SAN FRANCISCO STATE UNIVERSITY
Computer Science Department

CSC510 – Analysis of Algorithms
Extracredit Algorithm Challenge

Instructor: Jose Ortiz

Full Name: Edel Jhon Cenario
Student ID: 921121224

Assignment Instructions. Must read!

Note: Failure to follow the following instructions in detail will impact your grade negatively.

1. This algorithm challenge is worth 5% extracredit that will be added to the student's final grade at the end of the semester.
2. **No partial credit will be given for this algorithm challenge**. Credit will only be given for correct answers.

Your Work Here

1. Given the following pseudocode and assuming that the print is the basic operation, (1) set up the initial recurrence as a function of n, (2) solve the recurrent equation using the back substitution method, and (3) check your results with the Master Theorem.. **Note that this problem must be solved by using only Back Substitution and Master Theorem approaches. Other approaches than the ones mentioned above won't be considered for credit**

Algorithm 1 Extracredit Algo Challenge

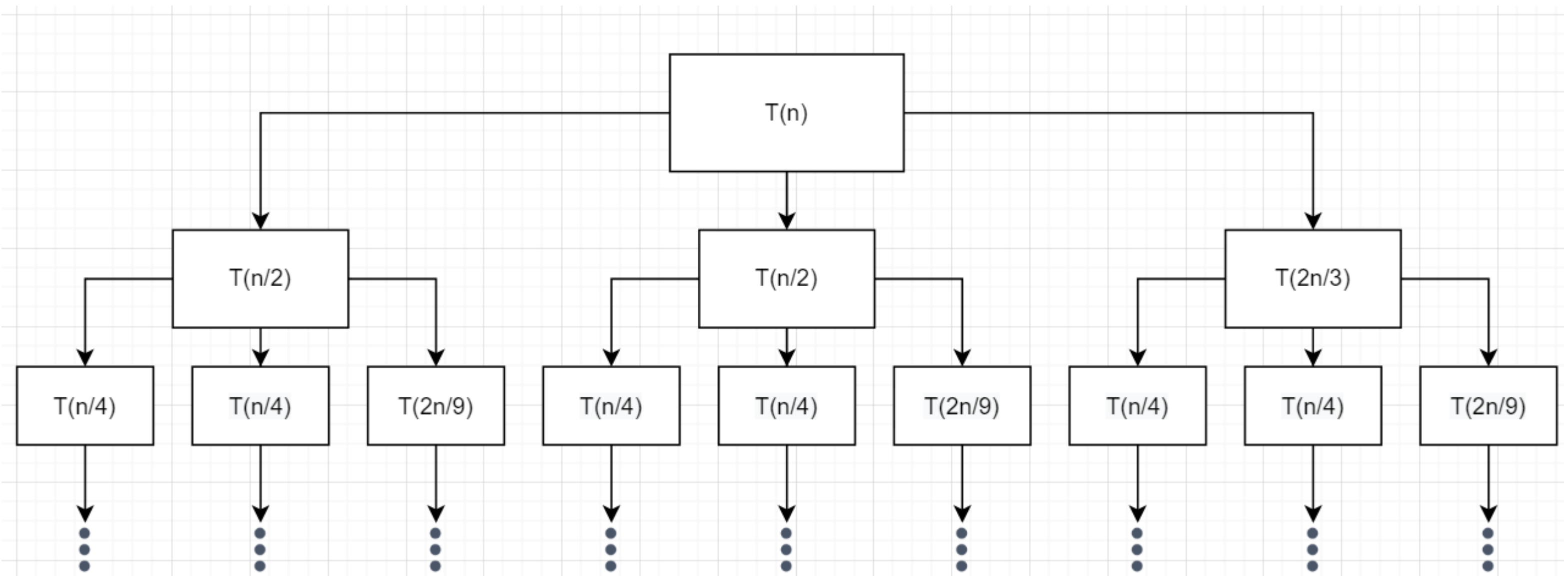
```
function EXTRACREDIT(n)
    INITIALIZE: i=0, j=0, k=0                    ▷ Indexes for loops
    if n ≤ 1 then
        Return                                   ▷ Base condition
    while i < n do
        while j < n do
            while k < n do
                Print("CSC510")
                INCREMENT: k + 1
            INCREMENT: j + 1
        INCREMENT: i + 1
    RECURSIVE CALL: EXTRACREDIT(n/3)
    RECURSIVE CALL: EXTRACREDIT(n/3)
    RECURSIVE CALL: EXTRACREDIT(2n/3)
```

Let us take the pseudocode and understand it better.
By deriving an equation from our pseudocode, it will allow us to create a tree which will give us a better idea of the process.

Let us take the equation:

$$T(n) = 2T + T(\ 2n/3\ ) + 3n$$

With that we can create our tree



Let us examine our tree,

We can see that we have a combination of 1/2 and 2/3 powers.
And with that we can derive:

$$T(n) = \sum_{i=1}^{k} T(\tfrac{1}{2}^{i} * \tfrac{2}{3}^{k-i} n)$$ At each level of tree we have 3n operations.