CSCI 3104 Spring 2022
Instructor: Profs. Chen and Layer

Problem Set 5

Due Date .... March 1
Name .... **Your Name**
Student ID .... **Your Student ID**
Collaborators .... **List Your Collaborators Here**

Contents

Instructions

- The solutions **must be typed**, using proper mathematical notation. We cannot accept hand-written solutions. Useful links and references on LaTeXcan be found here on Canvas.
- You should submit your work through the **class Canvas page** only. Please submit one PDF file, compiled using this LaTeX template.
- You may not need a full page for your solutions; pagebreaks are there to help Gradescope automatically find where each problem is. Even if you do not attempt every problem, please submit this document with no fewer pages than the blank template (or Gradescope has issues with it).
- You are welcome and encouraged to collaborate with your classmates, as well as consult outside resources. You must **cite your sources in this document. Copying from any source is an Honor Code violation. Furthermore, all submissions must be in your own words and reflect your understanding of the material.** If there is any confusion about this policy, it is your responsibility to clarify before the due date.
- Posting to **any** service including, but not limited to Chegg, Reddit, StackExchange, etc., for help on an assignment is a violation of the Honor Code.

1 Standard 14 - Analyzing Code III: (Writing down recurrences)

Problem 1. Write down the recurrence relation for the runtime complexity of these algorithms. **Don't forget to include the base cases.**

1.1 Problem 1(a)

Algorithm 1 Writing Recurrences 1

```
procedure Foo(Integer $n$)
    if $n \leq 3$ then return

    Foo($n/3$)
    Foo($n/3$)
    Foo($n/3$)

    for $i \leftarrow 1; i \leq 2 * n; i \leftarrow i + 1$ do
        print "Hi, Hi"
```

Answer. $T(n) = \left\{ \begin{array}{ll} \Theta(1) & n \leq 3 \\ 3T(n/3) + \Theta(n) & n > 3 \end{array} \right.$

□

1.2 Problem 1(b)

Algorithm 2 Writing Recurrences 2

```
procedure Foo(Integer $n$)
    if $n \le 4$ then return

    Foo($n/4$)
    Foo($n/4$)
    Foo($n/3$)

    for $i \leftarrow 1; i \le n; i \leftarrow i * 2$ do
        print "Hi, Hi"
```

Answer. $T(n) = \begin{cases} \Theta(1) & n \le 4 \\ 2T(n/4) + T(n/3) + \Theta(n) & n > 4 \end{cases}$ □

2 Standard 15 - Unrolling

Problem 2. Consider the following recurrences and solve them using the unrolling method (i.e. find a suitable function $f(n)$ such that $T(n) \in \Theta(f(n))$).

(a)

$$T(n) = \begin{cases} 2 & : n < 3, \\ 2T(n-3) + 2 & : n \geq 3. \end{cases}$$

Answer. First we determine the number of times that we need to unroll. We hit a base case when $n - 3k < 3$. Solving for k we get that.
$n < 3 + 3k$
$n - 3 < 3k$
$\frac{n-3}{3} < k$
So we need to unroll until $k = \lceil \frac{n-3}{3} \rceil$

When we unroll $T(n - 3i)$, we obtain $2T(n - 3(i + 1)) + 2$. So
$T(n - 3i) = 2 * \lceil 2T(n - 3(i + 2)) + 2 \rceil + 2 = 2^2 T(n - 3(i + 2)) + 2 * 2 + 2^0 * 2$.

Thus $T(n) =$ (cost of base case) * (number of times the base case is reached) + $\sum_{i=0}^{\lceil \frac{n-3}{3}} \rceil 2^i * 2$
$T(n) 1 * 2^{\lceil \frac{n-3}{3} \rceil} + 2 * \sum_{i=0}^{\lceil \frac{n-3}{3}} \rceil 2^i$
$T(n) = 2^{\lceil \frac{n-3}{3} \rceil} + 2 * \frac{2^{\lceil \frac{n-3}{3} \rceil + 1} - 1}{2 - 1}$
$T(n) = 2^{\lceil \frac{n-3}{3} \rceil} + 2 * 2^{\lceil \frac{n-3}{3} \rceil + 1}$
Taking the high-order term, we obtain that $T(n) \in \Theta 2^{\lceil \frac{n-3}{3} \rceil} = \Theta(2^{n/3})$ □

(b)

$$T(n) = \begin{cases} 3 & : n < 2, \\ T(n-2) + 4n & : n \geq 2. \end{cases}$$

Answer. We hit a base case when $n - 2k < 2$ solving for k we get that $k > (n-2)/2$ so we have to unroll until $\lceil k = (n-2)/2 \lceil$.
We will replace $4n$ with cn in this recurrence.

$$\begin{align} T(n) &= T(n-2) + cn \tag{1} \\ &= T(n-2) + c(n-2) + cn \tag{2} \\ &= T(n-4) + c(n-4) + c(n-2) + cn \tag{3} \\ Thiseventuallybecomes \tag{4} \\ &= T(n - 2 * \lceil (n-2)/2 \rceil) + 2c + 4c + ... + c(n-4) + c(n-2) + cn \tag{5} \\ Thus: \tag{6} \\ &= 3 + c * \sum_{i=0}^{\lceil (n-2)/2 \rceil} (n - 2i) \tag{7} \\ &= 3 + c * \sum_{i=0}^{\lceil (n-2)/2 \rceil} (n - 2c) \sum_{i=0}^{\lceil (n-2)/2 \rceil} i \tag{8} \\ &= 3 + nc * \lceil (n-2)/2 \rceil - 2c * \frac{\lceil (n-2)/2 \rceil * (\lceil (n-2)/2 \rceil + 1}{2} \tag{9} \\ &= 3 + nc * \lceil (n-2)/2 \rceil - c * [\lceil (n-2)/2 \rceil * (\lceil (n-2)/2 \rceil + 1] \tag{10} \end{align}$$

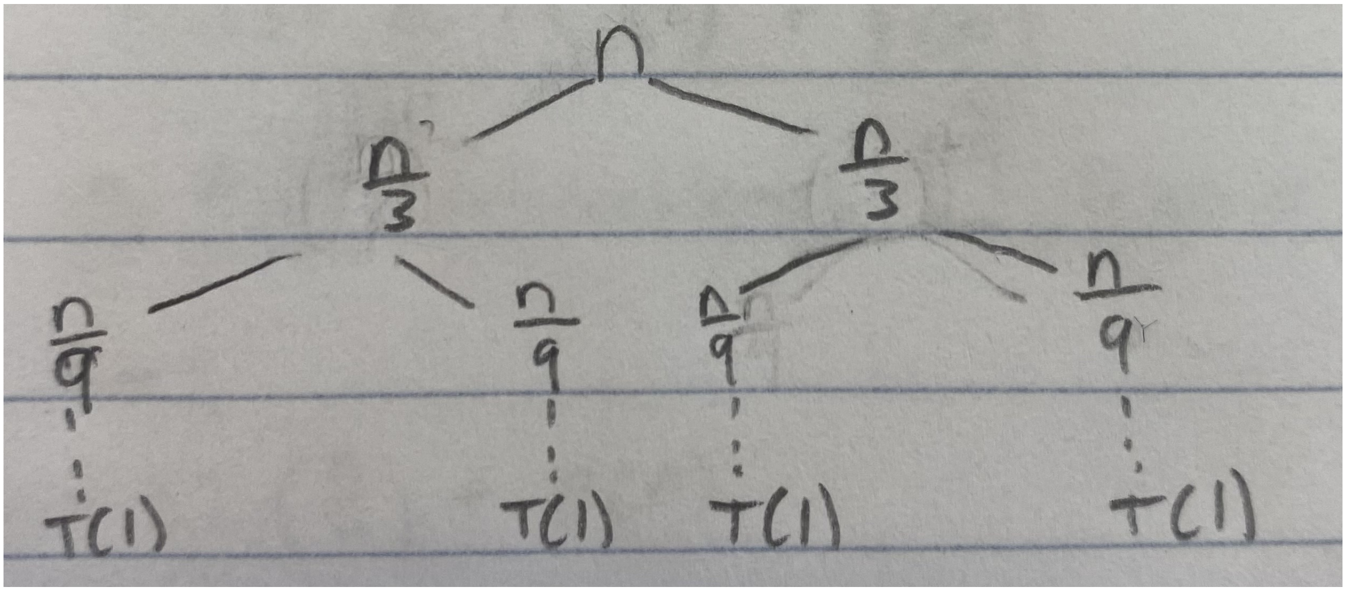
Thus, $T(n) \in \Theta(n^2)$ □

3 Standard 16 - Tree Method.

Problem 3. Consider the recurrence $T(n)$ below. Using the **tree method**, determine a suitable function $f(n)$ such that $T(n) \in \Theta(f(n))$. Clearly show all steps. Note the following:

- You may assume, without loss of generality, that n is a power of 3 (i.e., $n = 3^k$ for some integer $k \geq 0$).
- You may hand-draw your tree and embed it, provided it is legible and we do not have to rotate our screens to read it. However, **all your calculations must be typed**.

$$T(n) = \begin{cases} 2, & n < 3 \\ 2T(n/3) + n^2, & \text{otherwise.} \end{cases}$$



Answer.
First we have to determine the number of level of the tree. Let k denote the number of levels of the tree.
$\frac{n}{3^k}$
$k = \lceil \log_3(n) \rceil$
Therefore the total work done by the recurrence is.

$$T(n) \leq \sum_{i=0}^{\lceil \log_3(n) \rceil} n^2 \tag{11}$$
$$= n^2 \log_3 n \tag{12}$$

Thus $T(n) \in \Theta(n^2 \log_3 n)$ □