1. Solve the following recurrence relations.

a) $x(n) = x(n-1) + 5$ for $n > 1$ $x(1) = 0$

$$x(n) = x(n-1) + 5$$

by substitutional method $\qquad x(1) = 0 \rightarrow 1$

if $n = 2 \Rightarrow \quad x(2) = x(2-1) + 5 = x(1) + 5 \rightarrow 2$

$x(1) = 0 + 5 \qquad$ [substitute 1 in 2]

$x(1) = 5 \rightarrow 3$

if $n = 3 \Rightarrow \quad x(3) = x(3-1) + 5$

$= x(2) + 5 \rightarrow 4$

substitute 3 in 4

$x(3) = 5 + 5 = 10$

$\therefore x(n) = 5n$ for $n > 1$

b) $x(n) = 3x(n-1)$ for $x(1) = 4$

$x(1) = 4 \rightarrow 1$

if $n = 2 \Rightarrow x(2) = \cancel{3(2)}\ 3x(2-1) = 3x(1) \rightarrow 2$

$x(2) = 3x(1)$

substitute 1 in 2

$x(2) = 3(4) = 12 \rightarrow 3$

if $n = 3 \Rightarrow x(3) = 3x(3-1) = 3x(2)$

$x(3) = 3x(2) \rightarrow 4$

substitute 3 in 4

$$x(3) = 3 \times 12 = 36$$

$\vdots$

$$\therefore \; x(n) = 4 \times 3^{n-1}$$

c) $x(n) = x(n/2) + n$ for $n > 1$ $x(1) = 1$ (solve for $n = 2^k$)

$$x(n) = x(n/2) + n \qquad \text{i.e., } n = 2^k$$

$$x(1) = 1$$

substitute $n = 2^k$

$$x(2^k) = x\left(\frac{2^k}{2}\right) + 2^k$$

$$x(2^k) = x(2^{k-1}) + 2^k$$

$$x(2^0) = 1$$

$$x(2^1) = x(2^{1-1}) + 2^1$$

$$= x(2^0) + 2^1 = 1 + 2 = 3$$

$$x(2^1) = 3$$

$$x(2^2) = x(2^{2-1}) + 2^2$$

$$= x(2^1) + 4 = 3 + 4 = 7$$

$$x(2^2) = 7$$

$\vdots$

$$\therefore \; x(2^k) = 2^{k+1} - 1$$

d) $x(n) = x(n/3)+1$ for $n>1$ $x(1)=1$ (solve for $n=3k$)

$$x(n) = x(n|3)+1 \rightarrow ①$$

$$x(1)=1 \qquad n=3^k$$

substitute $n=3^k$ in ①

$$x(3^k) = x(3^k/3)+1$$

$$= x(3^{k-1})+1$$

$$x(3^0) = x(1) = 1$$

$$x(3^1) = x(3^{1-1})+1 = 1+1=2$$

$$x(3^1) = 2$$

$$x(3^2) = x(3^{2-1})+1 = x(3^1)+1$$

$$= 2+1$$

$$x(3^2) = 3$$

$$x(3^k) = k+1$$

2) Evaluate the following recurrences completely.

(i) $T(n) = T(n/2)+1$, where $n=2k$ for all $k \geq 0$

$$T(n) = T(n|2)+1$$

$$n=2^k$$

$$T(2^k) = T(2^k/2)+1$$

$$T(2^k) = T(2^{k-1})+1$$

~~k/2~~ $\cancel{T(2^k) = T(2^{2-1})+1}$

$$= \cancel{T(2^1)+1}$$

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

$$T(2^{k-2}) = T\left[\frac{2^{k-2}}{2}\right] + 1 = T\left[2^{k-3}\right] + 1$$

$$T[2^1] = T(2^0) + 1$$

Now,

$$n = 2^k \Rightarrow k = \log_2 n$$

$$T(2^k) = T(2^{k-1}) + 1 = T(2^{k-2}) + 1 + 1 \ldots T(2^0) + k$$

since,

$$n = \cancel{2^k} \Rightarrow k = \cancel{\log_2 n}$$

$$\cancel{T(2^k)} = T(2^{k-1}) \cancel{+1} = T(2^{k-2})$$

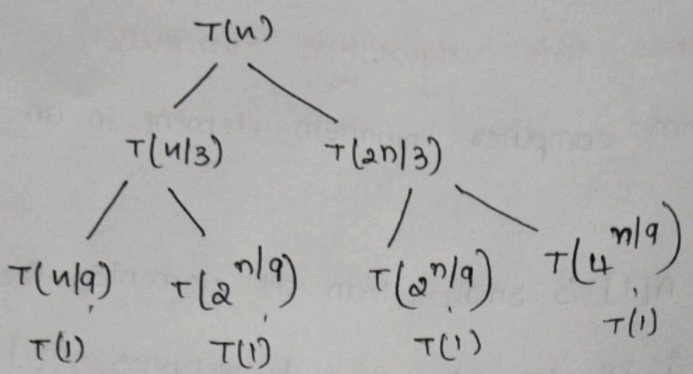
$$2^0 = 1, \quad T(2^0) = T(1) \qquad T(1) = 1$$

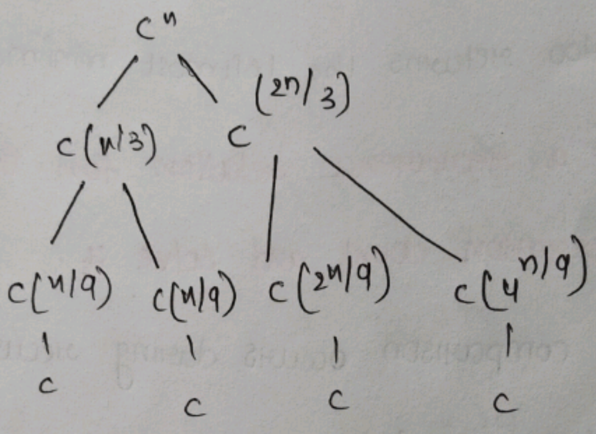
$$T(2^k) = 1 + k$$

$$T(n) = 1 + \log_2 n$$

(ii) $T(n) = T(n/3) + T(2n/3) + cn$

we use recursion tree method





length = $\log_3 n$ (divided by 3)

$T(n) = cn \log_3 n$

$\Rightarrow \omega(n \log n)$

3) consider the following recursion algorithm

```
Min1(A[0----n-1])
if n=1 return A[0]
Else temp = Min1(A[0... n-2])
        if temp<= A[n-1] return temp
Else
        Return A[n-1]
```

a) what does this algorithm compute?

This algorithm computes minimum element in an array A of size n.

If $i<n$, $A[i]$ is smaller than all elements, then $A[j]$, $j=i+1$ to $n-1$, then it returns $A[i]$.

It also returns the leftmost minimal element.

b) Setup a recurrence relation for the algorithms basic operation count and solve it:

Mainly comparison occurs during recursion.

So, $T(n) = T(n-1)+1$, where $n>1$ (one comparison at every step except, $n=1$)

$T(1)=0$ (when $n=1$ no comparison)

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

$= 0+(n-1)$

$T(n) = n-1$

$\therefore$ Time Complexity $= O(n)$

4. Analyse the order of growth

(i) $F(n) = 2n^2+5$ and $g(n) = 7n$

Use the $\Omega(g(n))$ notation

$F(n) = 2n^2+5$

$c.g(n) = 7n$

$F(n) \geq c.g(n)$

$n=1$

$F(1) = 2(1)^2+5$
$= 2+5$
$= 7$

$c.g(n) = 7n$
$= 7(1)$
$= 7$

$n=2$

$F(2) = 2(2)^2+5$
$= 8+5 = 13$

$c.g(n) = 7n$
$= 7\times2$
$= 14$

$n=3$

$F(3) = 2(3)^2+5$
$= 18+5 = 23$

$c.g(n) = 7n$
$= 7\times3 = 21$

$n=1, \quad 7=7$

$n=2, \quad 13=14$

$n=3, \quad 23=21$

$n\geq3, \quad F(n) \geq c.g(n).$

$F(n)$ is always greater than or equal to $c.g(n)$ when, n value is greater or equal to 3.

$$\therefore F(n) = \Omega(g(n)).$$

$F(n)$ is grows more than $g(n)$ from below asymptotically.