Q.32 T(n) = 0 if n = 1

T(n) = T(n-1) + n - 1 if n ≥ 2

= T(n-1) + n - 1

= T(n-2) +n-2 +n -1

=T(n-2) + 2n -3

=T(n-3) + n -3 + 2n -2

=T(n-3) + 3n – 5

=T(n-4) + n -4 +3n -5

=T(n-4) + n-1 + n-2 + n-3 + n-4

= (n^2 – n) / 2

= O(n^2)

Q.8 T(n) = 7* T * (n/2) + 500 * n2

Here a=7, b=2, d= 2 (f(n) = 500 * n2)

So, b^d = 2^2 = 4

Therefore, a > b^d

So, T(n) = O(n^ log2(2))

Q.5 Binary Search (Search 76)

6	12	28	29	45	54	62	76
l=0			**m=3**				**r= 7**

Here, m = l + (r- l)/2

So, m = 0 + (7-0)/2

m= 3.5 (Approx. 3)

76 is at 7th Index.

Q. 10 Merge Sort