Note:

1<logn<$\sqrt{n}$< n<nlogn<n2<n3<…….<2n<3n<nn

Mathematical Analysis of Recursive Algorithms

Q1: void Test(int n)

 {

 If (n>0)

 {

 Print (n)

 Test(n-1)

 }

 }

Q2: void Test(int n)

 {

 If (n>0)

 {

 For(i=0;i<n;i++)

 { Print (n) }

 Test(n-1)

 }

 }

Q3: void Test(int n)

 {

 If (n>0)

 {

 For(i=0;i<n;i=i\*2)

 { Print (n) }

 Test(n-1)

 }

 }

T(n)=T(n-1)+1 🡪 O(n)

T(n)=T(n-1)+n 🡪 O(n2)

T(n)=T(n-1)+logn 🡪 O(nlogn)

We can conclude:

T(n)=T(n-1)+n2 🡪 O(n3)

T(n)=T(n-2)+1 🡪 n/2 🡪O(n)

T(n)=T(n-100)+n 🡪 n2/100 🡪O(n2)

T(n)=2T(n-1) + 1 🡺 ???

Q4: void Test(int n)

 {

 If (n>0)

 {

 Print (n)

 Test(n-1)

 Test(n-1)

 }

 }

T(n)=T(n-1)+1 🡪 O(n)

T(n)=T(n-1)+n 🡪 O(n2)

T(n)=T(n-1)+logn 🡪 O(nlogn)

T(n)=T(n-1)+n2 🡪 O(n3)

T(n)=T(n-2)+1 🡪 n/2 🡪O(n)

T(n)=T(n-100)+n 🡪 n2/100 🡪O(n2)

Now

T(n)=2T(n-1) + 1 🡺 O(2n)

T(n)=3T(n-1)+1 🡪 O(3n)

T(n)= 2T(n-1)+n 🡪 O(n2n)

T(n) = 3T(n-1)+logn 🡪 O(3n X logn)

T(n) = T(n/2) + 1 ----???

Q5: Algorithm Test ( int n)

{

 If(n>1)

 {

 Print (n)

 Test(n/2)

 }

}

Q6: Algorithm Test ( int n)

{

 If(n>1)

 {

 For(i=0;i<n;i++)

 { Print (n) }

 Test(n/2)

 }

}

Q7: Algorithm Test ( int n)

{

 If(n>1)

 {

 For(i=0;i<n;i++)

 { Print (n) }

 Test(n/2)

 Test(n/2)

 }

}

T(n)=T(n/2) + 1 🡺 O(logn)

T(n)=T(n/2) + n 🡺 O(n)

T(n)=2T(n/2)+n 🡪O(nlogn)

Q8: Algorithm Test ( int n) **(HW)**

{

 If(n>1)

 {

 Print (n)

 Test($\sqrt{n}$)

 }

}