Code:

```
sum = 0;
for (i=1; i \le n; i++)
sum += n; \longrightarrow dominating instruction (i)
```

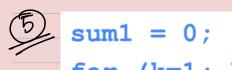
values of i	(i) executed	
1	1	
2	1	A / \
3	1	0(n)
•	:	•
•		
n	1	
	sum = n	

2

```
sum1 = 0;
for (i=1; i<=n; i++)
  for (j=1; j<=n; j++)
  sum1++;</pre>
```

value of i	of executes
1	\sim
2	\sim
n	\sim)
	Sum: $n \times (n)$
2/2	, total values of ,
(n ²)	= n ²

```
sum2 = 0;
    for (i=1; i<=n; i++)
      for (j=1; j<=i; j++)</pre>
       sum2++;
 values of i & executes
                    2
               \frac{n(n+1)}{2}
sum = 0;
    for (j=1; j<=n; j++)
       for (i=1; i<=j; i++)
        sum++; ----
    for (k=0; k<n; k++)
      A[k] = k;
```



	values of i	& executed	
	11 = 2°	<u> </u>	
	2 = 2'	n	
	$4 = 2^2$	n	
	8 = 23	γ	
	16 = 29	γ	
*	3 3		
2 = n	2 n = 2k	n.	
k=logn	<u> </u>	> kxn	$= \left(\log_2 n\right) \times$
\sim			V - /

total values of i, k= log2n

= O(nlogn)

```
for (k=1; k \le n; k \ge 2)
     for (j=1; j<=k; j++)
       sum2++; ☆
```

Values of i	$\frac{\lambda}{\lambda}$
1 = 2° 2 = 2'	
$4 = 2^2$	
~ · · · · · · · · · · · · · · · · · · ·	
/ -2-	
	total =
14 2 12 2 2	
ks log n	/>

) log 2 = log 2 x= n

1 = 20 2 = 2' n=2k

= 2"+2"+2"+ ... +2 = 1+2+22+1... +2 logh

number of terms = logn > geometric series

sum = a(rck-1) | a=finst term = 1x(2k-1) K= total term

> = 2 K - 1 = 2 log 2 - 1 = n - 1

Complexity = O(n)