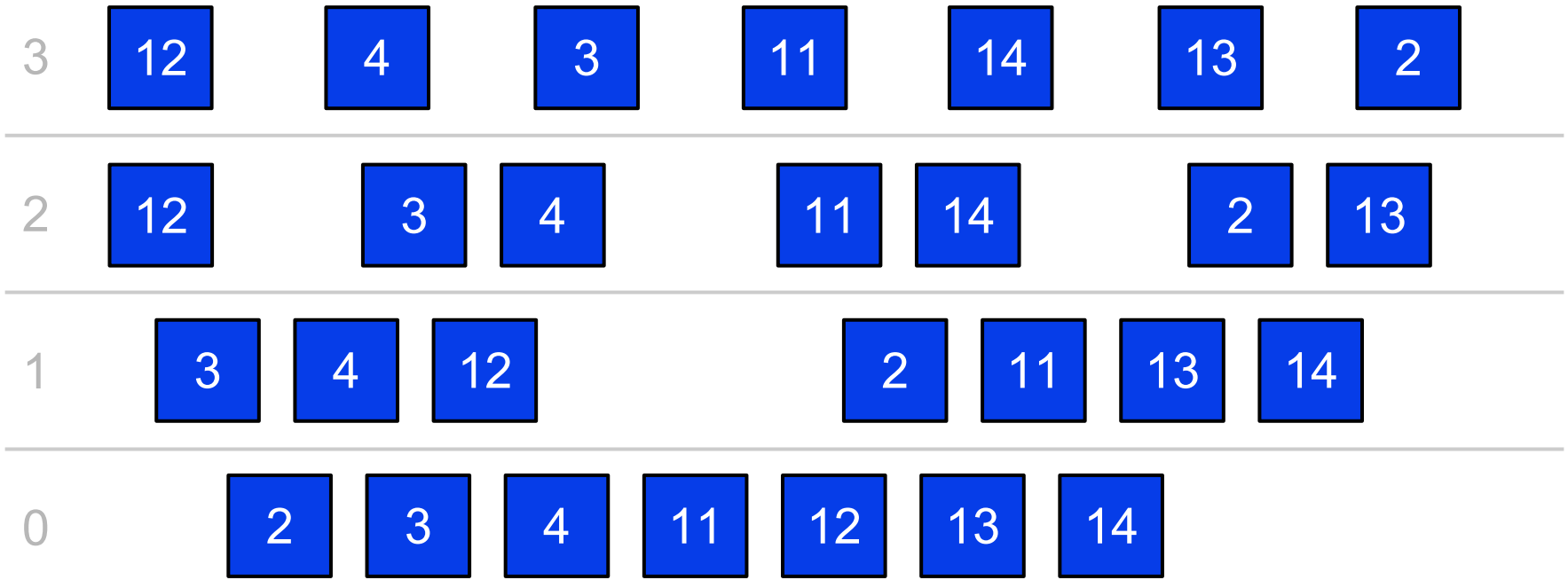
- Merge two halves together



Merge Sort Example (1)



Merge Sort Example (2)



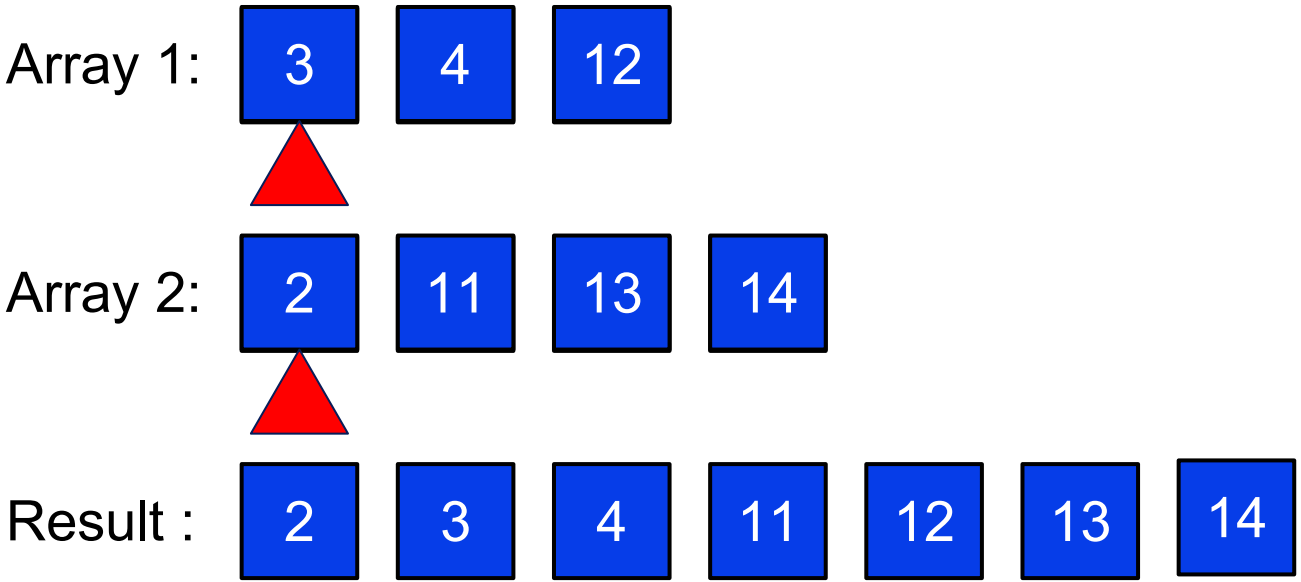
Merge Sort Example (3)

Let’s code!

Merge Sort Example (4)

- How to merge 2 sorted arrays?

  A: 2 pointers technique



Merge Sort Example (5)

Let’s implement it!

Parallel Merge Sort Using MPI (1)

- How to manage multiple processes together to work on huge arrays?
- Example: Sort 1’000’000’000 integers

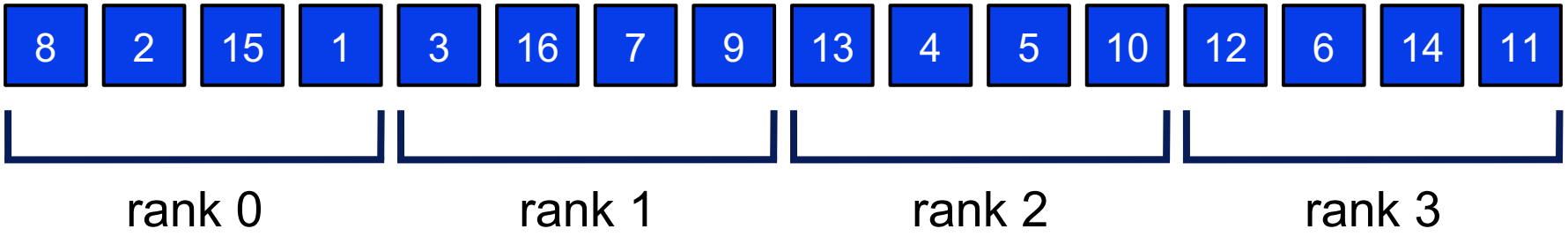
  Sequential version takes 258.4 seconds
- Can we do better?

  Spoiler: with 64 processes, ~12x speedup

(Ran on SCITAS)

Parallel Merge Sort Using MPI (2)

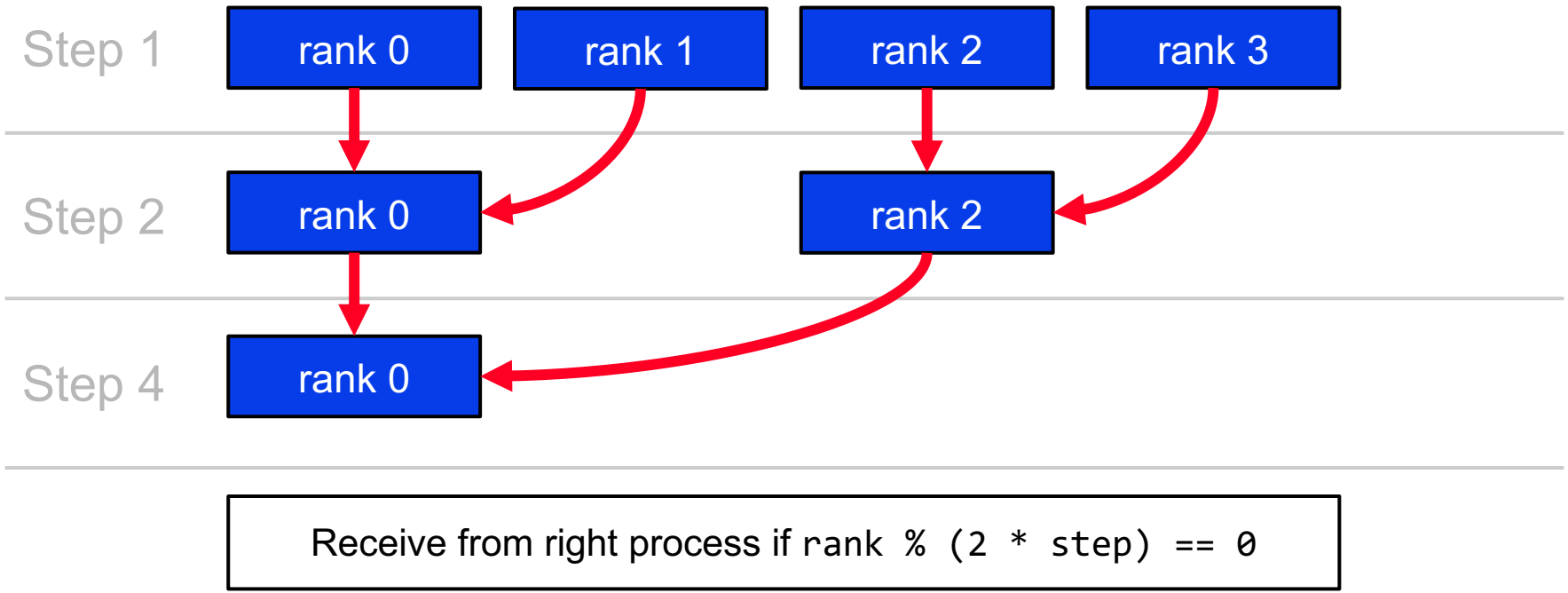
- Idea: split the array evenly across processes, sort locally the subarrays, and combine results back
- Example (with 4 processes)



- Split array evenly
- Sort locally
- Combine??

Parallel Merge Sort Using MPI (3)

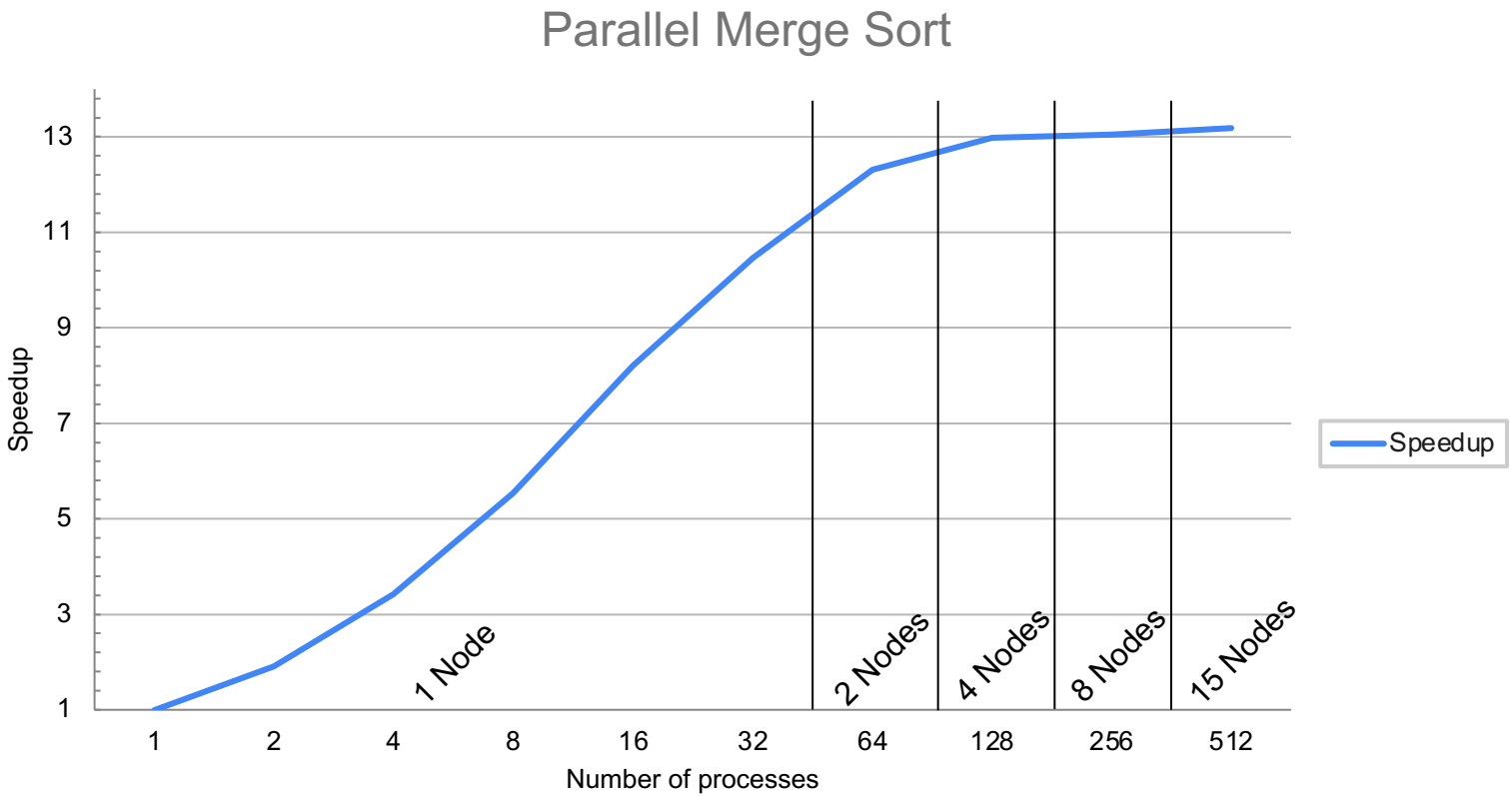
- How to combine arrays in parallel: Tree-like Merging



Merge Sort Example (5)

Let’s implement it!

Speedup


Parallel Merge Sort
Speedup
13
11
9
7
5
3
1
1 Node
2 Nodes
4 Nodes
8 Nodes
15 Nodes
1
2
4
8
16
32
64
128
256
512
Number of processes
Speedup

Merge Sort Example (5)

Questions?