Question 2

a) Time complexity

T(1) = 1

T(n) = 3T(n/3) + n

T(n/3) = 3T(n/9) + n/3

T(n) = 3(3T(n/9) + n/3) + n = 3^2 * T(n/(3^2)) + 2n

= 3^i * T(n/(3^i)) + i*n

⇨ $T(n) = n + n\log_3(n)$

$0(n\log_3(n))$

b) 3 way has better time complexity because $\log_3 n < \log_2 n$

c) in-place merge sort algorithm

int arr[]

int a = arr[0], int b = arr[1]

if(a < b) increase position of a by 1

else increase position of a and b by 1