Complexity theory Part2

Friday, October 29, 2021 10:09 AM

```
ADD (A, B, n)
{
  for(i=0; i<n, i++) ———————— n+1          n
  {
    for(j=0; j<n; j++) ———————— n*n+1 = n^2+n
    {
      c[i,j] = A[i,j] + B[i,j] ———— n*n    n^2
    }
  }                                         f(n)=2n^2+2n+1
           S(n) = O(n^2)          O(n^2)
}
```

```
Factorial(n)
{
  if (n<=1)
     return 1;
  else
     return n*factorial(n-1); ———— n
}
```

```
ARRAY OPERATION(A, n)
{
  for (int i=0; i<n; i++) ———————— n
      A[i]=0
  for (i=0; i<n; i++) ———————— n
  {
    for (j=0; j<n; j++) ———————— n×n      O(n^2)
                                 =n2
    {
      A[i] += A[i] + i + j; ——— n×n=n2
    }
  }
}
```

```
Multiply(A, B, n)
{
  for(i=0; i<n; i++) ———————————— n = n
  {
    for(j=0; j<n; j++) ———————————— n*n = n^2
    {
      c[i,j]=0;
      for(k=0; k<n; k++) ———————————— n*n*n = n^3
```

for (k=0; k<n; k++) —— $n*n*n = n^3$
{
C[i,j] = C[i,j] + A[i,k] * B[k,j] —— $n*n*n = n^3$
}
$O(n^3)$
}
}
}

SOME RECOGNIZABLE PATTERNS

```
for (i=0; i<n; i++)
    // simple STATEMENT  ——> n
for (i=n; i>0; i--)
    // SIMPLE STATEMENTS ——> n

for (i=1; i<n; i+=2)    n/2   O(n)
    // ____

for (i=0; i<n; i++) —— n+1
{ for (j=0; j<n; j++) —— n+1 * n+1
    { // ____          n x n     O(n^2)
    }
}

for (i=0; i<n; i++)
    for (j=n; j>0; j--)
        // ____                  O(n^2)
```

```
for (i=0; i<n; i++)
{
    for (j=0; j<i; j++)
    {
        // —— CODE GOES HERE
    }
}
```

$f(n) = 1+2+3+4 \cdots + n$

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

$O(n^2)$

i	j	CODE	
0	0	no exec	
1	0	exec	1
1	1	no exec	
2	0	exec	
2	1	exec	2
2	2	no exec	
3	0	exec	
3	1	exec	3
3	2	exec	
3	3	no exec	

$O(n^2)$

3	2	exec
3	3	no exec
⋮	⋮	⋮

4

```
p=0
for(i=1; p<=n; i++)
{
   p=p+i
}
```

i	p	CODE
1	0	0+1 = 1
2	1	0+1+2=3
3	3	0+1+2+3=6
4	6	0+1+2+3+4=10
⋮	⋮	⋮
K		1+2+3+4+....+K

SINCE P ACCORDING TO CODE EXECUTION = SUM OF ALL NATURAL NUMBERS

$$P = \boxed{\frac{K(K+1)}{2}}$$

ASSUMING END CONDITION

$P > n$

$$\frac{K(K+1)}{2} > n \Rightarrow \boxed{\frac{K^2+K}{2}} > n$$

$$\approx K^2 > n \Rightarrow K > \sqrt{n}$$

$O(\sqrt{n})$

```
for(i=1; i<n; i=i*2)
{
   "________
}
```

i
1
1*2 = 2
2x2 = 2^2
2^2x2 = 2^3
⋮

SINCE $i=i*2$ THE CODE DOES NOT EXECUTE FOR N TIMES. WE KNOW IT WILL EXECUTE FOR 2^K.

... $i = 2^K$

WILL EXECUTE FOR 2^k.

$i > n$ & $i = 2^k$

$2^k > n \Rightarrow k = \log_2 n \Rightarrow O(\log n)$