

1. Problem statement

What are the number of operations for the following function?

```
public static void func(int n)
{
    int sum=0;
    for(int i=2;i<n;i+=2)
        sum+=i;
    System.out.println(sum);
}
```

1.  $k_1 + k_2(n)$
2.  $k_1 + k_2$
3.  $k_1 + k_2(n^2)$
4.  $k_1 + k_2(\log n)$

2. Problem statement

What are the number of operations for the following function?

```
public static void func(int n)
{
    int sum=0;
    for(int i=1;i<n*n;i++)
        sum+=i;
    System.out.println(sum);
}
```

1.  $k_1 + k_2(n)$
2.  $k_1 + k_2$
3.  $k_1 + k_2(n^2)$
4.  $k_1 + k_2(\log n)$

3. Problem statement

What will be the Time Complexity of following code in terms of 'n'?

```
public static void func(int n)
{
    int sum=0;
    for(int i=1;i<n;i++)
    {
        for(;i<n*n;i++)
        {
            sum+=i;
        }
    }
    System.out.println(sum);
}
```

1.  $O(n)$
2.  $O(n^2)$
3.  $O(n^3)$
4.  $O(n^4)$

4. Problem statement

What will be the Time Complexity of following code in terms of 'n'?

```
public static void func(int n)
{
    int sum=0;
    for(int i=1;i<n;i++)
    {
        for(int j=1;j<n*n;j++)
        {
            sum+=i;
        }
    }
    System.out.println(sum);
}
```

1.  $O(n)$
2.  $O(n^2)$
3.  $O(n^3)$
4.  $O(n^4)$

5. Problem statement

What will be the Time Complexity of following code in terms of 'n'?

```
public static void func(int n)
{
    int sum=0;
    for(int i=1;i<n;i++)
    {
        for(int j=1;j<=i;j++)
        {
            sum+=i;
        }
    }
    System.out.println(sum);
}
```

1.  $O(n)$
2.  $O(n^2)$
3.  $O(n^3)$
4.  $O(n \log n)$

6. Problem statement

What will be the Time Complexity of following code in terms of 'n'?

```
public static void func(int n)
{
    int sum=0;
    for(int i=1;i<n;i*=2)
    {
        sum+=i;
    }
    System.out.println(sum);
}
```

1.  $O(\log n(\text{base } 2))$
2.  $O(n)$
3.  $O(\ln(n))$
4.  $O(n^2)$

7. What is the time complexity for merging two sorted arrays? Size of arrays are  $n$  and  $m$ .
- i.  $O(n*m)$
  - ii.  $O(n+m)$
  - iii.  $O(n)$
  - iv.  $O(m)$