DAY 4

Question 1

In the Binary Search algorithm, it is suggested to calculate the mid as

beg + (end - beg) / 2 instead of (beg + end) / 2. Why is it so?

mid = (beg+end)/2 is a correct logic in majority of the cases but if (beg+end) approches maximum int size it may yield -1 as a result of integer capacity overflow, which will as a result cause error in the mid point claculation.

So it is suggested to use mid = beg+(end-beg)/2 as it avoids any condition for integer overflow.

Question 2

Write the algorithm/function for Ternary Search.

int ternary\_search(int l,int r, int x)

{

if(r>=l)

{

int mid1 = l + (r-l)/3;

int mid2 = r - (r-l)/3;

if(ar[mid1] == x)

return mid1;

if(ar[mid2] == x)

return mid2;

if(x<ar[mid1])

return ternary\_search(l,mid1-1,x);

else if(x>ar[mid2])

return ternary\_search(mid2+1,r,x);

else

return ternary\_search(mid1+1,mid2-1,x);

}

return -1;

}