



PRÁCTICA 3: DIVIDE Y VENCERÁS

Gabriel García Martínez L.6



15 DE MARZO DE 2023
ALGORITMIA

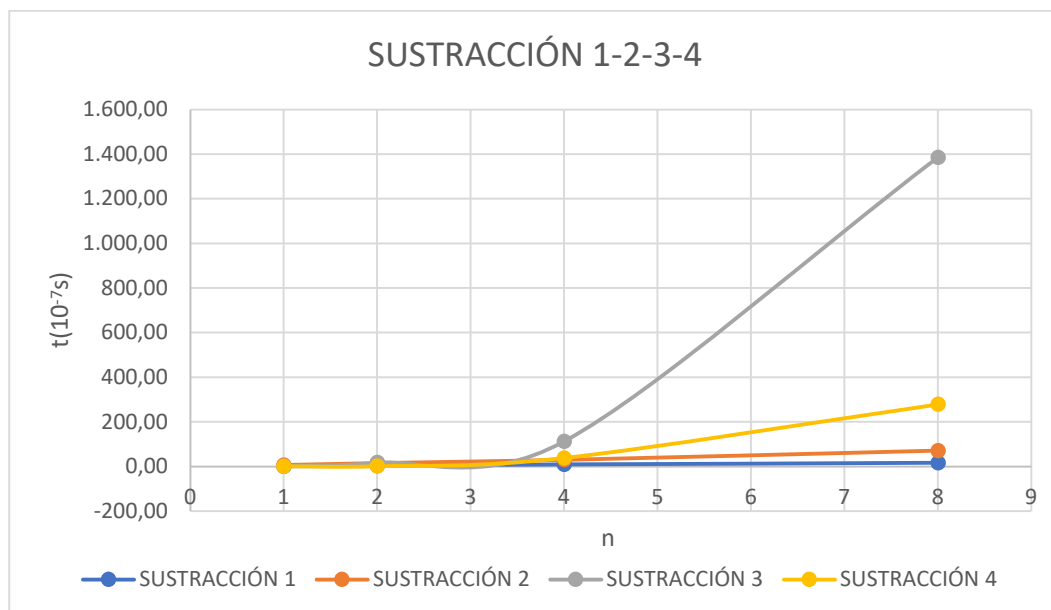
Las mediciones han sido realizadas con un ordenador con las siguientes características:

Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz 1.80 GHz

8,00 GB (7,87 GB usable)

SUSTRACCIÓN

n	SUSTRACCIÓN 1			SUSTRACCIÓN 2			SUSTRACCIÓN 3			SUSTRACCIÓN 4		
	t(10 ⁻⁹ s)	limite		t(10 ⁻⁹ s)	Limite		t(10 ⁻⁹ s)	limite		t(10 ⁻⁹ s)	limite	
1	0,28	1000000000		6,30	100000000		0,48	1000000000		0,26	1000000000	
2	3,42	1000000000		15,30	100000000		16,90	1000000000		0,92	1000000000	
4	8,81	1000000000		30,10	100000000		112,61	1000000000		38,00	100000000	
8	16,10	100000000		71,10	100000000		1.386,00	1000000		278,10	10000000	
16	31,00	100000000		245,80	100000000		346.500,00	10000		27.700,00	10000	
32	54,50	100000000		850,40	100000000		12.201.000.000,00	1		93.000.000,00	1	
64	124,60	100000000		3.232,20	100000000		>25 min			>25 min		
128	248,40	100000000		11.900,00	10000							
256	499,00	100000000		40.900,00	10000							
512	975,30	100000000		151.300,00	10000							
1024	1.620,00	100000		589.100,00	10000							
2048	4.520,00	100000		2.342.800,00	10000							
4096	8.550,00	100000		9.342.100,00	10000							
8192	17.330,00	100000		37.234.600,00	10000							
16384	35.560,00	100000		35.400.000,00	10							
32768	72.950,00	100000		126.300.000,00	10							



Como se puede observar en la gráfica, la complejidad de sustracción 1 es $O(n)$, la complejidad de sustracción 2 es $O(n^2)$, la de 3 es $O(2^n)$ y de 4 $O(3^{n/2})$. Por lo que la curva de Sustracción 3 es la más pronunciada. Comprobaciones:

Sustracción 1:

$$n_1=64 \quad n_2=128 \quad t_1=124,6 \text{ (} 10^{-9} \text{ s)}$$

$$t_2= 249,2(10^{-9} \text{ s}) \approx 248,4(10^{-9} \text{ s})$$

Sustracción 2:

$$n_1=16 \quad n_2=32 \quad t_1=245,8 \text{ (} 10^{-9} \text{ s)}$$

$$t_2= 983,2(10^{-9} \text{ s}) \approx 850,4(10^{-9} \text{ s})$$

Sustracción 3:

$$n_1=2 \quad n_2=4 \quad t_1=16,9 \text{ (} 10^{-9} \text{ s)}$$

$$t_2= 67,6(10^{-9} \text{ s}) \approx 112,61(10^{-9} \text{ s})$$

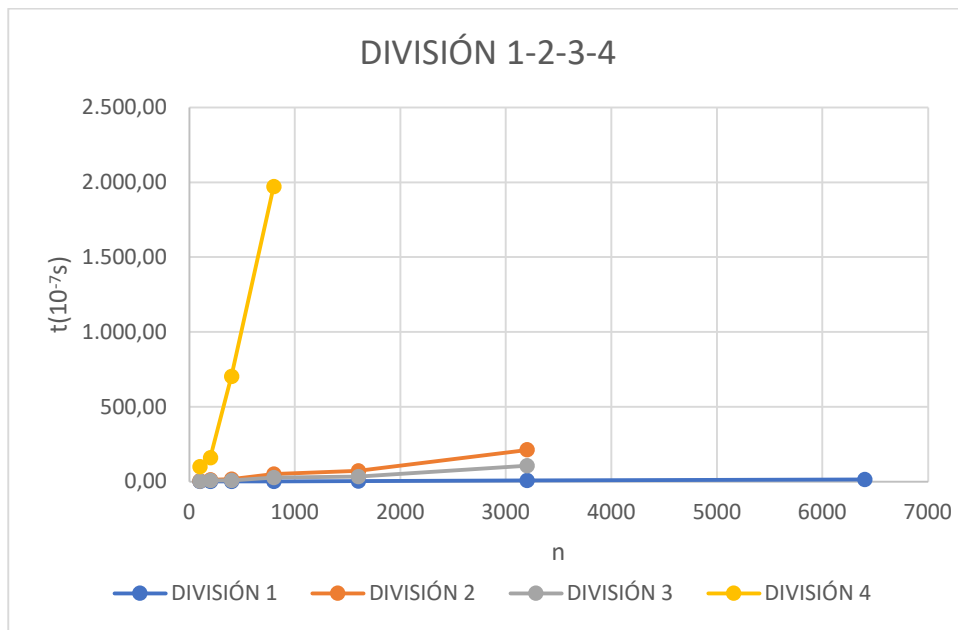
Sustracción 4:

$$n_1=4 \quad n_2=8 \quad t_1=38 \text{ (} 10^{-9} \text{ s)}$$

$$t_2= 342(10^{-9} \text{ s}) \approx 278,1(10^{-9} \text{ s})$$

DIVISIÓN

	DIVISIÓN 1		DIVISIÓN 2		DIVISIÓN 3		DIVISIÓN 4	
N	t (10 ⁻⁷ s)	Límite	t (10 ⁻⁷ s)	Límite	t (10 ⁻⁷ s)	Límite	t (10 ⁻⁷ s)	Límite
100	0,39	10000000	5,70	100000	2,40	100000	99,00	10000
200	0,57	10000000	12,10	100000	6,80	100000	159,00	10000
400	1,09	10000000	16,60	100000	8,50	100000	702,00	10000
800	2,04	10000000	50,00	100000	26,80	100000	1.972,00	10000
1600	3,76	10000000	72,50	100000	34,00	100000	6.434,00	10000
3200	7,70	100000	212,70	100000	106,40	100000	25.981,00	10000
6400	14,80	100000	309,70	100000	133,90	100000	156.000,00	10
12800	29,40	100000	897,40	100000	422,70	100000	770.000,00	1
25600	57,70	100000	1.347,00	100000	534,20	100000	2.290.000,00	1
51200	113,60	100000	3.789,90	100000	1.683,00	100000	8.630.000,00	1
102400	226,10	100000	5.749,80	100000	2.136,90	100000	36.860.000,00	1
204800	453,20	100000	16.500,00	100	6.980,00	1000	126.710.000,00	1
409600	908,80	100000	25.900,00	100	8.630,00	1000	449.870.000,00	1
819200	1.792,70	100000	68.100,00	100	27.140,00	1000	1.421.120.000,00	1
1638400	3.603,20	100000	109.600,00	100	34.450,00	1000	>15 min	
3276800	7.188,30	100000	284.200,00	100	108.650,00	1000		
6553600	11.400,00	100	461.600,00	100	137.630,00	1000		
13107200	22.300,00	100	1.186.500,00	100	437.000,00	10		
26214400	44.400,00	100	1.932.700,00	100	555.000,00	10		
52428800	88.600,00	100	4.936.300,00	100	1.745.000,00	10		
104857600	176.200,00	100	8.140.800,00	100	2.216.000,00	10		
209715200	352.800,00	100	19.840.000,00	1	7.003.000,00	10		
419430400	707.200,00	100	33.780.000,00	1	8.793.000,00	10		
838860800	1.412.100,00	100	82.930.000,00	1	27.812.000,00	10		



Como se puede observar en la grfica, la complejidad de Divisi3n 1 es $O(n)$, al tratarse de un esquema " $a < b^k$ ", la de 2 es $O(n \cdot \log n)$, al tratarse de un esquema " $a = b^k$ ", divisi3n 3 tiene una complejidad $O(n^{\log_2 2})$, con un esquema " $a > b^k$ " y divisi3n 4 tiene $O(n^2)$, ya que se trata de un esquema " $a < b^k$ " con $a=4$, $b=3$, $k=2$. Comprobaciones:

Divisi3n 1:

$$n_1=6400 \quad n_2=12800 \quad t_1=14,8 \text{ (} 10^{-7} \text{ s)}$$

$$t_2=12800^2 \cdot 6400^2 \cdot 14,8 = 29,6 \text{ (} 10^{-7} \text{ s)} \approx 29,4 \text{ (} 10^{-7} \text{ s)}$$

$$n_1=12800 \quad n_2=25600 \quad t_1=29,4 \text{ (} 10^{-7} \text{ s)}$$

$$t_2=25600^2 \cdot 12800^2 \cdot 29,4 = 58,8 \text{ (} 10^{-7} \text{ s)} \approx 57,7 \text{ (} 10^{-7} \text{ s)}$$

Divisi3n 2:

$$n_1=6400 \quad n_2=12800 \quad t_1=309,7 \text{ (} 10^{-7} \text{ s)}$$

$$t_2=668,39 \text{ (} 10^{-7} \text{ s)} \approx 897,4 \text{ (} 10^{-7} \text{ s)}$$

Divisi3n 3:

$$n_1=400 \quad n_2=800 \quad t_1=8,5 \text{ (} 10^{-7} \text{ s)}$$

$$t_2=17 \text{ (} 10^{-7} \text{ s)} \approx 26,8 \text{ (} 10^{-7} \text{ s)}$$

Divisi3n 4:

$$n_1=1600 \quad n_2=3200 \quad t_1=6434 \text{ (} 10^{-7} \text{ s)}$$

$$t_2=25736 \text{ (} 10^{-7} \text{ s)} \approx 25981 \text{ (} 10^{-7} \text{ s)}$$

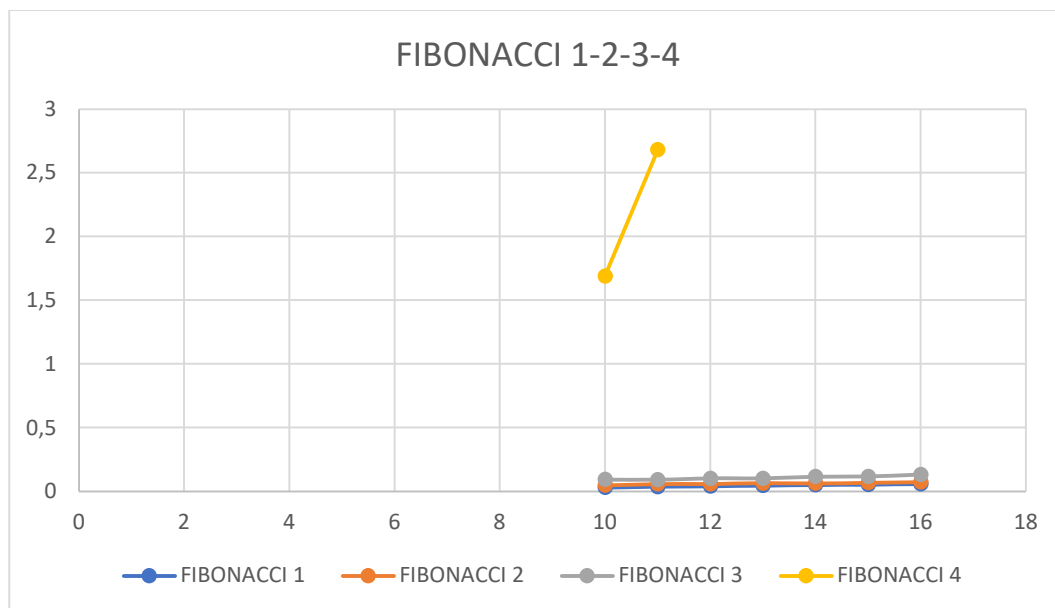
$$n_1=25600 \quad n_2=512000 \quad t_1=2290000 \text{ (} 10^{-7} \text{ s)}$$

$$t_2=9160000 \text{ (} 10^{-7} \text{ s)} \approx 8630000 \text{ (} 10^{-7} \text{ s)}$$

FIBONACCI

	FIBONACCI 1		FIBONACCI 2		FIBONACCI 3		FIBONACCI 4	
n	t (10 ⁻⁷ s)	Límite	t (10 ⁻⁷ s)	Límite	t (10 ⁻⁷ s)	nTimes	t (10 ⁻⁷ s)	nTimes
10	0,0311	100000000	0,048	10000000	0,094	10000000	1,69	1000000
11	0,0372	100000000	0,058	10000000	0,092	10000000	2,68	1000000
12	0,041	100000000	0,059	10000000	0,103	10000000	4,35	1000000
13	0,0458	100000000	0,065	10000000	0,103	10000000	7,09	1000000
14	0,0505	100000000	0,063	10000000	0,116	10000000	11,35	1000000
15	0,0537	100000000	0,068	10000000	0,118	10000000	18,36	1000000
16	0,058	100000000	0,073	10000000	0,133	10000000	29,64	1000000
17	0,0427	100000000	0,079	10000000	0,132	10000000	47,89	1000000
18	0,0428	100000000	0,077	10000000	0,151	10000000	77,52	1000000
19	0,0478	100000000	0,083	10000000	0,147	10000000	125,64	1000000
20	0,0523	100000000	0,089	10000000	0,167	10000000	203,08	1000000
21	0,055	100000000	0,095	10000000	0,175	10000000	328,68	1000000
22	0,0602	100000000	0,099	10000000	0,2	10000000	550	1000
23	0,0649	100000000	0,104	10000000	0,212	10000000	880	1000
24	0,0687	100000000	0,11	10000000	0,219	10000000	1410	1000
25	0,0538	100000000	0,116	10000000	0,222	10000000	2280	1000
26	0,0538	100000000	0,115	10000000	0,235	10000000	3720	1000
27	0,0576	100000000	0,123	10000000	0,234	10000000	5920	1000
28	0,0629	100000000	0,127	10000000	0,251	10000000	9660	1000
29	0,0667	100000000	0,137	10000000	0,251	10000000	15510	1000
30	0,0713	100000000	0,137	10000000	0,267	10000000	24970	1000
31	0,0745	100000000	0,142	10000000	0,265	10000000	40230	1000
32	0,0806	100000000	0,151	10000000	0,286	10000000	65160	1000
33	0,0655	100000000	0,157	10000000	0,298	10000000	105530	1000
34	0,0647	100000000	0,157	10000000	0,308	10000000	169470	1000
35	0,0692	100000000	0,166	10000000	0,309	10000000	273760	1000
36	0,0758	100000000	0,171	10000000	0,322	10000000	445830	1000
37	0,0786	100000000	0,178	10000000	0,329	10000000	740850	1000
38	0,0833	100000000	0,18	10000000	0,338	10000000	1240000	1
39	0,0877	100000000	0,186	10000000	0,342	10000000	1950000	1
40	0,0924	100000000	0,195	10000000	0,355	10000000	3110000	1
41	0,0828	100000000	0,201	10000000	0,356	10000000	5100000	1
42	0,0796	100000000	0,202	10000000	0,374	10000000	8100000	1
43	0,0827	100000000	0,21	10000000	0,382	10000000	13090000	1
44	0,086	100000000	0,218	10000000	0,395	10000000	21150000	1
45	0,0907	100000000	0,228	10000000	0,398	10000000	34220000	1
46	0,0953	100000000	0,228	10000000	0,414	10000000	55190000	1
47	0,1	100000000	0,235	10000000	0,423	10000000	89720000	1
48	0,1039	100000000	0,248	10000000	0,436	10000000	145100000	1
49	0,0965	100000000	0,253	10000000	0,431	10000000	234530000	1

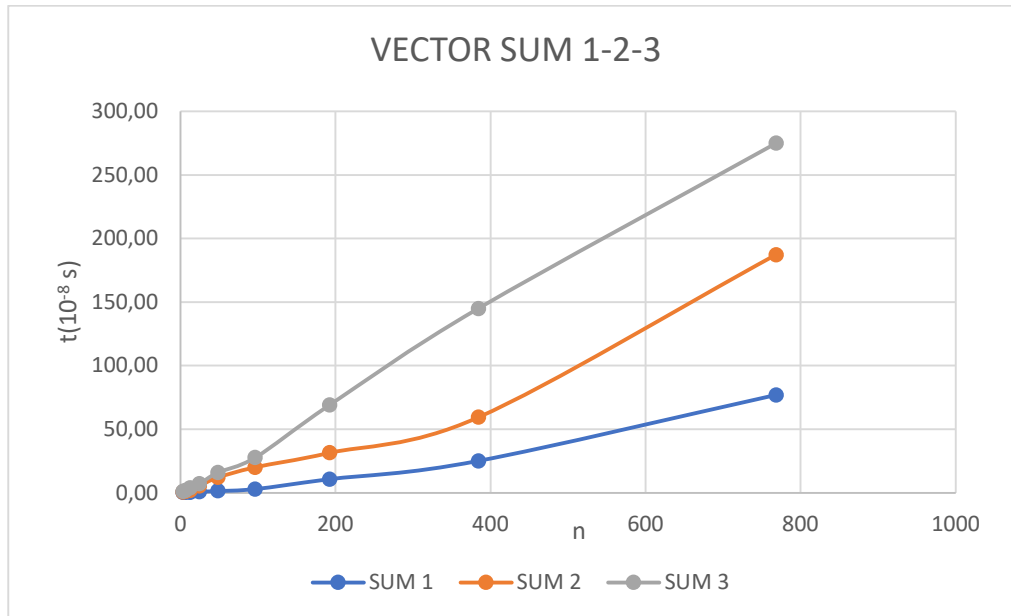
50	0,0977	100000000	0,255	10000000	0,446	10000000	382380000	1
51	0,0999	100000000	0,265	10000000	0,447	10000000	616440000	1
52	0,104	100000000	0,271	10000000	0,473	10000000	997360000	1
53	0,1079	100000000	0,282	10000000	0,485	10000000	1608070000	1
54	0,1121	100000000	0,287	10000000	0,485	10000000	> 25 min	
55	0,1167	100000000	0,295	10000000	0,485	10000000		
56	0,1215	100000000	0,306	10000000	0,502	10000000		
57	0,1149	100000000	0,317	10000000	0,512	10000000		
58	0,1143	100000000	0,317	10000000	0,583	10000000		
59	0,1188	100000000	0,329	10000000	0,515	10000000		



VECTOR SUM

n	sum 1		sum 2		sum 3	
	t(10 ⁻⁸ s)	Límite	t(10 ⁻⁸ s)	Límite	t(10 ⁻⁸ s)	Límite
3	0,30	100000000	0,54	10000000	1,02	10000000
6	0,35	100000000	1,08	10000000	1,88	10000000
12	0,49	100000000	1,90	10000000	3,97	10000000
24	0,77	100000000	5,40	1000000	7,13	10000000
48	1,52	100000000	12,20	1000000	16,07	10000000
96	2,92	100000000	20,10	1000000	27,79	10000000
192	10,76	100000000	31,50	1000000	69,00	100000
384	25,09	100000000	59,50	1000000	145,00	100000
768	77,00	100000	187,30	1000000	275,00	100000
1536	161,00	100000	389,30	1000000	545,00	100000
3072	312,00	100000	1.014,40	1000000	991,00	100000
6144	572,00	100000	1.895,70	1000000	1.839,00	100000
12288	1.112,00	100000	3.880,10	1000000	3.926,00	100000

24576	2.298,00	100000	9.300,00	1000	9.792,00	100000
49152	4.448,00	100000	Stack overflow		21.528,00	100000
98304	8.899,00	100000			30.400,00	1000



La complejidad del método Sum 1 es $O(n)$, ya que es un método iterativo con un bucle for. La complejidad del método Sum 2 es $O(n)$, ya que es un esquema por sustracción, con $a = 1$, $b = 1$ y $k = 0$. En el método Sum 2 se produce Stack Overflow debido a que cada vez que se llama al método recursivo solamente resta en uno el tamaño del problema, por lo que para $n = 49152$ elementos, tendría que realizar 49152 llamadas. La complejidad del algoritmo Sum 3 es $O(n^{\log_2 2})$, ya que es un esquema por división, donde $a > b^k$, con $a=2$, $b=2$ y $k=0$. Demostraciones:

VectorSum 1:

$$n_1=1536 \quad n_2=3072 \quad t_1=161 (10^{-8} s)$$

$$t_2= 322(10^{-8} s) \approx 312(10^{-8} s)$$

VectorSum 2:

$$n_1=192 \quad n_2=384 \quad t_1=31,5 (10^{-8} s)$$

$$t_2= 63(10^{-8} s) \approx 59,5(10^{-8} s)$$

VectorSum 3:

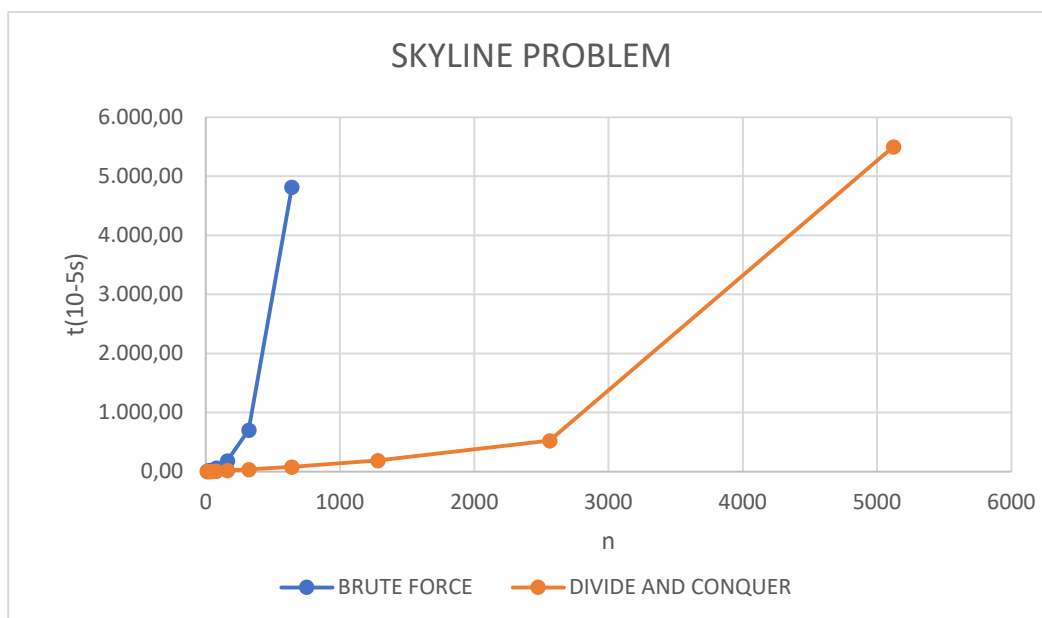
$$n_1=384 \quad n_2=768 \quad t_1=145 (10^{-8} s)$$

$$t_2= 290(10^{-8} s) \approx 275(10^{-8} s)$$

SKYLINE PROBLEM

n	BRUTE FORCE		DIVIDE AND CONQUER	
	t(10 ⁻⁵ s)	Límite	t(10 ⁻⁵ s)	Límite
10	6,80	1000	2,08	10000

20	13,40	1000	3,44	10000
40	22,90	1000	3,21	10000
80	63,20	1000	9,62	10000
160	180,70	1000	18,07	10000
320	701,10	1000	35,83	10000
640	4.819,00	1000	78,19	10000
1280	38.700,00	1	188,08	10000
2560	64.500,00	1	522,12	10000
5120	230.900,00	1	5.500,00	1
10240	902.500,00	1	14.700,00	1
20480	3.881.300,00	1	19.700,00	1
40960	38.206.300,00	1	13.400,00	1
81920	>25 min		25.600,00	1
163840			51.400,00	1



La complejidad del algoritmo, partiendo del código, Brute Force es $O(n^2)$, como se demuestra a continuación:

$$n_1=160 \quad n_2=320 \quad t_1=180,7(10^{-5} \text{ s})$$

$$t_2=320^2 * 160^2 * 180,7 = 722,8 (10^{-5} \text{ s}) \approx 701,10 (10^{-5} \text{ s})$$

La complejidad del algoritmo, partiendo del código, Divide and Conquer es $O(n * \log n)$ como se demuestra a continuación:

$$n_1=160 \quad n_2=320 \quad t_1=18,07(10^{-5} \text{ s})$$

$$t_2=320 * \log(320) / (160 * \log(160)) * 18,07 = 41,07 (10^{-5} \text{ s}) \approx 35,83 (10^{-5} \text{ s})$$

$$n_1=320 \quad n_2=640 \quad t_1=35,83(10^{-5} \text{ s})$$

$$t_2=640 * \log(640) / (320 * \log(320)) * 35,83 = 80,27 (10^{-5} \text{ s}) \approx 78,19 (10^{-5} \text{ s})$$