① Solve the following recurrence relations

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

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

let $x(1) = 0$:

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

$\boxed{\therefore x(n) = 5n-5}$

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

substituting:

$x(n) = 3^{n-1}x(1)$

$\Rightarrow x(1) = 4$:

$x(n) = 4 \cdot 3^{n-1}$

$\boxed{\therefore x(n) = 4 \cdot 3^{n-1}}$

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

$x(n) = n + n/2 + n/4 + \dots + 1$

here $1 + n/2 + n/4 + \dots + n/2\log n$ simplifies to $2n-1$

$\therefore x(n) = 2n-1$

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

$x(n) = 1 + 1 + 1 + \dots$ (for $\log_3 n$ times)

$x(n) = \log_3 n$

$\therefore \cancel{x} x(n) = \log_3 n$

(2) Evaluate the following recurrences completely

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

here

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

$T(n/2) = T(n/4) + 1$

$T(n/4) = T(n/8) + 1$

$\Rightarrow$

$T(n) = 1 + 1 + 1 + \cdots$ for $\log_2 n$ times

$\therefore T(n) = T(n/2) + 1$ for $n = 2^k$ is :

$T(n) = \log_2 n$

$\therefore T(n) = O(\log n)$

ii) $T(n) = T(n/3) + T(2n/3) + cn$, where 'c' is a constant and 'n' is the input size

$\Rightarrow$ $T(n) = aT(n/b) + f(n)$

$a = 1, b = 3, f(n) = cn.$

i, calculate $n^{\log_b a}$

$n^{\log_3 1} = n^0 = 1$

ii, compare $f(n)$ with $n^{\log_b a}$:

$f(n) = cn$

$f(n) = o(n^{0.1})$

iii, Apply case 3 for master theorem

if $f(n) = \theta(n^{\log_b a} \log^k n)$ for some $k \geq 0$, then $T(n) = \theta(n^{\log_b a} \log^{k+1} n)$

Since $f(n) = O(n)$

$T(n) = \theta(n \log n)$

$\therefore$ $T(n) = T(n/3) + T(2n/3) + cn$ is : $T(n) = \theta(n \log n)$

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 is designed to find the minimum element in an array 'A' of size (n).

(b) Set up a recurrence relation for the algorithms basic operation count and solve it

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

* $T(1) = 1$

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

Expand: $T(n) = T(n-2) + 2 + 2$

$T(n) = T(n-3) + 2 + 2 + 2$ [continue the pattern]

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

$\boxed{T(n) = 2n - 1}$ ⇒ Best case

4. Analyze the order of growth.

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

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

(5) Compute the limit:

$$\lim_{n \to \infty} \frac{F(n)}{g(n)} = \lim_{n \to \infty} \frac{2n^2+5}{7n}$$

Simplify the fraction:

$$\lim_{n \to \infty} \frac{2n^2+5}{7n} = \lim_{n \to \infty} \left(\frac{2n^2}{7n} + \frac{5}{7n}\right) = \lim_{n \to \infty} \left(\frac{2}{7}n + \frac{5}{7n}\right)$$

Evaluate the limit:

$$\lim_{n \to \infty} \left(\frac{2}{7}n + \frac{5}{7n}\right) = \infty$$

Conclusion:

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

$\therefore$ F(n) is asymtotically bounded below by g(n), meaning F(n) grows at least as fast a g(n). In simpler terms F(n) is a asymtotically quadratic.