CMP 338

11-3-09

In BSTNode class

```
public BSTNode parentOfMaximum()
{
    if (right == null)
        return null;
    if (right.right != null)
        return right.parentOfMaximum();
    else
        return this;
}
```

In remove()

```
BSTNode maxParent = target.left.parentOfMaximum();
BSTNode max;
if (maxParent == null)
{
    max = maxParent = target;
    max = maxParent.left;
}
else
{
    max = maxParent.right;
    max.left = target.left;
    maxParent.right = max.left;
}
max.right = target.right;
if (x < parent.x)
    parent.left = max;
else
    parent.right = max;
}
```

CMPT 338

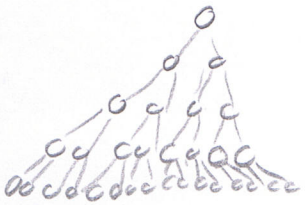
11.3.05

Heap

1) Binary tree

2) Each node has a value greater (less than) it's parent.

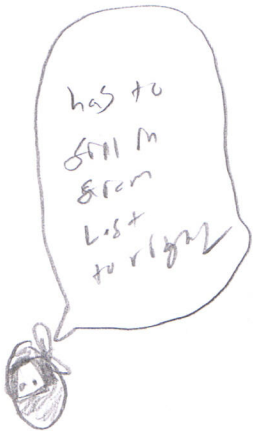
3) Complete



Full Binary tree - is a binary tree where all non-leaves have two children and all leaves are in the same level. In other word, every level in the tree has the maximum # of nodes for that level.

Maximum # of nodes n in level L:

$$n = 2^L$$



Balanced Binary tree - a binary tree where each level except perhaps the last level has the maximum number of nodes for that level

11.3.05

CMP 338

Complete Binary tree - a Balanced Binary tree such that the last Level is Filled from left to right with no gaps between the Left most node in that level and the right most node in that level

$Left = 2n + 1$

$right = 2n + 2$

$Parent = (n-1)/2$