Some Common Guesses

- $T(n)=T(n-1)+1$ $O(n)$
- $T(n)=T(n-1)+n$ $O(n^2)$
- $T(n)=T(n-1)+\log n$ $O(n\log n)$
- $T(n)=T(n-1)+n^2$ $O(n^3)$
- $T(n)=T(n-1)+1/n$ $O(\log n)$
- $T(n)=T(n/2)+1$ $0(\log n)$
- $T(n)=T(n/2)+n$ $0(n)$
- $T(n)=2T(n/2)+n$ $0(n\log n)$
- $T(n)=2T(n/2)+n^2$ $0(n^2)$

Question 1:

- **Input:** An sorted array of n-distinct element
- **Output:** Find any element A[i] such that A[i] == i

-50	-30	-20	-10	-2	0	2	4	6	9	11	12	14	15	17	19	20	25	27	30
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

- When we came to the middle element in Binary Search 1st place is 9 so if we go left then 9th place value cannot be 9 but if we go right i.e. 11th position value can be 10 or 11 so there is a hope of going right
- Answer- Binary Search (logn)

Question 2:

- **<u>Input:</u>** An array of n-elements in which until some places all are integers and afterwards all are infinity.
- **<u>Output:</u>** Find 1^{st} position of infinity

-50	-30	-20	5	-2	0	2	4	3	9	12	∞	∞	∞	∞	∞	∞	∞	∞	∞
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

Answer: After applying Binary Search, we reach at middle element and at this position we check whether it is integer or ∞, it says I am integer than ∞ must lie at its right side.
Now, it reach to right side's middle, then it compare with 1^{st} left element, ask are you ∞ or integer not then it says yes I am ∞……….

Question 3:

- **Input:** A sorted array of n-distinct element
- **Output:** find any 2 element a & b such that a+b=1000

100	200	300	400	500	600	700	800
1	2	3	4	5	6	7	8

- Linear Search = O(n^2)
- Binary Search = O(nlogn)

(D) **SUBSTITUTION METHOD**

step 1: Guess the form of solution

step 2: Use mathematical induction to find the constants and show that the solution works.

Substitution method can be used to establish upper or lower bounds on a recursion.

Ques- Solve the recurrence $T(n) = 2T(\lfloor n/2 \rfloor) + n$ by substitution method. Take $T(1)=1$

Guess the solution.

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

ie $T(n) \leq c \cdot (n \log n)$ — ①

Use Mathematical Induction.

Working.

$T(n) = 2T(\lfloor n/2 \rfloor) + n$

$n^{\log_2^2} = n$

$f(n) = n$

Case 2

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

① Base Condition $n = 1$

$T(1) \leq c \cdot 1 \cdot \log 1$

$1 \leq 0$ False

$\underline{\underline{n=2}}$

$T(2) \leq c \cdot 2 \cdot \log_2 2$

$2T(1) + 2 \leq c \cdot 2$

$2 + 2 \leq 2 \cdot c$ True

⋮

$n = k$

$T(k) \leq ck \log k$ for $2 \leq k \leq n$

② Inductive step.

Assume that it is true for $n/2$.

ie $T(n/2) \le c \cdot \frac{n}{2} \log \frac{n}{2}$ is true

Now, we have to show that it is true for $n = n$

$T(n) \le cn\log n$

we know that

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

$$\le \underbrace{2c\left\lfloor \frac{n}{2} \right\rfloor \log \left\lfloor \frac{n}{2} \right\rfloor}_{} + n$$

→ Replace $T(n/2)$ by $ck\log k$ for $k = n/2$

$$\le \underbrace{2 \cdot c \frac{n}{2} \log \frac{n}{2}}_{} + n$$

→ Remove floor as $\frac{n}{2} \ge \lfloor n/2 \rfloor$

$$\le cn\log \frac{n}{2} + n$$

$$\le cn(\log_2 n - \log_2 2) + n$$

$$\le cn\log_2 n - cn + n$$

$$\le cn\log_2 n + (1-c)n$$

$$\le cn\log_2 n \quad \forall \quad c \ge 1$$

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

Ques Consider the recurrence

$$T(n) = T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + 1. \qquad T(1) = 0$$

Guessing the solution

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

ie $T(n) \le c \cdot \log n$ — ①

Use Mathematical Induction

① Base Condition

$n = 1$

Working.

$n^{\log_2 1} = n^0 = 1$

case 2

$= 1 \cdot \log n$.

$T(1) \leq c \cdot \log 1$

$1 \leq 0$ False

When $n = 2$

$T(2) \leq c \cdot \log_2 2$

$T(1) + 1 \leq c$

$1 \leq c.$ True

$\vdots$

$n = k$

Now we assume it is true till k for $2 \leq k \leq n$

$T(k) \leq c \cdot \log k$

② Inductive step.

Assume that it is true for $n/2$

ie $T(n/2) \leq c \cdot \log \frac{n}{2}$ is true

Now we have to show that it is true for n

$T(n) \leq c \cdot \log n$

We know that

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

$$\leq c \cdot \log \left\lfloor \frac{n}{2} \right\rfloor + 1$$

$$\leq c \cdot \log \frac{n}{2} + 1$$

$$\leq c \cdot [\log_2 n - \log_2 2] + 1$$

$$\leq c \cdot \log_2 n - c + 1$$

$\therefore \quad T(n) = O(\log_2 n)$