INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE CÓMPUTO

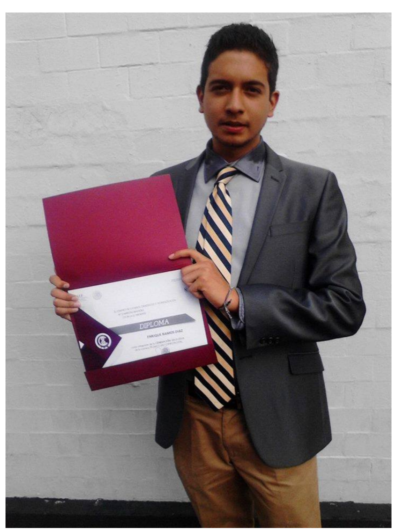
Ejercicio 04: Análisis de algoritmos no recursivos

Unidad de aprendizaje: Análisis de Algoritmos

Grupo: 3CM3

Alumno:
Ramos Diaz Enrique

Profesor(a):
Franco Martínez Edgardo Adrián



9 de Octubre 2018

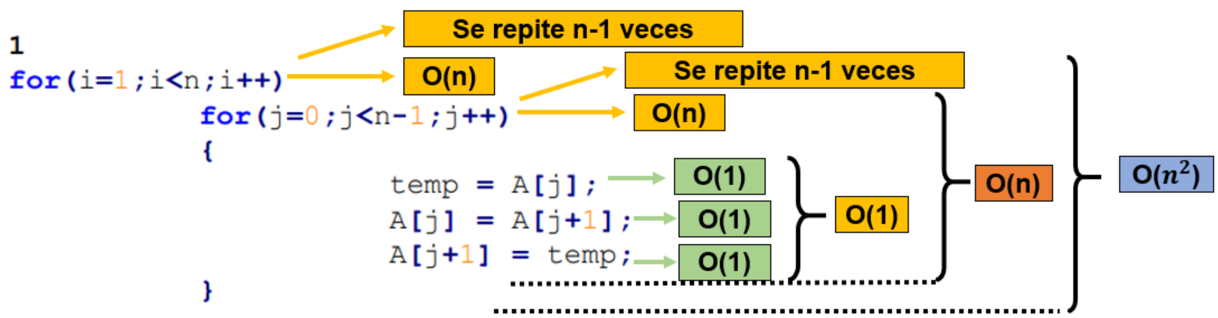
Índice

1. Para los siguientes 12 algoritmos determine la cota O()

Operaciones básicas para el análisis de la complejidad temporal: asignación, aritméticas, condicionales y saltos implícitos.

1.1. Algoritmo 1

Análisis de Cotas



Análisis de complejidad temporal

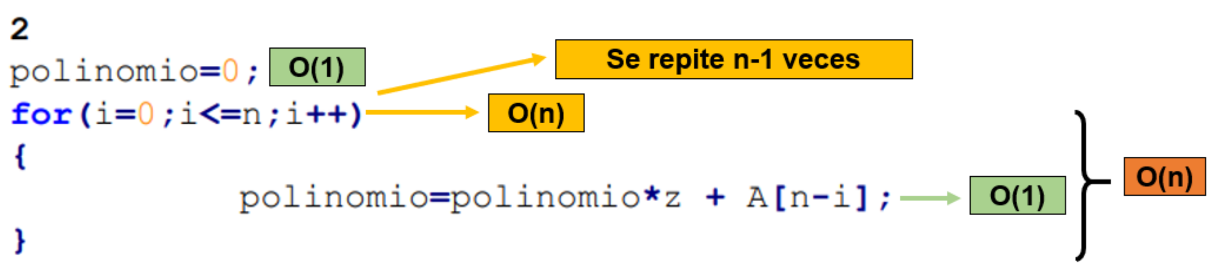
Observamos que el algoritmo esta conformado principalmente por dos ciclos for anidados.

La función temporal es $f_t(n) = 8n^2 - 10n + 5$

La función temporal del algoritmo obtenida en análisis previo del Ejercicio02 corresponde al orden de complejidad obtenido en el análisis de cotas $O(n^2)$.

1.2. Algoritmo 2

Análisis de Cotas



Análisis de complejidad temporal

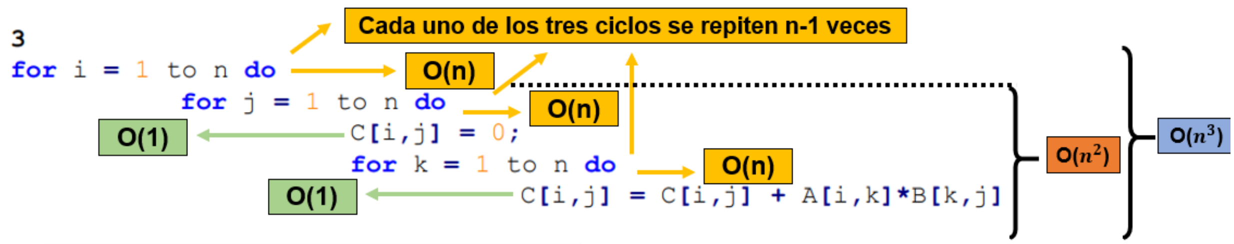
El algoritmo consiste de un unico for, y dentro de él existen par de asignaciones y operaciones aritméticas.

La función temporal es $f_t(n) = 7n + 11$

La función temporal del algoritmo obtenida en análisis previo del Ejercicio02 corresponde al orden de complejidad obtenido en el análisis de cotas $O(n)$.

1.3. Algoritmo 3

Análisis de Cotas



Análisis de complejidad temporal

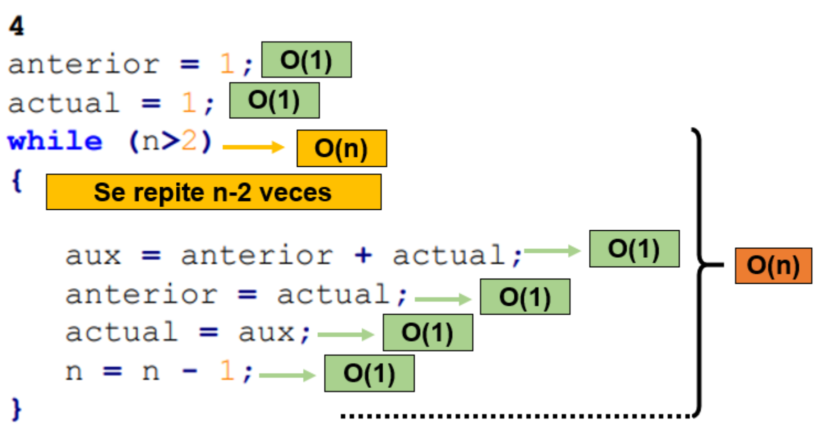
Existen 3 ciclos for anidados en este algoritmo, en donde la mayor parte de asignaciones y operaciones aritméticas entre los arreglos A, B y C ocurren en el tercer ciclo.

La función temporal es $f_t(n) = 6n^3 + 5n^2 + 7n + 3$

La función temporal del algoritmo obtenida en análisis previo del Ejercicio02 corresponde al orden de complejidad obtenido en el análisis de cotas $O(n^3)$.

1.4. Algoritmo 4

Análisis de Cotas



Análisis de complejidad temporal

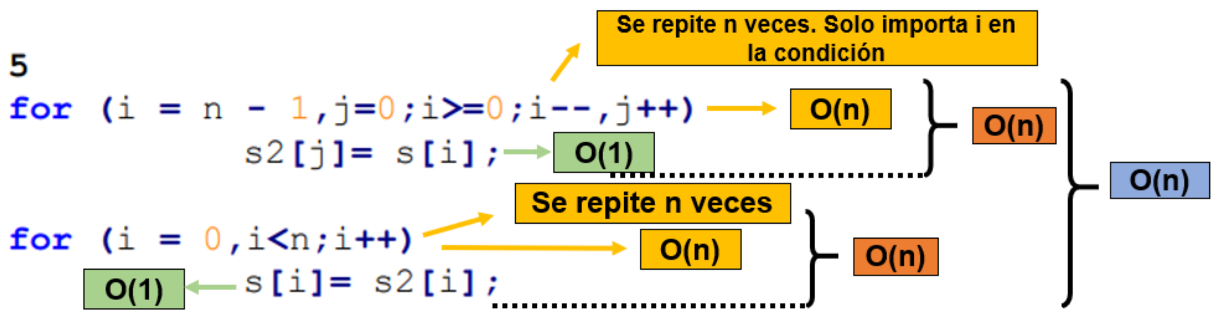
En este algoritmo únicamente existen un único ciclo while, en donde se hace la mayoría de asignaciones y operaciones matemáticas.

La función temporal es $f_t(n) = \begin{cases} 4 & \text{si } n \leq 2 \\ 8n - 12 & \text{si } n > 2 \end{cases}$

La función temporal del algoritmo (siendo el caso n>2) obtenida en análisis previo del Ejercicio02 corresponde al orden de complejidad obtenido en el análisis de cotas $O(n)$.

1.5. Algoritmo 5

Análisis de Cotas



Análisis de complejidad temporal

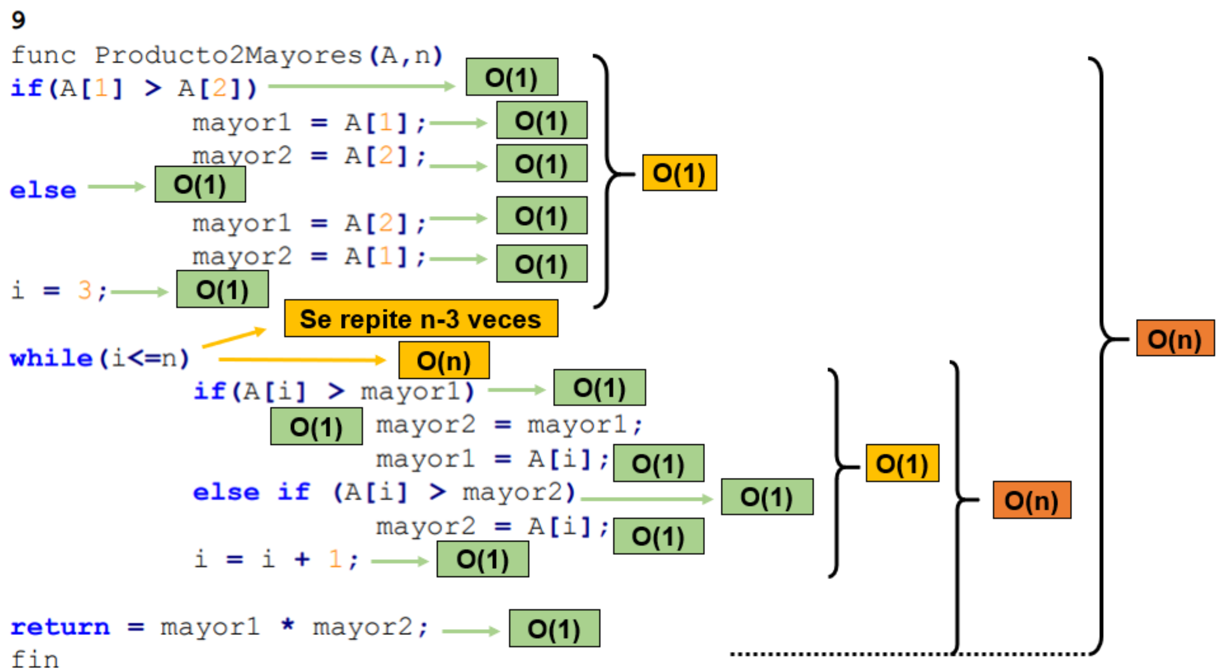
Similar a algoritmos anteriores, existen dos ciclos for. Sin embargo, estos no están anidados, sino son independientes uno de otro. Pareciera que la complejidad de este algoritmo esta en el orden de n^2, pero esto es incorrecto.

La función temporal es $f_t(n) = 9n + 8$

La función temporal del algoritmo obtenida en análisis previo del Ejercicio02 corresponde al orden de complejidad obtenido en el análisis de cotas $O(n)$.

1.6. Algoritmo 9

Análisis de Cotas



Análisis de complejidad temporal

Mejor Caso $f_{tmc}(n) = 3 + 2(n-2)$

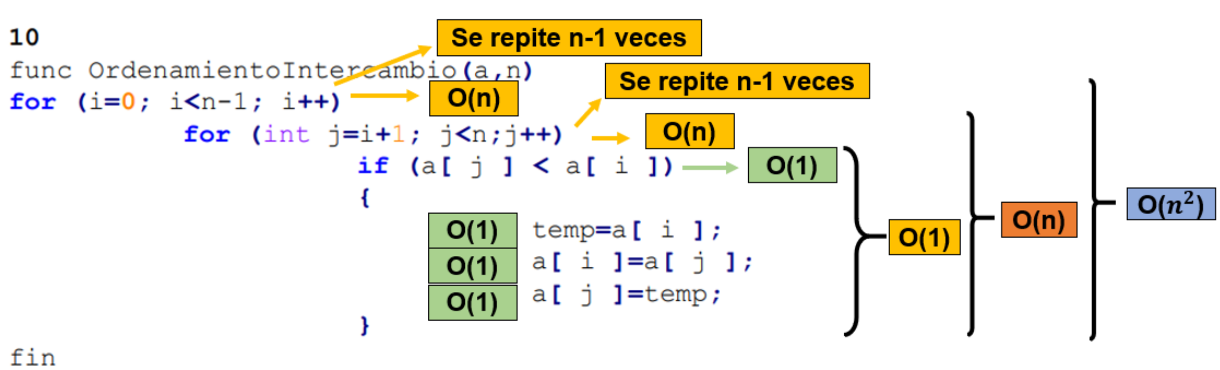
Peor Caso $f_{tpc}(n) = 3 + 3(n-2)$

Caso Medio $f_{tcm}(n) = \frac{8}{3}n - \frac{7}{3}$

La función temporal del algoritmo obtenida en análisis previo del Ejercicio02 corresponde al orden de complejidad obtenido en el análisis de cotas $O(n)$ (para los tres casos).

1.7. Algoritmo 10

Análisis de Cotas



Análisis de complejidad temporal

Mejor Caso $f_{tmc}(n) = (n-1)^2$

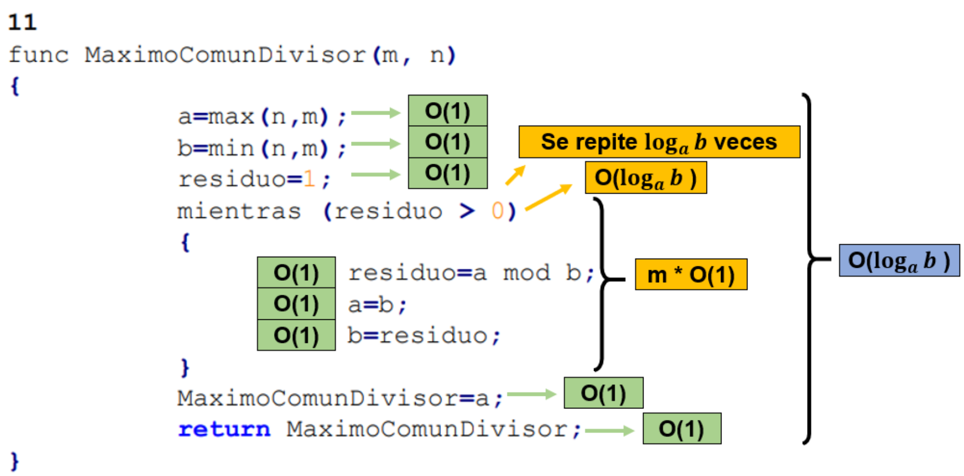
Peor Caso $f_{tpc}(n) = 4(n-1)^2$

Caso Medio $f_{tcm}(n) = \frac{5}{2}(n-1)^2$

La función temporal del algoritmo obtenida en análisis previo del Ejercicio02 corresponde al orden de complejidad obtenido en el análisis de cotas $O(n^2)$ (para los tres casos).

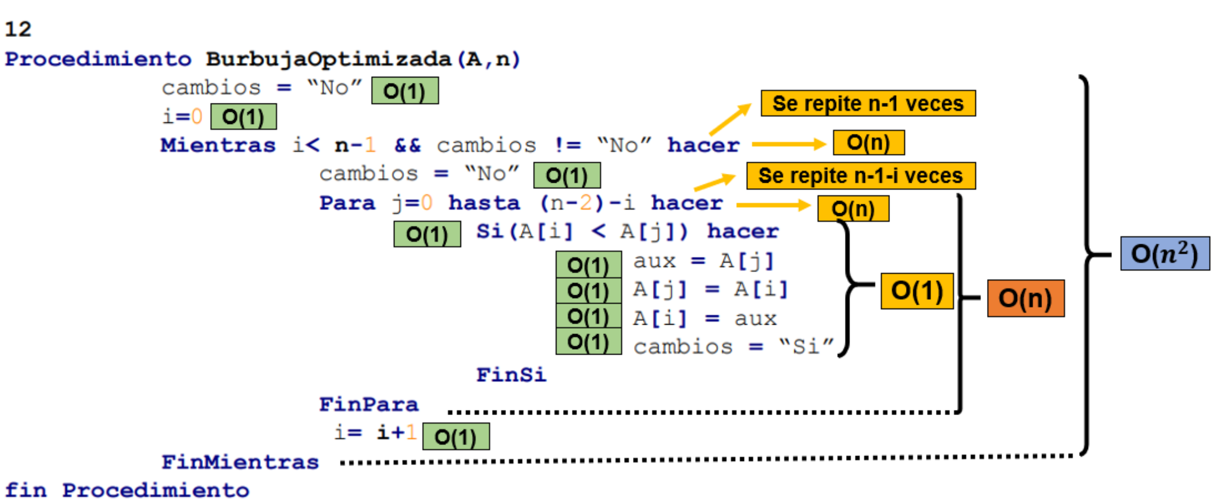
1.8. Algoritmo 11

Análisis de Cotas



1.9. Algoritmo 12

Análisis de Cotas



Análisis de complejidad temporal

Mejor Caso $f_{tmc}(n) = n + \frac{n(n-1)^2}{2}$

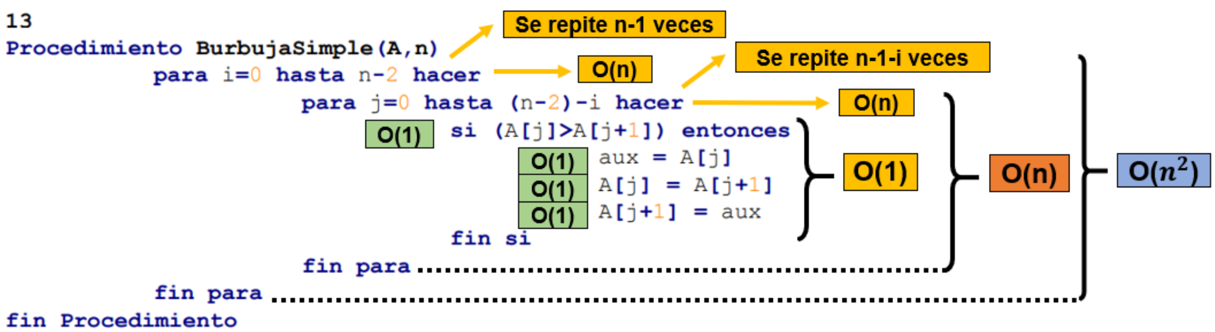
Peor Caso $f_{tpc}(n) = n + \frac{5}{2}n(n-1)^2$

Caso Medio $f_{tcm}(n) = n + \frac{3}{2}n(n-1)^2$

La función temporal del algoritmo obtenida en análisis previo del Ejercicio02 NO corresponde al orden de complejidad obtenido en el análisis de cotas $O(n^2)$, siendo de orden n^3.

1.10. Algoritmo 13

Análisis de Cotas



Análisis de complejidad temporal

Mejor Caso $f_{tmc}(n) = \frac{n(n-1)^2}{2}$

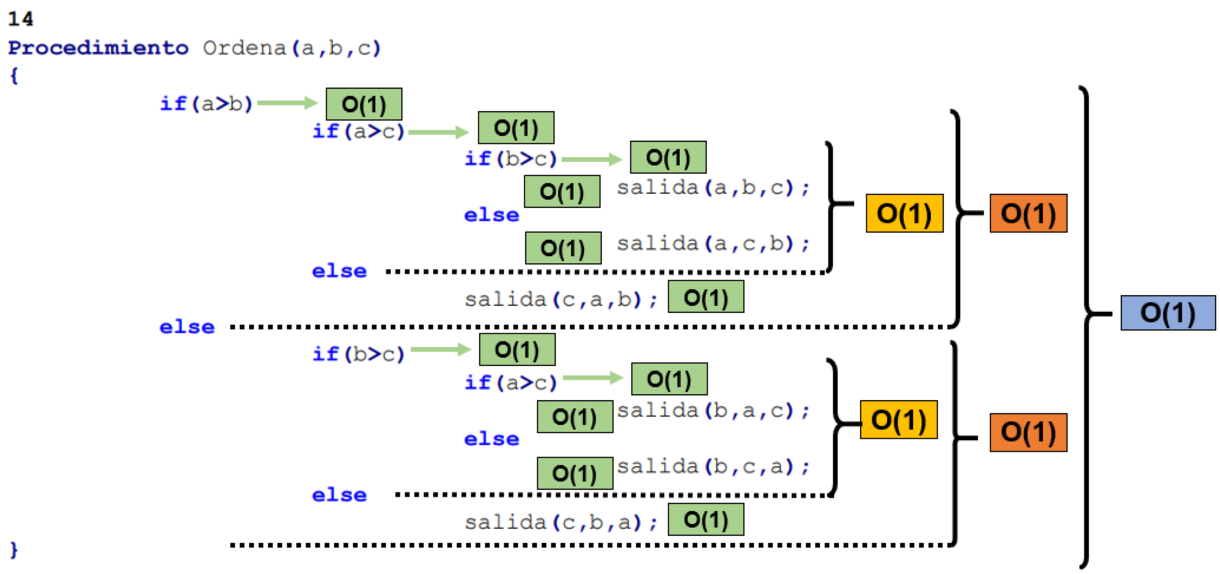
Peor Caso $f_{tpc}(n) = 2n(n-1)^2$

Caso Medio $f_{tcm}(n) = 5n(n-1)^2$

La función temporal del algoritmo obtenida en análisis previo del Ejercicio02 NO corresponde al orden de complejidad obtenido en el análisis de cotas $O(n^2)$, siendo de orden n^3.

1.11. Algoritmo 14

Análisis de Cotas



Análisis de complejidad temporal

Mejor Caso $f_{tmc}(n) = 2$

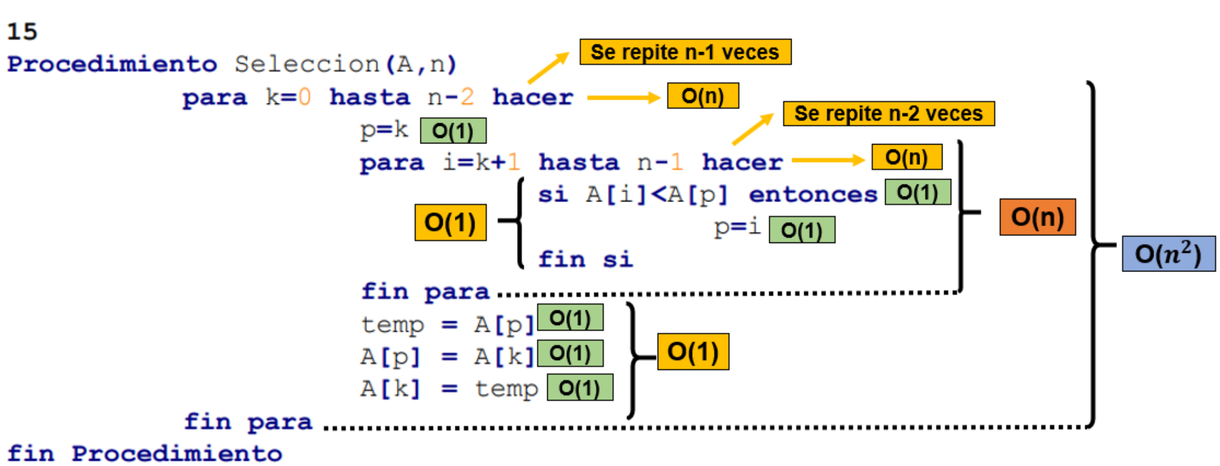
Peor Caso $f_{tpc}(n) = 3$

Caso Medio $f_{tcm}(n) = 2.66$

La función temporal del algoritmo obtenida en análisis previo del Ejercicio02 corresponde al orden de complejidad obtenido en el análisis de cotas $O(1)$ (para los tres casos).

1.12. Algoritmo 15

Análisis de Cotas



Análisis de complejidad temporal

Mejor Caso $f_{tmc}(n) = 2n(n-1)^2 + 4(n-1)$

Peor Caso $f_{tpc}(n) = 3n(n-1)^2 + 4(n-1)$

Caso Medio $f_{tcm}(n) = \frac{5}{2}n(n-1)^2 + 4(n-1)$

La función temporal del algoritmo obtenida en análisis previo del Ejercicio02 NO corresponde al orden de complejidad obtenido en el análisis de cotas $O(n^2)$, siendo de orden n^3.