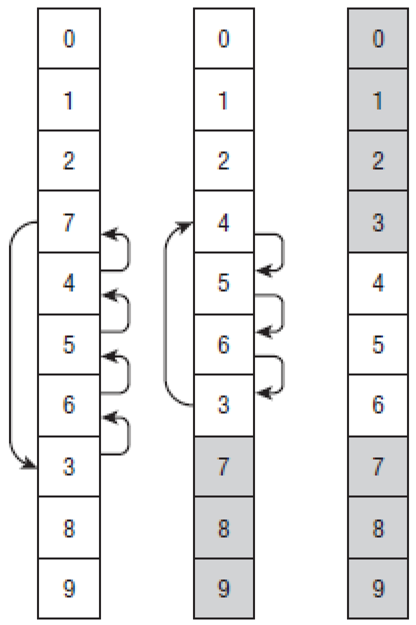
Sorting, Part 1



Agenda

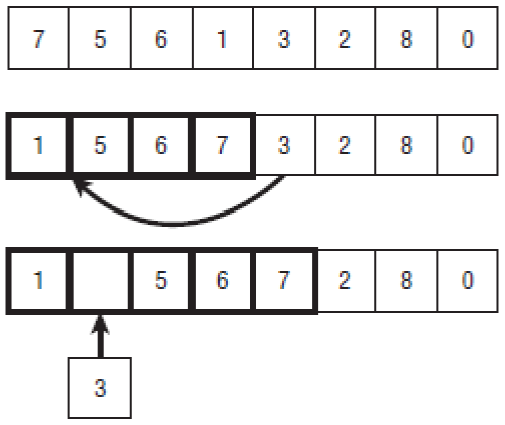
- O(N^2) Algorithms
- O(N log N) Algorithms
- Summary
- Exercises

$O(N^2)$ Algorithms

- Insertionsort
- Selectionsort
- Bubblesort

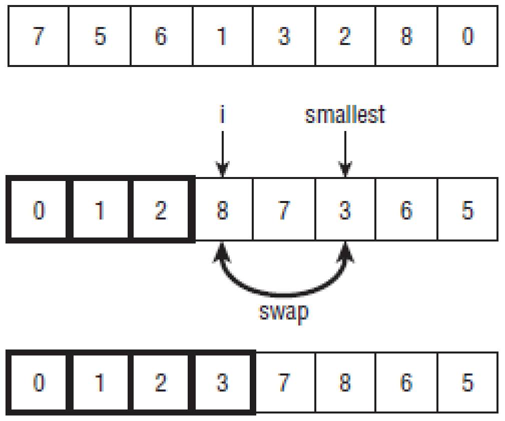
Insertionsort

```
Insertionsort(Data: values[])
    For i = 0 To <length of values> - 1
        // Move item i into position in the sorted part of the array.
        <Find the first index j where j < i and values[j] > values[i].>
        <Move the item into position j.>
    Next i
End Insertionsort
```



Selectionsort

```
Selectionsort(Data: values[])
    For i = 0 To <length of values> - 1
        // Find the item that belongs in position i.
        <Find the smallest item with index j >= i.>
        <Swap values[i] and values[j].>
    Next i
End Selectionsort
```

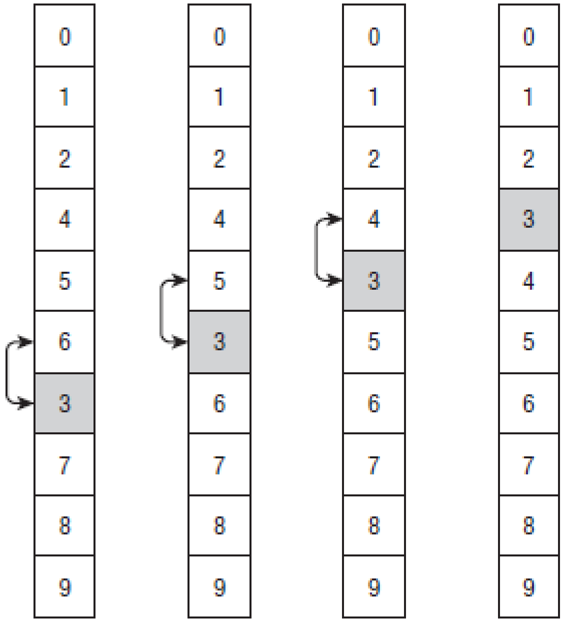


Bubblesort

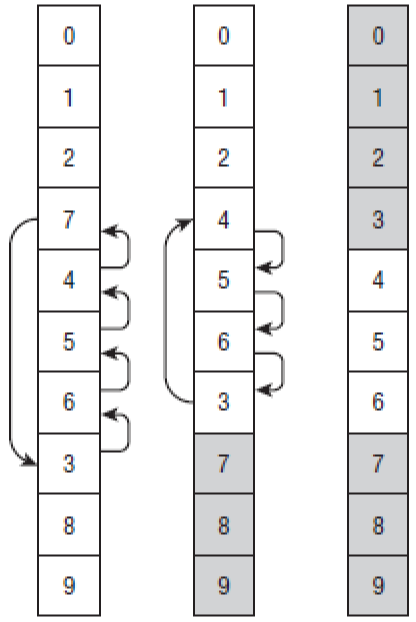
```
Bubblesort(Data: values[])
    // Repeat until the array is sorted.
    Boolean: not_sorted = True
    While (not_sorted)
        // Assume we won't find a pair to swap.
        not_sorted = False

        // Search the array for adjacent items that
        // are out of order.
        For i = 0 To <length of values> - 1
            // See if items i and i - 1
            // are out of order.
            If (values[i] < values[i - 1]) Then
                <Swap them.>

                // The array isn't sorted after all.
                not_sorted = True
            End If
        Next i
    End While
End Bubblesort
```



Improved Bubblesort

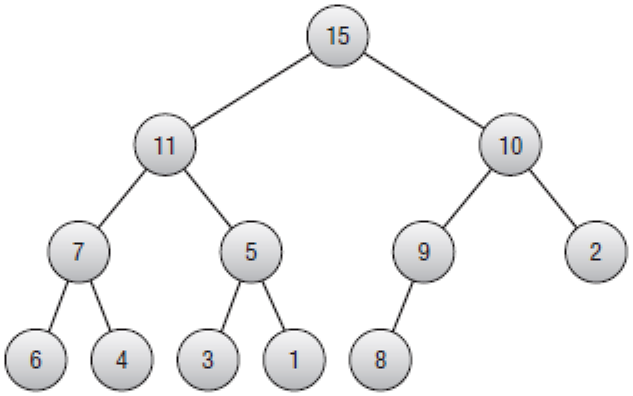


O(N log N) Algorithms

- Heapsort

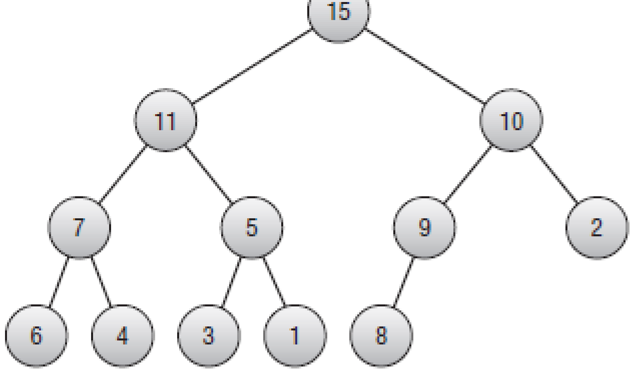
Heapsort

- Stores values in a tree data structure called a heap



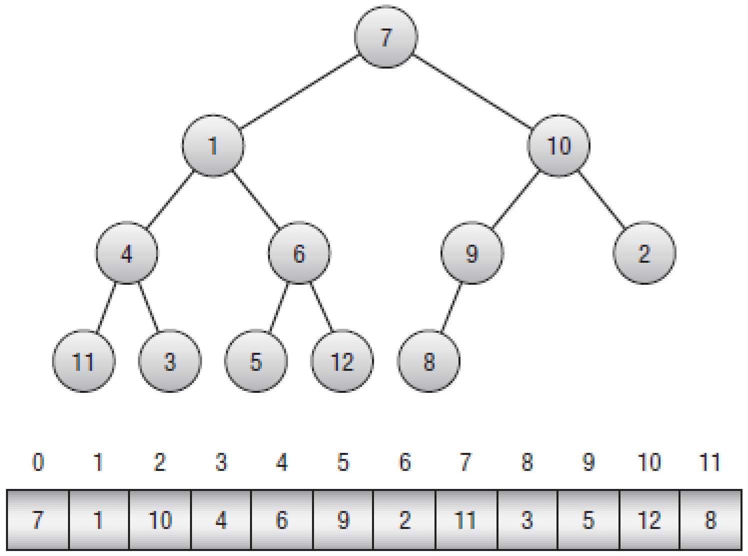
Heap

- Every node's value is at least as large as the values of its children



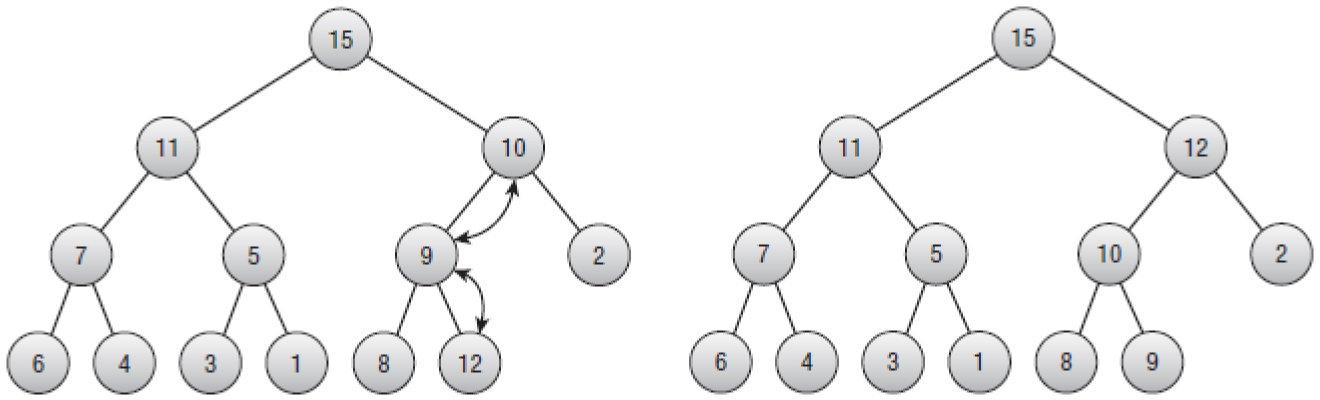
Heapsort Storage

- Storing a complete binary tree in an array



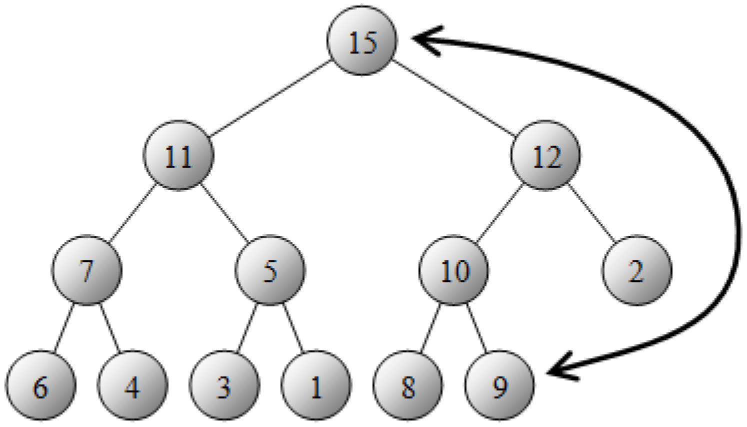
Adding to a Heap

- Add the value at the end
- Move it up to restore the heap property

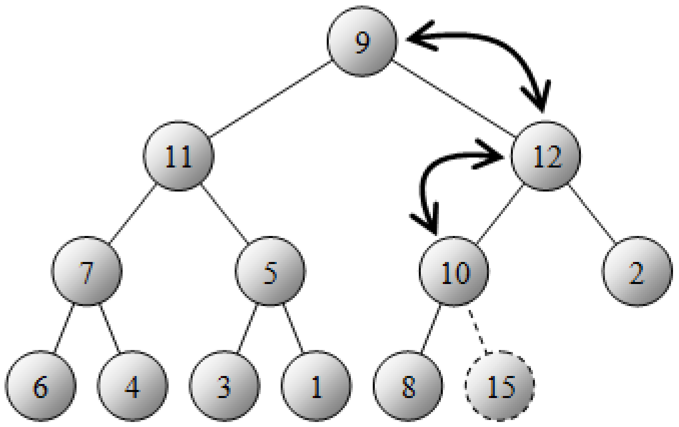


Removing from a Heap

- Swap the first and last values
- Remove the last position from the heap



- Push the top value down to restore the heap property



Implementing Heapsort

- Make a heap
- Repeat:
  - Remove the first item from the heap
  - Restore the heap property

Summary

- O(N^2) Algorithms
  - Insertionsort
  - Selectionsort
  - Bubblesort
- O(N log N) Algorithms
  - Heapsort
    - Heaps
    - Storing complete binary trees
    - Adding to a heap
    - Removing from a heap

Exercises

- Chapter 6 Exercises 1 – 10.
- Bonus: Chapter 6 Exercises 11 – 12.
- Bonus: Chapter 6 Exercises 13 – 14.
- Read *Essential Algorithms, 2e* Chapter 6 pages 145 – 162. (The rest of Chapter 6.)