Computer Science I – Summer 2011
Recitation #5: Recurrence Relations & Summations (Solutions)

Solve the following recurrence relations using the iteration technique:

1) $T(n) = T(n-1) + 7, \quad T(1) = 4$
==

T(n) = T(n-1) + 7
T(1) = 4

T(n) = T(n-1) + 7 = [T(n-2) + 7] + 7 = T(n-2) + 7 + 7
T(n) = T(n-2) + 2*7
T(n) = T(n-2) + 2*7 = [T(n-3) + 7] + 2*7 = T(n-3) + 7 + 2*7
T(n) = T(n-3) + 3*7
T(n) = T(n-3) + 3*7 = [T(n-4) + 7] + 3*7 = T(n-4) + 7 + 3*7
T(n) = T(n-4) + 4*7

<u>Substituting Equations</u>
<u>n → n-1</u>

T(n-1) = T(n-2) + 7
T(n-2) = T(n-3) + 7
T(n-3) = T(n-4) + 7
T(n-4) = T(n-5) + 7

Do it one more time…

T(n) = T(n-5) + 5*7

<u>So now rewrite these five equations and look for a pattern:</u>

T(n) = T(n-1) + 1*7	←	1st step of recursion
T(n) = T(n-2) + 2*7	←	2nd step of recursion
T(n) = T(n-3) + 3*7	←	3rd step of recursion
T(n) = T(n-4) + 4*7	←	4th step of recursion
T(n) = T(n-5) + 5*7	←	5th step of recursion

<u>Generalized recurrence relation at the kth step of the recursion:</u>
T(n) = T(n-k) + k*7

We want T(1). So we let n-k = 1. Solving for k, we get k = n - 1. Now plug back in.

T(n) = T(n-k) + k*7
T(n) = T(1) + 7*(n-1), and we know T(1) = 4
T(n) = 4 + 7*(n-1) = 7n - 7 + 4 = 7n - 3

We are done. Right side does not have any T(…)'s. This recurrence relation is now solved in its closed form, and it runs in O(n) time.

2) $T(n) = 2T\left(\frac{n}{2}\right) + 2, T(1) = 2$

T(n) = 2T(n/2) + 2
T(1) = 2

T(n) = 2T(n/2) + 2 = 2[2T(n/4) + 2] + 2 = 4T(n/4) + 4 + 2
T(n) = 4T(n/4) + 6

T(n) = 4T(n/4) + 6 = 4[2T(n/8) + 2] + 6 = 8T(n/8) + 8 + 6
T(n) = 8T(n/8) + 14

<u>Substituting Equations</u>
<u>n → n/2</u>

T(n/2) = 2T(n/4) + 2
T(n/4) = 2T(n/8) + 2
T(n/8) = 2T(n/16) + 2
T(n/16) = 2T(n/32) + 2

T(n) = 8T(n/8) + 14 = 8[2T(n/16) + 2] + 14 = 16T(n/16) + 16 + 14
T(n) = 16T(n/16) + 30

T(n) = 16T(n/16) + 30 = 16[2T(n/32) + 2] + 30 = 32T(n/32) + 32 + 30
T(n) = 32T(n/32) + 62

Do it again:
T(n) = 64T(n/64) + 126

<u>So now rewrite these five equations and look for a pattern:</u>

T(n) = 2T(n/2) + 2	= $\mathbf{2^1T(n/2^1) + 2^2 - 2}$	⟵	1^{st} step of recursion
T(n) = 4T(n/4) + 6	= $\mathbf{2^2T(n/2^2) + 2^3 - 2}$	⟵	2^{nd} step of recursion
T(n) = 8T(n/8) + 14	= $\mathbf{2^3T(n/2^3) + 2^4 - 2}$	⟵	3^{rd} step of recursion
T(n) = 16T(n/16) + 30	= $\mathbf{2^4T(n/2^4) + 2^5 - 2}$	⟵	4^{th} step of recursion
T(n) = 32T(n/32) + 62	= $\mathbf{2^5T(n/2^5) + 2^6 - 2}$	⟵	5^{th} step of recursion
T(n) = 64T(n/64) + 126	= $\mathbf{2^6T(n/2^6) + 2^7 - 2}$	⟵	6^{th} step of recursion

In general, after k iterations, we have:

$$T(n) = 2^kT\left(\frac{n}{2^k}\right) + 2^{k+1} - 2$$

We're not done since we still have T(…)'s on the right side of the equation. We need to get down to T(1). How?

We have $T(n/2^k)$, and we want T(1). So let $n = 2^k$. We will then have $T(2^k/2^k)$, which equals T(1). So use that substitution ($n = 2^k$) throughout the entire generalized, kth recurrence relation.

$$T(n) = 2^kT\left(\frac{n}{2^k}\right) + 2^{k+1} - 1 = n * T\left(\frac{2^k}{2^k}\right) + 2n - 1 = n * T(1) + 2n - 1$$

$$T(n) = n * 2 + 2n - 1 = 4n - 1$$

So, T(n) = 4n – 1 and runs in O(n) time.

3) $T(n) = T\left(\frac{n}{2}\right) + n$, $T(1) = 1$, Hint: $\sum_{i=0}^{\infty} \frac{n}{2^i} = 2n$ (Just get an approximate solution here.)
==

$$T(n) = T\left(\frac{n}{2}\right) + n$$
$$T(n) = T\left(\frac{n}{4}\right) + \frac{n}{2} + n$$
$$T(n) = T\left(\frac{n}{8}\right) + \frac{n}{4} + \frac{n}{2} + n$$

Here's the generalized recurrence relation for the kth step of the recursion:

$$T(n) = T\left(\frac{n}{2^k}\right) + \boldsymbol{\frac{n}{2^{k-1}} + \frac{n}{2^{k-2}} + \cdots \frac{n}{2^3} + \frac{n}{2^2} + \frac{n}{2^1} + n}$$

Look at the bolded portion: let's refer to it as the "stuff remaining". Unfortunately, there is no direct way to represent this series. Like on previous, simple examples, if we had "k" 7's on step k of the recursion, we would simply represent that as k*7. But, it is not that easy here.

So what do you do? **Express (write out) that series as a summation**, where the step of the recursion is part of the limit of the summation. So on the 3rd step of the recursion, we had these three expressions that were added together:

$$\frac{n}{4} + \frac{n}{2} + n$$

Remember, this is the 3rd step of the recursion. How can you write this as a summation?

$$\sum_{i=0}^{2} \frac{n}{2^i} = \frac{n}{2^0} + \frac{n}{2^1} + \frac{n}{2^2} = \frac{n}{1} + \frac{n}{2} + \frac{n}{4} = n + \frac{n}{2} + \frac{n}{4}$$

So now that we've represented the "stuff" for the 3rd step of recursion. Try it for the 117th step!

$$\sum_{i=0}^{116} \frac{n}{2^i}$$

Finally, express this "remaining stuff" for the kth step:

$$\sum_{i=0}^{k-1} \frac{n}{2^i}$$

So, at the kth step of the recursion, we have the following generalized recurrence relation:

$$T(n) = T\left(\frac{n}{2^k}\right) + \sum_{i=0}^{k-1} \frac{n}{2^i}$$

Now, we can solve; but we will need to make use of the hint given at the beginning!

Using the hint, we find that this summation is approximately 2n. (It's more accurately 2n – 2, but this is a very minor difference from the infinite sum given.)

$$T(n) \sim T\left(\frac{n}{2^k}\right) + 2n$$

We have $T(n/2^k)$, and we want T(1). So like on the last example, let $n = 2^k$. We will then have $T(2^k/2^k)$, which equals T(1). Plug in:

$$T(n) \sim T\left(\frac{2^k}{2^k}\right) + 2n = T(1) + 2n$$

$$T(n) \sim 1 + 2n$$

So this also runs in O(n) time.