CS 344: Homework #2

Due on October 2, 2017

Professor Bahman Kalantari Section #1

Tina Janulis (trj31), Caleb Rodriguez, and Rui Zhang (rz187)

Problem 1

Consider the binary numbers $x = 11011010$ and $y = 10011011$. As in class, decomposing x into a , b and y into c, d, compute the product xy using the divide-and-conquer method by forming $w1 = a+b$, $w2 = c+d$, $u = w1w2$, $v = ac$, $w = bd$. Recall $xy = 2^n v + 2^{n/2}(u - v - w) + w$. Each multiplication by 2 amounts to a shift.

We know that $x = 11011010$ and $y = 10011011$.
Let us separate these into 4-bit strings such that $a = 1101$, $b = 1010$, $c = 1001$, and $d = 1011$.
Furthermore, we know that x and y are both 8-bit, so $n = 8$.
Next, we convert each one of these 4-bit numbers into decimal numbers.
Thus, we can say $a = 13$, $b = 10$, $c = 9$, $d = 11$.
Next, we introduce variables $w1$, $w2$, v, u, and w.
We let $w1 = a + b = 23$, $w2 = c + d = 20$, $u = w1w2 = ac + ad + bc + bd = 460$.
Next, we let $v = ac = 117$ and $w = bd = 110$.
Plugging this into the longer equation, we get $xy = (2^8)(117) + (460 - 117 - 110)(2^4) + 110$.
$= (256)(117) + (233)(16) + 110$
$= 29952 + 3728 + 110$
$= 33790$
As a bit string, this could be represented as 1000001111111110.

Problem 2

You are given an infinite array $A[i]$ in which the first n elements are integers in sorted order and the rest are filled with , You are not given n. Describe an algorithm that takes as input an integer x and finds a position in the array containing x, if such a position exists, in $O(logn)$ time.

1. Take an array $A[i]$ with infinite size.
2. Let $right = 1$, $left = 1$, and $mid = 0$.
3. While($A[right] \neq$).
4. $left = right$
5. $right = right * 2$
6. endWhile
7. While($left - right > 1$)
8. $mid = left + \lfloor \frac{x}{2} \rfloor$.
9. if($A[mid] =$)
10. $right = mid$
11. else

12. $left = mid$
13. endWhile
14. $n = left + 1$ //n is the size of the integer portion of the array.
15. Call $binarySearch(A, n, x)$
16. if(x is found)
17. return index of x
18. else
19. return
20. end

Problem 3

Solve each of the recurrence relations and give Θ bound for each. You can use the master theorem if applicable.

(a) $T(n) = 5\mathrm{T}(n/4) + n$

Let $n = 4^k$ where $k = \log_4 n$. Let's apply this to the equation:

$$\begin{aligned} T(4^k) &= 5T(4^{n-1}) + 4^n \\ &= 5(5T(4^{k-2}) + 4^{k-1}) + 4^k \\ &= 5^2T(4^{k-2}) + 5 * 4^{k-1} + 4^k \\ &= 5^2T(4k-2) + 5 * 4^k(4^{-1} + 1) \\ &= 5^2(5T(4^{k-3} + 4^{k-2}) + 4^k(4^{-1} + 1) \\ &= 5^3T(4^{k-3}) + 4^k(5^{-2} + 5^{-1} + 1) \end{aligned}$$

As k approaches ∞, we can conclude that:
$T(4^k) = 5^kT(4^0) + 4^k * \sum_{i=1}^{n-1}(4/5)^i$.
This then translates to:
$T(4^k) = 0 + 4^k\frac{5/4^{k-1}}{5/4-1})$
And finally:
$n\Theta(5^k) = n\Theta(n \log_4 5) = \Theta(n \log_4 5)$

(b) $T(n) = 7T(n/7) + n$

Let $n = 7^k$ where $k = \log_7 n$. Let's apply this to the equation:

$$\begin{aligned} T(7^k) &= 7T(7^{k-1}) + 7^k \\ &= 7 * 7(T(7^{k-2}) + 7^{k-1}) + 7^k \\ &= 7^2T(7^{k-2}) + 2(7^k) \\ &= 7^3T(7^{k-3}) + 3(7^k) \end{aligned}$$

As k approaches ∞, we can conclude that:
$T(7^k) = 7^k(7^0) + k7^k = k7^k = n \log_7 n$
Finally, this means:
$\Theta(n \log_7 n)$

(c) $T(n) = 9T(n/4) + n^2$

Let $n = 4^k$ where $k = \log_4 n$. Let's apply this to the equation:

$$\begin{aligned} T(4^k) &= 9T(4^{k-1}) + 4^{2k} \\ &= 9 * 9(T(4^{k-2}) + 9 * 4^{2k-2}) + 4^{2k} \\ &= 9^2T(4^{k-2}) + 4^{2k}(9 * 4^{-2} + 1) \\ &= 9^3T(4^{k-3}) + 9^2 * 4^{2k-6} + 4^{2k}(9 * 4^{-2} + 1) \\ &= 9^3T(4^{k-3}) + 4^{2k}(9^2 * 4^{-3} + 9 * 4^{-2} + 1) \end{aligned}$$

As k approaches ∞, we can conclude that:
$T(4^k) = 9^k(T(4^0)) + \sum_{i=1}^{k-1}(9/4^2)^i$
Furthermore:
$T(4^k) = 0 + n^2((9/16)^{k-1} - 1) = \Theta(n^2)$.

(d) $T(n) = 8T(n/2) + n^3$

Let $n = 2^k$ where $k = \log_2 n$. Let's apply this to the equation:

$$\begin{aligned} T(2^k) &= 8T(2^{k-1}) + 2^{3k} \\ &= 8T(2^{k-1}) + 8^k \\ &= 8^2T(2^{k-2}) + 2 * 8^k \end{aligned}$$

As k approaches ∞, we can conclude that:
$T(2^k) = 8^k(T(2^0)) + k8^k$
Therefore:
$\Theta(n \log_2 n)$.

(e) $T(n) = 49T(n/25) + n^3.5 \log n$

Let $n = 25^k$ where $k = \log_{25} n$. Let's apply this to the equation:
$T(25^n) = 49T(25^{k-1}) + 25^{3.5} \log 25^k$
$= 49^2T(25^{k-2}) + 49 * 25^{3.5(k-1)} \log 25^{k-1} + 25^{3.5k} \log 25^k$
$= 49^2T(25^{k-1}) + 25^{3.5k} log 25^k(49 * 25^{-3.5} log 25^{-1} + 1)$
As k approaches ∞, we can conclude that:
$T(25^n) = 49^kT(1) + n^{3.5} \log n \sum i = 1^{k-1} \frac{49 \log 1/25^i}{25^i}$ Therefore:
$\Theta(n^{log_{25} 49})$

(f) $T(n) = T(\sqrt{n}) + 1$

Let $n = 2^k$ where $k = log_2 n$. Let's apply this to the equation:
$T(2^k) = T(2^{k/2}) + 1$
Let $S(n) = T(2^k$.
This implies that $2T(2^{k/2}) = S(k/2)$S(k)=2S(k/2)+1
As k approaches ∞, we can conclude that:
$S(k) = \Theta(\log k) = \Theta(\log \log n)$

(g) $T(n) = T(\sqrt{n}) + \log n$

$T(2^k) = T(2^{k/2}) + \log k/2$
Let $S(k) = T(2^k)$
$S(k) = 2S(k/2) + k$
$S(k) = \Theta(k \log k)$ Therefore:
$\Theta(\log n \log \log n)$

Problem 4

Find coefficient of polynomial of deg. 2, $p(x) = a_2x^2 + a_1x + a_0$, such that p(1) = -1, p(2) = 1, p(3) = 0.

$$p(1) \rightarrow a_2 + a_1 + a_0 = -1$$
$$p(2) \rightarrow 4a_2 + 2a_1 + a_0 = 1$$
$$p(3) \rightarrow 9a_2 + 6a_1 + a_0 = 0$$

Converting to matrix

$$\begin{pmatrix} 1 & 1 & 1 & -1 \\ 4 & 2 & 1 & 1 \\ 9 & 3 & 1 & 0 \end{pmatrix}$$

Performing RREF on the matrix would yield

$$a_0 = -6$$
$$a_1 = \frac{13}{2}$$
$$a_2 = -\frac{3}{2}$$

Problem 5

$$25, 14, 63, 29, 63, 47, 12, 21 \text{ Start} \tag{1}$$
$$25(piv), 14(<), 63(>), 29(>), 63(>), 47(>), 12(<), 21(<) \tag{2}$$
$$25(piv), 14(<), 12(<), 21(<), 63(>), 47(>), 63(>), 29(>) \tag{3}$$
$$14, 12, 21, 25, 63, 47, 63, 29 \tag{4}$$
$$14(piv), 12(<), 21(>), 25(>), 63(>), 47(>), 63(>), 29(>) \tag{5}$$
$$12, 14, 21, 25, 63, 47, 63, 29 \tag{6}$$
$$12, 14, 21, 25, 63(piv), 47(<), 63(>), 29(<) \tag{7}$$
$$12, 14, 21, 25, 47, 29, 63, 63 \tag{8}$$
$$12, 14, 21, 25, 47(piv), 29(<), 63(>), 63(>) \tag{9}$$
$$12, 14, 21, 25, 29, 47, 63, 63 \tag{10}$$
$$12, 14, 21, 25, 29, 47, 63(piv), 63(>) \tag{11}$$
$$12, 14, 21, 25, 29, 47, 63, 63 \tag{12}$$