

# 5. Divide & conquer

---

(just a brief introduction)

# Divide-and-Conquer

## Divide-and-conquer.

- Break up problem into several parts.
- Solve each part recursively.
- Combine solutions to sub-problems into overall solution.

## Most common usage.

- Break up problem of size  $n$  into **two** equal parts of size  $\frac{1}{2}n$ .
- Solve two parts recursively.
- Combine two solutions into overall solution in **linear time**.

## Consequence.

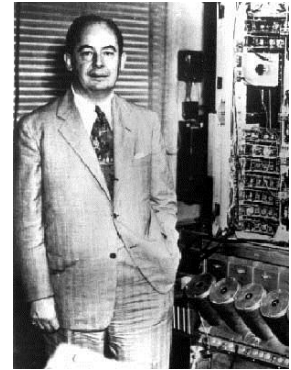
- Brute force:  $n^2$ .
- Divide-and-conquer:  $n \log n$ .

Divide et impera.  
Veni, vidi, vici.  
- *Julius Caesar*

Q. Give an example of a divide-and-conquer algorithm.

# Mergesort

Q. How does mergesort work?



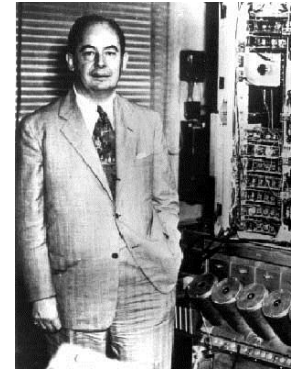
Jon von Neumann (1945)  
[http://en.wikipedia.org/wiki/John\\_von\\_Neumann](http://en.wikipedia.org/wiki/John_von_Neumann)

A L G O R I T H M S

# Mergesort

## Mergesort.

- Divide array into two halves.
- Recursively sort each half.
- Merge two halves to make sorted whole.



Jon von Neumann (1945)

[http://en.wikipedia.org/wiki/John\\_von\\_Neumann](http://en.wikipedia.org/wiki/John_von_Neumann)

A L G O R I T H M S

A L G O R                      I T H M S

divide  $O(1)$

A G L O R                      H I M S T

sort  $2T(n/2)$

A G H I L M O R S T

merge  $O(n)$

# Merging

**Merging.** Combine two pre-sorted lists into a sorted whole.

How to merge efficiently?



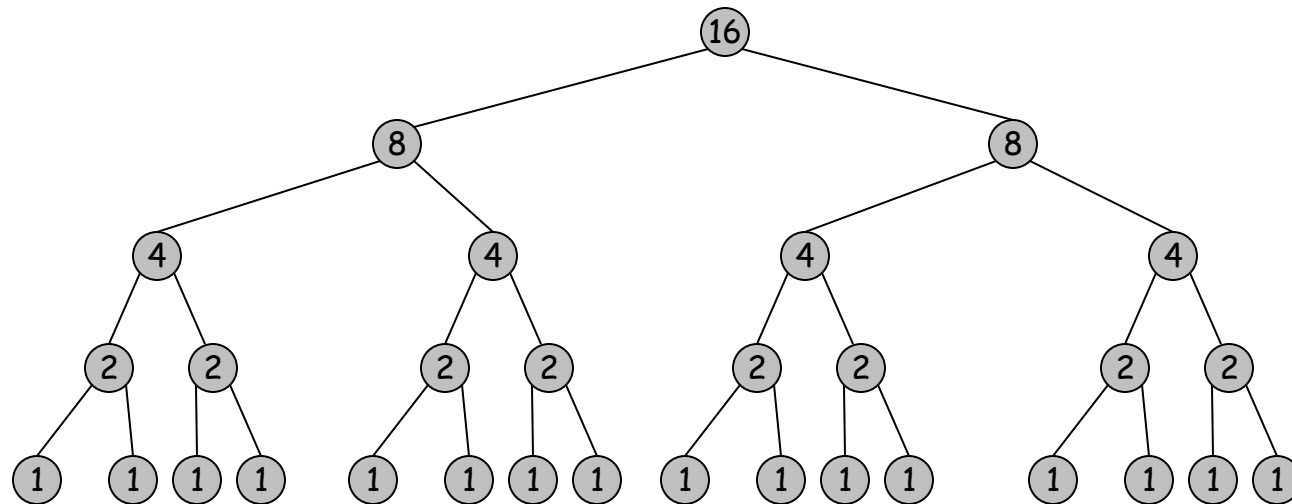
- Linear number of comparisons.
- Use temporary array.



**Challenge for the bored.** In-place merge. [Kronrod, 1969]

↑  
using only a constant amount of extra storage

# Call graph of Mergesort



Call graph of Mergesort of a string of length 16  
(the nodes contain size of substring)