CS 381: Assignment #2

Due on Thursday, September 30th, 2014

Prof. Grigorescu 12:00pm

Yao Xiao(xiao67)

Problem 1

Give the solution to the following recurrences by applying the Master theorem when possible, or by any other of the methods weve learned when the Master theorem doesnt apply. Assume T (1) = 1.

1. T (n) = T (n/5) + T (7n/10) + 1
2. T (n) = 2T (n - 1) + 5
3. T (n) = 6T (2n/5) + log 2 n
4. T (n) = 2T (n/2) + $n^{1.4}$
5. T (n) = 12T (n/12) + 11n
6. T (n) = T ($\sqrt{n}$) + 12.

1.$T(n) = T(n/5) + T(7n/10)$ We can't use master therom. Assume $T(n) <= cn$
So $T(n) <= 1/5cn + 7/10cn + 1$
=$0.9cn + 1$
=$cn - 0.9cn + 1$
$<= cn$
So $T(n) = \Theta(n)$

2.$T(n) = 2T(n-1) + 5 = 4T(n-2) + 10 = 8T(n-3) + 20 = 2^k * T(n-k) + 5^{2^{k-1}}$

$when\ k = n-1\ T(n) = 2^{n-1} * T(1) + 5^{2^{n-2}} = 2^{n-1} + 5^{2^{n-2}}$

3.$T(n) = a\ T\left(\frac{n}{b}\right) + f(n) \quad$ where $a \geq 1,\ b > 1$
$a = 6$
$b = 5/2$
$f(n) = log_2 n$
$c = log_{5/2} 6 \approx 1.95$
$f(n) = \Omega(n^c)$
$T(n) = \Theta(log_2 n)$

4.$a = 2$
$b = 2$
$f(n) = n^{1.4} log_2 2 = 1 < 1.4$
$SoT(n) = \Theta(n^{1.4})$

5.$a = 12$
$b = 12$
$f(n) = 11n$
$log_{12} 12 = 1 = c$
$SoT(n) = \Theta(nlogn)$

6.$T(n) = T(\sqrt{n}) + O(1)$
$Letm = log_2 nn = 2^m$
$T(2^m) = T(2^{m/2}) + O(1)$
$SetK(m) = T(2^m),\ K(m) = K(m/2) + O(1),\ wherea = 1b = 2$
$K(m) = \Theta(log2(m))T(n) = \Theta(log2(log2(n)))$

Problem 2

Dan and Alex play the following game: Dan picks an integer from 1 to m and Alex is trying to figure it out by asking Dan as few questions as possible. Dan is willing to respond truthfully, but he announces that he will only answer yes or no to any of Alexs questions. Design a strategy such that Alex can always guess the number using only o(m) questions

Ask every time if the number is bigger than (upperbound+lowerbound)/2, and divide change the upper bound and lower bound based on the answer.

Problem 3

You are given two n-digit numbers a, b ¿ 0, and have to come up with a fast algorithm to multiply them.
1. What is a trivial upper bound on the number of steps it takes to compute the product ab ?
2. Design a divide-and-conquer procedure that uses O(n1.58) digit-operations.

1. The upper bound will be $O(n^2)$ since you have to multiple by each number.
2. Set $A = x1 * 10^n + x0$ $B = y1 * 10^n + y0$
$So\ AB = x1y1 * 10^{2n} + (x1y0 + x0y1)10^n + x0y0$
$where(x1y0 + x0y1) = (x1 + x0)(y1 + y0) - x1y1 - x0y0$
So we just need to compute for every step $x0y0x1y1(x1 + x0)(y1 + y0)$
there are three component and each time we divide the number by half

The formula will be $T(n) = 3(t(n/2)) + c$

The master theorem gives $T(n) = \Theta(log_2 3)$

Source: Piazza post @47

Problem 4

1. The time would become $T(n) = T(\lceil n/11 \rceil) + T(9/10) + \Theta(n) = 1.11cn + \Theta(n)$
2. The time would become $T(n) = T(\lceil n/3 \rceil) + T(5/6n) + \Theta(n) \neq cn + \Theta(n)$
Since after selecting, the element which could be abandoned is only 1/6 of the total elements

Problem 5

I don't know , but $http://en.wikipedia.org/wiki/Closes\_pair\_of\_points\_problem$