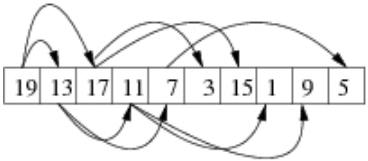
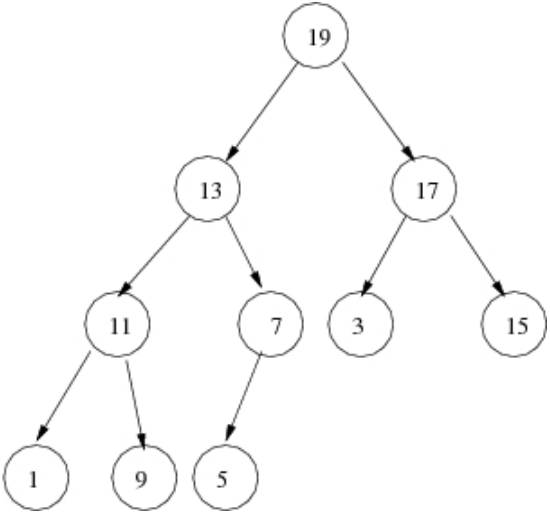
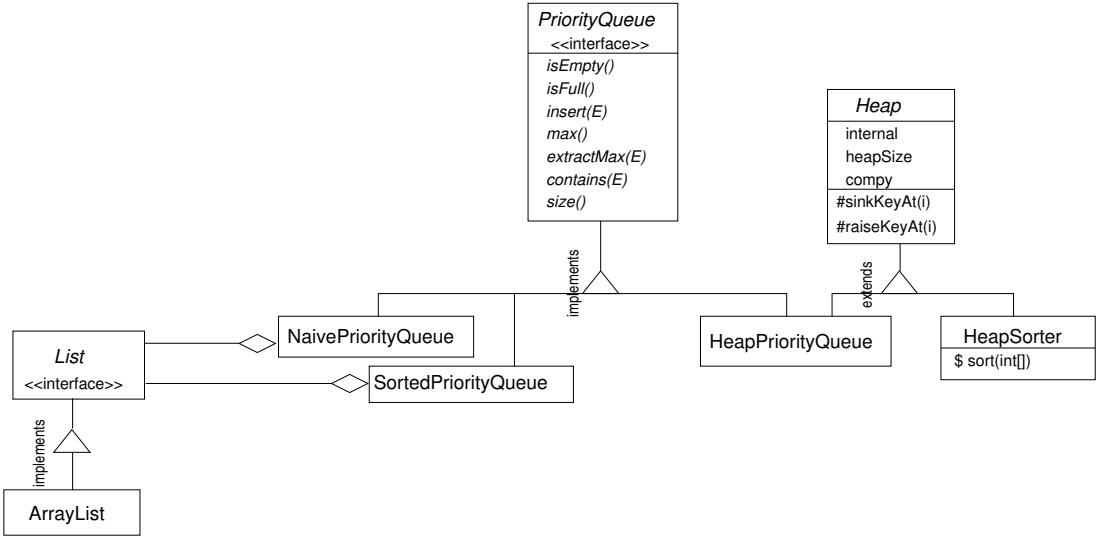
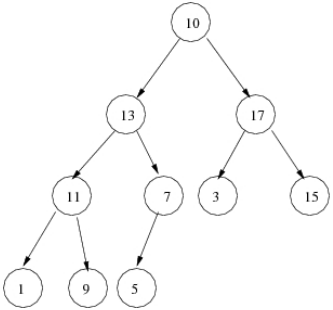
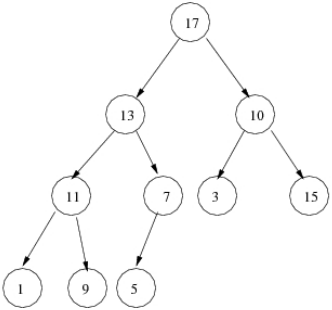
	Naïve	Sorted
`insert`	$\Theta(1)$	$\Theta(n)$
`max`	$\Theta(n)$	$\Theta(1)$
`extractMax`	$\Theta(n)$	$\Theta(1)$
`contains`	$\Theta(n)$	$\Theta(n)$
`increaseKey`	$\Theta(1)$	$\Theta(n)$

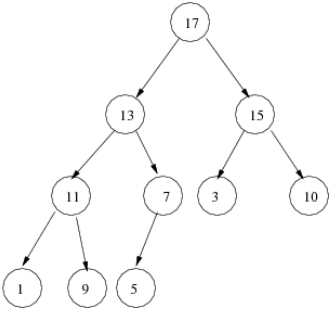

19
13
17
11
7
3
15
1
9
5


19
13
17
11
7
3
15
1
9
5

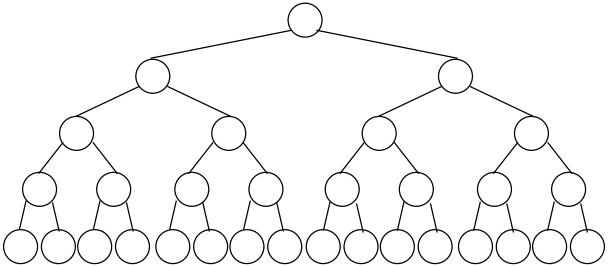

PriorityQueue
<<interface>>
isEmpty()
isFull()
insert(E)
max()
extractMax(E)
contains(E)
size()
Heap
internal
heapSize
compy
#sinkKeyAt(i)
#raiseKeyAt(i)
implements
extends
NaivePriorityQueue
HeapPriorityQueue
HeapSorter
$ sort(int[])
List
<<interface>>
SortedPriorityQueue
implements
ArrayList


10
13
17
11
7
3
15
1
9
5

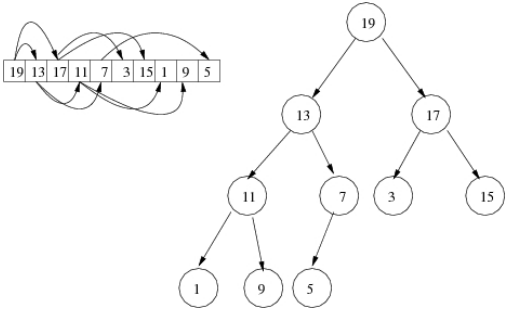

17
13
10
11
7
3
15
1
9
5

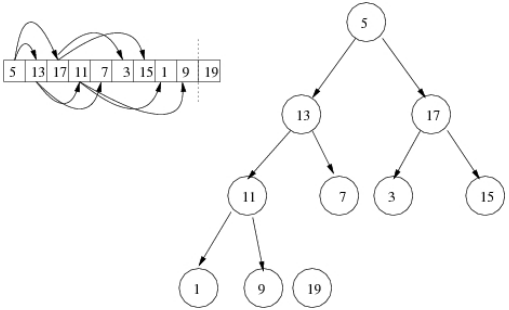

17
13
15
11
7
3
10
1
9
5

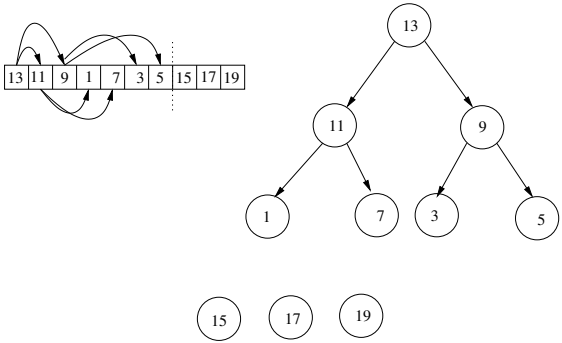
Level	Nodes
0	1
1	2
2	4
3	8
4	16



$$
\begin{aligned}
& \textstyle\sum_{i=0}^{h} 2^i (h+1-i) \\
= \quad & (h+1) \textstyle\sum_{i=0}^{h} 2^i \;-\; \sum_{i=0}^{h} i 2^i \\
= \quad & (h+1)(2^{h+1}-1) - (2 + (h-1)2^{h+1}) \\
= \quad & h2^{h+1} - h + 2^{h+1} - 1 - 2 - h2^{h+1} + 2^{h+1} \\
= \quad & 2 \cdot 2^{h+1} - h - 3 \\
= \quad & 2^{h+2} - h - 3 \\
= \quad & 2^{\lg(n+1)-1+2} - \lg(n+1) + 1 - 3 \\
= \quad & 2^{\lg(n+1)+1} - \lg(n+1) - 2 \\
= \quad & 2 \cdot (n+1) - \lg(n+1) - 2 \\
= \quad & 2 \cdot n - \lg(n+1)
\end{aligned}
$$


19 13 17 11 7 3 15 1 9 5
19
13
17
11
7
3
15
1
9
5


5 13 17 11 7 3 15 1 9 19
5
13
17
11
7
3
15
1
9
19


13 11 9 1 7 3 5 15 17 19
13
11
9
1
7
3
5
15
17
19

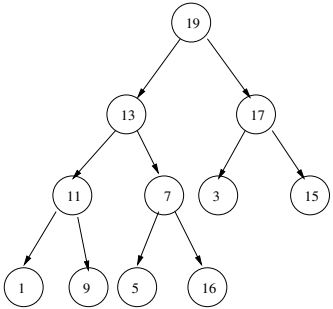
Invariant (Outer loop of heapsort)

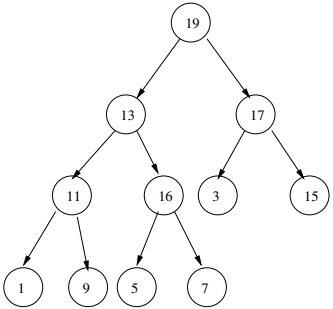
After i iterations,

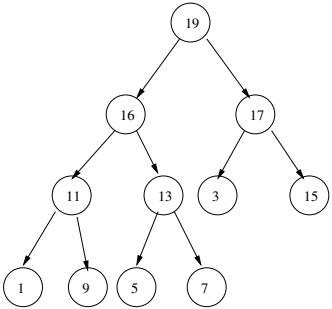
(a) *The range $[n-i, n)$ is sorted.*

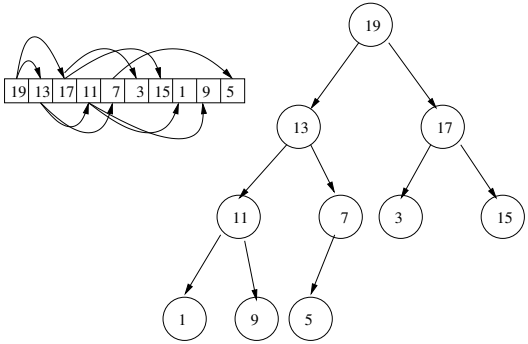
(b) *Everthing in the range $[n-i, n)$ is greater than everything in the range $[0, n-i)$.*

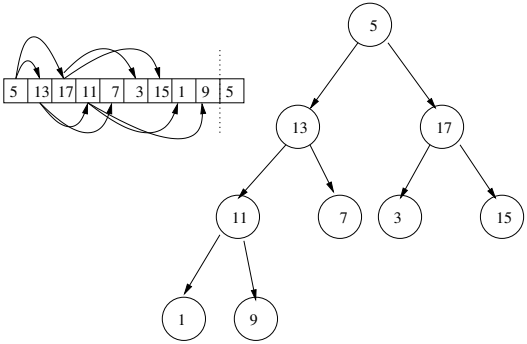
(c) *The range $[0, n-i)$ is a heap.*


19
13
17
11
7
3
15
1
9
5
16

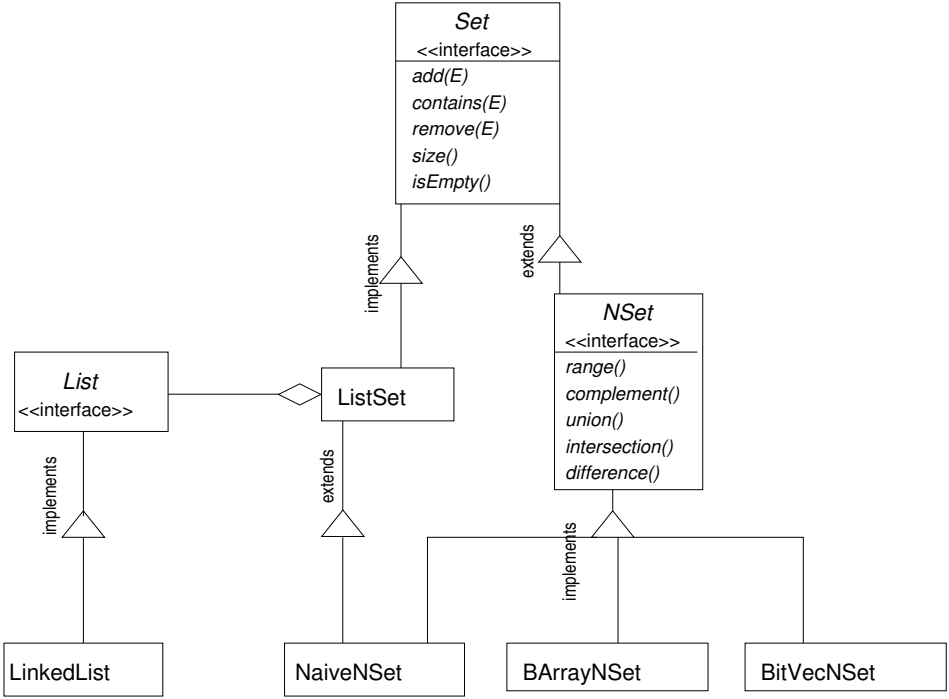

19
13
17
11
16
3
15
1
9
5
7


19
16
17
11
13
3
15
1
9
5
7


19 13 17 11 7 3 15 1 9 5
19
13
17
11
7
3
15
1
9
5


5 13 17 11 7 3 15 1 9 5
5
13
17
11
7
3
15
1
9

	Naïve	Sorted	Heap
`insert`	$\Theta(1)$	$\Theta(n)$	$\Theta(\lg n)$
`max`	$\Theta(n)$	$\Theta(1)$	$\Theta(\lg n)$
`extractMax`	$\Theta(n)$	$\Theta(1)$	$\Theta(\lg n)$
`contains`	$\Theta(n)$	$\Theta(n)$	$\Theta(n)$
`increaseKey`	$\Theta(1)$	$\Theta(n)$	$\Theta(n)$


Set
<<interface>>
add(E)
contains(E)
remove(E)
size()
isEmpty()
implements
extends
NSet
<<interface>>
range()
complement()
union()
intersection()
difference()
List
<<interface>>
ListSet
implements
extends
implements
LinkedList
NaiveNSet
BArrayNSet
BitVecNSet