Quicksort

Weiss, from section 8.6 to the end of chapter 8 (pages 364–391)

I am indebted to my colleagues, John David Stone and Peter-Michael Osera, for assistance with this handout.
This work is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License. To view a copy of this license, visit https://creativecommons.org/licenses/by-sa/4.0/
or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.
public static <AnyType extends Comparable<? super AnyType>>
void insertionSort( AnyType [ ] a )
{
    for( int p = 1; p < a.length; p++ )
    {
        AnyType tmp = a[ p ];
        int j = p;

        for( ; j > 0 && tmp.compareTo( a[ j - 1 ] ) < 0; j-- )
            a[ j ] = a[ j - 1 ];
        a[ j ] = tmp;
    }
}
Insertion Sorting (cont.)

- The running time of insertion sort depends on the number of inversions.
  - For a pair \((i, j)\): \(i < j\) but \(A_i > A_j\).
- There are \(N(N - 1)/4\) inversions.
- The running time of any algorithm that performs sorting by exchanging adjacent elements is \(\Omega(N^2)\).
- Because of nested loops, the insertion sort takes \(O(N^2)\) time.
Quicksort

\[
\begin{array}{|c|c|c|}
\hline
\text{mid} & \text{[ ]} & \text{[ ]} \\
\hline
\end{array}
\begin{array}{|c|c|c|}
\hline
\text{mid} & \text{< mid} & \text{[ ]} \\
\hline
\text{mid} & \text{[ ]} & \text{> mid} \\
\hline
\end{array}
\]

\[
\begin{array}{c}
3, 9, 2, 8, 6, 4, 1, 7, 5 \\
\end{array}
\]

\[
\begin{array}{c}
3, 9, 2, 8, 5, 4, 1, 7, 6 \\
\end{array}
\]

\[
\begin{array}{c}
3, 9, 2, 8, 5, 4, 1, 7, 6 \\
\end{array}
\]

\[
\begin{array}{c}
3, 9, 2, 8, 5, 4, 1, 7, 6 \\
\end{array}
\]

\[
\begin{array}{c}
3, 9, 2, 8, 5, 4, 1, 7, 6 \\
\end{array}
\]

\[
\begin{array}{c}
3, 9, 2, 8, 5, 4, 1, 7, 6 \\
\end{array}
\]
**Quicksort**

[3, 1, 2, 8, 5, 4, 9, 7, 6]

[3, 1, 2, 8, 5, 4, 9, 7, 6]

[3, 1, 2, 8, 5, 4, 9, 7, 6]

[3, 1, 2, 8, 5, 4, 9, 7, 6]

[3, 1, 2, 4, 5, 8, 9, 7, 6]

[3, 1, 2, 4, 5, 8, 9, 7, 6]

[3, 1, 2, 4, 5, 6, 9, 7, 8]

< 6 > 6

- Choosing pivot impacts the running time of quicksort
- Choose the middle element
- Median-of-three (median of the first, middle, and last elements)