

Análisis de algoritmos de búsqueda

Búsqueda Secuencial										
No.	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10
71	4	2	3	3	3	3	3	2	2	2
51015	39	39	37	39	39	39	39	39	32	39
999993	998	909	904	754	875	797	894	733	583	999
1236194	997	931	790	735	1003	584	1005	770	606	735

Búsqueda Mejorada 1										
No.	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10
71	3	2	4	2	2	3	2	2	2	2
51015	162	68	54	135	57	57	87	85	69	70
999993	1513	1296	1326	1398	1323	1242	1379	1112	871	2175
1236194	1503	1163	1510	1109	1559	890	1511	1398	915	1205

Búsqueda Mejorada 2										
No.	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10
71	3	2	4	2	2	3	2	3	2	3
51015	63	37	29	55	31	91	59	56	46	53
999993	818	741	692	596	695	612	712	582	479	1017
1236194	833	827	1286	582	815	466	793	577	625	734

Búsqueda Binaria										
No.	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10
71	4	3	6	3	3	4	3	4	3	6
51015	4	3	5	4	3	5	4	7	3	5
999993	3	4	3	4	3	5	4	2	2	6
1236194	3	2	2	2	4	3	3	3	2	3

Gráficas de Búsquedas:

Número 71: <https://www.desmos.com/calculator/xmilyxnr2>

Número 51015: <https://www.desmos.com/calculator/gxvnqeublo>

Número 999993: <https://www.desmos.com/calculator/frpyg2u7fs>

Número 1236194: <https://www.desmos.com/calculator/d3divkkda2>

¿Qué algoritmo fue más rápido?

El algoritmo de búsqueda binaria

¿Qué tan grande es la diferencia entre el más rápido y el más lento?

Tomando en cuenta la notación de Big O, el más rápido es el de búsqueda binaria con $O(\log n)$, mientras que el más tardado es el de búsqueda mejorada 1, donde tiene una notación de $O(n)$, pero cuando comparamos el tiempo en el que se tardaron en encontrar un número, o darse cuenta de que no está en el arreglo, la búsqueda binaria gana por aproximadamente 1200 milisegundos

¿Qué tan grande es la diferencia entre los dos más rápidos?

Siendo búsqueda binaria y búsqueda mejorada 2 los más rápidos, tienen una diferencia de alrededor de 850 milisegundos, con esto es evidente la increíble velocidad de la búsqueda binaria, incluso en los peores escenarios

¿Por qué no se usa siempre búsqueda secuencial?

Porque, por más de que es un método de búsqueda bastante seguro, no es el más óptimo, ya que con una complejidad de $O(n)$, el tener una cantidad excesiva de datos, hace que este algoritmo de búsqueda no sea eficiente, en comparación de la búsqueda binaria.