Adnar Lozano
CSE 330

06/06/17

Homework #3

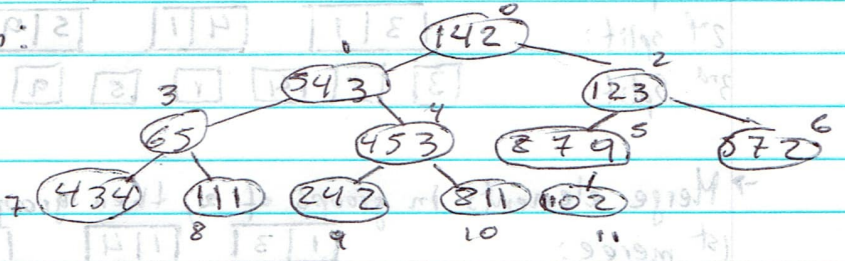
1. Show how heapsort processes the input.

Input array:

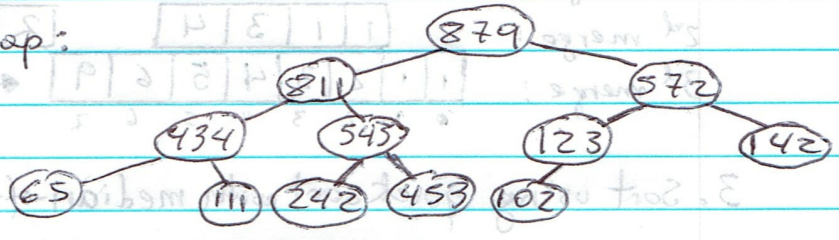
0	1	2	3	4	5	6	7	8	9	10	11
142	543	123	65	453	879	572	434	111	242	811	102

$n = 12$

→ Build the heap:



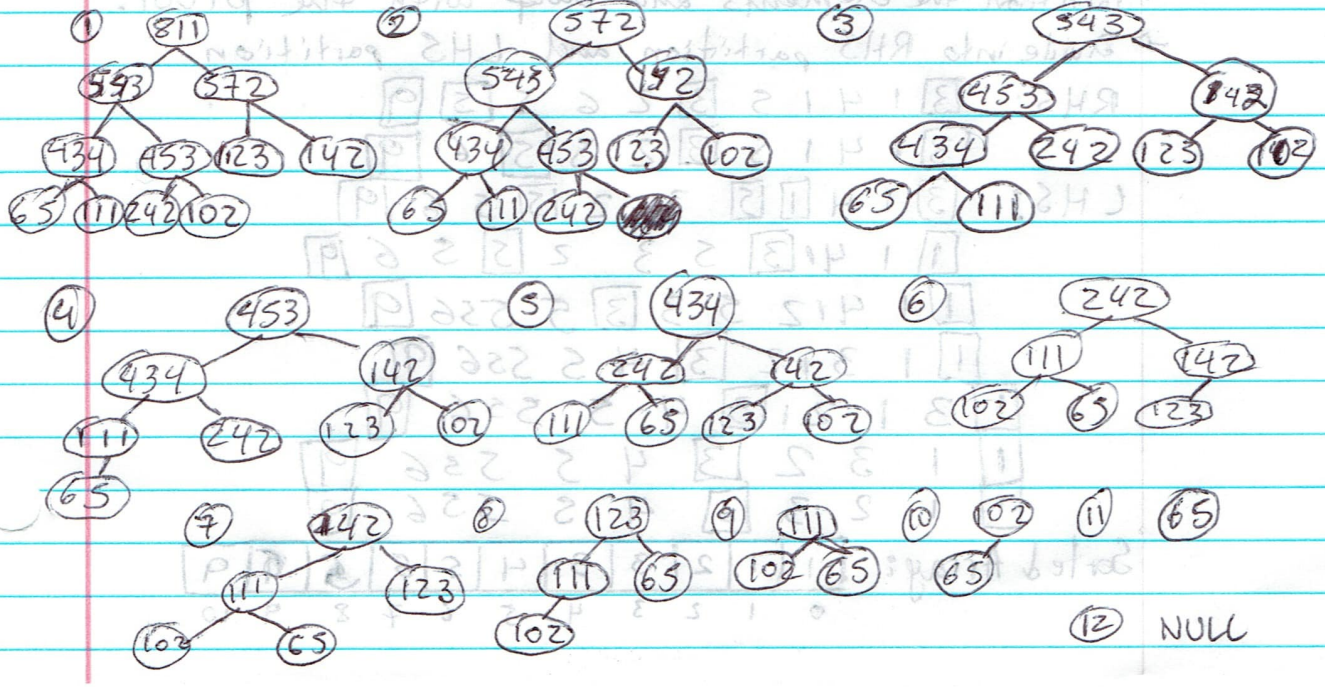
→ Build Max heap:



→ Remove the max element n times and store in an array in reverse order. Heapify after each deletion

Sorted array:

0	1	2	3	4	5	6	7	8	9	10	11
65	102	111	123	142	242	434	453	543	572	811	879



2. Sort using mergesort:

Input Array:

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

$n = 8$

→ Split the array in half until all elements are split individually

1st split: [3 1 4 1] [5 9 2 6]

2nd split: [3 1] [4 1] [5 9] [2 6]

3rd Split: [3] [1] [4] [1] [5] [9] [2] [6]

→ Merge elements in groups of 2, then groups of 4, then 8, and so on

1st merge: [1 3] [1 4] [5 9] [2 6]

2nd merge: [1 1 3 4] [2 5 6 9]

3rd merge: [1 1 2 3 4 5 6 9] ← Sorted Array

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

3. Sort using quicksort with median-of-three partitioning & cutoff 3

Input Array:

0	1	2	3	4	5	6	7	8	9	10
3	1	4	1	5	9	2	6	5	3	5

$n = 11$

FIRST, MIDDLE, LAST

→ Sort First, middle, last:

3	1	4	1	5	5	2	6	5	3	9

MEDIAN (PIVOT)

→ Partition the elements and swap with the pivot.

→ Divide into RHS partition and LHS partition

RHS: [3] 1 4 1 5 [3] 2 6 5 [5] [9]

[3] 1 4 1 5 [3] 2 5 [5] 6 [9]

LHS: [3] 1 4 [1] [5] 3 2 [5] 5 6 [9]

[1] 1 4 [3] 5 3 2 [5] 5 6 [9]

[1] 1 4 2 5 3 [3] 5 5 5 6 [9]

[1] 1 3 2 [3] 4 5 5 5 6 [9]

[1] 3 1 2 [3] 4 5 5 5 6 [9]

[1] 1 3 2 [3] 4 5 5 5 6 [9]

[1] 1 2 3 [3] 4 5 5 5 6 [9]

Sorted Array:

0	1	2	3	4	5	6	7	8	9	10
1	1	2	3	3	4	5	5	5	6	9