**Chapter 11-3 – Faster Sorting Algorithms**

Internal Sort – collection of data fits in memory

External Sort – collection of data does not fit in memory

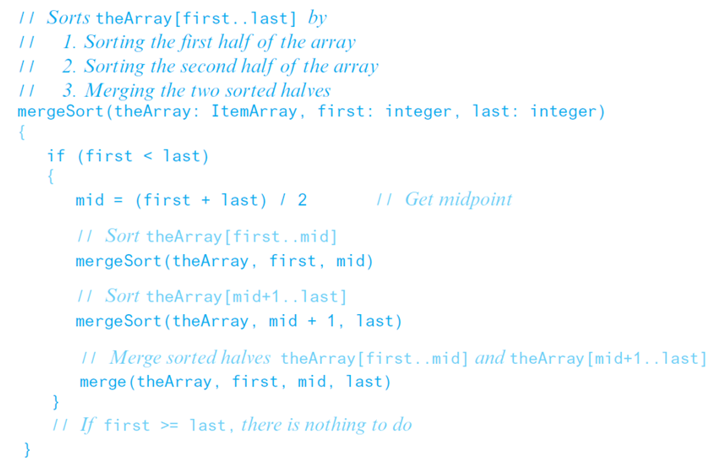
* Must reside on secondary storage

1. **Merge Sort – Divide and Conquer sorting algorithm**

**Divide**: divide the n-element sequence to be sorted into two subsequences of n/2 elements each

**Conquer**: sort the two subsequences recursively using merge sort. IF the length of a sequence is 1, do nothing since it is already in order.

**Combine**: merge two sorted subsequences to produce the sorted answer.



1. **The Quick Sort**

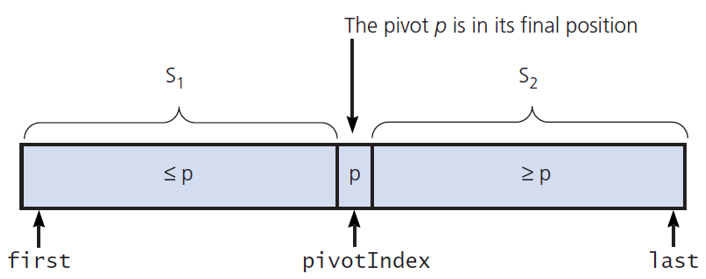
* Another divide-and-conquer algorithm



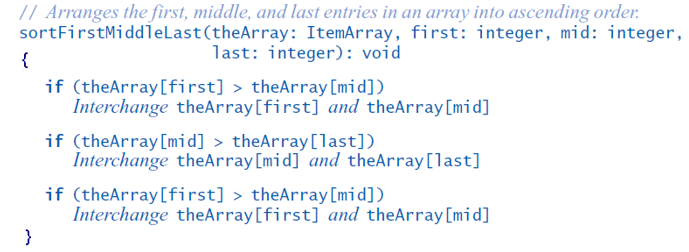
* Partitions an array into items that are
  + Less than or equal to the pivot and
  + Those that are greater than or equal to the pivot
* Partitioning places pivot in its correct position within the array

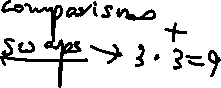


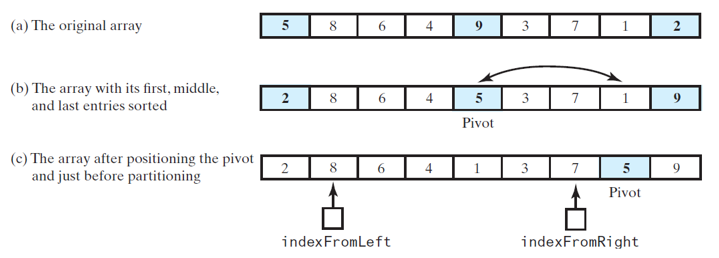
* + Place chosen pivot in theArray[last] before partitioning



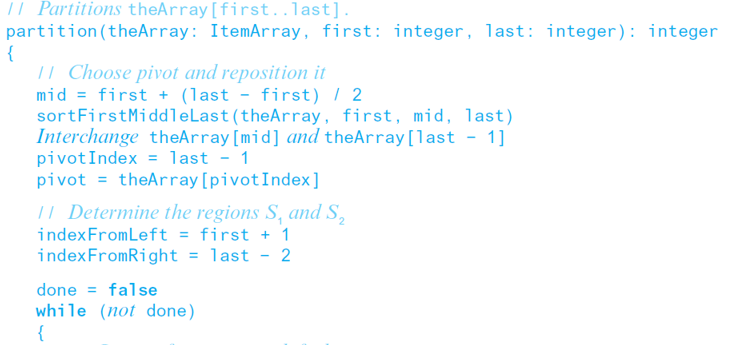
Median-of-three pivot selection

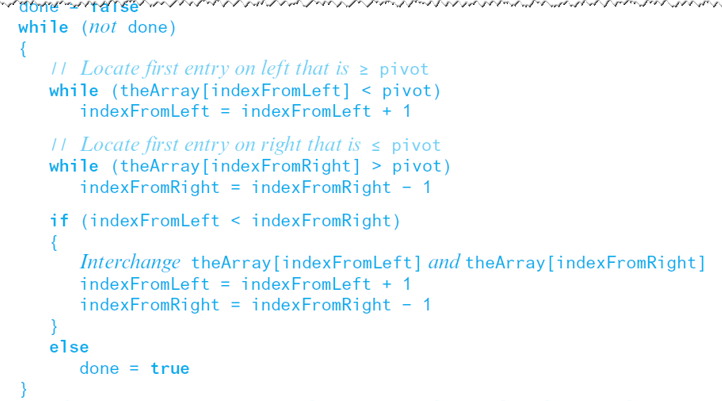
****

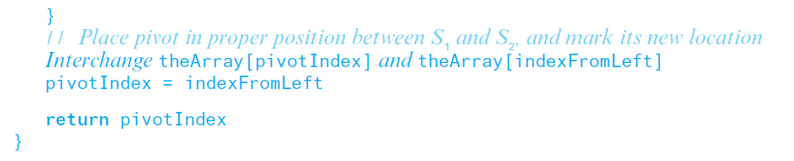


****



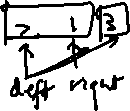
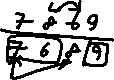
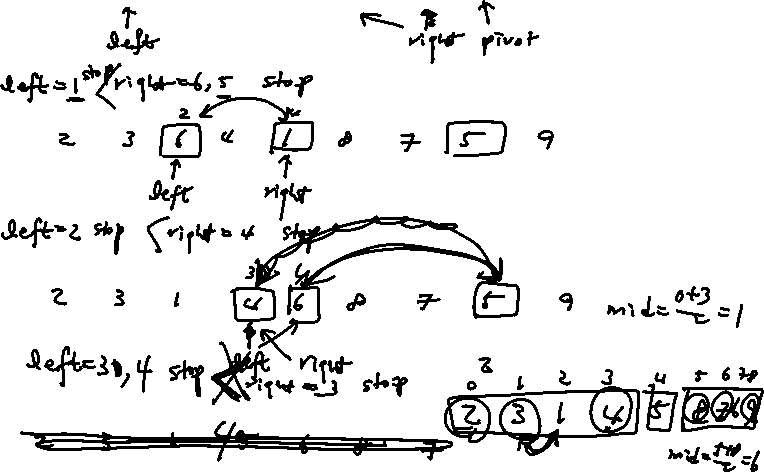




****



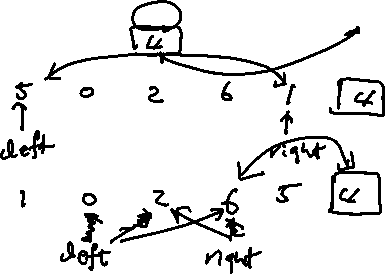
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | 8 | 6 | 4 | 1 | 3 | 7 | 5 | 9 |

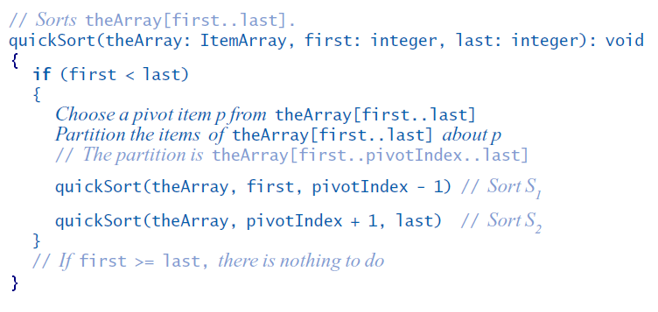


**Example** A partitioning of an array during a quick sort



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

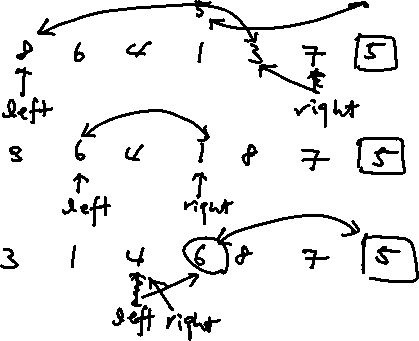




**Example** Trace the quicksort’s partitioning algorithm as it partitions the following array



|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 5 | 8 | 6 | 4 | 9 | 3 | 7 | 1 | 2 |



**Example)** Trace the quicksort’s portioning algorithm as it partitions the following array

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 38 | 16 | 40 | 39 | 12 | 27 |

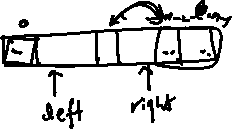
**Analysis**

Partitioning – O(n)

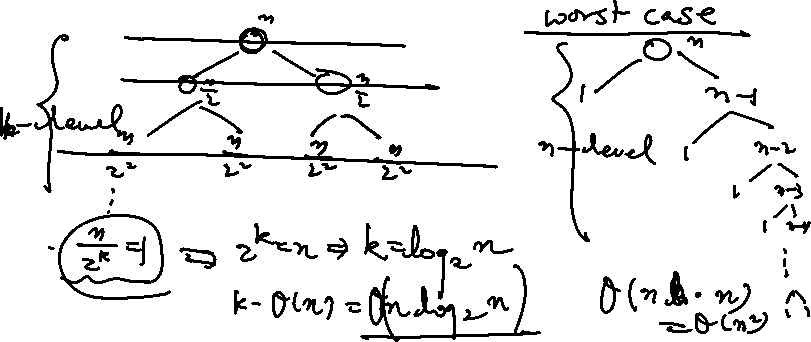
1. Find pivot: median –of-three selection



1. partitioning

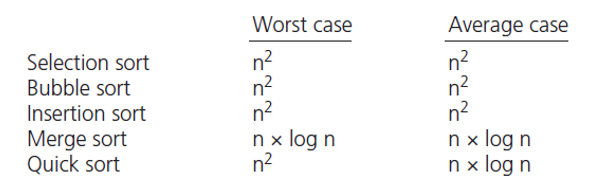


Average case



Worst case

**A Comparison of sorting algorithms**

****

**Quiz**

1. Trace the merge sort algorithm as it sorts the following array into ascending order.

21, 83, 45, 20, 66, 31

(2) Trace the quick sort algorithm as it sorts the following array into ascending order**.**

21, 83, 45, 20, 66, 13, 17