Analysis of Algorithms (Summer 2012)

Home Work # 1

Start date: Friday, 27th of July 2012

Submission: Monday, 6th of August 2012, (In Class)

Note-1: No homework will be accepted after class.

Note-2: A group of 4 students are allowed so discuss/solve the homework. But after discussion, every student has to write the solution on his/her own without getting help of the group members and submit the homework individually . You should not have group-discussion notes in front of you when you will be writing the solutions.

Note-3: Write the name of group members who collaborate in the solution-discussion.

Note-4: Do not write the solution if you cannot explain it orally. I will randomly pick any solution and ask for explanation. Getting failed in explaining the written-solution may cause the cancellation of the whole homework.

Note-5: Start early, no extensions will be entertained.

1. **[30]** Prove the following

   i. Suppose $a \in Z$. If $a^2 - 2a + 7$ is even, then a is odd.
   ii. Product of any five consecutive integers is divisible by 120.
   iii. Suppose $a, b \in Z$. If $4|(a^2 + b^2)$, then a and b are not both odd.
   iv. At least one of the real numbers $a_1, a_2, a_3, \ldots, a_n$ is greater than or equal to the average of these numbers.
   v. Prove that there are no integers $a > 1$ and $n > 0$ such that $a^2 + 1 = 2^n$
   vi. Prove that $\sqrt{2} + \sqrt{3}$ is irrational.

2. **[15]** Prove the following using induction

   i. A set of n elements has exactly $\frac{n(n-1)(n-2)}{6}$ subsets containing exactly three elements, for all integers $n \geq 3$
   ii. 21 divides $4^{n+1} + 5^{2n-1}$ for all integers $n \geq 1$
   iii. 3 divides $n^3 + 2n$ for all integers $n \geq 1$

3. **[10]** Give an asymptotically tight bound for the summation $\sum_{k=0}^{n} k2^k$

4. **[20]** Exercise 3-3(a) from CLRS

3-3 Ordering by asymptotic growth rates

a. Rank the following functions by order of growth; that is, find an arrangement $g_1, g_2, \ldots, g_{30}$ of the functions satisfying $g_1 = \Omega(g_2)$, $g_2 = \Omega(g_3)$, $\ldots$, $g_{29} = \Omega(g_{30})$. Partition your list into equivalence classes such that $f(n)$ and $g(n)$ are in the same class if and only if $f(n) = \Theta(g(n))$.

$\lg(\lg^* n)$	$2^{\lg^* n}$	$(\sqrt{2})^{\lg n}$	n^2	$n!$	$(\lg n)!$
$(\frac{3}{2})^n$	n^3	$\lg^2 n$	$\lg(n!)$	2^{2^n}	$n^{1/\lg n}$
$\ln \ln n$	$\lg^* n$	$n \cdot 2^n$	$n^{\lg \lg n}$	$\ln n$	1
$2^{\lg n}$	$(\lg n)^{\lg n}$	e^n	$4^{\lg n}$	$(n+1)!$	$\sqrt{\lg n}$
$\lg^*(\lg n)$	$2^{\sqrt{2 \lg n}}$	n	2^n	$n \lg n$	$2^{2^{n+1}}$

5. **[25]** Find tight asymptotic bounds for the following:

a. $T(1) = 1$, and $T(n) = T\left(\frac{n}{2}\right) + T\left(\frac{n}{3}\right) + T\left(\frac{n}{6}\right) + n$

b. $T(2) = 2$, and $T(n) = 2T\left(\frac{n}{2}\right) + \frac{n}{\lg n}$

c. $T(2) = 1$, and $T(n) = 2T(\sqrt{n}) + 1$

d. $T(2) = 1$, and $T(n) = \sqrt{n}T(\sqrt{n}) + n$

e. $T(1) = 1$, and $T(n) = \frac{1}{4}T\left(\frac{n}{4}\right) + \frac{3}{4}T\left(\frac{3n}{4}\right) + 1$

6. **[10]** Consider a queer version of merge sort in which we divide the array at the middle for "k" steps ("k" constant), then use bubble sort to sort the sub-arrays at that level, and then go back up the tree, merging. What is the running time, in theta notation, for this queer algorithm?

7. **[15]** Solve the following problem:

15 pts An $(N-1)$ by M matrix contains 0's and 1's only. Each row is a binary representation of an integer from the set $\{1, \dots, N\}$, with no number repeating. But since there are only $N-1$ rows, one of the numbers from $\{1, \dots, N\}$ is missing. The rows are <u>not</u> necessarily sorted according to the numbers they represent. Report the missing number in $O(N)$. [Hint: Since the rows are representing integers in the range $\{1, \dots, N\}$, we have $M \approx \lg N$, which is the time it will take you to compare two rows, if you wished to do so].

8. **[10]** Describe a $\Theta(n \lg n)$-time algorithm that, given a set S of n integers and another integer x, determines whether or not there exist two elements in S whose sum is exactly x.

9.[15] Let $A[1{:}n]$ be an array of n distinct numbers. If $i < j$ and $A[i] > A[j]$, then the pair (i, j) is called an ***inversion*** of A. Give an algorithm that determines the number of inversions in any permutation on n elements in $\Theta(n \lg n)$ worst-case time. (*Hint:* Modify merge sort.)

10. [25] Assume $T(1) = \Theta(1)$ i.e constant. Use Recursion Tree method or Master method (Which ever possible) to tight-bound the following:

a. $T(n) = 2T(n/2) + n^3$.
b. $T(n) = T(9n/10) + n$.
c. $T(n) = 16T(n/4) + n^2$.
d. $T(n) = 7T(n/3) + n^2$.
e. $T(n) = 7T(n/2) + n^2$.