James Tessitore HeapSortNotes

* Heap Sort
  + Considered BigO(log2N)
  + Each heap starts with a root, and then branches off into two elements, which also branch off into two elements and so on.
  + Sorts a heap by:
    - Take root element off the heap, and put it in its place
    - Reheap the remaining elements (put next largest element in loop position)
    - Repeat until there are no more elements
  + Assume there are no duplicate values
  + Two Types:
    - Min heap
      * value in location n is < than either of its children
    - Max heap
      * value in location n is > than either of its children
  + **To sort a min heap**

Array of values

Index: 1 2 3 4 5

Value: 3

3

Index: 1 2 3 4 5

Value: 3 2

3 2

2 Compare 2 and 3 - Switch 3

Index: 1 2 3 4 5

Value: 2 3 -1

2 -1

3 -1 Compare 2 and -1 - Switch 3 2

Index: 1 2 3 4 5

Value: -1 3 2 10

-1

3 2

10 Compare 10 and 3 - No Switch

Index: 1 2 3 4 5

Value: -1 3 2 10 6

-1

3 2

10 6 Compare 6 and 3 – No Switch

* + **To sort a max heap**

Index: 1 2 3 4 5

Value: 10

10

Index: 1 2 3 4 5

Value: 10 12

10 12

12 Compare and 10 and 12 – Switch 10

Index: 1 2 3 4 5

Value: 12 10 -5

12

10 -5 Compare -5 and 12 – No switch

Index: 1 2 3 4 5

Value: 12 10 -5 20

12 12 20

10 -5 20 -5 12 -5

20 Compare 20 and 10 – Switch 10 Compare 20 and 12 – Switch 10

Index: 1 2 3 4 5

Value: 20 12 -5 10 15

20 20

12 -5 15 -5

10 15 Compare 15 and 12 – Switch 10 12

Index: 1 2 3 4 5

Value: 20 15 -5 10 12