HEAP SORT LAB

5	22	9	76	63	81	48	92	54	28
0	1	2	3	4	5	6	7	8	9

list

Build Heap

1) index = length/2 - 1 = $\frac{10}{2}$ - 1 = 4   leaves

Parent of last node in tree: list[4] = 63   list[4]=

2) a) Left Subtree: list[b] ⇒ list[4·2+1] = list[9] = 28

Right Subtree: list[c] → list[4·2+2] = list[10] = DNE or null

b) list[largerIndex] = list[b] → list[9]

→ greater child b/c list[b] > list[c]

c)

63 — 28

b/c list[a] > list[b]

already a heap

d) Skip

___

list[3]   index=3   list[3] = 76

2) a) Left ST – list[b] → list[3·2+1] → list[7] = 92

Right ST – list[c] → list[3·2+2] → list[8] = 54

b) list[largestIndex] = list[b]   largestIndex=7

b/c   list[b] > list[c]

c) list[a] < list[largestInd], so swap occurs

5	22	9	76	63	81	48	92	54	28
0	1	2	3	4	5	6	a.7	8	9

Before

| 5 | 22 | 9 | 92 | 63 | 81 | 48 | 76 | 54 | 28 | After
|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

* .

d) Repeat for list [7]

a) RST - list [15] = null
   LST - list [16] = null
   so move on

---

list [2] → list [7] = 9

2) a) RST - list [5] = 81   → list [b]
      LST - list [6] = 48   → list [c]

   b) list[b] > list [c] so largerInd = 5

   c) list [largerInd] > list [7] (list[2]), so swap!

5	22	9	92	63	81	48	76	54	28
0	1	2	3	4	5	6	7	8	9

5	22	81	92	63	9	48	76	54	28
0	1	2	3	4	5	6	7	8	9

d) list [2] = 81
  a) LST - list [5] = 9
  b) RST - list [6]   so already heap

---

a) 2 =   list [1] = 22

LST - list [3] = 92   b
RST - list [4] = 63   c
b) list [3] > list [4]   so largerInd = 3
        b   c
c) list [3] > list [1] so swap

5	92	81	22	63	9	48	76	54	28
0	1	2	3	4	5	6	7	8	9

d) Repeat since swap list [3] = 22

a) LST - list [7] = 76   b
  RST - list [8] = 54   c
b) list [7] > list [8]   so largerInd = 7
      b   c
c) list [7] > list [3]   so swap

5	92	81	76	63	9	48	22	54	28
0	1	2	3	4	5	6	7	8	9

---

2) a) list [0] = 5 a
LST - list [1] = 92 b
RST - list [2] = 81   c

b) list[1] > list[2] so largerInd=1
     b   c

c) list[1] > list[0] so swap

92	5	81	76	63	9	48	22	54	28
0	1	2	3	4	5	6	7	8	9

d) Repeat   list[i] = 5   a

LST - list[3] = 76 b
RST - list[4] = 63 c

b) list[3] > list[4]   so   largerInd = 3
    b   c

c) list[3] > list[1]   so swap

92	76	81	5	63	9	48	22	54	28
0	1	2	3	4	5	6	7	8	9

d) Repeat   list[3] = 5   a
LST - list[7] = 22   b
RST - list[8] = 54   c

b) list[8] > list[7]   largerInd = 8

c) list[8] > list[3] so swap

92	76	81	54	63	9	48	22	5	28
0	1	2	3	4	5	6	7	8	9

✓

HEAP SORT

1) SWOP
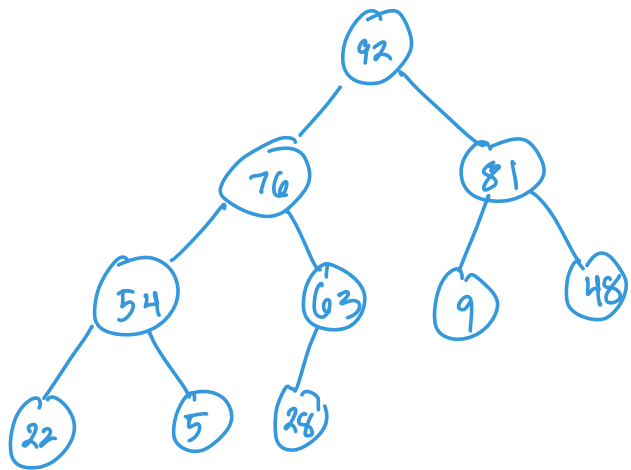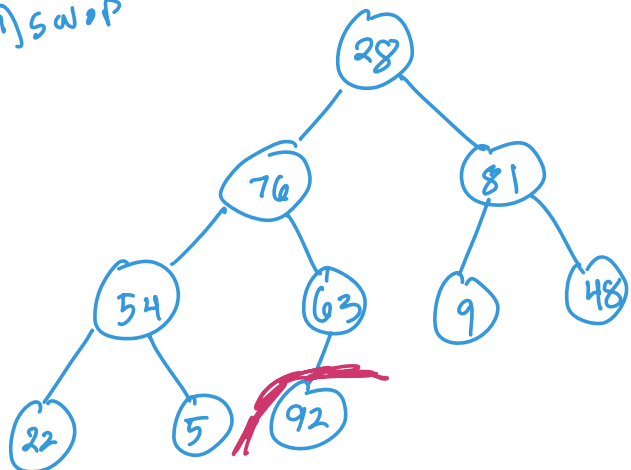
2) HEPIFY

28	76	81	54	63	9	48	22	5	92
0	1	2	3	4	5	6	7	8	9

28 81

76 81

54 63 9 48

22 5 92

81

76 28

54 63 9 48

22 5 92

81 0

76 48 2
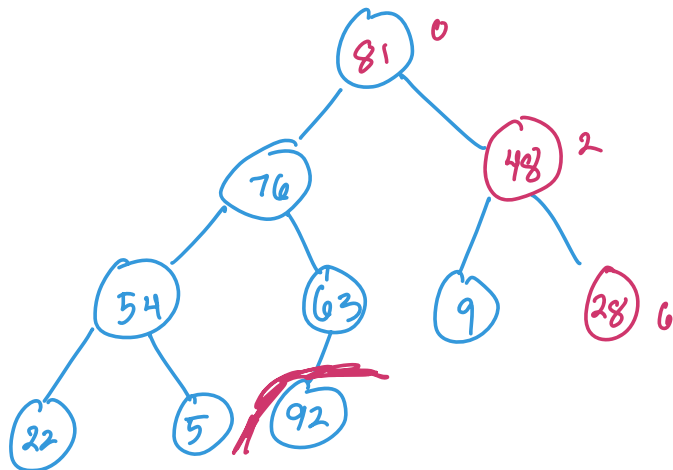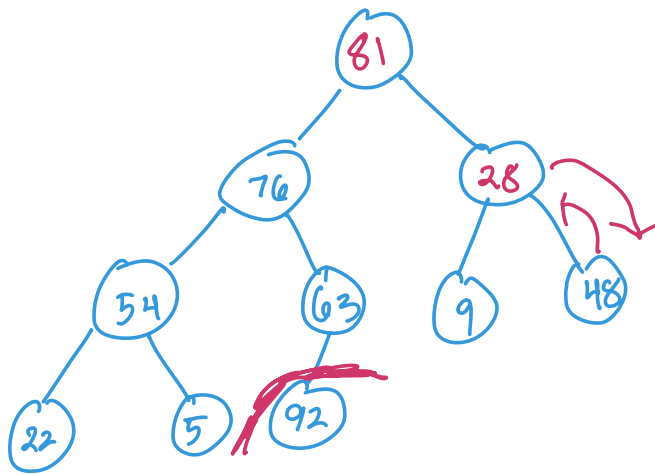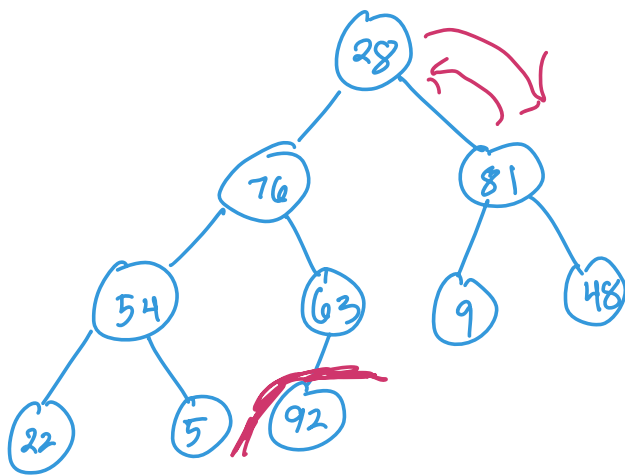
54 63 9 28 6

22 5 92

81	76	48	54	63	9	28	22	5	92
0	1	2	3	4	5	6	7	8	9

Repeat
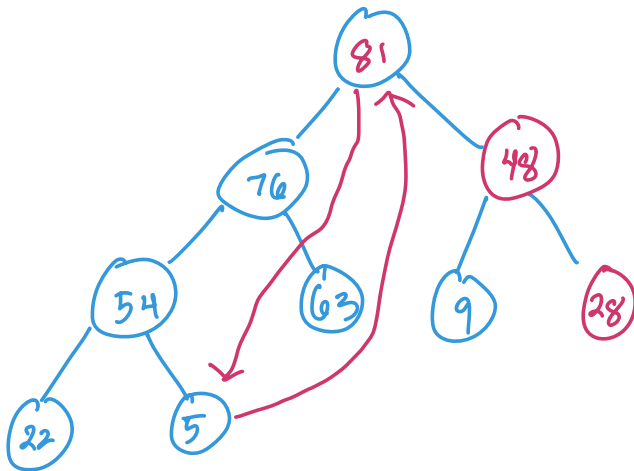
5	76	48	54	63	9	28	22	81	92
0	1	2	3	4	5	6	7	8	9
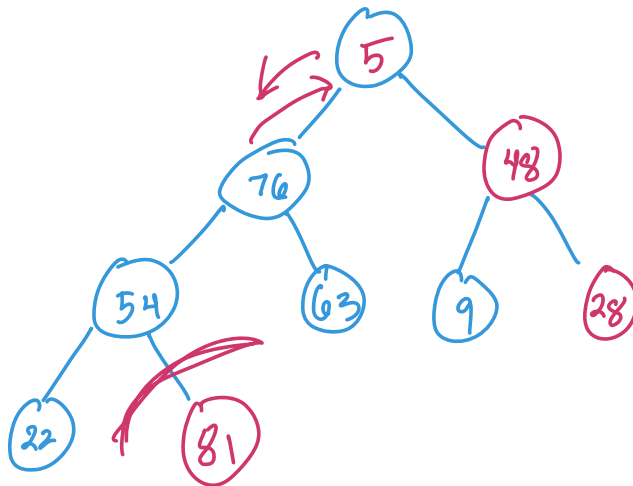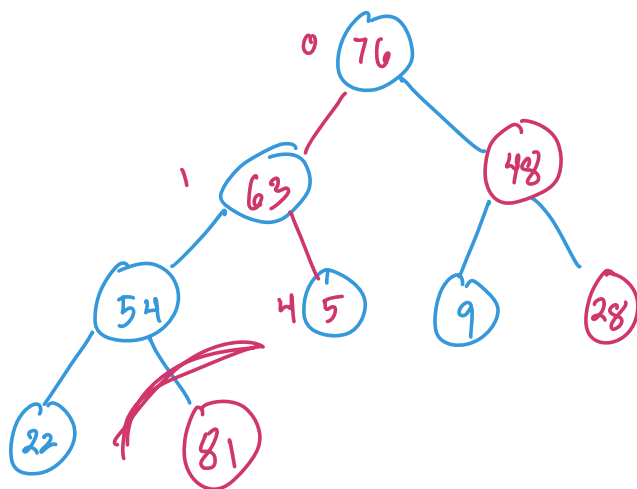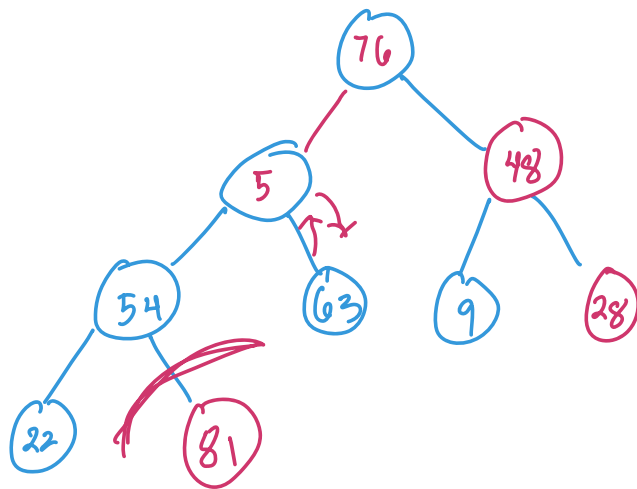
Repeat

76	63	48	54	5	9	28	22	81	92
0	1	2	3	4	5	6	7	8	9
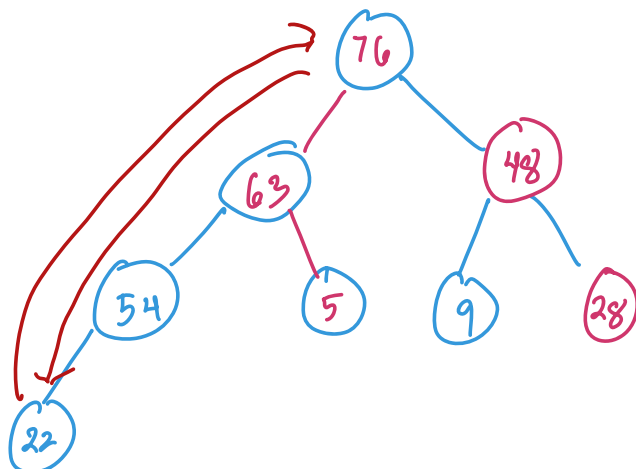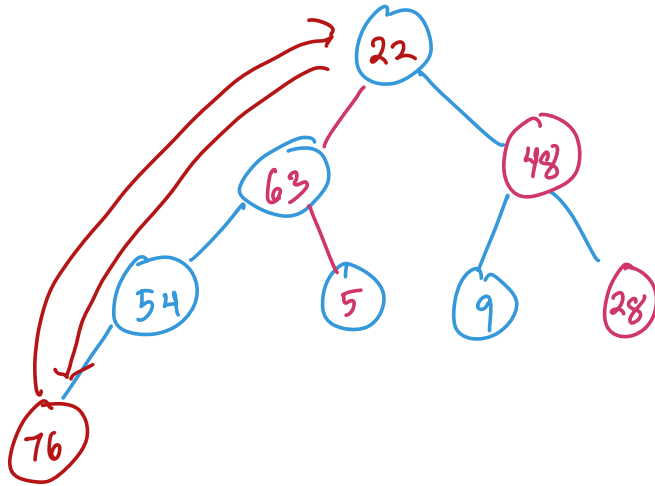
22	63	48	54	5	9	28	76	81	92
0	1	2	3	4	5	6	7	8	9

Binary tree (top):
- Root: 63 (index 0)
- 54 (index 1) — left child of 63
- 48 (index 2) — right child of 63
- 22 (index 3) — left child of 54
- 5 — right child of 54
- 9 — left child of 48
- 28 — right child of 48
- 76 — child of 22 (crossed out)

Array 1:
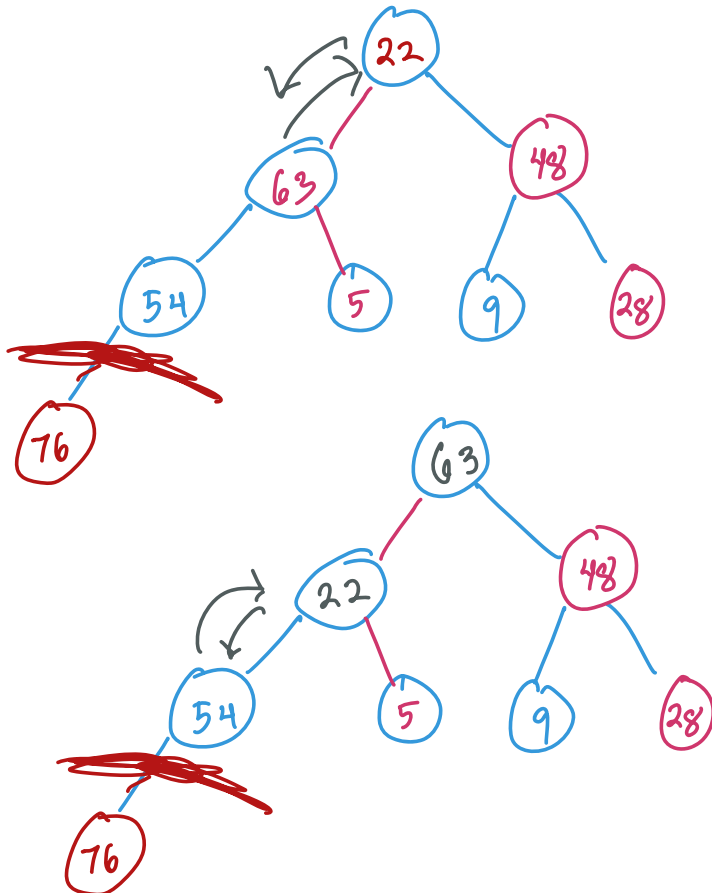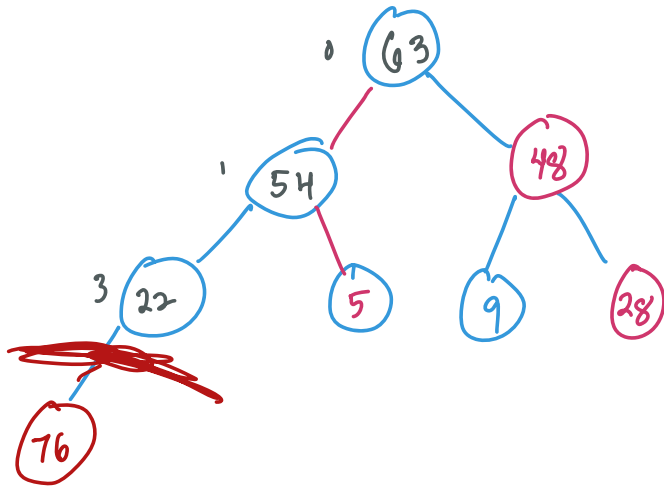
63	54	48	22	5	9	28	76	81	92
0	1	2	3	4	5	6	7	8	9

Array 2:

28	54	48	22	5	9	63	76	81	92
0	1	2	3	4	5	6	7	8	9

Binary tree (middle):
- Root: 28
- 54 — left child
- 48 — right child
- 22 — left child of 54
- 5 — right child of 54
- 9 — left child of 48
- 63 — right child of 48 (crossed out)

Binary tree (bottom):
- Root: 54 (index 0)
- 28 (index 1) — left child
- 48 — right child (crossed out child)
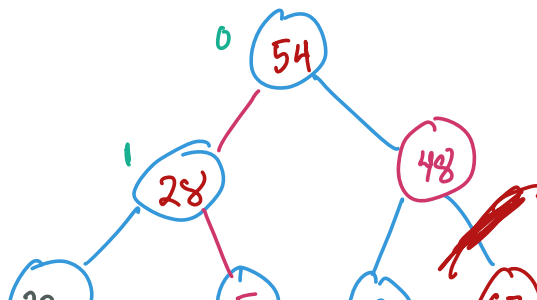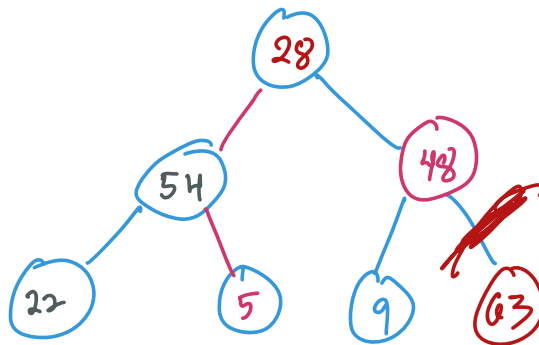
54 | 28 | 48 | 22 | 5 | 9 | 63 | 76 | 81 | 92
0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

9 | 28 | 48 | 22 | 5 | 54 | 63 | 76 | 81 | 92
0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
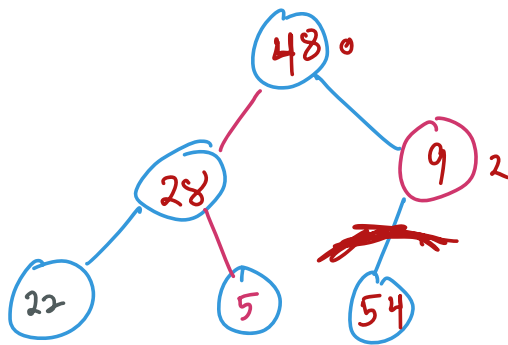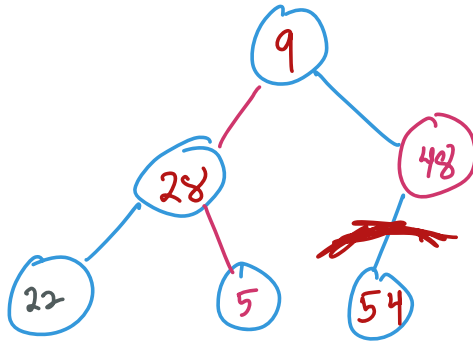
48 | 28 | 9 | 22 | 5 | 54 | 63 | 76 | 81 | 92
0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

5 | 28 | 9 | 22 | 48 | 54 | 63 | 76 | 81 | 92
0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

Binary tree with root 5, left child 28 (with children 22 and 48), right child 9.

Binary tree with root 28 (index 0), left child 22 (index 1) with children 5 (index 3) and 48, right child 9.
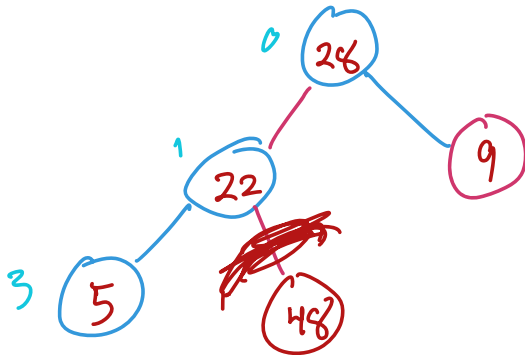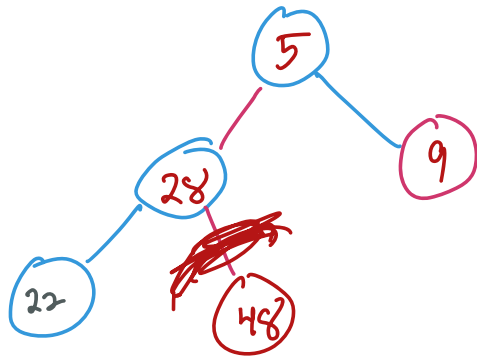
28	22	9	5	48	54	63	76	81	92
0	1	2	3	4	5	6	7	8	9

5	22	9	28	48	54	63	76	81	92
0	1	2	3	4	5	6	7	8	9

Binary tree with root 22 (index 0), left child 5 with child 28, right child 9.
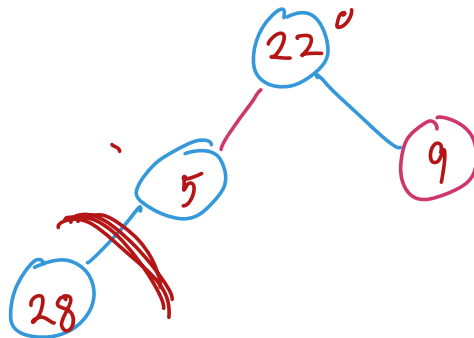
22	5	9	28	48	54	63	76	81	92
0	1	2	3	4	5	6	7	8	9

9	5	22	28	48	54	63	76	81	92
0	1	2	3	4	5	6	7	8	9

5	9	22	28	48	54	63	76	81	92
0	1	2	3	4	5	6	7	8	9

5	9	22	28	48	54	63	76	81	92
0	1	2	3	4	5	6	7	8	9

✓