German University in Cairo
Faculty of Media Engineering and Technology
Dr. Amr Desouky

CSEN 703 Analysis and Design of Algorithms
Winter Semester 2014
Quiz 1 Solution

October 21^{st}, 2014

Name: ....

ID: ....

Tutorial: ....

Instructions: Read carefully before proceeding.

1. Write your name, ID and tutorial number in the provided space above.
2. No books or any extra aids are permitted for this quiz.
3. When you are told that time is up, stop working on the quiz.

Good Luck! :)

Exercise	1	2	3	Total	Bonus
Grade					
Max Grade	5	10	5	20	5

German University in Cairo
Faculty of Media Engineering and Technology
Dr. Amr Desouky

Exercise 1: Asymptotic Notations (5 marks = 2.5 + 2.5)

Prove or disprove the following:

a. $2^{n+100} = O(2^n)$

Solution:

Using the limit test:

$lim_{n\to\infty} \frac{2^n.2^{100}}{2^n} = 2^{100} \in R^+$

Therefore, $f(n) = \Theta(g(n))$

Since $f(n) = \Theta(g(n)) \implies f(n) = O(g(n))$

Therefore, $f(n) = O(g(n)$

b. $2^{2n} = O(2^n)$

Solution:

Using the limit test:

$lim_{n\to\infty} \frac{2^2n}{2^n} = 2^n = \infty$

Therefore, $f(n) \neq O(g(n))$

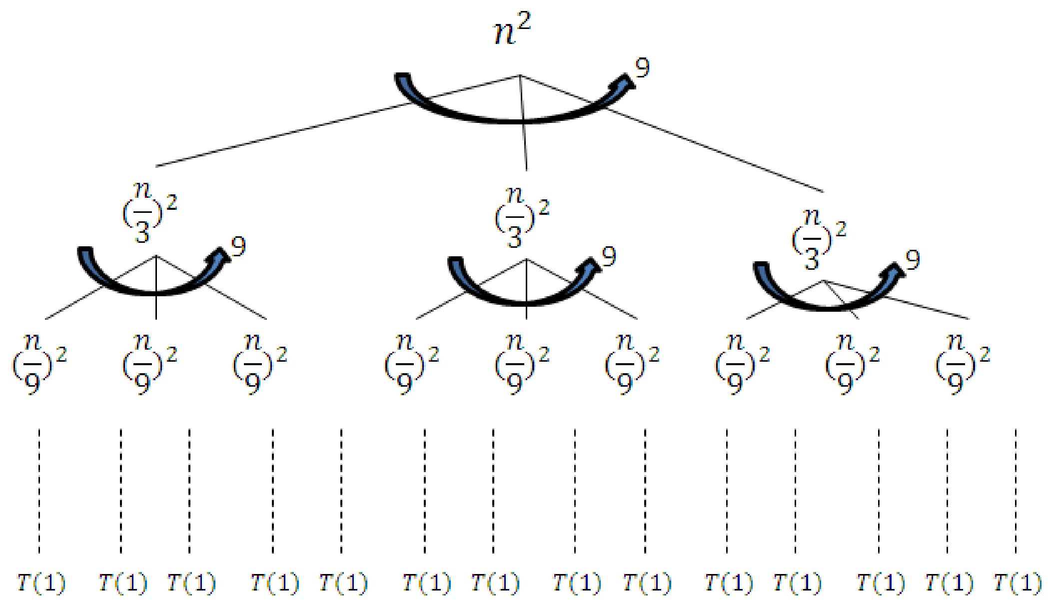
German University in Cairo
Faculty of Media Engineering and Technology
Dr. Amr Desouky

Exercise 2: Solving Recurrences (12.5 marks = 5 + 5 + 2.5 (bonus))

a. Use a recursion tree to determine an upper bound on the following recurrence:

$$T(n) = 9T(\frac{n}{3}) + n^2$$

Solution:



Height of the tree $= log_3(n)$.
Cost of each level in the tree $= n^2$.
Total cost $=$ Cost of each level $\times$ height of the tree $= n^2 log_3(n)$.
Therefore, $T(n) = O(n^2 log_3(n))$.

b. Use the substitution method to verify your answer.

Solution:
Base Case: $T(1) = O(1)$.
Induction hypothesis: $T(k) = k^2 log_3(k) \quad \forall k. 1 \leq k < n$
Induction step: $T(n) = 9T(\frac{n}{3}) + n^2$
$T(n) = 9[(\frac{n}{3})^2 log_3(\frac{n}{3})] + n^2 = n^2 log_3(\frac{n}{3}) + n^2 = n^2(log_3(n) - 1) + n^2$
$T(n) = n^2 log_3(n) \leq cn^2 log_3(n) \quad \forall c \geq 1$
Therefore,$T(n) = O(n^2 log_3(n)) = O(n^2 lg(n))$.

c. **Bonus:** Can you use the master method to solve the given recurrence? If yes, use the master theorem to prove the upper bound you obtained in (a). Otherwise, explain why the master theorem can't be used to solve the recurrence.
Solution:
Yes, the master theorem can be used to solve the given recurrence.
$a = 9, b = 3, f(n) = n^2$
Cost of leaves $= n^{log_b(a)} = n^{log_3(9)} = n^2$
Therefore, we are in case 2.
Total Cost $= T(n) = \Theta(n^2 lg(n))$

Exercise 3: Fibonacci Numbers (7.5 marks = 5 + 2.5 (bonus))
Consider the following divide-and-conquer approach for calculating the n^{th} fibonacci number in the fibonacci sequence:

```
function FIB(int n)
    if n ≤ 2 then
        return 1
    else
        return FIB(n-1)+FIB(n-2)
    end if
end function
```

a. Write a recurrence expressing the above pseudo code.
Solution:

$$T(n) = \begin{cases} T(n-1) + T(n-2) + \Theta(1) & \text{if } n > 2; \\ \Theta(1) & \text{if } n \leq 2. \end{cases}$$

b. **Bonus:** Can you use the master theorem to solve the recurrence that you obtained in (a)? If yes, solve it. Otherwise, explain why the master theorem can't be used to solve the recurrence.
Solution:
No, we can't use the master method to solve the above recurrence as it is not in the form $aT(\frac{n}{b}) + f(n)$ where b>1.

German University in Cairo
Faculty of Media Engineering and Technology
Dr. Amr Desouky

<u>**Useful Formulas:**</u>

Logarithms:

- $lgn = log_2(n)$
- $lnn = log_e(n)$
- $lg^k(n) = (lg(n))^k$
- $log_c(ab) = log_c(a) + log_c(b)$
- $log_c(\frac{a}{b}) = log_c(a) - log_c(b)$
- $log_b(a^n) = nlog_b(a)$
- $a^{log_b(c)} = c^{log_b(a)}$
- Logarithmic change of base: $log_b(a) = \frac{log_c(a)}{log_c(b)}$

Summations:

- $\sum_{i=j}^{k} a = a(k - j + 1)$
- Arithmetic series: $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$
- Finite Geometric series: $\sum_{i=0}^{n-1} r^i = \frac{1-r^n}{1-r}$