

## COMPARATIVA DE EFICIENCIA

Antonio Manuel De Pablos Pancorbo  
Laura Zafra Alarcos

En esta práctica hemos implementado dos formas distintas de recorrer las palabras de un diccionario almacenado como árbol con el objetivo de resolver una partida del juego de las letras.

- La primera de ellas es el iterador *iterator*. Este recorre el diccionario palabra a palabra, sin descartar ninguna de ellas y sin tener en cuenta las letras por la que está formada cada una.
- La segunda forma es el iterador *possible\_words\_iterator*. Este método recorre el diccionario palabra a palabra teniendo en cuenta el conjunto de letras que se encuentra disponible en cada partida, ignorando aquellas palabras que contengan letras que no se encuentren en el conjunto. De esta forma no visitamos todos los nodos del árbol y se logra un resultado igual de bueno que el anterior y más rápido.

A continuación mostramos una comparativa de la eficiencia de ambos iteradores. Para ello hemos ejecutado 250 veces el programa correspondiente a cada iterador midiendo el tiempo que tarda en segundos, usando las funciones disponibles en la librería chrono. Hemos tomado varias muestras modificando las variables que más influyen en el tiempo de ejecución, como la modalidad de juego (longitud o puntuación), el número de letras que se usan en cada partida (en nuestro caso, 4 ó 9 letras) y el tamaño del diccionario (grande, con 78399 palabras; mediano con 20 palabras y mini, con 8 palabras).

A partir de estos datos hemos elaborado 12 gráficas en las que modificamos estas variables. En ellas, podemos observar en el eje x el número de ejecuciones del programa y en el eje y el tiempo acumulado transcurrido en tras cada ejecución. En cada gráfica se encuentra en color azul el iterador menos eficiente, *iterator*, y en color naranja el iterador más eficiente, *possible\_words\_iterator*.



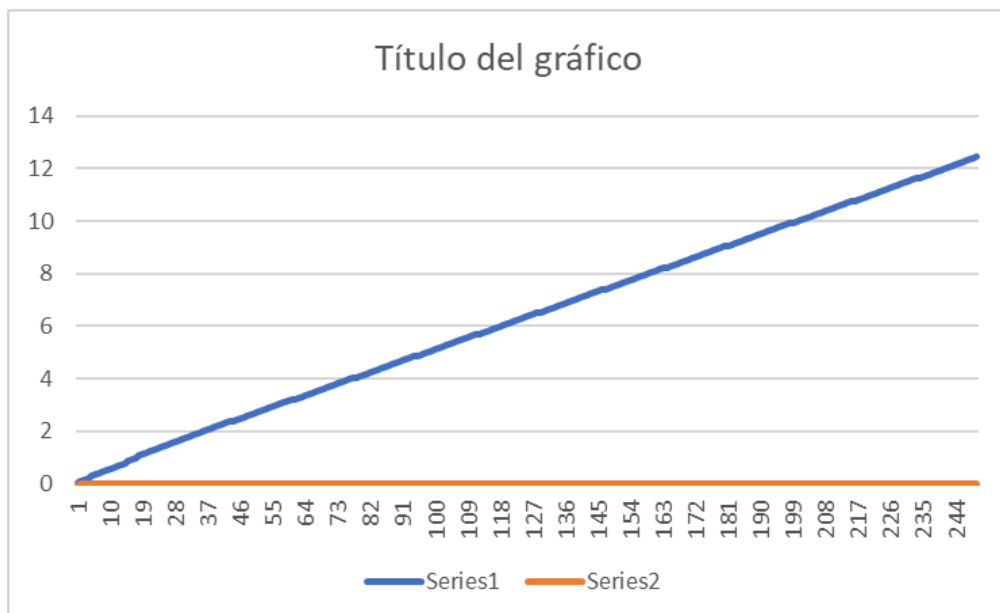
Modalidad de juego: longitud

Número de letras: 9

Tamaño del diccionario: grande

Tiempo medio de ejecución para **iterator**: 0.0559778

Tiempo medio de ejecución para **possible\_words\_iterator**: 0.000541679



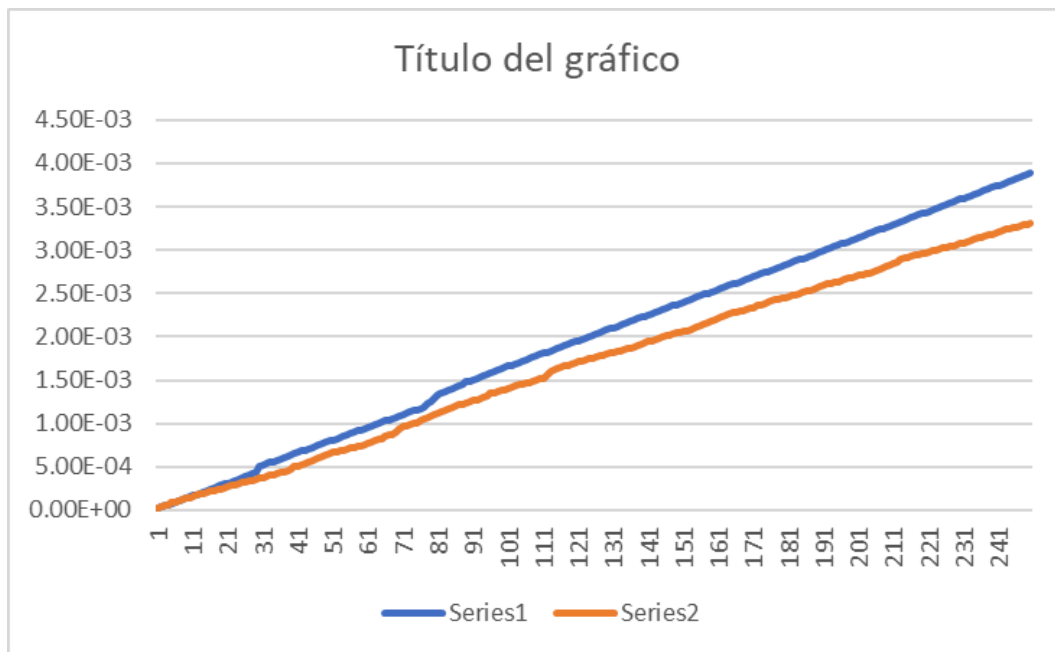
Modalidad de juego: longitud

Número de letras: 4

Tamaño del diccionario: grande

Tiempo medio de ejecución para **iterator**: 0.0497216

Tiempo medio de ejecución para **possible\_words\_iterator**: 3.063e-05



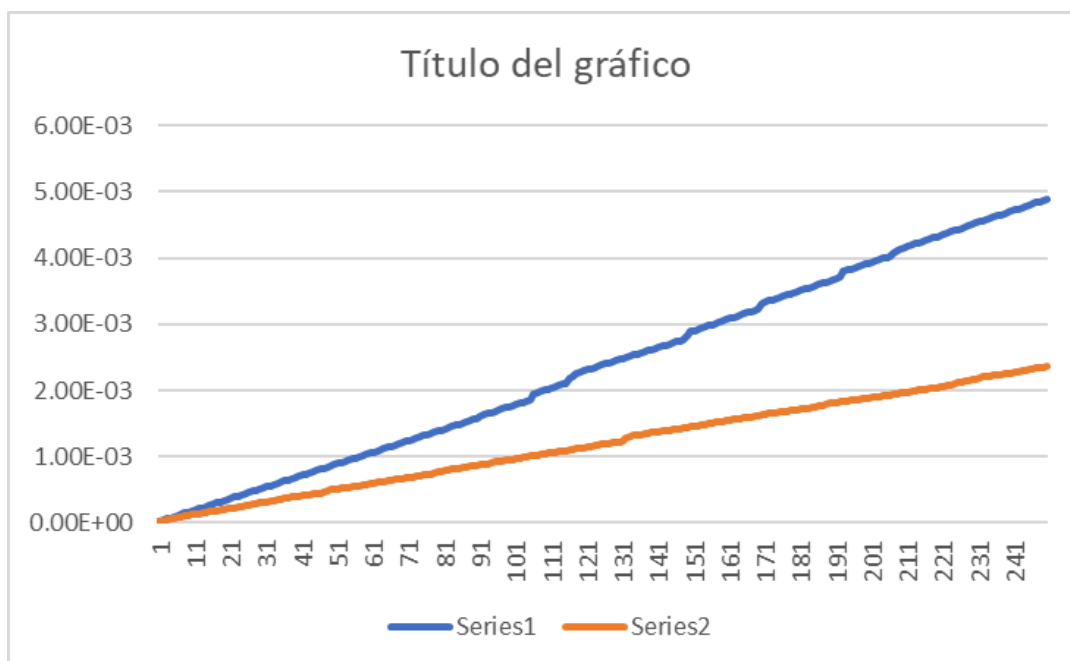
Modalidad de juego: longitud

Número de letras: 9

Tamaño del diccionario: mediano

Tiempo medio de ejecución para **iterator**: 1.55368e-05

Tiempo medio de ejecución para **possible\_words\_iterator**: 1.32276e-05



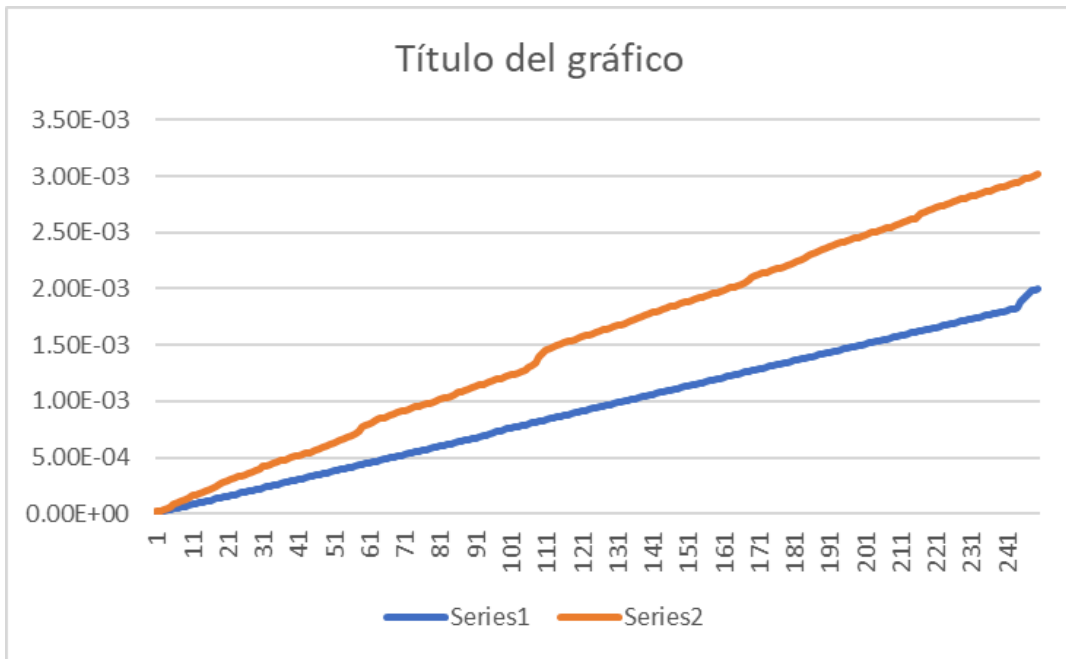
Modalidad de juego: longitud

Número de letras: 4

Tamaño del diccionario: mediano

Tiempo medio de ejecución para **iterator**: 1.95542e-05

Tiempo medio de ejecución para **possible\_words\_iterator**: 9.4424e-06



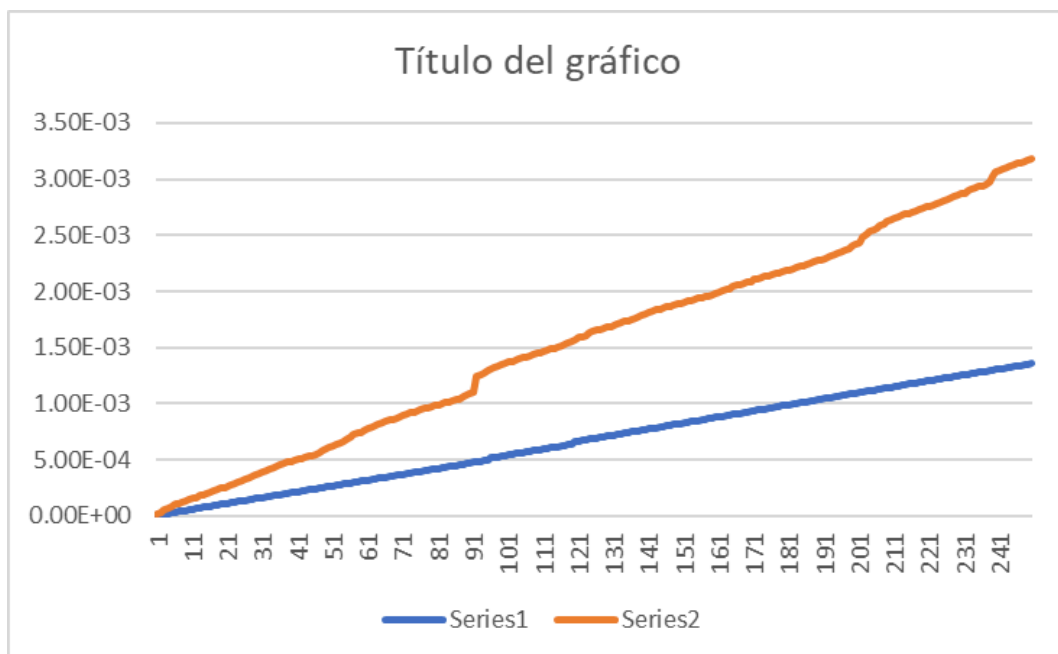
Modalidad de juego: longitud

Número de letras: 9

Tamaño del diccionario: mini

Tiempo medio de ejecución para [iterator](#): 1.20656e-05

Tiempo medio de ejecución para [possible\\_words\\_iterator](#): 7.99284e-06



Modalidad de juego: longitud

Número de letras: 4

Tamaño del diccionario: mini

Tiempo medio de ejecución para [iterator](#): 5.4242e-06

Tiempo medio de ejecución para [possible\\_words\\_iterator](#): 1.27096e-05



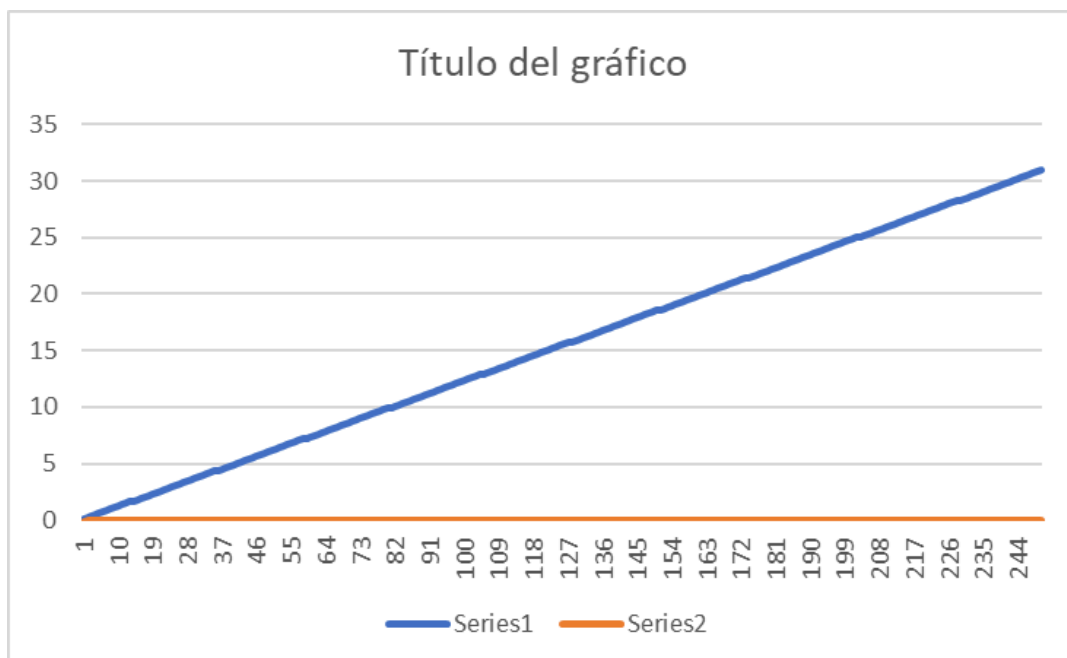
Modalidad de juego: puntuación

Número de letras: 9

Tamaño del diccionario: grande

Tiempo medio de ejecución para [iterator](#): 0.137715

Tiempo medio de ejecución para [possible\\_words\\_iterator](#): 0.000406832



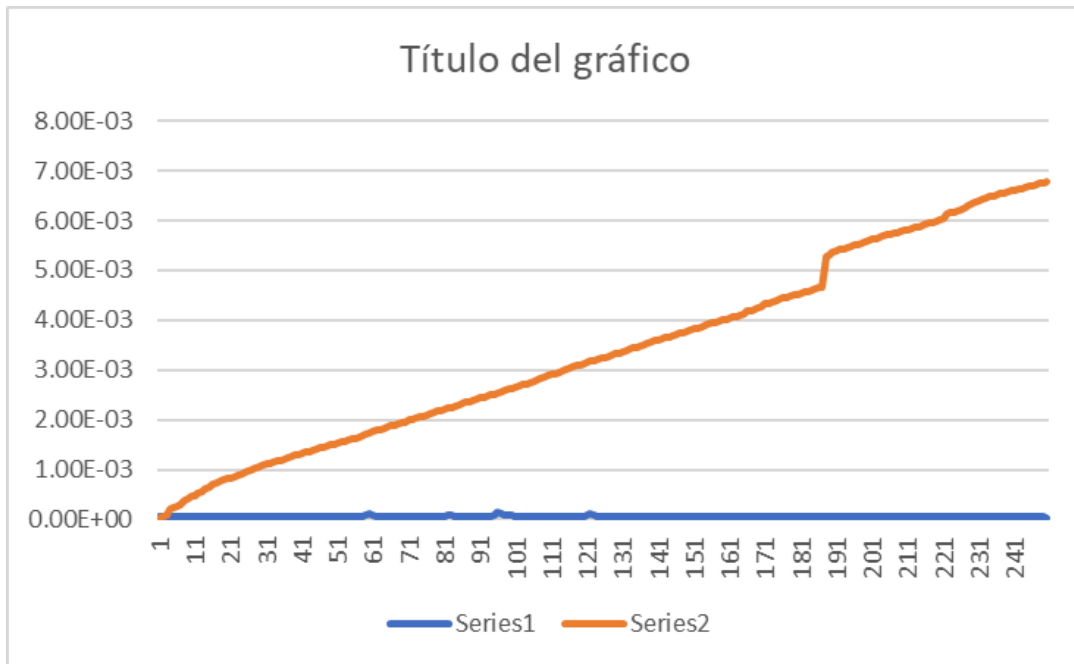
Modalidad de juego: puntuación

Número de letras: 4

Tamaño del diccionario: grande

Tiempo medio de ejecución para [iterator](#): 0.12377

Tiempo medio de ejecución para [possible\\_words\\_iterator](#): 3.04036e-05



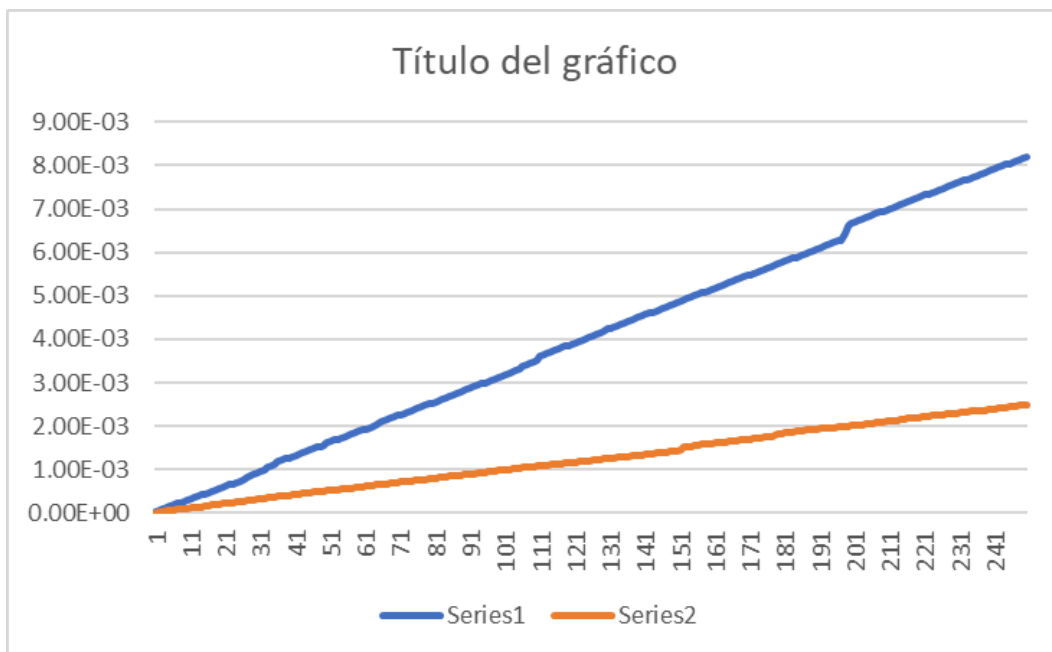
Modalidad de juego: puntuación

Número de letras: 9

Tamaño del diccionario: mediano

Tiempo medio de ejecución para [iterator](#): 2.88832e-05

Tiempo medio de ejecución para [possible\\_words\\_iterator](#): 2.71416e-05



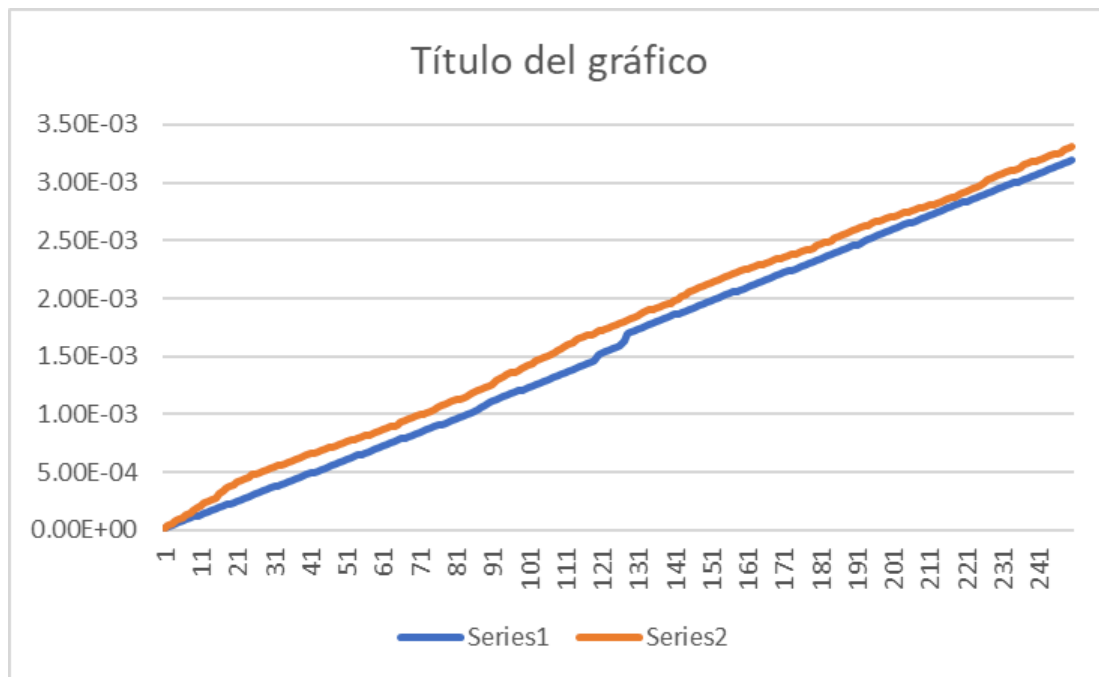
Modalidad de juego: puntuación

Número de letras: 4

Tamaño del diccionario: mediano

Tiempo medio de ejecución para [iterator](#): 9.96519e-06

Tiempo medio de ejecución para [possible\\_words\\_iterator](#): 1.94673e-05



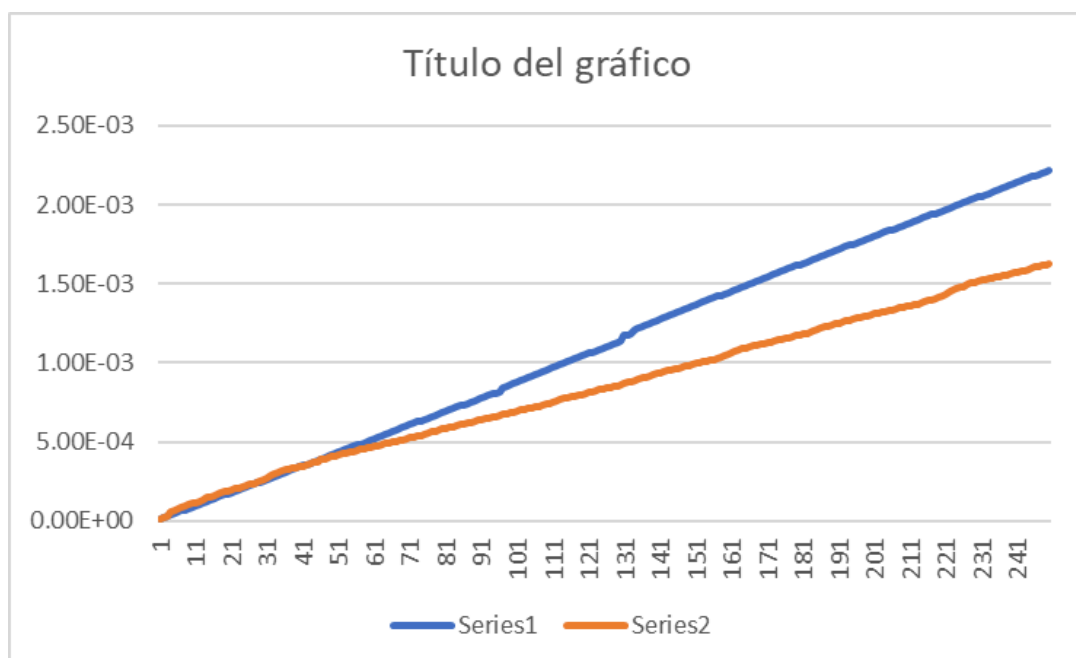
Modalidad de juego: puntuación

Número de letras: 9

Tamaño del diccionario: mini

Tiempo medio de ejecución para [iterator](#): 1.27756e-05

Tiempo medio de ejecución para [possible\\_words\\_iterator](#): 1.32216e-05



Modalidad de juego: puntuación

Número de letras: 4

Tamaño del diccionario: mini

Tiempo medio de ejecución para [iterator](#): 8.86831e-06

Tiempo medio de ejecución para [possible\\_words\\_iterator](#): 6.49881e-06

Vemos a continuación una comparativa de los tiempos medios de ejecución en segundos con las distintas variables, en la que podemos ver resaltados en verde los resultados menores para cada caso, es decir, en qué condiciones es más eficiente cada iterador:

<b>Variables usadas: tamaño / nº letras / modo</b>	<b>Tiempo medio iterator</b>	<b>Tiempo medio possible_words_iterator</b>
grande / 9 / longitud	0.0559778	0.000541679
grande / 4 / longitud	0.497216	3.063e-05
mediano / 9 / longitud	1.55368e-05	1.32276e-05
mediano / 4 / longitud	1.95542e-05	9.4424e-06
mini / 9 / longitud	1.20656e-05	7.99284e-06
mini / 4 / longitud	5.4242e-06	1.27096e-05
grande / 9 / puntuación	0.137715	0.000406832
grande / 4 / puntuación	0.12377	3.04036e-05
mediano / 9 / puntuación	2.88832e-05	2.71416e-05
mediano / 4 / puntuación	9.96519e-06	1.94673e-05
mini / 9 / puntuación	1.27756e-05	1.32216e-05
mini / 4 / puntuación	8.86831e-06	6.49881e-06

Es claro que, en la mayoría de casos, la velocidad del iterador possible\_words iterator es mayor que la del iterador iterator tan solo observando la tabla de tiempos medios. Veamos esto en más detalle.

Como se puede ver en las gráficas, se produce una mayor diferencia entre los tiempos de ejecución de ambos iteradores cuando hay más palabras sobre las que iterar, es decir, cuando el diccionario es más largo, siendo más rápido possible\_words\_iterator. En el caso de los diccionarios medianos esta diferencia es menos notable, decreciendo hasta algunas cienmilésimas de segundo, aunque sigue manteniendo su puesto líder en la eficiencia el segundo iterador. Respecto a los diccionarios pequeños, el liderazgo en la eficiencia se tambalea, puesto que la victoria entre iteradores está repartida en partes iguales.

Además, en la medida que aumenta el número de letras disponible para el juego lo hacen también los tiempos de ejecución. Por lo general, en igualdad de condiciones el tiempo requerido para ejecutar una partida con 9 letras es mayor que el necesario para una de 4. No se ha podido encontrar ninguna variable que explique este fenómeno. Por último, en cuanto al modo de juego, son más rápidas las ejecuciones en el modo longitud que en el modo puntuación, excepto en algunos casos que generalmente ocurren con diccionarios grandes.