

Cálculo de T(N)

Teniendo en cuenta que los cuatro siguientes algoritmos realizan la misma tarea: Cargar un arreglo de N componentes:

- Calcular el T(N) de cada uno de ellos e indicar a qué orden de complejidad pertenecen.
- Recomendar el uso de uno de ellos justificando la respuesta.

Void Carga1(entero arre[N])

Comienzo

Entero i,num;

Para i desde 0 hasta N-1 $1+N+1+N = 2N+2$

Leer(num); $1N$

Arre[i]=num; $2N$

Finpara

Retorna(); 1

Fin

$T_1(N) = 5N+3$, $T_1(N) \in O(N)$

Void Carga2(entero arre[N])

Comienzo

Entero i;

Para i desde 0 hasta N-1 $1+N+1+N = 2N+2$

Leer(Arre[i]); $2N$

Finpara

Retorna(); 1

fin

$T_2(N) = 4N+3$, $T_2(N) \in O(N)$

Comparemos $T_1(N)$ y $T_2(N)$: $\text{¿}T_1(N) < T_2(N)\text{?}$ ó $\text{¿}T_1(N) > T_2(N)\text{?}$

- Consideremos que: $T_2(N) < T_1(N)$, entonces: $4N+3 < 5N+3$, luego $3-3 < 5N-4N$, esto es: $N > 0$; esta expresión relacional es una tautología, pues es verdadera para todo número natural N.
- Consideremos que: $T_2(N) > T_1(N)$, entonces: $4N+3 > 5N+3$, luego $3-3 > 5N-4N$, esto es: $N < 0$; esta expresión relacional es un absurdo, pues es falsa para todo número natural N.

Por lo tanto, de las dos cargas anteriores (ambas usan el para_finpara) elegimos la Carga2.

Void Carga3(entero arre[N])

Comienzo

Entero i,num;

i=0; 1

Mientras(i<N) $N+1$

Leer(num); $1N$

Arre[i]=num; $2N$

i=i+1; $2N$

Finmientras

Retorna(); 1

fin

$T_3(N) = 6N+3$, $T_3(N) \in O(N)$

Void Carga4(entero arre[N])

Comienzo

Entero i;

I=0; 1

Mientras(i<N) $N+1$

Leer(Arre[i]); $2N$

I=i+1; $2N$

Finmientras

Retorna(); 1

fin

$T_4(N) = 5N+3$

$T_4(N) \in O(N)$

Comparemos $T_3(N)$ y $T_4(N)$: $\text{¿}T_3(N) < T_4(N)\text{?}$ ó $\text{¿}T_3(N) > T_4(N)\text{?}$

- Consideremos que: $T_4(N) < T_3(N)$, entonces: $5N+3 < 6N+3$, luego $3-3 < 6N-5N$, esto es: $N > 0$; esta expresión relacional es una tautología, pues es verdadera para todo número natural N.
- Consideremos que: $T_4(N) > T_3(N)$, entonces: $5N+3 > 6N+3$, luego $3-3 > 6N-5N$, esto es: $N < 0$; esta expresión relacional es un absurdo, pues es falsa para todo número natural N.
- Por lo tanto de las dos cargas anteriores (ambas usan el mientras_finmientras) elegimos la Carga4.

Cabe hacernos ahora la siguiente pregunta: entre la Carga2 y la Carga4 ¿cuál recomendaríamos?

Deberíamos comparar las siguientes dos funciones: $T_2(N) = 4N+3$ y $T_4(N) = 5N+3$

- Consideremos que: $T_4(N) > T_2(N)$, entonces: $5N+3 > 4N+3$, luego $5N-4N > 3-3$, esto es: $N > 0$, con lo cual concluimos que la Carga2 tiene mejor eficiencia que la Carga4, para cualquier tamaño N del arreglo.
- Consideremos que: $T_4(N) < T_2(N)$, entonces: $5N+3 < 4N+3$, luego $5N-4N < 3-3$, esto es: $N < 0$, con lo cual concluimos que nunca la Carga4 tiene mejor eficiencia que la Carga2.

Conclusión: De las cuatro Cargas vistas la de mejor eficiencia es la Carga2.