Names: _____

CS Logins: _____

Worksheet #8

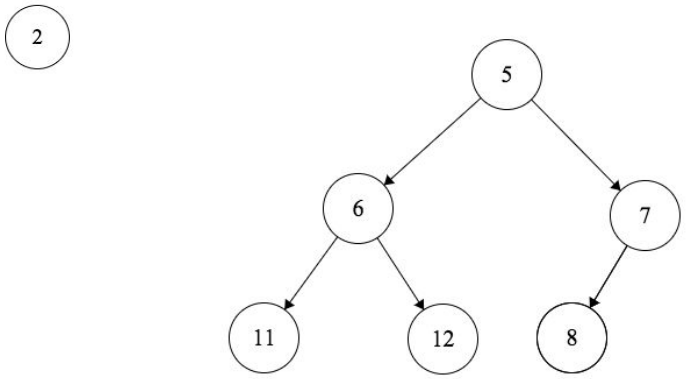
Heaps and Priority Queues Activities

As always, sit with a partner and work through these together.

Activity #1: Fill in the missing values:

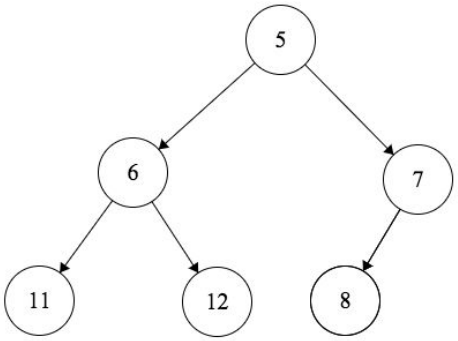
Implementation	add	removeMin
Unsorted Array		**O(n)**
Sorted Array		**O(1)**
Unsorted Linked List	**O(1)**	
Sorted Linked List	**O(n)**	
Hash Table	**O(1)**	**O(n)**
Heap		

Activity #2a: Draw how inserting the node below into the heap would look **keeping heap properties in mind**:



Activity #2b: What is the runtime for insert()? Please explain.

Activity #3a Also keeping heap properties in mind, draw what you removeMin() will look like:



Activity #3b What is the runtime for removeMin()? Please explain.

Activity 4: Improving Pseudocode

1. Make an improvement to this code
2. Briefly summarize the improvement in the space provided

```
function LCA(u, v):
     lca = null
     udepth = T.depth(u)
     vdepth = T.depth(v)
     if (T.isroot(u) == true) or (T.isroot(v) == true) then
           lca = T.root
     while (lca == null) do
           if (u == v) then
                 lca = u
           else
                 if udepth > vdepth then
                       u = T.parent(u)
                       udepth = udepth – 1
                 else if vdepth > udepth
                       v = T.parent(v)
                       vdepth = vdepth – 1
                 else
                       u = T.parent(u); udepth = udepth - 1
                       v = T.parent(v); vdepth = vdepth – 1
     return lca
```

Improvements: