National University of Computer and Emerging Sciences, Lahore Campus

	Course Name:	Design and Analysis of Algorithms	Course Code:	CS2009
	Degree Program:	BSCS	Semester:	SPRING 2023
	Quiz Date:	Monday, February 15, 2023	Total Marks:	10
	Section:	ALL	Page(s):	1
	Exam Type:	Quiz 1		

Student : Name:_____ Roll No._____ Section:_____

Instruction/Notes : Solve your own quiz.

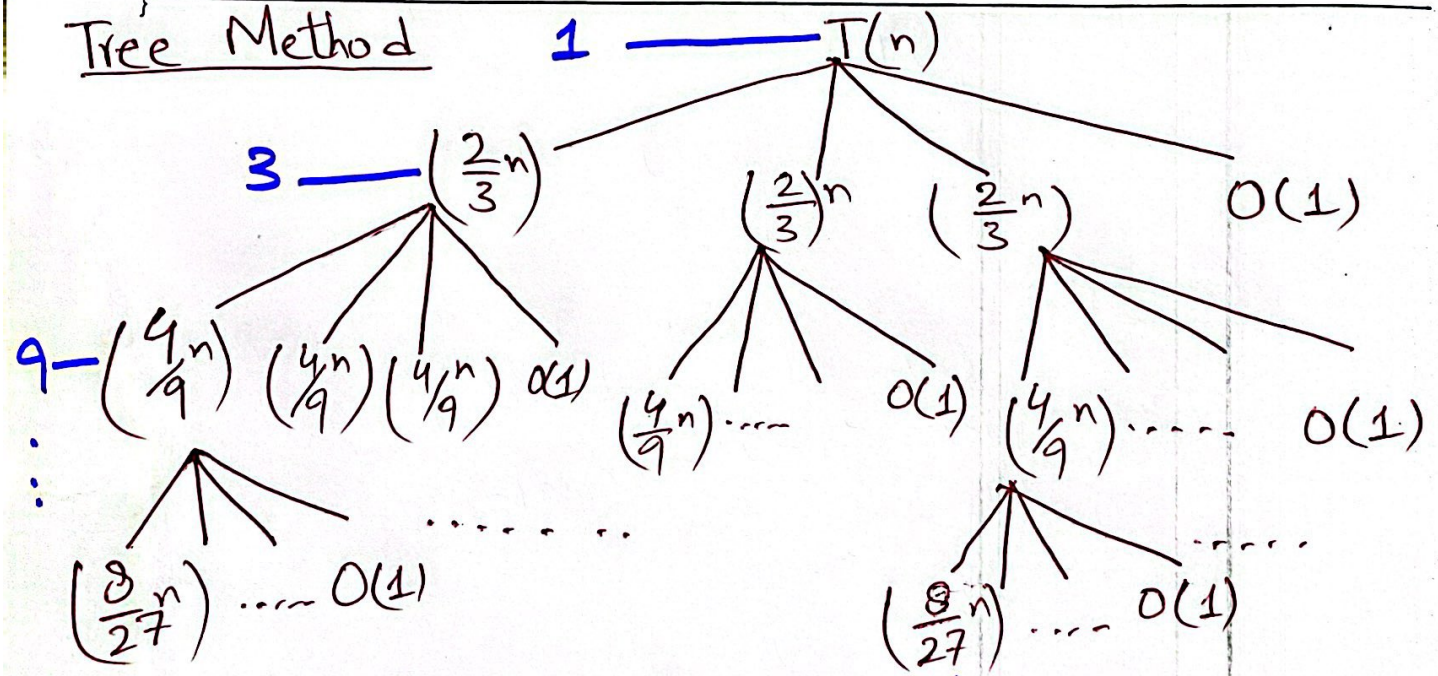
Pseudocode of Stooge-Sort is given below. Derive the time complexity of Stooge-Sort

```
STOOGE-SORT (left, right) {
        IF (A[left] > A[right])
            SWAP A[left] with A[right]
        IF (right-left > 1)              RETURN
        ELSE
            third = (right-left+1)/3
            STOOGE-SORT (left + third, right)
            STOOGE-SORT (left, right - third)
            STOOGE-SORT (left + third, right)
}
```

Recurrence Relation:

$$T(n) = 3T\left(\frac{2}{3}n\right) + O(1)$$

Tree Method



$3^0 + 3^1 + 3^2 + \dots + 3^k$

Geometric Series Sum $= \frac{a(r^k - 1)}{r - 1}$

$a = 1,\ r = 3,\ k = \log_{3/2} n$

$\left(\frac{2}{3}\right)^k n = 1$

$n = \left(\frac{3}{2}\right)^k$

$K = \log_{3/2} n$

$3^{\log_{3/2} n} = n^{\log_{3/2} 3} = n^{2.709}$

National University of Computer and Emerging Sciences, Lahore Campus

	Course Name:	Design and Analysis of Algorithms	Course Code:	CS2009
	Degree Program:	BSCS	Semester:	SPRING 2023
	Quiz Date:	Monday, February 15, 2023	Total Marks:	10
	Section:	ALL	Page(s):	1
	Exam Type:	Quiz 1		

Student : Name:_____ Roll No._____ Section:_____

Instruction/Notes : Solve your own quiz.

Pseudocode of Stooge-Sort is given below. Derive the time complexity of Stooge-Sort

```
STOOGE-SORT (left, right) {
        IF (A[left] > A[right])
            SWAP A[left] with A[right]
        IF (right-left > 1)              RETURN
        ELSE
            third = (right-left+1)/3
            STOOGE-SORT (left + third, right)
            STOOGE-SORT (left, right - third)
            STOOGE-SORT (left + third, right)
}
```

Recurrence Relation

$T(n) = 3T(\frac{2}{3}n) + O(1)$

Master Theorem $T(n) = aT(\frac{n}{b}) + O(n^c)$

here $a = 3$, $b = \frac{3}{2}$, $c = 0$

$(a >= 1, \quad b > 1, \quad c >= 0)$

when $a > b^c$, we have solution

$O(n^{\log_b a})$

$O(n^{\log_{3/2} 3}) = O(n^{2.709})$