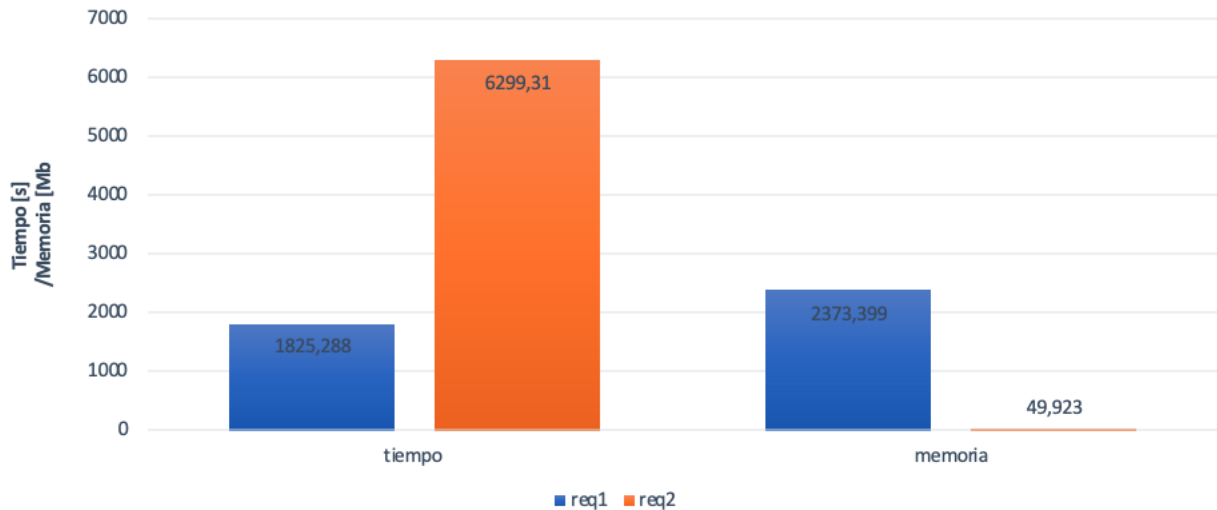


OBSERVACIONES RETO 4

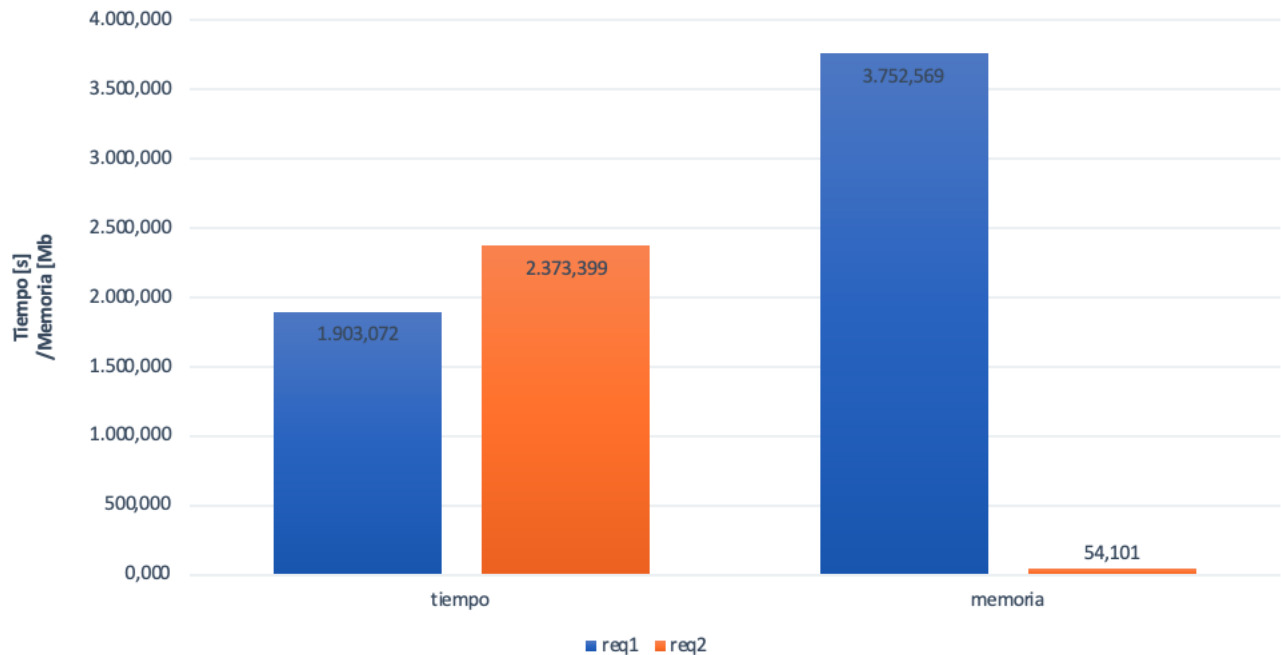
A continuación, se presenta la comparación entre el rendimiento del código implementado en el reto 4:

Lo primero que se hizo fue elegir 5 parejas de países al azar y se realizaron los dos requerimientos en cada uno:

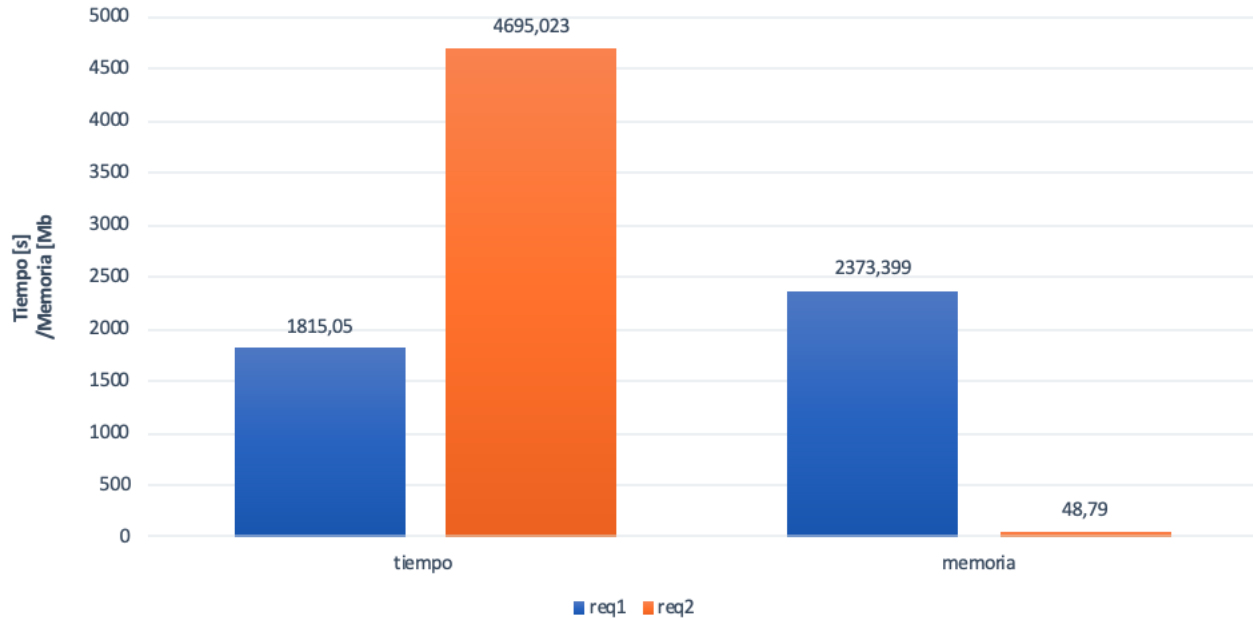
Prueba de rendimiendo usando Belgica y Alemania como
países base.



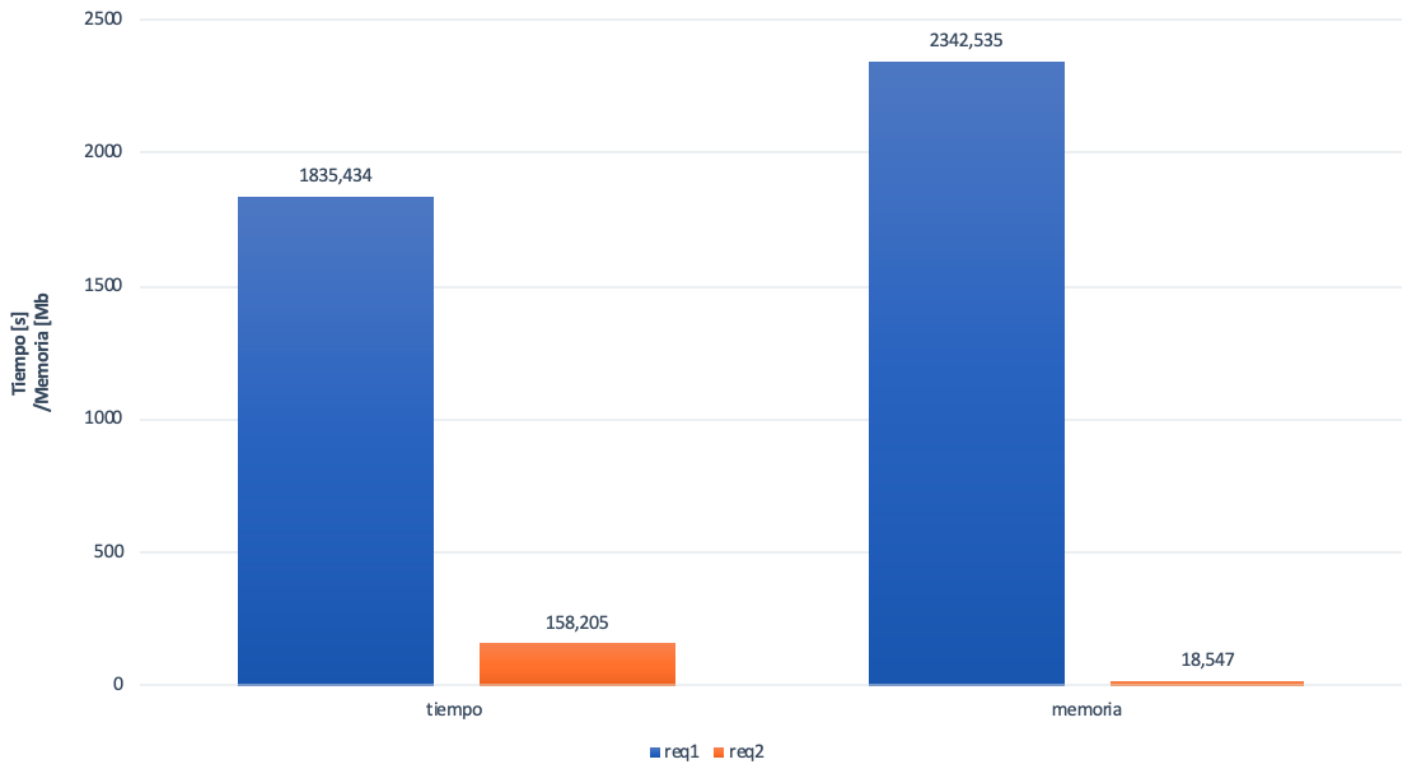
Prueba de rendimiendo usando Indonesia y Singapur como países base.

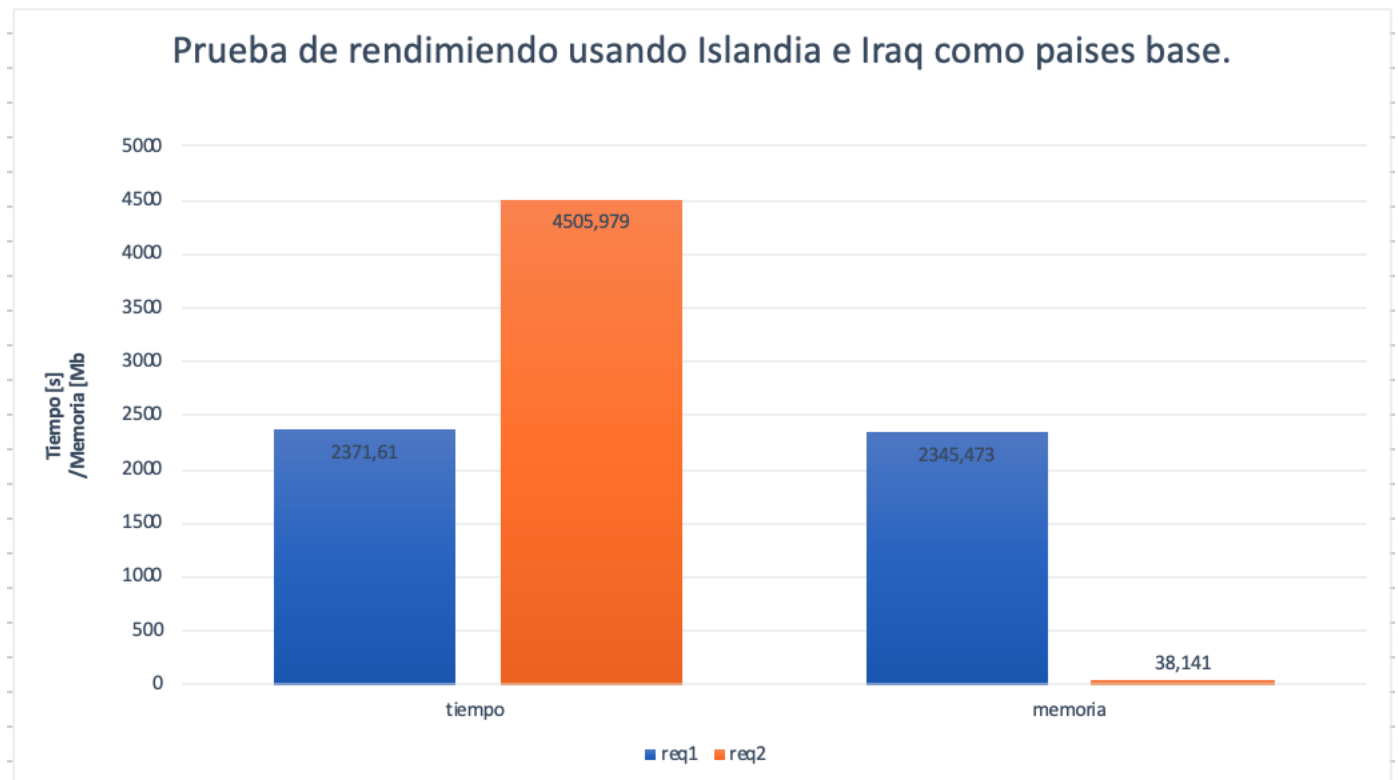


Prueba de rendimiento usando Colombia y Japon como paises base.

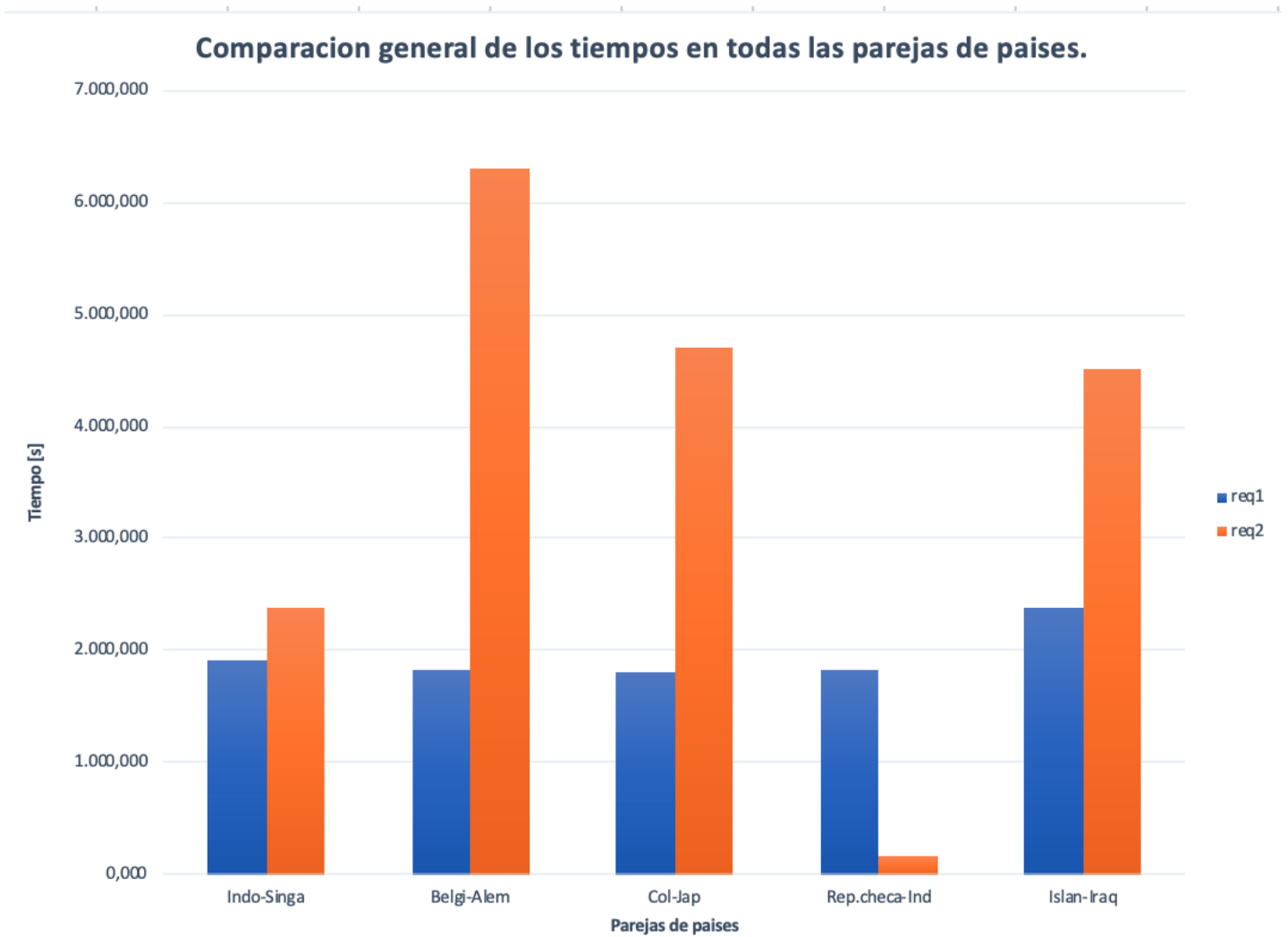


Prueba de rendimiento usando Republica checa e India como paises base.

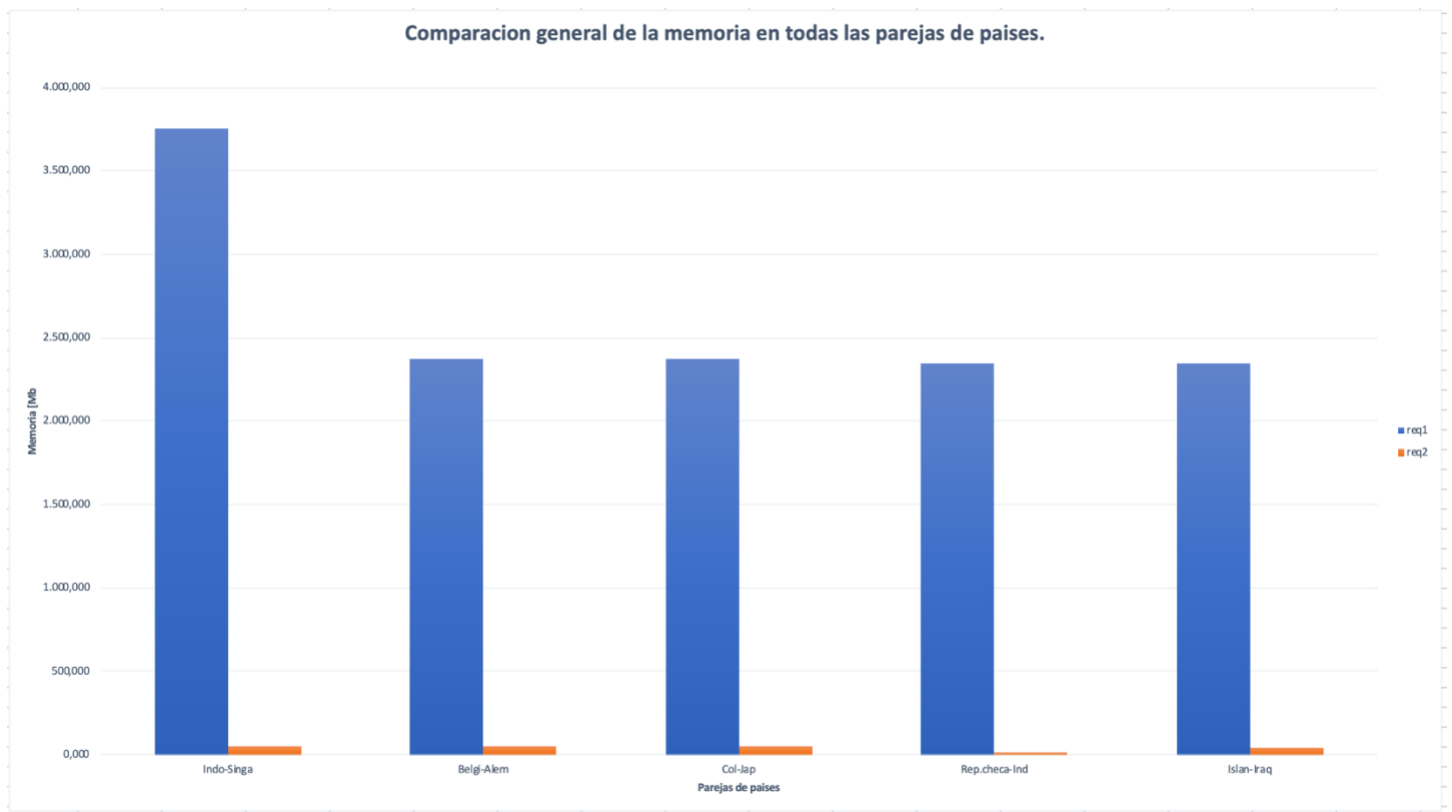




Teniendo la información recolectada por los 5 pares de países podemos hacer dos gráficas donde se comparen los tiempos y usos de memoria:



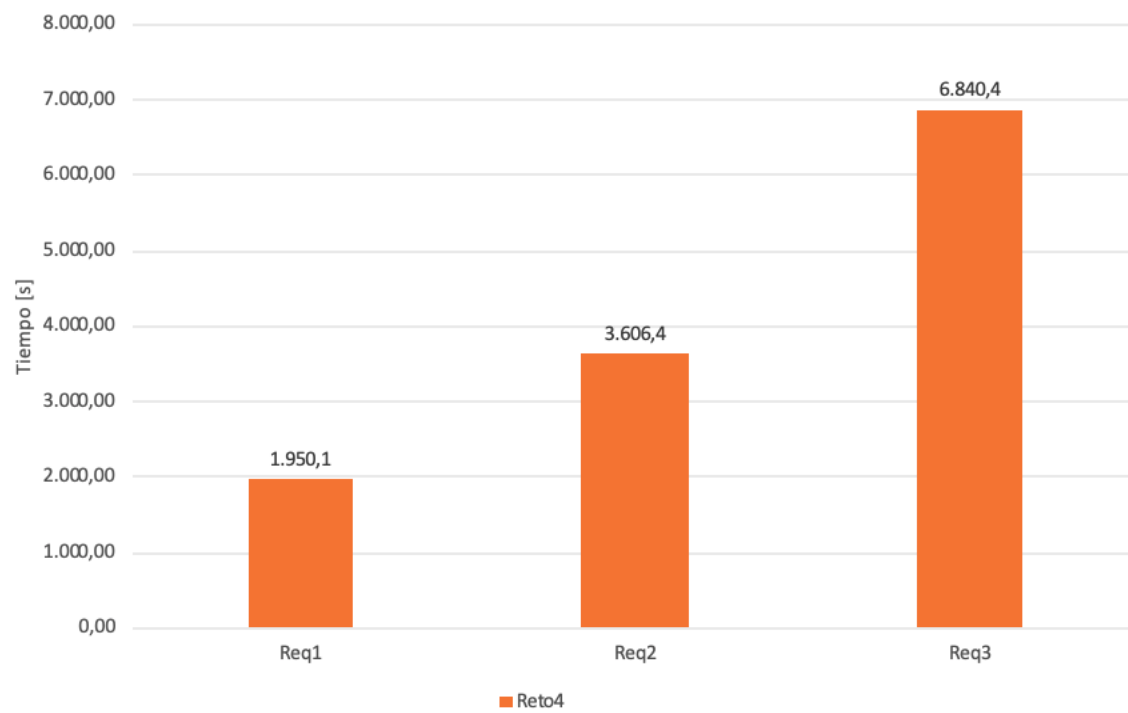
Es evidente que en la mayoria de casos es consistente como los tiempos del requerimiento 2 son los mas altos pero no los mas consistentes por otro lado en el requerimiento 1 los tiempos son los menores y los mas consistentes.



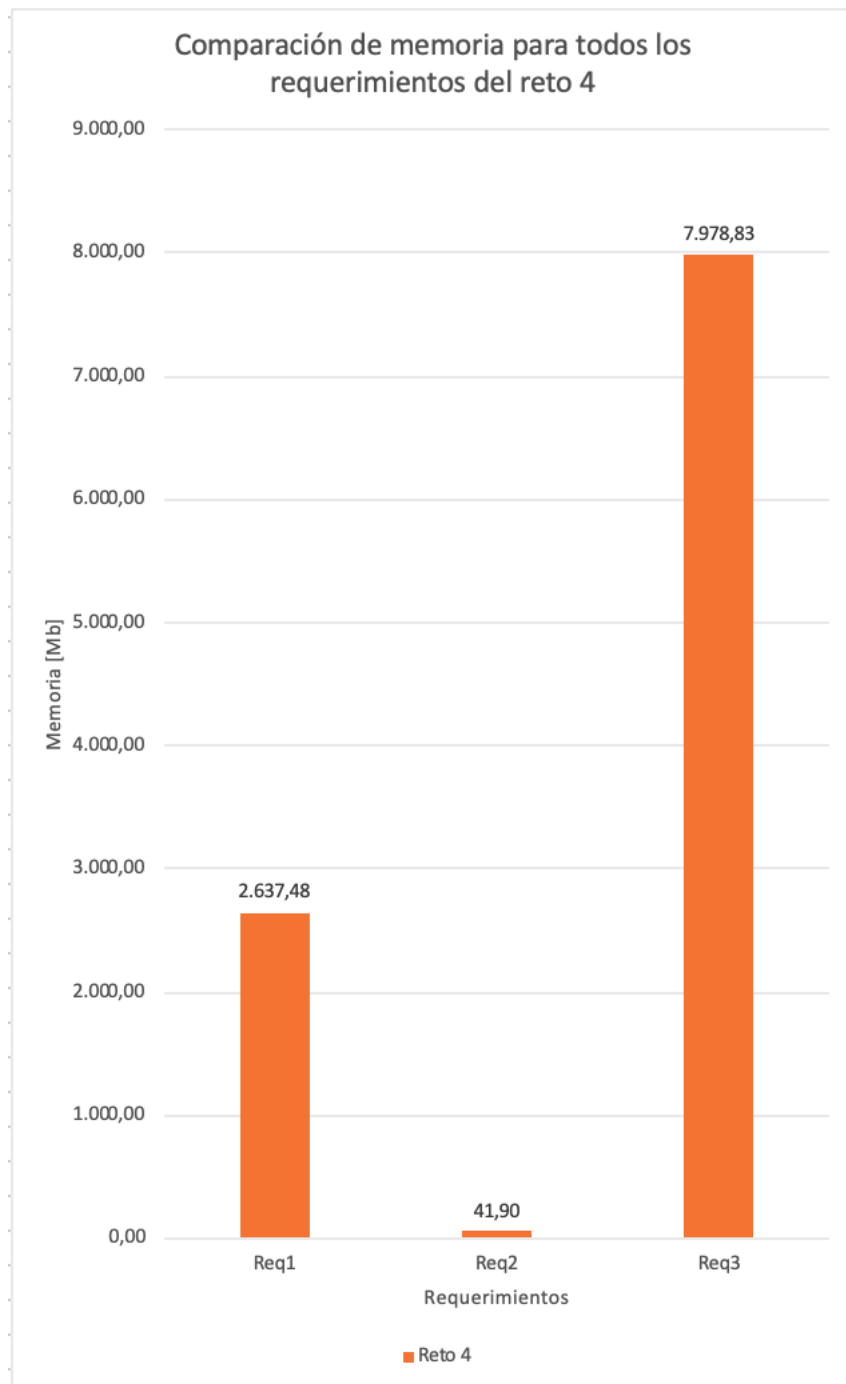
Es rotundamente notorio el alto uso de memoria que tiene el requerimiento uno frente al requerimiento 2.

De las dos graficas anteriores se obtuvieron los promedios se grafican junto al requerimiento 3 el cual se realizo 5 veces y tambien se promedio.

Comparación de tiempo para todos los requerimientos del reto 4



Como se puede observa en la grafica es notorio que los tiempos van aumentando de manera progresiva.



El req3 es el requerimiento que mas memoria consume seguido del req1.

Complejidad $O(n)$:

Requerimiento 1:

El requerimiento 1 de los componentes fuertemente conectados presenta dos procesos diferentes el primero es el uso del algoritmo de Kosaraju que tiene de complejidad $O(v+e)$ donde V son los vertices y E los arcos del grafo , y en el segundo proceso del requerimiento 1 es complejidad $O(1)$.

Requerimiento 2:

Para el segundo requerimiento el cual es el de la ruta mínima que se compone principalmente del algoritmo de Dijkstra que tiene como complejidad $O(\log(V))$ donde V son los vértices y E los arcos del grafo, además hay que hallar la ruta entre dos puntos y esa complejidad dependerá de la cantidad de vértices entre los dos puntos entonces si hay pocos puntos la complejidad sería $O(1)$ pero si hay demasiados vértices y se acerca a la cantidad total de vértices esto nos representará una complejidad $O(V)$.

Requerimiento 3:

El requerimiento 3 que consiste en la red de expansión mínima se usará el algoritmo de Prim de complejidad $O(\log(V))$ donde V son los vértices y E los arcos del grafo, ahora para encontrar la rama más larga en el peor de los casos tendríamos $O(V)$ y todo esto debido a que tendríamos que recorrer todos los vértices.