

Ejercicio 1

(Usando 1 repetición)

Pondré FdT si pasa de 2 minutos

n	TBucle1	TBucle2	TBucle3	TBucle4
100	0	0	1	2
200	0	1	6	9
400	0	7	29	77
800	0	24	130	670
1600	0	95	525	7716
3200	1	452	2217	FdT
6400	2	1930	16622	FdT
12800	3	17587	85165	FdT
25600	11	67206	FdT	FdT
51200	14	FdT	FdT	FdT

Los tiempos tienen sentido de acuerdo a la complejidad temporal de cada uno, siendo $O(n\log n)$, $O(n^2\log n)$, $O(n^2\log n)$ y $O(n^3)$, respectivamente.

Ejercicio 2

(Ya están implementados bucle5 6 y 7 pero comprobar si están bien)

Pondré FdT cuando pase de 3 minutos (En bucle7 puse una más porque me parecía muy poco solo dos iteraciones)

n	TBucle5	TBucle6	TBucle7
100	9	119	935
200	50	1076	25325
400	221	14896	309816
800	1048	1037	FdT
1600	7750	136939	FdT
3200	40724	FdT	FdT
6400	123965	FdT	FdT

Los tiempos concuerdan con lo esperado ya que las mediciones siguen la misma tendencia de la complejidad de sus respectivas funciones.

Ejercicio 3

n	TBucle1 (t1)	TBucle2 (t2)	T1/T2
100	0	0	0
200	0	1	0
400	0	7	0
800	0	24	0
1600	0	95	0
3200	1	452	0,002
6400	2	1930	0,001
12800	3	17587	$1,7 \cdot 10^{-4}$
25600	11	67206	$1,7 \cdot 10^{-4}$
51200	14	FdT	FdT

Los tiempos tienen sentido ya que el bucle 1 es de una complejidad menor que el bucle 2.

Ejercicio 4

Asumimos t=0 como t=1 para dividir

n	TBucle3	TBucle2	T3/T2
100	1	0	1
200	6	1	6
400	29	7	4,14
800	130	24	5,41
1600	525	95	5,52
3200	2217	452	4,904
6400	16622	1930	8,61
12800	85165	17587	4,85
25600	FdT	67206	FdT
51200	FdT	FdT	FdT

El tiempo y su cociente concuerdan ya que aunque sean algoritmos de la misma complejidad, el bucle3 tardará más debido a la implementación del código, que matemáticamente se puede comprobar que va a dar resultados mayores que el bucle2.

Ejercicio 5

(Uso 10 repeticiones para Python y 1 para Java)

Marcaremos FdT aquellos tiempos que superen los 5 minutos.

Asumimos t=0 como t=1 para la división

n	Bucle4 py (t1)	Bucle4 java sin OPT (t2)	t1/t2
100	12	0	12
200	82	12	6,833
400	675	92	7,33
800	8925	630	14,16
1600	114080	7319	15,58
3200	FdT	72378	FdT
6400	FdT	FdT	FdT

Los tiempos tienen sentido ya que Python es interpretado, lo que hace que tarde más en ejecutarse, eso sumado a la complejidad cúbica del código, hará que el bucle 4 de Python tarde mucho más que el código de Java.