GU 2

CS 341 Assignment 1

Gautam Gupta
20588082

1.a Let $f(n) = n^{3.41} - 2017 n^{1.28} + 1$

$g(n) = n^{3}$

~~For~~ $f(n) \in \omega(g(n))$ if $\forall\, c > 0,\ \exists\, n_0 > 0$

s.t. $0 \le c\, g(n) < f(n) \quad \forall\, n \ge n_0$

or $c n^{3} < n^{3.41} - 2017 n^{1.28} + 1$

$\le n^{3.41} \qquad \forall\, n \ge n_0,\ n_0 > 0$

$\Rightarrow \quad c < n^{0.41}$

$\Rightarrow \quad n > c^{2.44}$

or choose n_0 s.t. $\underline{n_0 > c^{2.44}}$

& then $f(n) \in \omega\, g(n))$

__/__/__

1\. b) To prove: $\log_{b_1}(n) = \Theta(\log_{b_2}(n))$ $\quad b_1, b_2 > 1$

Proof: $\log_{b_1} n = \dfrac{\log_{b_2} n}{\log_{b_2} b_1} \rightarrow$ Constant

$\therefore \log_{b_1} n \in \Theta(\log_{b_2} n)$

c) To prove: $n^{\log_{b_1} a} \notin O(n^{\log_{b_2} a})$ $\quad a, b_1, b_2 > 1$

Proof by contradiction

Claim: $n^{\log_{b_1} a} \in O(n^{\log_{b_2} a})$

Let $a = 8,\ b_1 = 2,\ b_2 = 8$

$n^{\log_{b_1} a} = n^{\log_2 8} = n^3$

$n^{\log_{b_2} a} = n^{\log_8 8} = n$

But $n^3 \notin n$

$\therefore$ this is a contradiction

& $n^{\log_{b_1} a} \notin O(n^{\log_{b_2} a})$

d) i Biggest term in $f(n)$: n^{2017}

$g(n)$: $\underbrace{2^{2017}}_{\text{constant}} \cdot n^{2017}$

$\therefore f(n) \in \underline{\Theta}\ g(n)$

ii Biggest term in $f(n)$: $300n^5$

$g(n)$: $4n^6$

$\therefore f(n) \in \underline{o}\ g(n)$

iii Factorial terms dominate in both functions but $f(n)$'s $(n+1)! > g(n)$'s $n!$

$\therefore f(n) \in \underline{\omega}\ g(n)$

iv $\log(f(n)) = \sqrt{n} \cdot \log n$ $\qquad \log(g(n)) = n/2$

Clearly $\sqrt{n} \log n < n$

$\therefore f(n) \in \underline{o}\ g(n)$

A1 Q2-4

Q2. a

Lines	Complexity
(5)	$\theta(1)$
(4) - (6)	$\theta(n+3)$
(3) - (7)	$\theta\left(\sum_{k=1}^{n+3}(n+3)\right)$ *or* $\theta(n^2+9+6n)$
(8) - (10)	$\theta(i^3-3+1)$ *or* $\theta(i^3-2)$
(3) - (10)	$\theta(n^2+6n+i^3+7)$
(2) - (11)	$\theta\left(n^2+6n+\sum_{j=1}^{i} i^3+7\right)$ *or* $\theta(n^2+6n+7+i^4)$
(1) - (12)	$\sum_{i=1}^{n}\theta(n^2+6n+7+i^4)$ *or* $\theta\left(\frac{1}{30}\begin{matrix}(6n^5+15n^4+40n^3\\+180n^2+209n)\end{matrix}\right)$ or $\boldsymbol{\theta(n^5)}$

b

Lines	Complexity
(1), (3), (5), (8)	$\theta(1)$
(3) - (7)	$j=n, n-n^{\frac{1}{3}}, n-2n^{\frac{1}{3}}, \ldots, n-(n^{\frac{2}{3}}-1)*n^{\frac{1}{3}}$ $n^{\frac{2}{3}}$ *iterations or* $\theta(n^{\frac{2}{3}})$
(1) - (9)	$i=n^3, \frac{n^3}{3}, \frac{n^3}{9} \ldots, 1$ *or* $(\log_3 n^3+1)$ *iterations* $\theta\left(n^{\frac{2}{3}}*3*\log_3 n\right)$ *or* $\theta\left(n^{\frac{2}{3}}*\log n/\log 3\right)$ *or*

	$\theta\left(n^{\frac{2}{3}} * \log n\right)$

Q3.

Algorithm

```
1  L: [l_1,l_2,…,l_k]                            // individually sorted
2  n: total number of elements in all lists
3  k: total number of lists
4  returns sorted merged list
5  function mergeLists(L, n, k) {
6    newList = []
7    for i = 0 to n {
8      for j = 0 to k {
9        if L[j][0] < nextElem {
10         nextElem = L[j].shift()  // pop first element
11       }
12     }
13
14     newList.push(nextElem)
15   }
16   return newList
17 }
```

Correctness

Invariant for loop on i:

```
newList[i] <= newList[j]    where 0 <= i < j < n
L[m] is sorted              for all 0 <= m < k
```

Analysis

Lines	Complexity
(6), (9)-(11), (14), (16)	$\theta(1)$
(8) - (12) for loop on j	$\theta(k)$
(5) - (17) for loop on i	$\theta\left(\sum_{i=0}^{n} k\right) or$ $\theta(nk)$

Divide and Conquer

```
L: [l_1,l_2,…,l_k]                          // k sorted lists
returns: A sorted merged list
function mergeLists(L, k) {
  if (L.len == 1) {
    return L[0]
  } else {
    mid = floor(k / 2)
    L1 = mergeLists(L[0..mid])
    L2 = mergeLists(L[(mid+1)..k])
    return mergeSortedLists(L1, L2)
  }
}

L1, L2: Two sorted lists
returns: A merged sorted list
runtime: Θ(max(L1.len, L2.len))
function mergeTwoSortedLists(L1, L2) {
  newList = []
  i, j = 0, 0
  while (i < L1.len && j < L2.len) {
    if (L1[i] < L2[j]) {
      newList.push(L1[i])
      i++
    } else {
      newList.push(L2[j])
      j++
    }
  }
  // Add leftover elements
  newList.pushElements(L1[i..(L1.len)])
  newList.pushElements(L2[j..(L2.len)])
  return newList
}
```

Q4.

```
grid: 2D, sorted left-right & top-bottom
m: # of rows in grid
n: # of cols in grid
num: To check the existence of in the grid
returns: true if num exists in grid, false if not
function numExistsInGrid(grid, m, n, num) {
```

```
7    i, j = m-1, 0             // start at bottom left
8    while (i >= 0 && j < n) {
9      if (num == grid[i][j])
10       return true
11     else if (num > grid[i][j])
12       j++
13     else
14       i--
15     end
16   }
17   return false
18 }
```

Correctness

If we start at the bottom left of the grid,

- If `num` is equal to grid cell, we found its existence
- If `num` is greater than the grid cell, we go one column right as that's where we will find bigger numbers (and we have already excluded numbers below)
- If `num` is less than the grid cell, we go one row up as numbers in the current row on the right are greater than `num` (and we have already excluded numbers on the left)

Analysis

Lines	Complexity
(7), (9)-(15), (17)	$\theta(1)$
(6) - (18) while loop on i and j	$\theta(m+n)$

__/__/____

5. a) $T(n) = \begin{cases} 3T(n/5) + \sqrt{n} & n > 1 \\ 7 & n \leq 1 \end{cases}$ $\qquad c = 1$

Recursive Call	# nodes		
$T(n)$	1	$c\sqrt{n}$	$\rightarrow \sqrt{n}$
$T\left(\frac{n}{5}\right)$	3	$c\sqrt{\frac{n}{5}}$ $c\sqrt{\frac{n}{5}}$ $c\sqrt{\frac{n}{5}}$	$\rightarrow \frac{3}{\sqrt{5}}\sqrt{n}$
$T\left(\frac{n}{5^2}\right)$	3^2	$c\sqrt{\frac{n}{25}}$ $c\sqrt{\frac{n}{25}}$ $c\sqrt{\frac{n}{25}}$ $c\sqrt{\frac{n}{25}}$ $c\sqrt{\frac{n}{25}}$ $c\sqrt{\frac{n}{25}}$ $c\sqrt{\frac{n}{25}}$ $c\sqrt{\frac{n}{25}}$ $c\sqrt{\frac{n}{25}}$	$\rightarrow \frac{9}{5}\sqrt{n}$
$T\left(\frac{n}{5^i}\right)$	3^i		$\left(\frac{3}{\sqrt{5}}\right)^i\sqrt{n}$
$T(1)$	$3^{\log_5 n}$ or $n^{\log_5 3}$	$T(1)$ $T(1)$ $T(1)$ $T(1)$ $T(1)$ $T(1)$ $T(1)$ $T(1)$ $T(1)$	$\rightarrow \Theta\left(n^{\log_5 3}\right)$

$h = \log_5 n$

$$\therefore T(n) = \sum_{i=0}^{\log_5 n - 1} \left(\frac{3}{\sqrt{5}}\right)^i \sqrt{n} + \Theta\left(n^{\log_5 3}\right)$$

$$= \sqrt{n}\,\Theta\left(\left(\frac{3}{\sqrt{5}}\right)^{\log_5 n}\right) + \underbrace{\Theta\left(n^{\log_5 3}\right)}_{\text{dominates}}$$

$$\therefore T(n) \in \Theta\left(n^{\log_5 3}\right)$$

b) $T(n) = 5T(2n/5) + n^2 \qquad n > 1$

$T(n)$	n^2	n^2
$T\left(\frac{2}{5}n\right)$	$\left(\frac{2}{5}n\right)^2$ $\left(\frac{2}{5}n\right)^2$ $\left(\frac{2}{5}n\right)^2$ $\left(\frac{2}{5}n\right)^2$ $\left(\frac{2}{5}n\right)^2$	$n^2 \cdot 5 \cdot \frac{4}{25}$
$T\left(\left(\frac{2}{5}\right)^2 n\right)$	$\left(\frac{4}{25}n\right)^2 \left(\frac{4}{25}n\right)^2 \ldots$	$5 \cdot \left(\frac{2}{5}\right)^{2i} n^2$
$h = \log_{2.5} n$	$T(1)$ $T(1)$ $T(1)$ $T(1)$ $T(1)$	$\Theta\left(n^{\log_{2.5} 5}\right)$

$$\therefore T(n) = \sum_{i=0}^{\log_{2.5} n - 1} 5n^2\left(\frac{2}{5}\right)^{2i} + \Theta\left(n^{\log_{2.5} 5}\right)$$

$$= \underbrace{n^2\,\Theta\left(2^{\log_{2.5} n}\right)}_{\text{dominates}} + \Theta\left(n^{\log_{2.5} 5}\right)$$

$$\therefore T(n) \in \Theta\left(n^2\right)$$

__/__/___

6. a $T(n) = \begin{cases} 3T(\frac{n}{5}) + \sqrt{n} & n \geq 1 \\ 7 & n \leq 1 \end{cases}$

$a = 3, \ b = 5, \ y = \frac{1}{2}, \ \alpha = \log_5 3 \approx 0.863$

Since $y < \alpha$, by MT,

$T(n) \in \underline{\Theta(n^{\log_5 3})}$

b $T(n) = \begin{cases} 5T(\frac{2}{5}n) + n^2 & n > 1 \\ 1 & n \leq 1 \end{cases}$

$a = 5, \ b = 5/2, \ y = 2, \ \alpha = \log_{2.5} 5 \approx 1.757$

Since $y > \alpha$, by MT

$T(n) \in \underline{\Theta(n^2)}$