Q1]

a] $T(n) = T\left(\frac{n}{2}\right) + n$

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

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

$=$ ⋮ After k iteration

$= T(n/2^k) + (n/2^{k-1}) + \dots + n$

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

lets sg after k iteration it reaches it's bottom

$\therefore \frac{n}{2^k} = 2 \Rightarrow n = 2^{k+1}$

$\therefore \log n = k+1 \log 2 \Rightarrow \boxed{k = \log n + 1}$

$\therefore$ After reaching the end.

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

Page No.
Date

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

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

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

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

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

$\therefore$ The both lower bound and upperbond given linear growth

$\therefore$ lower bond $\Omega(n)$
upper bound $O(n)$

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

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

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

$$= 2^2\left(2T\left(\frac{n}{2^3}\right) + 2\left(\frac{n}{2^2}\right)\right) + 2^2 n$$

$$= 2^3 T\left(\frac{n}{2^3}\right) + 2(3n)$$

$$= 2^3\left(2T\left(\frac{n}{2^4}\right) + 2\left(\frac{n}{2^3}\right)\right) + 2(3n)$$

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

$$\vdots$$

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

let consider it terminate after k itherations

$\therefore = 2^k T(2) + 2kn$

$\therefore \quad \frac{n}{2^k} = 2$

$n = 2^{k+1}$

$\log n = k+1$

$\underline{k = \log n + 1}$

$\therefore T(n) = 2^{\log n} \cdot 2 \cdot T(2) + 2n \log n$

$= 2n\, T(2) + 2n \log n$

$\therefore \quad O(\bullet) = O(n \log n)$

$\Omega(\bullet) = \Omega(n \lg n)$

Page No.
Date

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

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

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

$= T(n^{\frac{1}{2^3}}) + 3$

$\vdots$

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

let us say it bottoms at k iteration.

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

$\therefore n^{\frac{1}{2^k}} = 2$

$\frac{1}{2^k} \log n = 1$

$\cancel{k = \log n}$

$\therefore \cancel{T(n) = T(2) + (\log n - 1)}$

$\log n = 2^k$

$\log \log n = k.$

$\therefore T(n) = T(2) + \log(\log n) - 1$

upper bound $\Rightarrow O(\log(\log n))$
lower bound $\Rightarrow \Omega(\log(\log(n)))$

Date

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

$= 2(2T(n-2)+1)+1$
$= 4T(n-2)+2$
$= 4(2T(n-3)+1)+2$
$= 8T(n-3)+3$
$= 2^3 T(n-3)+3$
$\vdots$
$= 2^k T(n-k)+k.$

Let it bottom out after k iteration.

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

$\therefore n-k = 2$
$k = n-2$

$\therefore T(n) = 2^{(n-2)} T(2) + (n-2)$
$= 2^n \cdot 4 T(2) + n - 2$

$\therefore$ upper bound $\Rightarrow O(2^n)$
lower bund $\Rightarrow \Omega(2^n)$

Q2]

a] ① → Sort the given set using merge sort.
② → let i point first element of S and j point last element.
③ → while ($i < j$)

sum = $S[i] + S[j]$
if (sum > z)
    j --
else
    i + +.
if (sum = = z)
    return true and end,

④ → Return False

For this Algo $O \Rightarrow \underbrace{n \log n}_{\text{(merge sort)}} + \underbrace{n}_{\text{(while)}}$

$= n \log n$

$\Omega \Rightarrow n \log n$

$\Theta = n \log n$

b] → The same Algo as 'a'
→ But now we don't need to sort it as it is already sorted.
→ We will use same Algo as part a But from step 2

∴ run time ⇒ $\underbrace{O(n)}_{\text{while loop}}$

Date

Q3] Assuming index start from 1 not 0 and n is size

// ① → Sort arry A using merge sort (No need to add this as already sorted)
② → if n=1 return true End
③ → let count = 1.
④ → for (i=2 ; i<=n ; i++)
{ if (A [i] = A [i-1])
count ++ ;
else
count = 1 ;
if (count > n/2)
return true
}
⑤ → return False

runn time ⇒ O (n log n) // if not sorted
O (n) // as it is sorted

Date

Q4) ① → Compute the median of X and the median of Y, let them be x_m and y_m

② → low = 1 high = n

median will be present as $(1+n)/2^{th}$ index,

i.e $(low + high)/2^{th}$ index.

③ if $x\_m < y\_m$

then make

low in x = x_m

high in y = y_m

if $x\_m > y\_m$

high in x = x_m

low in y = y_m.

④ Continue the upper part repeatedly until one of high − low = 0

⑤ return median = $\frac{x\_m + y\_m}{2}$

As there always will be two medians as there are total even number of element.