TIME COMPLEXITY

10 September 2024 21:42

ARITHMETIC SERIES.

$$\sum_{K=1}^{n} K = 1+2+3+ \cdots + n = \frac{n(n+1)}{2}$$

GEOMETRIC SERIES.

$$\sum_{K=0}^{n} x^K = 1+x+x^2+ \cdots + x^n = \frac{x^{n+1}-1}{x-1} \quad (x \neq 1)$$

HARMONIC SERIES

$$\sum_{K=1}^{n} \frac{1}{K} = 1+\frac{1}{2}+ \cdots +\frac{1}{n} \approx \log n.$$

other important formula.

$$\sum_{K=1}^{n} \log K \approx n \log n.$$

Compute Time Complexity of following.

ⓐ

```
int a=0, b=0;
    for( i=0 ; i<N; i++){
            a=a+rand();
        }
    for(j=0; j<M; j++){
            b=b+rand();
        }
```

TIME COMPLEXITY = O(N+M) time

SPACE COMPLEXITY = O(1) Space.

(b)

```
int a=0,
for(i=0; i<N; i++){
    for( j=N; j>i; j--){
        a= a+i+j;
    }
}
```

TIME COMPLEXITY = O(N*N).

SPACE COMPLEXITY = O(1)

(c)

```
int i, j, k=0
for( i=n/2 ; i<=n ; i++) {
    for( j=2 ; j<=n ; j= j*2){
        k= k+n/2 ;
    }
}
```

TC = O(n.logn).

(d)

```
int a=0, i=N
while (i>0){
    a+=i;
    i/=2;
}
```

TIME COMPLEXITY = O(log N).

(e)

```
int fun( int n)
{ for(int i=1; i<=n; i++)
```

j=1, j= 1+i, j=1+2i, j=1+3i, ---

(e)

```
int func ...
{
  for(int i=1; i<=n; i++)
  {
    for(int j=1; j<n; j+=i)
    {
         //some O(1) task
    }
  }
}
```

$j=1, j=1+i, j=1+2i, j=1+3i, \ldots$

$j=1+ki$

$1+ki<n$

$k<\frac{n-1}{i} \Rightarrow k \approx \frac{n}{i}$

$T(n)=\sum_{i=1}^{n} \frac{n}{i}$

$T(n)=n\cdot\sum_{i=1}^{n} \frac{1}{i}$

$\boxed{T(n)=n.\log n}$

(f)

```
int fun(int n)
{
  int count =0;
  for( int i=0 ;i<n; i++)
      for (int j=i ; j>0 ; j--)
            count = count+1;
    return count;
}
```

TIME COMPLEXITY = O(n^2).

(g)

```
A( )
{
 for(i=1 to n)
  {
   for (j=1 to n)
    { print "hello";
    }
  }
}
```

TIME COMPLEXITY - O(n^2).

TIME COMPLEXITY

(h)

① $f_1(n) = 2^n$.

② $f_2(n) = n^{(3/2)}$

③ $f_3(n) = n\log n$.

④ $f_4(n) = n^{(\log n)}$

$$\boxed{3 < 2 < 4 < 1}$$

(i)

```
A()
{
 for(i=1 to n)
  {
  for (j=1 to i)
   {
    for( K=1 to 100)
     print hello
    }
  }
}
```

$$\sum_{i=1}^{n} \sum_{j=1}^{i} (100 \text{ times}) = 100 \sum_{i=1}^{n} \sum_{j=1}^{i} 1$$

$$100 \sum_{i=1}^{n} i \Rightarrow 100\left(\frac{n(n+1)}{2}\right)$$

$= \underline{O(n^2)}$. TIME COMPLEXITY.

(j)

```
A()
{
 for (i=1 to n)        -> O(n)
  {
  for (j=1 to i^2)     ->
   {
    for( K=1 to n/2)   -> O(n)
     print hello
```

$$\boxed{1 + 2^2 + 3^2 + \cdots + n^2}$$

$$\sum_{i=1}^{n} i^2 * n \Rightarrow n\sum_{i=1}^{n} i^2$$

$n \,/\, n\,(n+1)\,(2n+1))$

print hello
}
}
}

$\frac{n}{2}\left(n\frac{(n+1)(2n+1)}{6}\right)$

$O(n^4)$

TIME COMPLEXITY = $O(N^4)$.

(k) A( )
```
{
  for (i=1 to n) --------> O(n)
  { for (j=1 to j<=n/2) -----> O(n)
    {
     for(K=1; K<=n; K=K*2) ---> O(log2 n)
     }
    }
  }
```

(l) A( )
```
{
 for(i=10 to n/2)  ——— O(n)
  {
   for(j=1; j<=n; j=2*j) — O(logn)
    {
     for(K=1; K<=n; K=K*2) — O(logn)
          print hello;
     }
    }
  }
```

TIME COMPLEXITY = $O(n.\log(\log n))$.

(m)
```
function(int n) {
   if (n==1) return;
   for(i=1; i<=n; i++) {
```
----> outer loop iterate n

----> for every n values of [illegible] loop runs 1 ;

```
if (n==1) return;
for(int i=1; i<=n; i++) {
    for(int j=1; j<=n; j++) {
        printf("*");
        break;
    }
}
}
```

for every n value of inner loop runs 1; because of Break Statement.