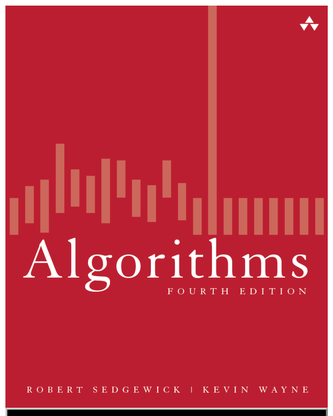

2. Sorting

Overview. Sorting is the process of rearranging a sequence of objects so as to put them in some logical order. Sorting plays a major role in commercial data processing and in modern scientific computing. Applications abound in transaction processing, combinatorial optimization, astrophysics, molecular dynamics, linguistics, genomics, weather prediction, and many other fields.

In this chapter, we consider several classical sorting methods and an efficient implementation of a fundamental data type known as the priority queue. We discuss the theoretical basis for comparing sorting algorithms and conclude the chapter with a survey of applications of sorting and priority-queue algorithms.

- *2.1 Elementary Sorts* introduces selection sort, insertion sort, and shellsort.
- *2.2 Mergesort* describes megesort, a sorting algorithm that is guaranteed to run in linearithmic time.
- *2.3 Quicksort* describes quicksort, which is used more widely than any other sorting algorithm.
- *2.4 Priority Queues* introduces the priority queue data type and an efficient implementation using a binary heap. It also introdues heapsort.
- *2.5 Applications* describes applications of sorting, including using alternate orderings, selection, the system sort, and stability.

Java programs in this chapter. Below is a list of Java programs in this chapter. Click on the program name to access the Java code; click on the reference number for a brief description; read the textbook for a full discussion.

REF	PROGRAM	DESCRIPTION / JAVADOC
2.1	Insertion.java	insertion sort
-	InsertionX.java	insertion sort (optimized)
-	BinaryInsertion.java	binary insertion sort
2.2	Selection.java	selection sort
2.3	Shell.java	shellsort
2.4	Merge.java	top-down mergesort
-	MergeBU.java	bottom-up mergesort
-	MergeX.java	optimized mergesort
-	Inversions.java	number of inversions
2.5	Quick.java	quicksort
-	Quick3way.java	quicksort with 3-way partitioning
-	QuickX.java	optimized 2-way quicksort
-	QuickBentleyMcIlroy.java	optimized 3-way quicksort
-	TopM.java	priority queue client
2.6	MaxPQ.java	max heap priority queue
-	MinPQ.java	min heap priority queue
-	IndexMinPQ.java	index min heap priority queue
-	IndexMaxPQ.java	index max heap priority queue
-	Multiway.java	multiway merge
2.7	Heap.java	heapsort

Sorting demos. Below are some interesting sorting demos.

- Sorting Algorithm Animations by David Martin.
- Audibilization and Visualization of Sorting Algorithms by Timo Bingmann.
- The Sound of Quicksort.
- Robot visualizations of quicksort and mergesort.
- Sorting visualizations by Carlo Zapponi, using inversion count as a measure of progress.

Last modified on August 26, 2016.