

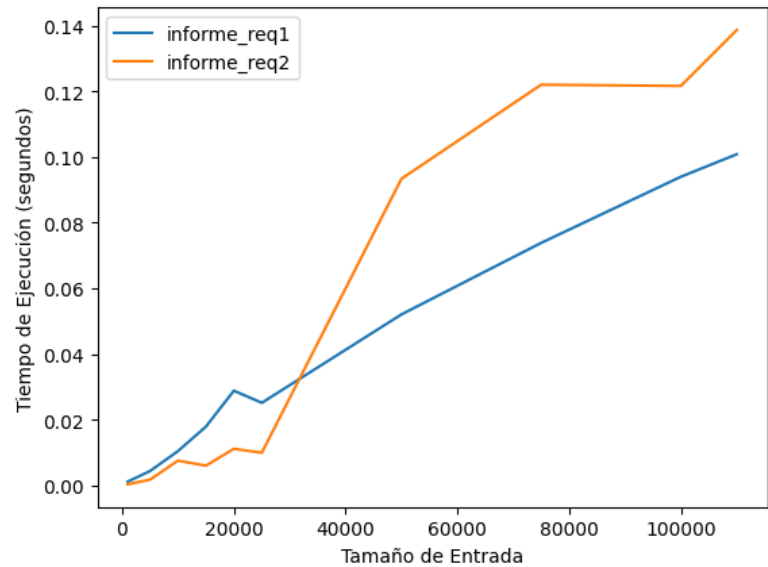
ANÁLISIS DEL RETO

Requerimiento <<n>>

Tabla de datos:

Tamaño muestra	Tiempo ms req1	Tiempo ms req2
1000	0.0011594295501708984,	[0.0003674030303955078,
5000	0.004405021667480469,	0.0017693042755126953,
10000	0.010476827621459961,	0.0075702667236328125,
15000	0.017925739288330078,	0.006027936935424805,
20000	0.028856754302978516,	0.01116943359375,
25000	0.025166034698486328,	0.009964942932128906,
50000	0.052100419998168945,	0.09341812133789062,
75000	0.07387375831604004,	0.12204384803771973,
100000	0.09401154518127441,	0.12167739868164062,
110000	0.10087347030639648	0.13869500160217285]

Graficos:



Análisis req1

Para req1, el tiempo de ejecución parece crecer de manera casi lineal con el tamaño de la muestra. Esto es evidente al observar que al duplicar, triplicar y cuadruplicar el tamaño de la muestra (por ejemplo, de 10000 a 20000, de 25000 a 50000, y de 50000 a 100000), el tiempo de ejecución

aumenta en una cantidad que parece proporcional. La complejidad computacional de req1, basada en esta observación, parece ser $O(n)$, donde n es el tamaño de la muestra.

Análisis req2

Para req2, el crecimiento del tiempo de ejecución con respecto al tamaño de la muestra es menos consistente en comparación con req1. En algunas instancias, el tiempo disminuye al aumentar el tamaño de la muestra (por ejemplo, de 15000 a 20000), lo cual podría deberse a variaciones en la carga del sistema, optimizaciones del compilador, o particularidades de cómo se ejecuta el algoritmo. Sin embargo, hacia las muestras más grandes (de 50000 en adelante), vemos un incremento más significativo y consistente, especialmente en el salto de 100000 a 110000. Esto sugiere que el algoritmo req2 podría tener una complejidad peor que lineal en algunas condiciones, pero las irregularidades en el patrón de tiempos sugieren que su comportamiento es más complejo de caracterizar con estos datos solos.

Comparación y análisis general

req1 muestra un patrón de crecimiento de tiempo que se alinea con una complejidad lineal ($O(n)$), donde el tiempo de ejecución aumenta de manera proporcional al tamaño de la entrada.

req2, por otro lado, muestra un patrón menos predecible. Aunque en general el tiempo aumenta con el tamaño de la muestra, las irregularidades sugieren que otros factores (como la eficiencia de operaciones específicas que realiza el algoritmo) pueden estar influyendo en el tiempo de ejecución. Es posible que req2 tenga una complejidad que varía dependiendo de la naturaleza de los datos de entrada y las operaciones específicas que realiza, aunque hacia el final parece tender a un comportamiento superlineal o posiblemente cuadrático en las muestras más grandes.