



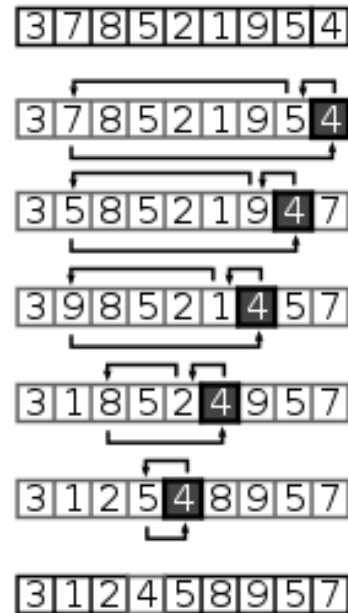
SIMON FRASER UNIVERSITY  
ENGAGING THE WORLD

# CMPT 125 - Introduction to Computing Science and Programming II

## Sorting Algorithms

- Some examples of popular sorting algorithms:
  - Insertion sort
  - Selection sort
  - Merge sort
  - Quicksort (next slide)
- Comparison sorts/Non-comparison sorts
- In place sort vs out of place
- There are also other algorithms:
  - Heapsort, Bubble sort, Bucket sort, Counting sort, Radix sort, etc.
- For some visualizations, please see the following link:
  - <https://visualgo.net>

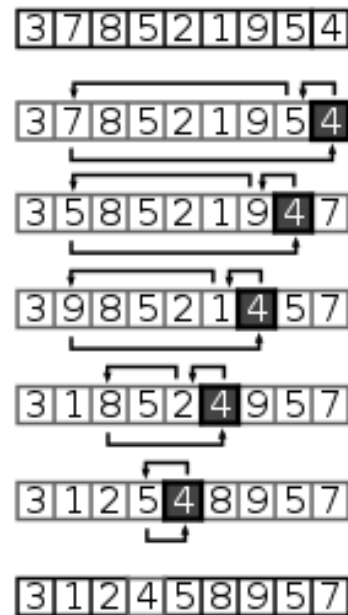
- A divide and conquer algorithm similar to MergeSort
- Idea:
  - Pick a pivot and partition the array based on that
    - Fixed pivot (first or last)
    - Random pivot



Source:

<https://en.wikipedia.org/wiki/Quicksort>

- A divide and conquer algorithm similar to MergeSort
- Idea:
  - Pick a pivot and partition the array based on that
    - Fixed pivot (first or last)
    - Random pivot
- Usually two functions:
  - Quicksort
  - Partition (helper function for Quicksort)
- Worst case:
  - Pivot always be the smallest or the largest element
- Best case:
  - Pivot always be the middle element



Source:

<https://en.wikipedia.org/wiki/Quicksort>

- Make sure you understand the different sorting algorithms
- Compare times of different algorithms using `gettimeofday`. Change the input length by modifying `LENGTH`
- For each algorithm, try to find the best and the worst inputs considering the running time.
- Learn the syntax of `qsort`. Note that `qsort` uses function pointers we discussed in class
- Implement `merge_sort` we saw in class, and compare its running time to other algorithms