Report for assignment 6

Manne Naga Nithin (14EE10026)

February 12, 2016

Creating an Interval Tree

1. **Function Outline**
   A list is created using the values of l, u and n.
   $size = (u - l + 1)/n$
   $remainder = (u - l + 1)\%n$
   $listnode- > l = prenode- > u + 1$
   $listnode- > u = node- > l + size + ((remainder - -) > 0)?1 : 0$
   A recursive function is run using this list, this function creates a tree node with the middle element in the list and recursively runs the function for the remaining two halves of the list and assigns them to left and right pointers of this middle node.
   $listnode- > left = createtree(list)$
   $listnode- > right = createtree(mediannode- > nextnode)$

2. **Time Complexity**
   For creating the list,
   $T(n) = O(n)$
   For making the tree,
   $T(n) = 2 * T(n/2) + O(n)$
   $T(n/2)$ from the recursive call
   $O(n)$ from traversing the list to find the median node
   Therefore, $T(n) = O(nlog(n))$