Analytical - 2

5) Big Oh notations:

$f(n) = n^2 + 3n + 5 \Rightarrow O(n^2)$

	$f(n)$		$g(n)$
$n=1$	1	=	9
$n=2$	2	=	15

$\therefore f(n)$ is best case when $n \geq 1$

$g(n) = n^2$

$\therefore O(n) \Rightarrow f(n) \leq c \cdot g(n)$

$c = N * 9 \neq k$.

$n \geq 1$:

$n^2 + 3n + 5 \leq n^4 + 3n^2 + 5n^2$

$n^2 + 3n + 5 \leq 9n^2$

$\therefore c = 9, g(n) = n^2$

$\therefore O(n^2)$

Hence Proved.

1) $t_1(n) \in O(g_1(n))$

$t_2(n) \in O(g_2(n))$

Prove $t1(n) + t2(n) \in O(\max\{g_1(n), g_2(n)\})$

Soln:

$\because$ Case is Worst (O)

$f(n) \leq c \cdot g(n)$

$f(n) \leq t_1(n) + t_2(n)$

$c \cdot g(n) \leq c_1(g_1(n)) + (c_2 \cdot g_2(n))$

$\leq c\{(g_1(n)) + (g_2(n))\}$

$\leq c(\max\{g_1(n), g_2(n)\})$

Hence Proved.

a) find time complexity:

3. $T(n) = \begin{cases} 2T(n/2)+1 & \text{if } n>1 \\ 1 & \text{otherwise} \end{cases}$

$$\boxed{T(n) = aT(n/b) + f(n)}$$

$\therefore a=2,\ b=2,\ f(n)=1$

$f(n) = n^k \log_b^a$

$\therefore 1 = n^0 \log_2^2$

$1 = 1$

$\boxed{k=0,\ \log_b^a = 1}$

$\therefore \log_b^a > k$

$\Rightarrow \theta(n \log_b^a)$

$\Rightarrow \theta(n \cdot 1)$

$\Rightarrow \theta(n)$

4) $T(n) = \begin{cases} 2T(n-1) & \text{if } n>0 \\ 1 & \text{otherwise} \end{cases}$

forward substitution:

$T(n) = 2T(n-1)$ [Assume $T(0)=0$]

$n=1$

$T(1) = 2T(1-1) = 2T(0)$

Backward Subn

$T(n) = 2T(n-1)$ $[T(0)=0]$

$n=n-1$

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

$T(n-1) = 2T(n-2)$ → (2)

(2) in (1):

$T(n) = 2[2T(n-2)]$

$T(n) = 2^2 T(n-2)$ → (3)

$2 = n-2$

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

$T(n-2) = 2T(n-3)$ → (4)

(4) in (3):

$T(n) = 2^2[2T(n-3)]$

$T(n) = 2^3 T(n-3)$ → (5)

$n = n-3$:

$T(n-3) = 2T((n-3)-1)$

$T(n-3) = 2T(n-4)$ → (6)

(6) in (5):

$T(n) = 2^3[2T(n-4)]$

$T(n) = 2^4 T(n-4)$ → (7)

$\therefore \boxed{T(n) = 2^k T(n-k)}$

$(n-k) = 0 \Rightarrow n = k$

$\therefore T(n) = 2^k (T(k-k))$

$\boxed{T(n) = 2^k T(0)}$

$T(0) = 1$:

$T(n) = 2^k$

$\therefore O(2^n)$

6) Big Omega:

$g(n) = n^3 + 2n^2 + 4n \Rightarrow \Omega(n^3)$

$g(n)$ $\quad$ $g(n)$

$f(n)$ $\quad$ $f(n) \geq c \cdot g(n)$

$n \geq 1, C = 1$

$n^3 + 2n^2 + 4n \geq n^3$

$g(n) \geq c \cdot n^3$

$1 \geq 1$

$\therefore \Omega(n^3)$

Hence Proved.

7) Big theta:

$h(n) = 4n^2 + 8n \Rightarrow \Theta(n^2)$

we add Big Omega & Big O:

$c_1 g(n) \leq f(n) \leq c_2 g(n)$

Big Ω:

$f(n) \geq c_1 \cdot g(n)$

$4n^2 + 8n \geq c_1 \cdot n^2 \Rightarrow 4 + 8/n \geq c_1$

$n = 1, c_1 = 1$

$7 \geq 1$

$\therefore \Omega(n^2)$

Big O:

$f(n) \leq c_2 g(n)$

$\Rightarrow 4n^2 + 8n \leq c_2 n^2$

$4 + \frac{8}{n} \leq c_2$

$n = 3, c_2 = 5$

$5 \leq 5$

$\therefore O(n^2)$

$\because$ Since it ($h(n)$) is both $\Omega \geq 0$, it is $\Theta(n^2)$

Hence Proved.

8. $f(n) = n^3 - 2n^2 + n \quad g(n) = n^2$

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

$\rightarrow f(n) \geq c \cdot g(n)$

$n^3 - 2n^2 + n \geq c \cdot n^2$

$n = 2, c = 1$

$(2)^3 - 2(2)^2 + 2 \geq 1 \cdot (2)^2$

$6 \geq 4$

$\therefore$ when $n \geq 4$ $f(n)$ is $\Omega(g(n))$

9) $h(n) = n \log n + n \in \Theta(n \log n)$

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

$a = 1, \; b = 1 \rightarrow f(n) = n$

$\therefore \; n = n^k \log_b^a$

$n = n^1 \log_1^1$

$n = n$

$\therefore k = 1$

$\log_b^a = 1$ $\quad \boxed{k = \log_b^a}$

$\therefore \Theta(n^k \log^{p+1} n)$

$\Theta(n^1 \log_n^2)$

$\therefore \Theta(n \log n)$ Hence Proved.

10) Order of growth:

$T(n) = 4T(n/2) + n^2, \; T(1) = 1$

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

$a = 4, \; b = 2, \; f(n) = n^2$

$n^2 = n^k \log_b^a$

$k = 2, \; \log_b^a = 2 \quad \therefore \boxed{k = \log_b^a}$

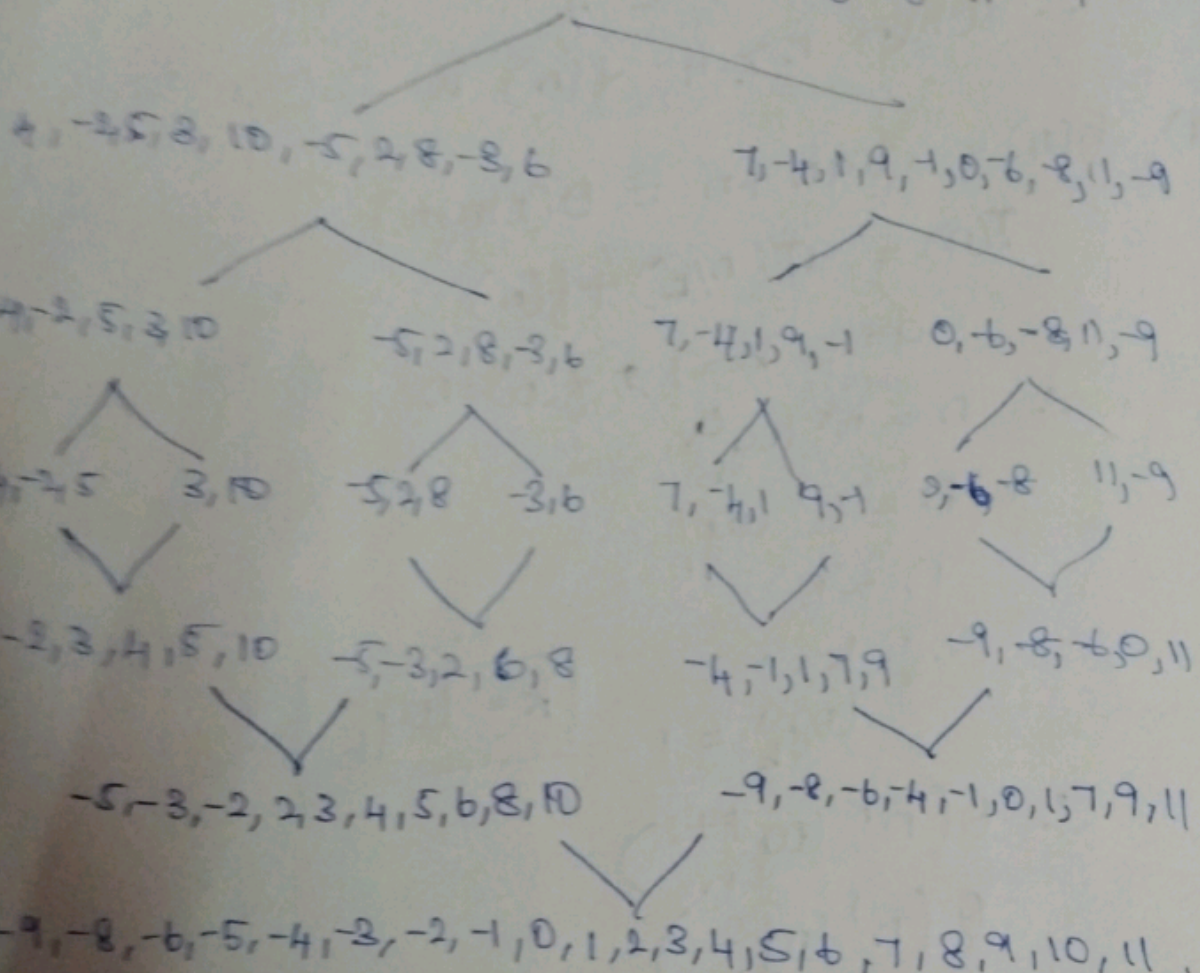
$T(n) = \theta(n^k \log^p n)$

$\therefore T(n) = \theta(n^2 \log n)$

$T(n) = 4^k T(n/2^k) + kn^2$.

A) find max & min:

4 -2 5 2 10 -5 2 8 -3 6 7 -4 1 9 -1 0 -6 -8 11 -9



$\therefore$ Minimum = $-9 \times 11 = -99$

Maximum = $10 \times 11 = 110$

B) Binary Search: key = 23

2	5	8	12	16	23	38	56	72	91
0	1	2	3	4	5 mid	6	7	8	9

$\therefore$ mid = $\frac{low + high}{2}$

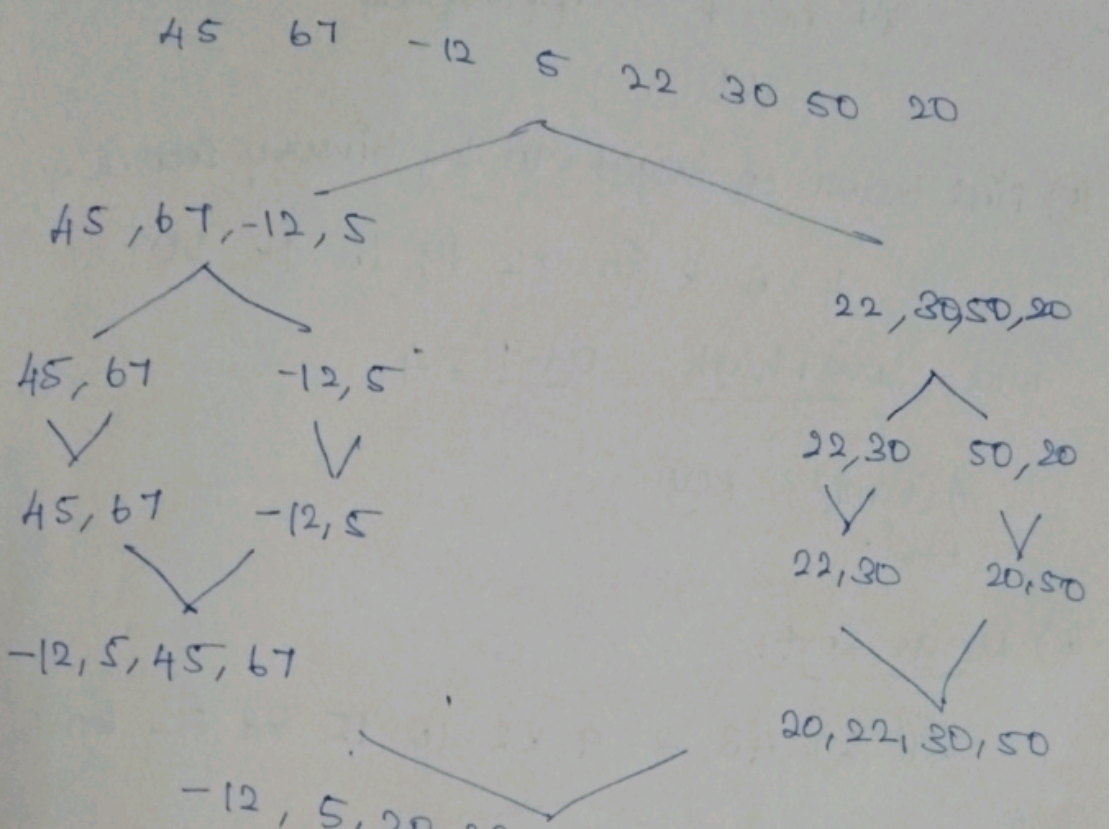
$\therefore$ mid = $\frac{0+9}{2}$ = 5

A[mid] = key

23 = key

End:

13) Merge Sort:



Recurrence relation:

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

$\log_b a = 1, k = 1 \Rightarrow k = \log_b a$

$T(n) = \theta(n^k \log^{k+1}_n)$

$= \theta(n \log n)$

14) Selection Sort:

(12) 7 5 (-2) 18 6 13 4

1) $\Rightarrow$ -2 | (7) 5 12 18 6 13 (4)

2) $\Rightarrow$ -2 4 | (5) 12 18 6 13 7

-2 4 5 | (12) 18 (6) 13 7

3) -2 4 5 6 | 12 12 7

4) -2 4 5 6 7 12 12 18

∴ Tot. no of swapping = 4

∴ $O(n^2)$

15) Find index of target = 10 by binary search.

2 4 6 8 (10) 12 14 16 18 20

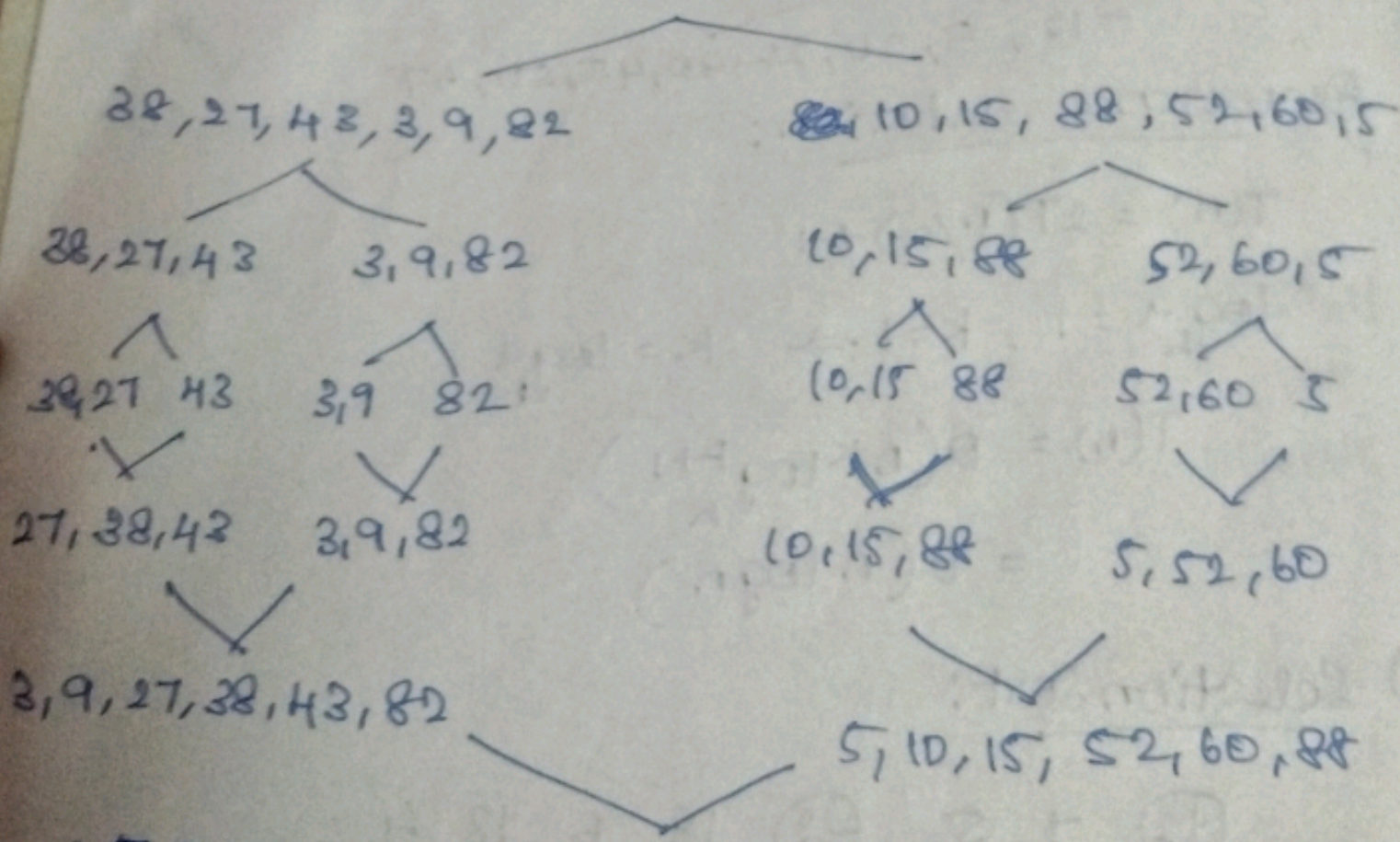
$mid = \frac{low + high}{2} = \frac{0+9}{2} = 4$

A[mid] = key

End.

16) Merge Sort:

38 27 43 3 9 82 10 15 88 52 60 5



3,5,9,10,15,27,38,43,52,60,82,88.

∴ $O(n \log n)$

(7) Bubble sort:

64 34 25 12 22 11 90

Pass 1:

64 34 25 12 22 11 90

34 64 25 12 22 11 90

34 25 64 12 22 11 90

34 25 12 64 22 11 90

34 25 12 22 64 11 90

34 25 12 22 11 64 90

Pass 2:

34 25 12 22 11 64 90

25 34 12 22 11 64 90

25 12 34 22 11 64 90

25 12 22 34 11 64 90

25 12 22 11 34 64 90

Pass 3:

25 12 22 11 34 64 90

12 25 22 11 34 64 90

12 22 25 11 34 64 90

12 22 11 25 34 64 90

Pass 4:

12 22 11 25 34 64 90

12 11 22 25 34 64 90

Pass 5:

12 11 22 25 34 64 90

11 12 22 25 34 64 90

Time Complexity:

Best case, Avg case, Worst case → n^2

(8) Selection sort:

64 25 12 22 11

11 | 25 12 22 64

11 12 | 25 22 64

11 12 22 25 64

Time Complexity:

Best, Avg, Worst ⇒ n^2

(9) Insertion sort:

28 27 43 3 9 82 10 15 88 52 60 5

27 28 43 3 9 82 10 15 88 52 60 5

27 28 3 43 9 82 10 15 88 52 60 5

27 3 28 43 9 82 10 15 88 52 60 5

3 27 28 43 9 82 10 15 88 52 60 5

3, 9, 27, 38, 43, 82, 10, 15, 88, 52, 60, 5

3 9 27 38 43 10 82 15 88 52 60 5

3 9 10 27 38 43 82 15 88 52 60 5

3 9 10 15 27 38 43 82 88 52 60 5

3 9 10 15 27 38 43 52 82 88 60 5

3 9 10 15 27 38 43 52 60 82 88 5

3 5 9 10 15 27 38 43 52 60 82 88.

Time Complexity: $O(n^2)$

20) Insertion sort:

4, 2, 5, 3, 10, -5, 2, 8, -3, 6, 7, -4, 1, 9, -1, 0, -6, 8, 11, 9

-2 4 5 3 10 -5 2 8 -3 6 7 -4 1 9 -1 0 -6 8 11 9

-2 3 4 5 10 -5 2 8 -3 6 7 -4 1 9 -1 0 -6 8 11 9

-5 -2 3 4 5 10 2 8 -3 6 7 -4 1 9 -1 0 -6 8 11 9

-5 -2 2 3 4 5 10 8 -3 6 7 -4 1 9 -1 0 -6 8 11 9

-5 -2 2 3 4 5 8 10 -3 6 7 -4 1 9 -1 0 -6 8 11 9

-5 -3 -2 2 3 4 5 8 10 6 7 -4 1 9 -1 0 -6 8 11 9

-5 -3 -2 2 3 4 5 6 8 10 7 -4 1 9 -1 0 -6 8 11 9

-5 -3 -2 2 3 4 5 6 7 8 10 -4 1 9 -1 0 -6 8 11 9

-5 -4 -3 -2 2 3 4 5 6 7 8 10 1 9 -1 0 -6 8 11 9

-5 -4 -3 -2 1 2 3 4 5 6 7 8 10 9 -1 0 -6 8 11 9

-5 -4 -3 -2 1 2 3 4 5 6 7 8 9 10 -1 0 -6 8 11 9

-5 -4 -3 -2 -1 1 2 3 4 5 6 7 8 9 10 0 -6 8 11 9

-5 -4 -3 -2 -1 0 1 2 3 4 5 6 7 8 9 10 -6 8 11 9

-8 -6 -5 -4 -3 -2 -1 0 1 2 3 4 5 6 7 8 9 10 11 9

-8 -6 -5 -4 -3 -2 -1 0 1 2 3 4 5 6 7 8 9 10 11

Time Complexity: $O(n^2)$