**INSHEAP(TREE,N,ITEM)**

‘ITEM’ is value to be inserted in TREE with Size ‘N’.

1. Set N = N + 1 and PTR = N.
2. Set TREE[PTR] = ITEM and PAR = ⌊PTR/2⌋.
3. Repeat Step 4 and 5 While TREE[PTR] < TREE[PAR].
4. [Swapping Node with its Parent.]

Set TEMP = TREE[PAR].

Set TREE[PAR] = TREE[PTR].

Set TREE[PTR] = TEMP.

1. Set PTR = PAR and PAR = ⌊PTR/2⌋.

[End Step 4 Loop.]

1. Exit.

**SEARCHHEAP(TREE,N,LOC,ITEM)**

‘LOC’ is Index of ITEM after Searching in TREE with Size ‘N’.

1. Repeat Step 2 For K = 1 to N.
2. If TREE[K] = ITEM then:

Set LOC = K and Return.

[End of If statement.]

[End of Step 1 Loop.]

1. Set LOC = NULL.
2. Return.

**MINHEAPIFY(TREE,N,K)**

‘K’ is INDEX of TREE of Size ‘N’ where MINHEAPIFY is to be applied.

1 Set LEFT = 2\*K and RIGHT = 2\*K+1.

2 Set MIN = K.

3 If LEFT < N and arr[LEFT] < arr[K] then:

Set MIN = LEFT.

4 If RIGHT < N and arr[RIGHT] < arr[MIN] then:

Set MIN = RIGHT.

5 If (MIN ≠ K)

[Swapping Nodes.]

Set TEMP = TREE[K].

Set TREE[K] = TREE[MIN].

Set TREE[MIN] = TEMP.

**Call MINHEAPIFY(TREE,N,MIN).**

[End of If statement.]

6 Return.

**EXTMIN(TREE,N,ITEM)**

‘ITEM’ is minimum Extracted Node of TREE with Size ‘N’.

1. Set ITEM = TREE[1].
2. [Last Node to first Node.]

Set TREE[1] = TREE[N] and N = N - 1.

1. **Call MINHEAPIFY(TREE,N,1).**
2. Return.

**DELHEAP(TREE,N,ITEM)**

‘ITEM’ is the Node to be Deleted from TREE with Size ‘N’.

1. **Call SEARCHHEAP(TREE,N,LOC,ITEM).**
2. [Decreasing value of ITEM to Minimum.]

TREE[LOC] = TREE[1]-1.

1. [Building Heap after Modification,]

Set TREE[PTR] = ITEM and PAR = ⌊PTR/2⌋.

1. Repeat Step 5 While TREE[PTR] < TREE[PAR].
2. [Swapping Node with its Parent.]

Set TEMP = TREE[PAR].

Set TREE[PAR] = TREE[PTR].

Set TREE[PTR] = TEMP.

[End of Step 4 Loop.]

1. **Call EXTMIN(TREE,N,ITEM).**
2. Return.

**HEAPSORT(TREE,N)**

1. Repeat Step 2 For K = 1 to N.
2. **Call EXTMIN(TREE,N,ITEM)** and TEMP[K] = ITEM.

[End of Step 1 Loop.]

1. Repeat Step 4 For K = 1 to N.
2. TREE[K] = TEMP[K].

[End of Step 3 Loop.]

1. Exit.

**BUILDHEAP(TREE,N)**

1. Repeat Step 2 For K = 1 to N.

2. ITEM = TREE[K] and **Call INSHEAP(TEMP,N,ITEM)**.

[End of Step 1 Loop.]

3. Repeat Step 4 For K = 1 to N.

4. TREE[K] = TEMP[K].

[End of Step 3 Loop.]

5. Exit.