Design and Analysis of Algorithm (CS 412)
Instructor: Dr. Ayesha Enayet
Date: _______________
CS 6^{th}

SIS ID:________________ Name:_________________

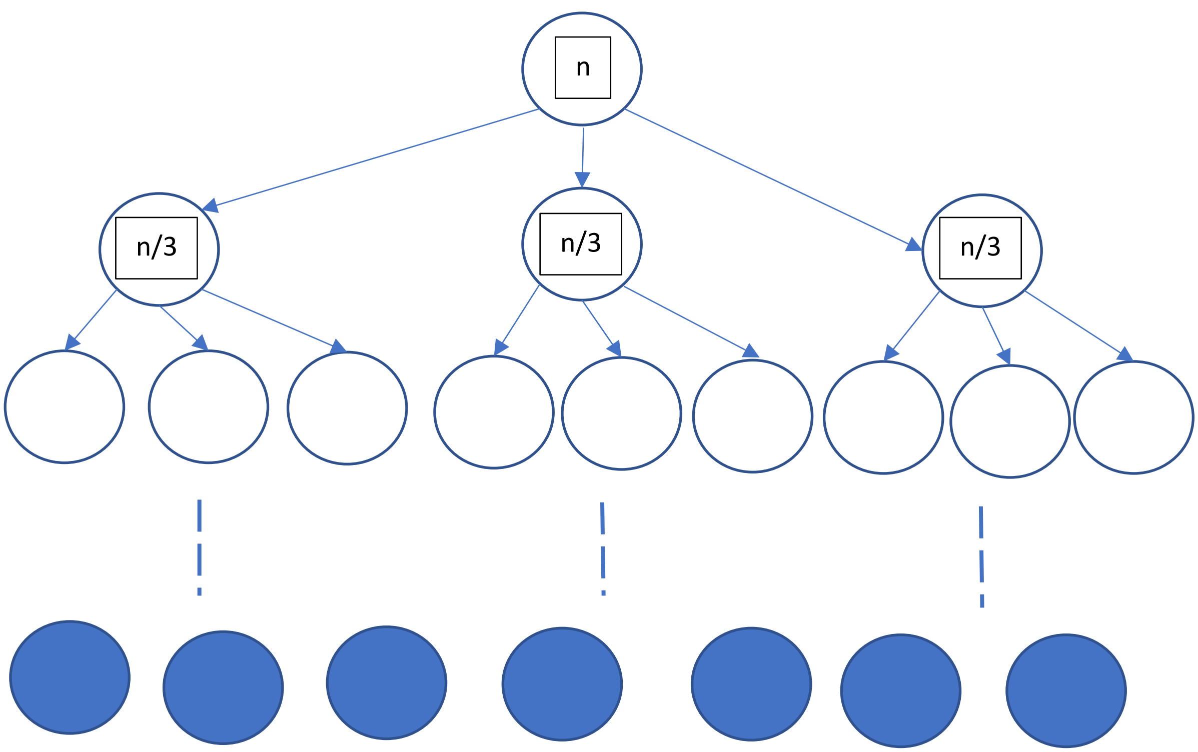
Note: Attempt all the questions

A. Choose the correct answer

1. The best-case complexity of the merge sort is __________________. [1]
   a) $\Omega(n)$ b) $\Omega(nlgn)$
   c) $\Omega(n^2)$ d) $\Omega(2^n)$
2. The solution to a recurrence T(n)=2T(n/2) +1 is: [1]
   a) $O(n)$ b) $O(nlgn)$
   c) $O(n^2)$ d) $O(2^n)$

B. Consider the following recurrence tree of an algorithm, where the size of each subproblem is n/3 and the complexity of the driving function f(n) is O(n):

1. Find out the worst-case time complexity of the algorithm. [1]
2. Verify your solution with the help of the Master theorem. [1]
3. The total cost of all the internal nodes of the tree (including the root node)?[Hint: use summation to define the solution] [1]



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

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

T(n)= 3[3T(n/3^2)+n/3]+n

T(n)= 3[3T(n/3^2)]+n+n

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

……

T(n)= 3^kT(n/3^k)]+k.n

K=$lg_3 n$

T(n)= 3^kT(n/3^k)]+n.($lg_3 n$)

T(n)=O(n.($lg_3 n$))

Case 2:f(n)= $n^{lg_b a}$

- *If y>-1 then T(n)=Θ($n^x lg^{y+1}$n)*
- T(n)= *Θ(nlgn)*

$$\sum_{k=0}^{lg_3 n-1} 3^k\, T(\frac{n}{3^k})$$