CSE 2320 Notes 5: Heapsort and Priority Queues

(Last updated 9/9/06 7:38 AM)

CLRS, Chapter 6

(BINARY) HEAP PROPERTIES

1. Binary tree.
2. Like a complete tree, but missing some "rightmost" leaves in deepest level.
3. a. Each parent has a priority ≥ the priority of its children - maxheap.

   b. Each parent has a priority ≤ the priority of its children - minheap.

Derived property:

4. a. The root priority is the _____.

   b. The root priority is the _____.

Common Mapping of a Heap to an Array

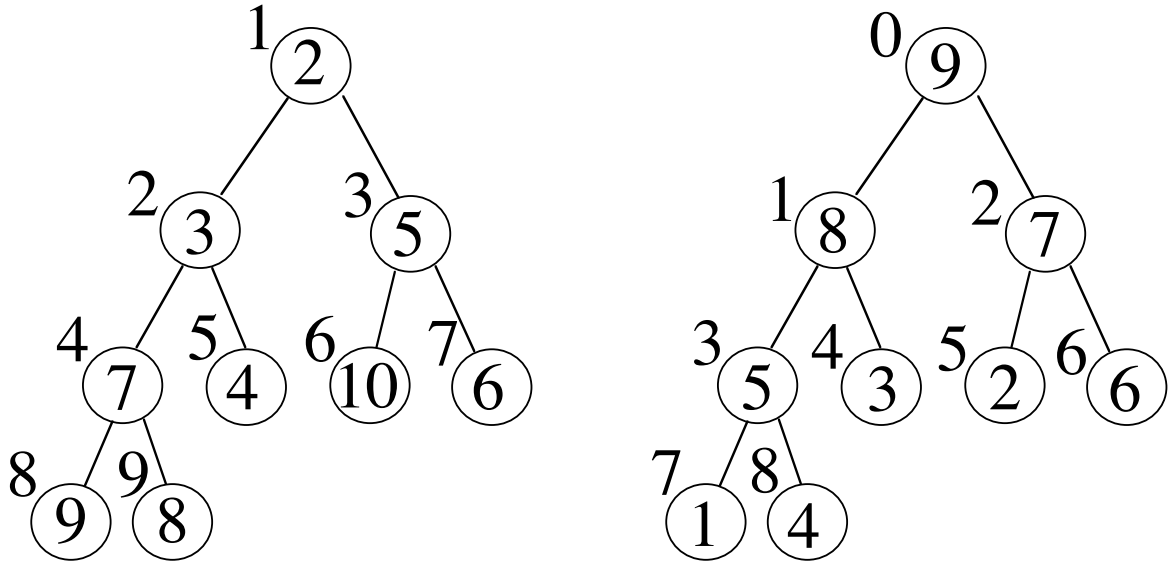
1. Subscript 0 is unused.
2. Subscript 1 stores the root of the heap.
3. If it exists, the left child for subscript i is stored at subscript $2i$.
4. If it exists, the right child for subscript i is stored at subscript $2i+1$.
5. Parent of node with subscript i is $\left\lfloor \frac{i}{2} \right\rfloor$.

Alternate Mapping of a Heap to an Array

1. Subscript 0 stores the root of the heap.
2. If it exists, the left child for subscript i is stored at subscript $2i + 1$.
3. If it exists, the right child for subscript i is stored at subscript $2i + 2$.
4. Parent of node with subscript i is $\left\lfloor \frac{i-1}{2} \right\rfloor$.

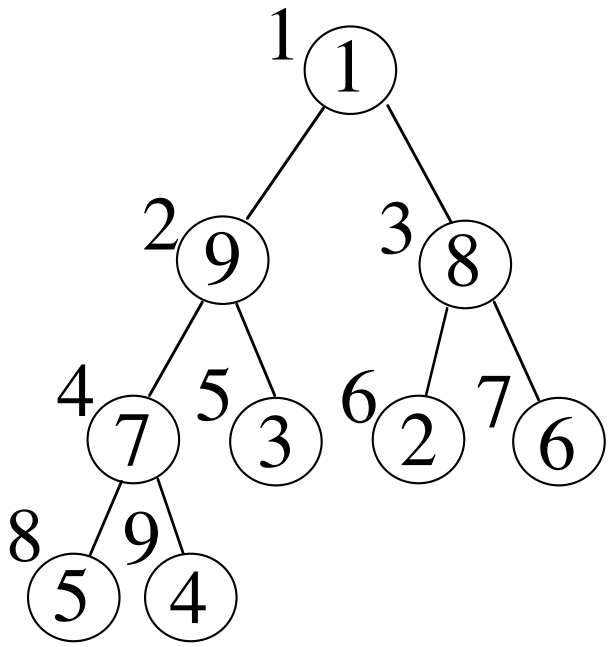
Under either mapping, the number of nodes/slots used (n) must be stored.

No pointers are needed.



CONVERTING AN UNORDERED ARRAY INTO A MAXHEAP (BUILD-MAX-HEAP)

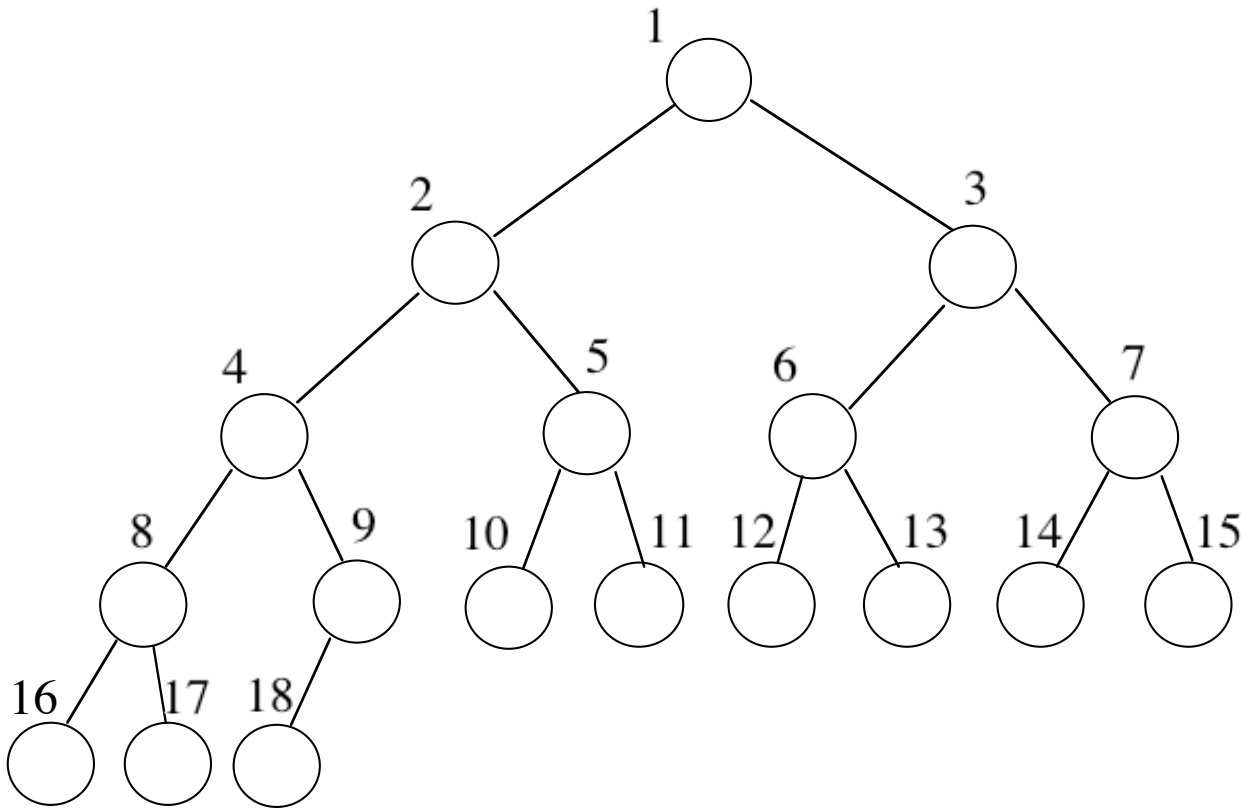
Special Case: Parent Node with Two Subtrees Having Maxheap Properties (MAX-HEAPIFY)



Worst case, $2^k \le n \le 2^{k+1} - 1$, processes _____ parents.

General Case: Move through the parents in descending subscript order, applying the special case at each parent.

Example:



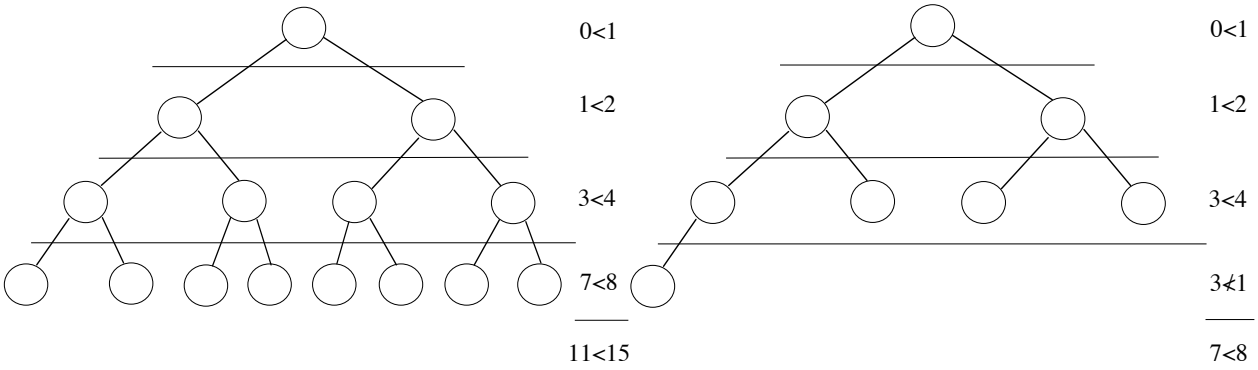
Time

Lower bound?

Worst case?

Worst case time [Different from book]

Each level may receive values from all ancestors in shallower levels.



SORTING USING A HEAP (HEAPSORT)

BUILD-MAX-HEAP leaves the maximum at the root.

Swap root with element at subscript *n*. Remove subscript *n* from heap.

MAX-HEAPIFY at root to restore maxheap property.

. . .

Time

BUILD-MAX-HEAP takes $\Theta(n)$ for worst case.

Even though the heap loses one node after each call to MAX-HEAPIFY, 1/2 the nodes are leaves taking $\Theta(\log n)$ worst-case time for each.

$\Theta(n \log n)$ for entire sort in worst case.

PRIORITY QUEUES

Entry with the maximum priority is the next item to be removed.

Operations are naturally supported by a heap in $\Theta(\log n)$ worst-case time.[Ignoring unions]

At least the following two operations are required to support a PQ:

MAX-HEAP-INSERT – New item is placed at next available leaf and is swapped with ancestors having lower priorities.

HEAP-EXTRACT-MAX – Remove item with maximum priority and fix heap, just like HEAPSORT.

Other convenient PQ operations, based on an "external" *dictionary*:

Dictionaries – Topic covered in detail for test 2.

Supports finding a desired (key, satellite data) pair

Some possibilities

(Hash) Table

Linked List

(Binary) Search Tree

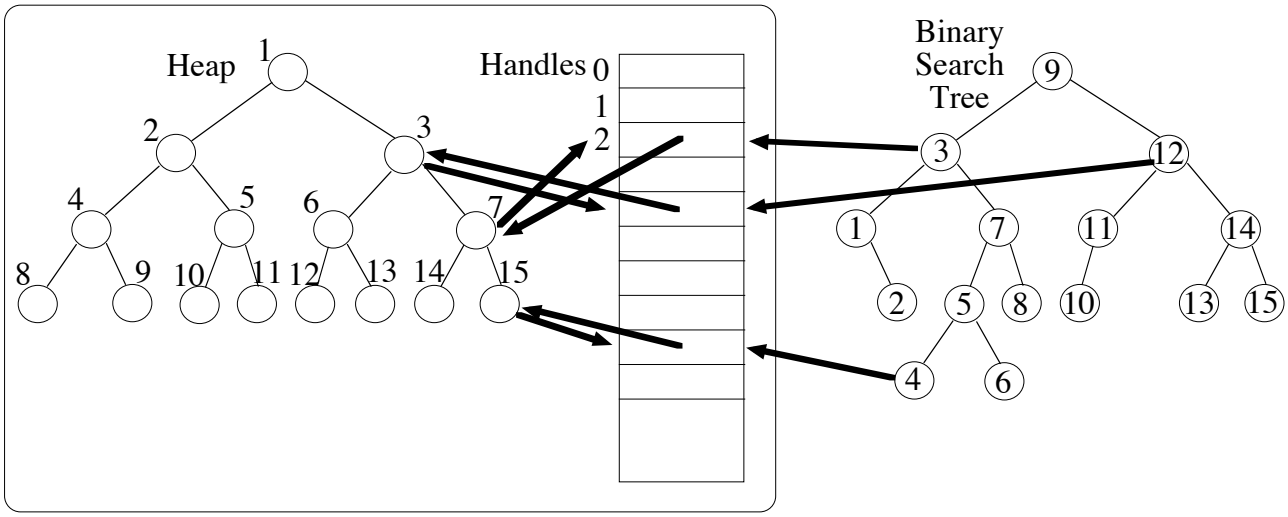
Linking Between Dictionary and Heap

Handles are used to track movement of a heap entry.

Satellite data in each dictionary entry includes subscript of handle.

Each heap entry includes the number of its handle.

[An efficient scheme for managing handles is considered in Notes 8.]



Change Priority -

HEAP-DECREASE-KEY

HEAP-INCREASE-KEY

Delete Entry – Replace with contents of last leaf.

See `minHeap.cpp` for an implementation (with limited implementation of handles).

APPLICATIONS

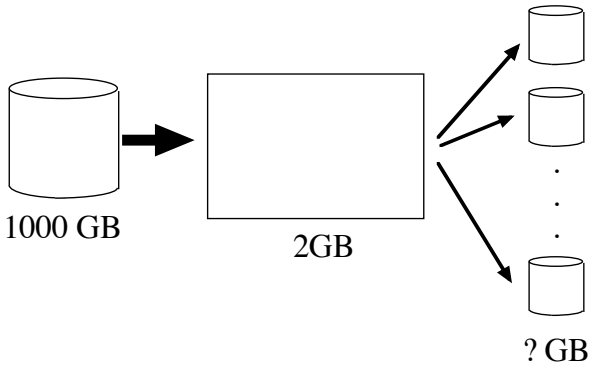
Elementary

1. Choosing a junior high basketball team.
2. Choosing jockeys to ride your horses.

For Later - Greedy Methods Using Heaps: Prim's Minimum Spanning Tree, Dijkstra's Shortest Path, Maximum Capacity Paths, Huffman Codes

Two Heap-Based Techniques Related to Mergesort with External Devices

1. Run/string/subfile production:



   a. Build initial minheap.

   b. Attempt to include minheap root in current run.

   c. Minheap root could be smaller than last element appended!?!?!?

2. Multiway merging from *k* devices using minheap with *k* initial entries: