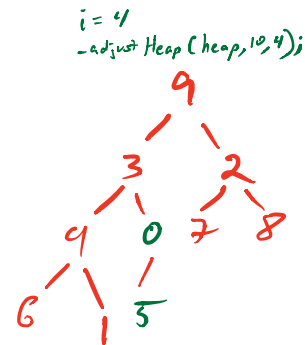
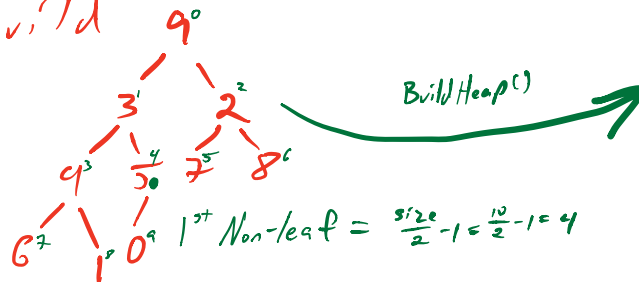
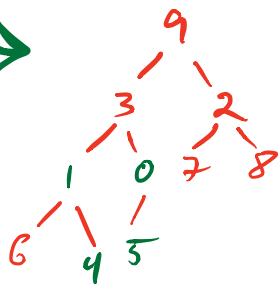


Unsorted array: [9][3][2][4][5][7][8][6][1][0]
 index: 0 1 2 3 4 5 6 7 8 9

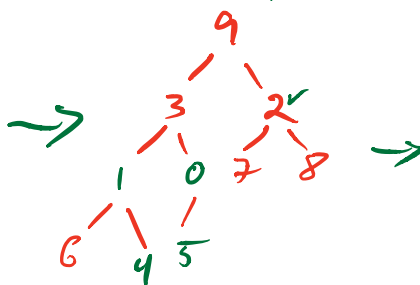
1) Build



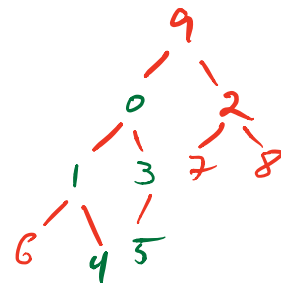
$i = 3$
 -adjustHeap(heap, 10, 3);



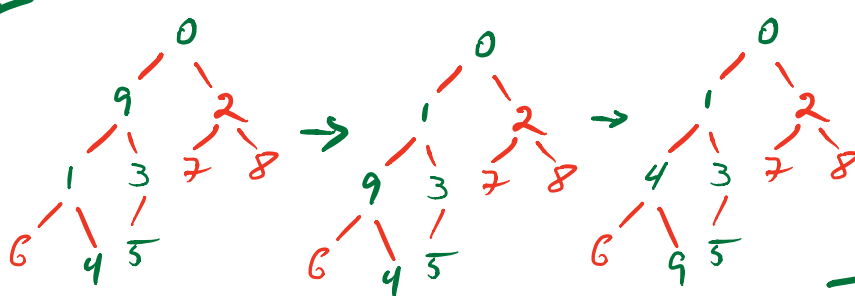
$i = 2$
 -adjustHeap(heap, 10, 2);



$i = 1$
 -adjustHeap(heap, 10, 1);



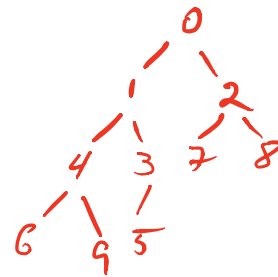
$i = 0$
 -adjustHeap(heap, 10, 0);



Finished
 Heap

2) Heap Sort:

0	1	2	4	3	7	8	6	9	5
0	1	2	3	4	5	6	7	8	9

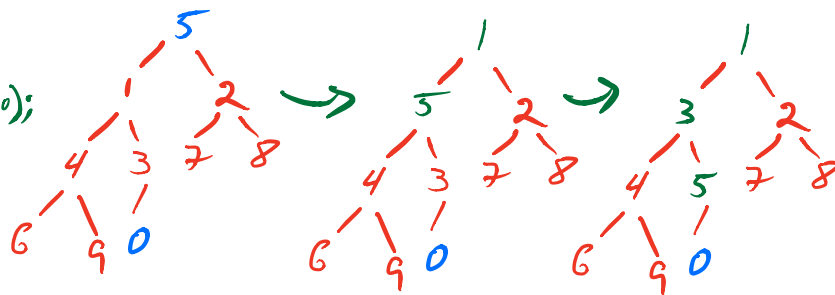


5	1	2	4	3	7	8	6	9	0
0	1	2	3	4	5	6	7	8	9

→

1	3	2	4	5	7	8	6	9	0
0	1			4	5	6	7	8	9

$i=9$
 $Swap(h, 0, i);$
 $- adjustHeap(h, i, 0);$

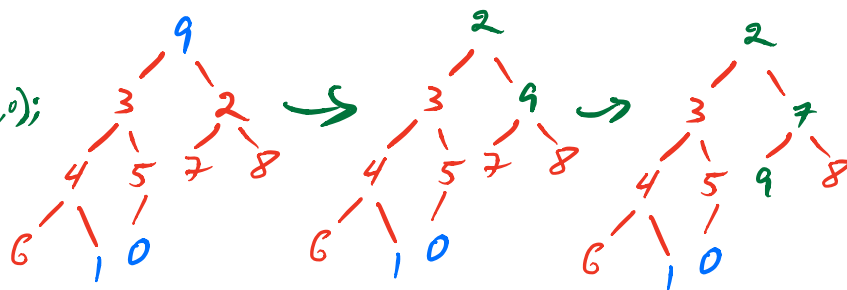


9	3	2	4	5	7	8	6	9	0
0	1	2	3	4	5	6	7	8	9

→

2	3	7	4	5	9	8	6	1	0
0	1	2	3	4	5	6	7	8	9

$i=8$
 $Swap(h, 0, i);$
 $- adjustHeap(h, i, 0);$

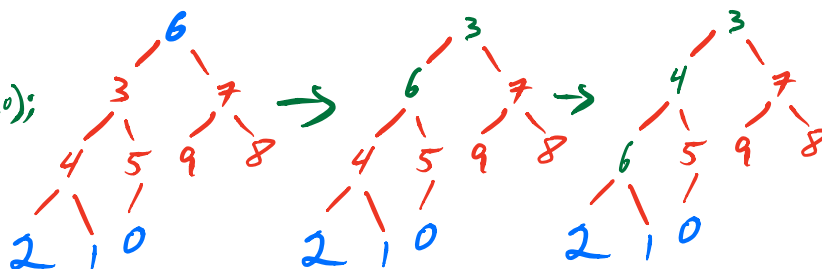


6	3	7	4	5	9	8	2	1	0
0	1	2	3	4	5	6	7	8	9

→

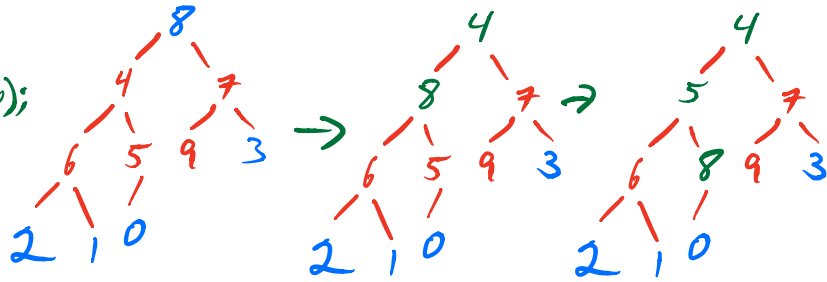
3	4	7	6	5	9	8	2	1	0
0	1	2	3	4	5	6	7	8	9

$i=7$
 $Swap(h, 0, i);$
 $- adjustHeap(h, i, 0);$

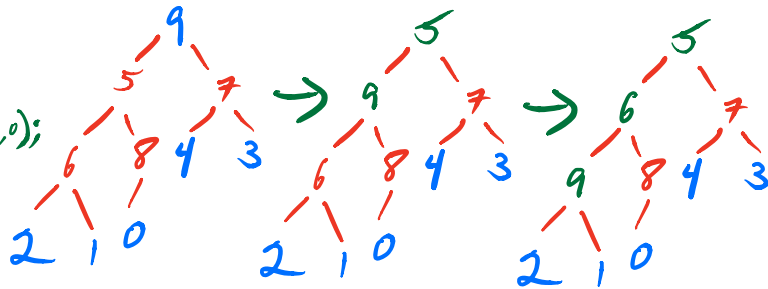




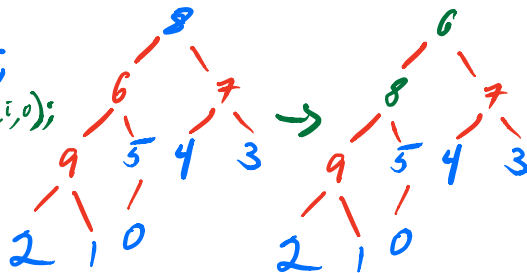
$i=6$
 $Swap(h, 0, i);$
 $-adjustHeap(h, i, 0);$



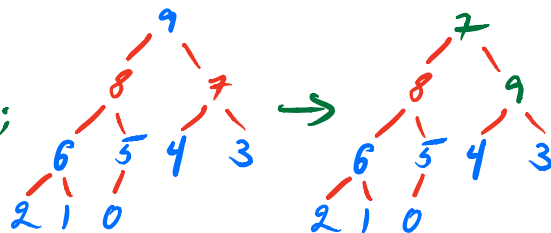
$i=5$
 $Swap(h, 0, i);$
 $-adjustHeap(h, i, 0);$



$i=4$
 $Swap(h, 0, i);$
 $-adjustHeap(h, i, 0);$



$i=3$
 $Swap(h, 0, i);$
 $-adjustHeap(h, i, 0);$

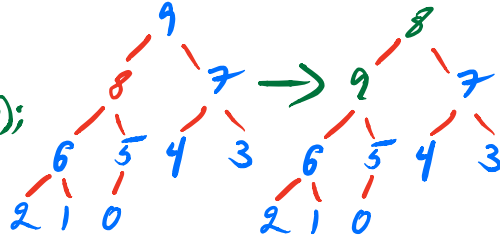




$i = 2$

Swap(h, 0, i);

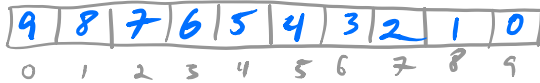
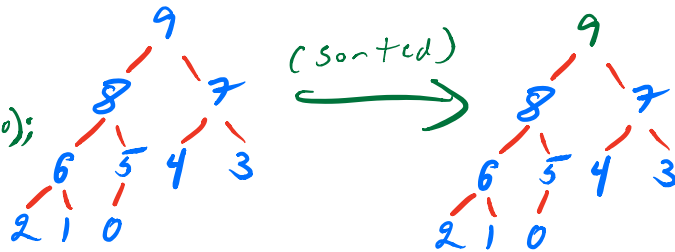
- adjustHeap(h, i, 0);



$i = 1$

Swap(h, 0, i);

- adjustHeap(h, i, 0);



$i = 0$

Array is
sorted

