Análisis de algoritmos recursivos

Estimar el tiempo de ejecución del siguiente algoritmo recursivo.

public int calcular (int x, int n) **T(1)=a**

{

 if (n==0) **T(1) = a**

 {

 return 1;

 }

 else

 {

 if (n%3==0) **T(1)=a**

 {

 return calcular(x, n/2) \* calcular(x, n/2); **→ 2(T(n/2));**

 }

 else

 {

 return calcular(x, n/2) \* calcular(x, n/2) \* x;  }

 }

}

**T(n)= 2(T(n/2))+T(1)**

**T(n)= 2(T(n/2))+b**

**T(n)= 2(T(2/2))+b**

**T(n)= 2(T(1))+b**

**T(n)= 2a+b**

Estimar el tiempo de ejecución del siguiente algoritmo recursivo.

public int recursivo(int n ) { **T(1)=a**

if (n == 1) { **T(1)=a**

return n;

} else {

return recursivo(n-1)+recursivo(n-1)+recursivo(n-1)+

recursivo(n-1); **🡪 4(T(n-1))**

}

}

**T(n)= 4(T(n-1)) + T(1)**

**T(n)= 4(T(2-1)) + b**

**T(n)= 4(T(1)) + b**

**T(n)= 4a+ b**

Estimar el tiempo de ejecución del siguiente algoritmo recursivo.

public int recursivo(int n )

{

if (n == 1)

{

return n;

}

else

{

return recursivo(n-1)+recursivo(n-1)+recursivo(n-1) +

recursivo(n-1) + recursivo(n-1);

}

}

Estimar el tiempo de ejecución del siguiente algoritmo recursivo.

public int recursivo( int n )

{

if ( n <= 1 ){

return 1;

}

else

{

for( i = 1 ; i <= n ; i++ )

{

j = j + 1;

}

return ( recursivo(n/2) + recursivo(n/2) );

}

}

Suponga que tiene las siguientes funciones de base e inducción. Deduzca el tiempo de ejecución.

1. T(1) = a

T(n) = 2T(n-1) + bn

T(n) = 2T(2-1) + bn

T(n) = 2T(1) + bn

T(n) = 2a + 2b

T(n) = 2T(3-1) + bn

T(n) = 2T(2) + 3b

T(n) = 2(2T(2-1)+2b)+3b

T(n) =  2(2T(1)+2b)+3b

T(n) =  2(2a+2b)+3b

T(n) = 4a+7b

T(n) = 2T(4-1) + bn

T(n) = 2T(3) + 4b

T(n) = 2(2T(3-1)+3b)+4b

T(n) = 2(2T(2)+3b)+4b

T(n) = 2(2(2T(2-1)+2b)+3b)+4b

T(n) = 2(2(2T(1)+2b)+3b)+4b

T(n) = 8a+18b

T(n) = 2T(5-1) + 5b

T(n) = 2T(4) + 5b

T(n) = 2(2T(4-1)+4b) +5b

T(n) = 2(2(2T(3-1)+3b)+4b)+5b

T(n) = 2(2(2(2T(2-1)+2b)+3b)+4b)+5b

T(n) = 16a+41b

T(n) = 2^(n-1)a+((1/8)\*((n-1)-1)\*((n-1)^3-5\*(n-1)^2+18\*(n-1)-8))\*b

1. T(1)= a

T(n) = T(n/2) + bn

3. T(1)= a

T(n) = 2T(n/2) + bn

4. T(1)= a

T(n) = 4T(n/2) + b

5. T(1)= a

T(n) = 7T(n-1) + a