

### Assignment 5 Written Questions

1. If a full binary tree has  $N$  nodes. Then it will have  $(N/2 + N/4 + N/8 + \dots N/N)$  interior nodes.

$$\sum_{i=1}^{\sqrt{N}} N/2^i$$

It would have this amount because full binary trees are sums of powers of two. Squaring  $N$  would give the row number and the sum of the powers up to the square root of  $N$  would give the sum.

2. Six is not a power of two, so it is impossible to make a tree with six nodes. Only powers of two are able to be full binary trees.
3. If a full binary tree has a height of 5 then it would have  $2^5$  or 32 number of leaf nodes.
4. AA tree.

1. Binary heaps have a worst case of  $O(\log n)$  for add and remove because it is either a max heap or min heap. For min heaps the parent is always smaller than the children and the order of the children don't matter. This makes it so each step of the adjustments will divide amount of data to look over in half.
2. If a heap received multiple copies of the same value, FIFO order would still be observed with those values. This is because the first value input into the heap would always sit higher on the tree than the second value. So, when removing the first value will percolate up first.
3. Create two heaps using the first two values. The larger value being a max heap root and the smaller being the min heap root. Add elements smaller than the max root to the max heap and elements larger than the min root to the min heap. If one heap has more than one element than the other, remove the root from that heap and add it to the other. When the heaps have equal elements the average of the two roots is the median. If there is not a even number of values, then the median is the root of the heap that has more elements.
4. Binomial Heap