

INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE CÓMPUTO
ANÁLISIS DE ALGORITMOS
EVALUACIÓN A TÍTULO DE SUFICIENCIA

EJERCICIO 1. Considere el pseudocódigo del BLOQUE 1 y considerando la notación asintótica Big O exprese su complejidad.

EJERCICIO 2. En el BLOQUE 2 se presentan una forma de realizar la impresión en consola de los elementos de un arreglo, proponga una distinta, calcule la complejidad de ambas y, en función de ello, determine ¿qué opción es mejor?

EJERCICIO 3. La función recursiva expuesta en el BLOQUE 3 permite calcular el producto de todos los números enteros positivos desde 0 hasta N, empleando ecuaciones de recurrencia determine su complejidad.

EJERCICIO 4. Calcule la complejidad del pseudocódigo expuesto en el BLOQUE 4.

EJERCICIO 5. En el BLOQUE 5, se presenta un código en lenguaje de programación c++ mediante el cual es posible calcular la sumatoria de los elementos de un arreglo, proponga uno distinto, evalúe las complejidades de ambas propuestas y, en función de ello, determine ¿qué opción es mejor?

EJERCICIO 6. Determine la complejidad de las líneas de pseudocódigo expuestas en el BLOQUE 6.

EJERCICIO 7. Al conjunto de problemas en los que podemos comprobar en un tiempo razonable si una respuesta al problema es correcta o no, se les conoce como:

EJERCICIO 8. Al conjunto de problemas en los que podemos encontrar una respuesta al problema en un tiempo razonable se les denomina:

EJERCICIO 9. Determina la complejidad del algoritmo del BLOQUE 7.

EJERCICIO 10. En el BLOQUE 8 se presenta un pseudocódigo que realiza una determinada tarea, proponga una estrategia alternativa que realice la misma actividad, evalúe ambas y determine ¿cuál es mejor opción en función de sus complejidades?

BLOQUE 1	BLOQUE 2
<pre> for i = 0 hasta N: j= 1 while j < N: print j j=j * 2 </pre>	<pre> for(int i =0; i<N;i++){ cout<<lista[i]<<endl; } </pre>
BLOQUE 3	BLOQUE 4
<pre> int funcion(N){ if(n==0) return 1; else return N*funcion(N-1) } </pre>	<pre> cont=1; do{ x = x + a[cont]; x = x + b[cont]; cont = cont + 1; }while(cont <= n) </pre>
BLOQUE 5	BLOQUE 6
<pre> int suma=0; for(int i=0;i<N;i++) suma=suma+arreglo[i] return suma </pre>	<pre> i=2 while(i<n) j=3 while(j<n) j=j+1 i++ </pre>
BLOQUE 7	BLOQUE 8
<pre> for(int i=n/2; i<n; i++) cout<<"Hola mundo"<<endl; </pre>	<pre> funcion(valor, A, n){ int i=1; while(i<=n && A[i]!=valor){ i=i+1; } return i; } </pre>