DATE / 20
PAGE No

Ans 1 =>
```
int a=0;
for(int i=0; i<n; i++)
{
  for(int j=n; j>i; j++)
  {
    a=a+i+j;
  }
}
```

i	j
1	n, (n-1) - - to 1
2	n - - - i+1
3	n - - - i+1
!	n - - - i+1
n	

$t(n) = n \times n = O(n^2)$

Ans 2 =>
```
int count=0
for(int i=1; j<=n; i=i*2)
{
  for(int j=1; j<=i; j++)
  {
    count=count+1;
  }
}
```

Outer loop =>

$i = \underbrace{1 \times 2 \times 2 \times 2 \times 2 \times \ldots}_{k} = n \Rightarrow 2^k = n$

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

i	j
1	1
2	1, 2
3	1, 2, 3, 4
4	
1	
.	
n	1, 2, 3, 4 - - - , n

~~i = i + 1~~

$j = 2^0 + 2^1 + 2^2 - - - + 2^k$

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

$= 2^{\log n}$

$= n$

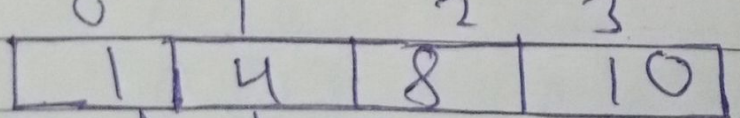
Total time complexity = $O(n)$

Ans3 -> Linear Search =>

worst case => when item to be searched is not present in the list or at the end of list.

— N comparisions

$O(n)$



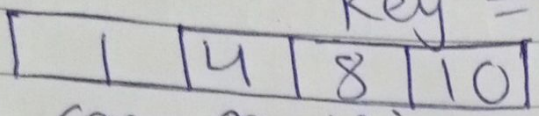
Key = 10

To find Key we have to make n comparisions.

$\therefore O(n)$

Best Case =>

Key = 1



- 1 comparision
- O(1) complexity

Average Case =>

$$\text{Average case} = \frac{\text{best case} + \text{worst case}}{2}$$

$$= \frac{1+n}{2}$$

$$\text{Average case} = O(n)$$