

# OBSERVACIONES RETO 4

Juan José Osorio (202017020)

Thais Tamaio Ramírez (202022213)

	Máquina 1	Máquina 2
Procesadores	3,1 GHz Intel Core i5 de dos núcleos	Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz 2.59 GHz
Memoria RAM (GB)	8 GB 2133 MHz LPDDR3	32 GB
Sistema Operativo	macOS Big Sur versión: 11.2.1	Windows 10 Home 19042.1023

Tabla 1. Especificaciones de las máquinas para ejecutar las pruebas de rendimiento.

Para realizar el análisis de los resultados en términos de memoria y tiempos de ejecución, decidimos medir el tiempo de ejecución y la memoria consumida para la carga de datos y para los cinco requerimientos del reto 4, en dos máquinas diferentes. En las siguientes tablas se muestran los resultados al ejecutar la carga de datos y los cinco requerimientos del reto 4:

## Resultados en la máquina 1

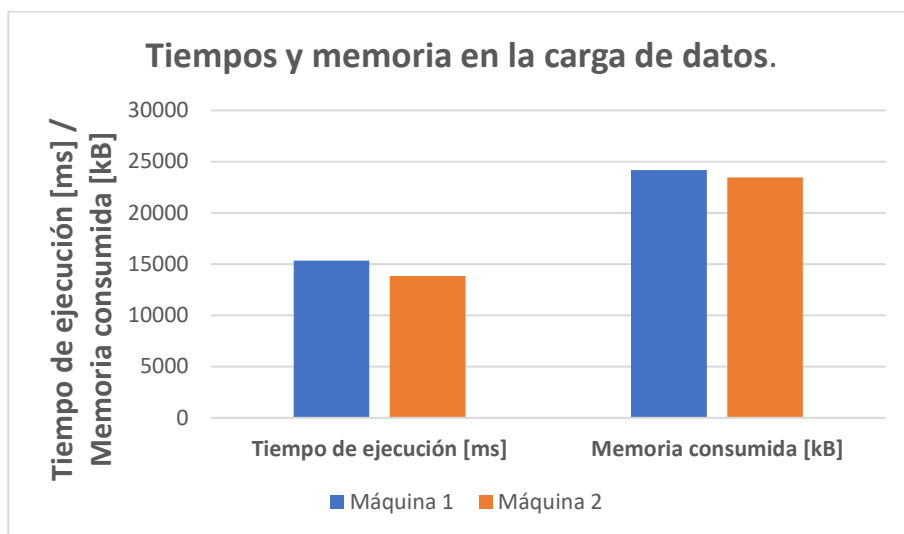
Máquina 1	Carga de datos	Req. 1	Req. 2	Req. 3	Req. 4	Req. 5
Tiempo de ejecución [ms]	15341,59	65,8	67,82	808,71	44,04	0,19
Memoria consumida [kB]	24189,09	20,14	7,68	2708,15	5,25	1,95

## Resultados en la máquina 2

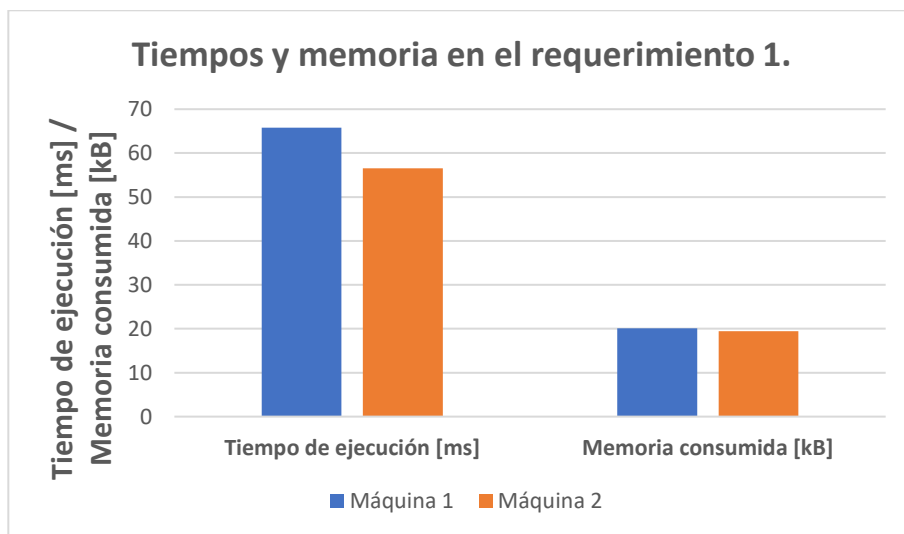
Máquina 2	Carga de datos	Req. 1	Req. 2	Req. 3	Req. 4	Req. 5
Tiempo de ejecución [ms]	13856,29	56,55	56,67	728,8	41,75	0,15
Memoria consumida [kB]	23464,07	19,48	8,78	2624,75	7,23	1,9

Por medio de las tablas que se mostraron anteriormente, se obtuvieron las siguientes gráficas para la máquina 1 y la máquina 2:

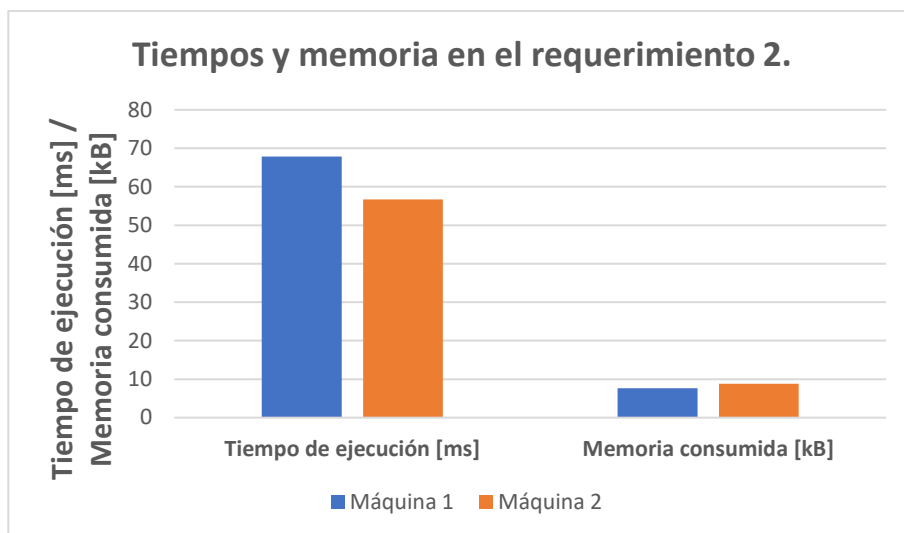
**Gráfica carga de datos:**



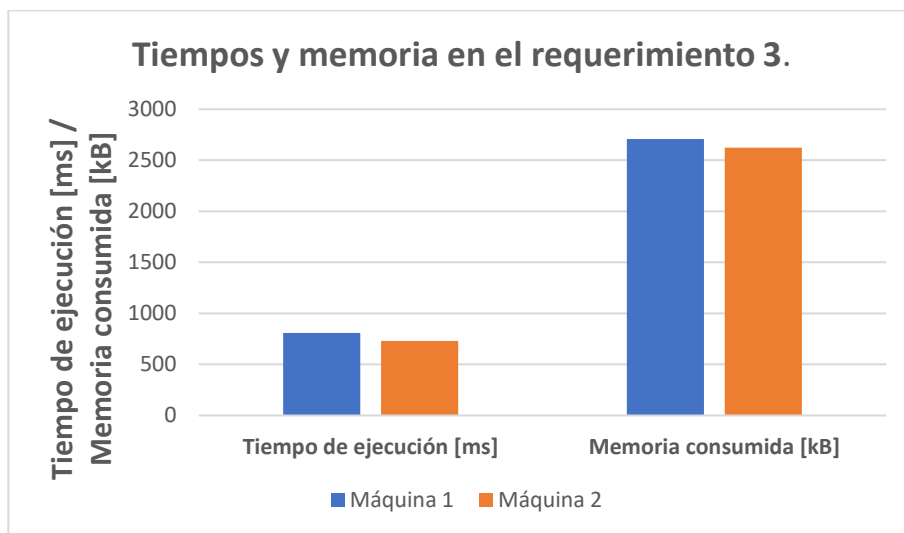
**Gráfica requerimiento 1:**



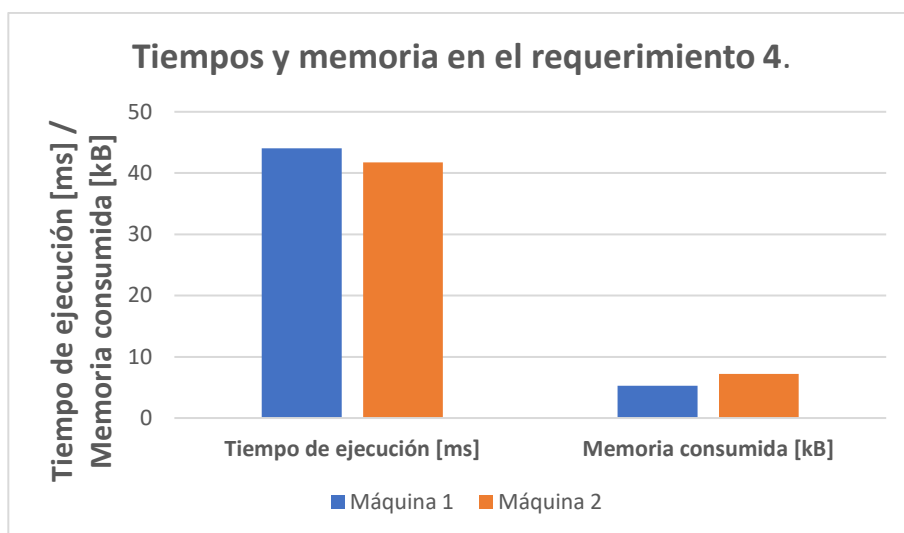
Gráfica requerimiento 2:



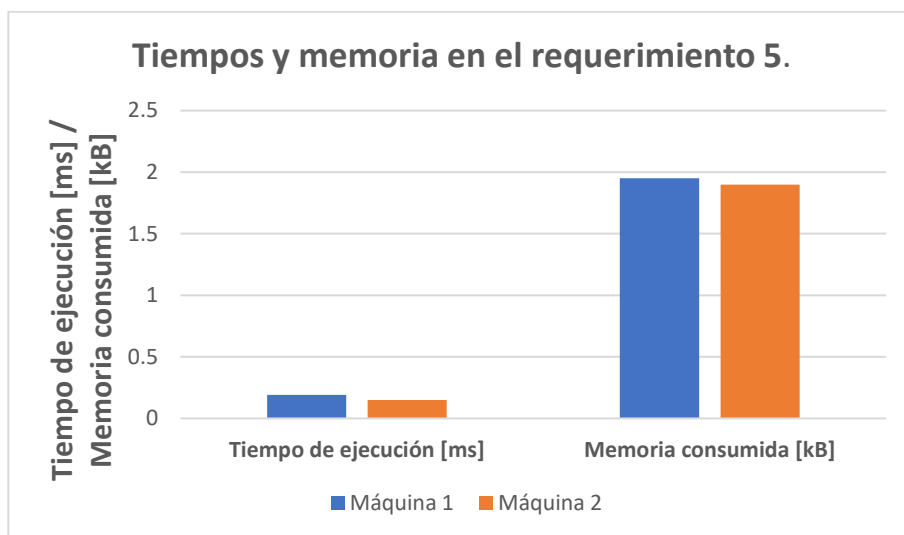
Gráfica requerimiento 3:



**Gráfica requerimiento 4:**



**Gráfica requerimiento 5:**



De manera general se puede determinar lo siguiente:

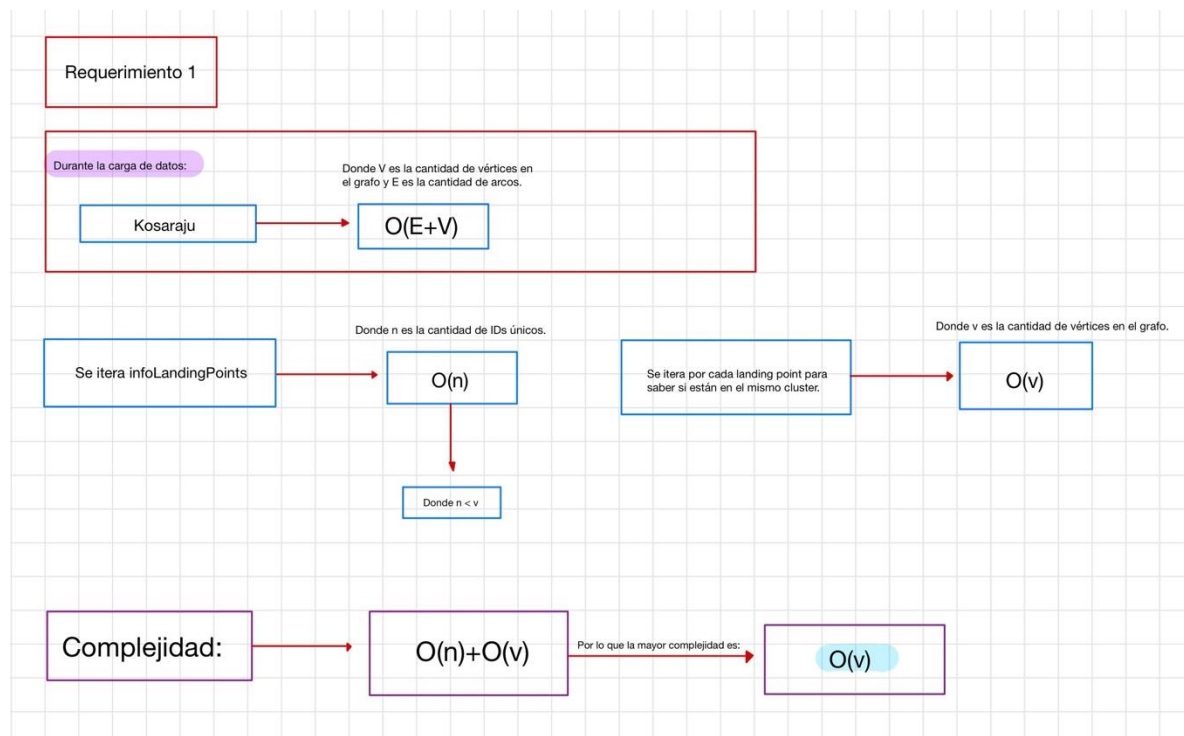
Al observar los resultados expuestos con anterioridad, se puede determinar que para ambas máquinas y para todos los requerimientos, junto con la carga de datos, se puede apreciar que existe una diferencia significativa entre los tiempos de ejecución y la memoria consumida. Esto se debe principalmente a que el obtener bajos tiempos de ejecución suele traer como consecuencia un incremento en la memoria consumida, y lo mismo en el sentido contrario.

Por otro lado, sobre la variedad de estructuras de datos en nuestro programa, el grafo está basado en una lista de adyacencias, por lo que solo se guarda información relevante del grafo, es decir, los

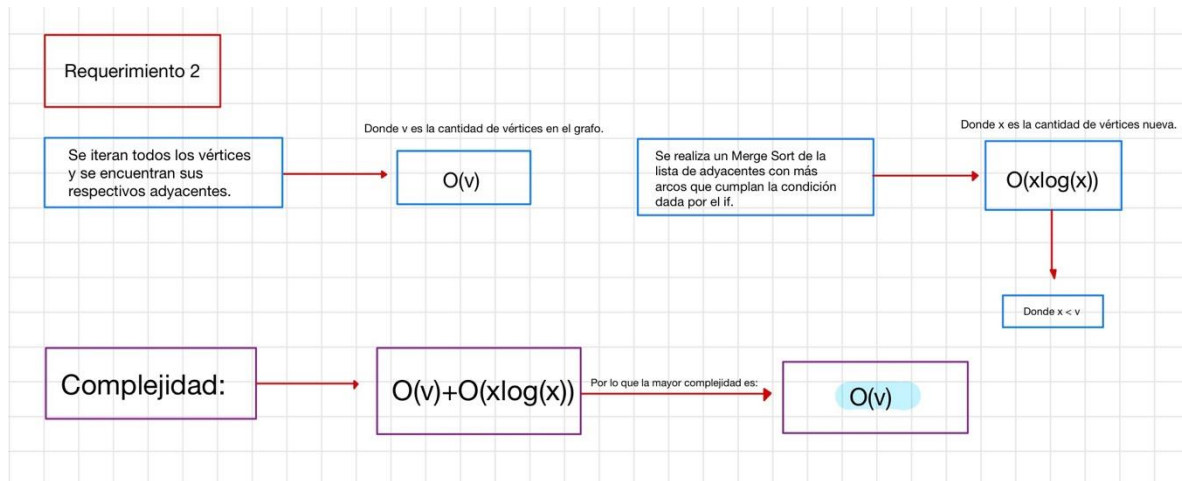
vértices y los acros presentes, lo que lleva a que el consumo de memoria sea menor y a que agregar un vértice se realice con tiempo constante. De igual manera, se escogieron únicamente tablas de hash de tipo PROBING ya que consumen menos memoria puesto que, a diferencia de las tablas de tipo CHAINING, estas no crean una lista encadenada cuando ocurre una colisión, sino que buscan un espacio disponible. Además, se eligió factor de carga 0.5 para mantener la complejidad lo más constante posible. Finalmente, cuando hablamos de listas utilizamos ARRAY's, ya que consumen menos memoria y además es más fácil llegar a un elemento puesto que éstos se guardan en posiciones contiguas en memoria. Para cada arreglo la función de comparación varía dependiendo del requerimiento.

A continuación se presenta el análisis de complejidad de cada uno de los requerimientos del reto 4:

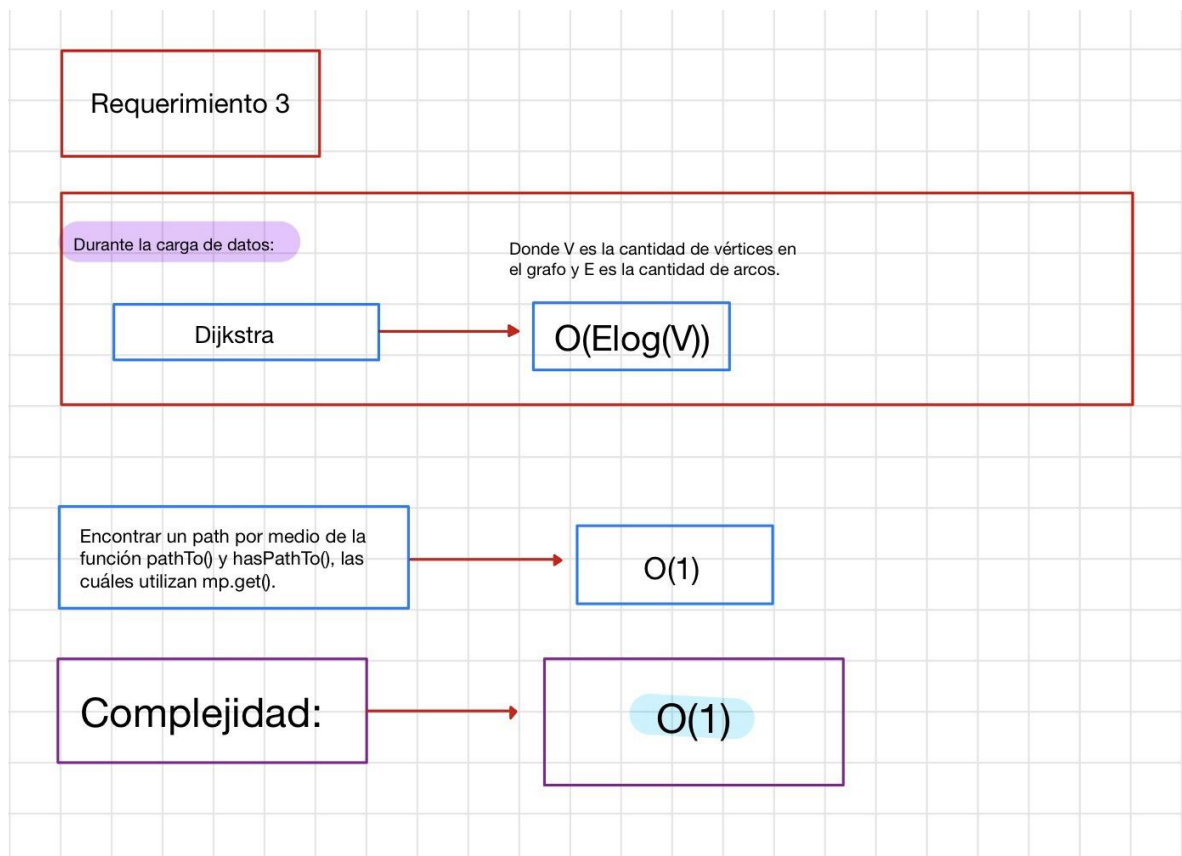
### 1. Requerimiento 1:



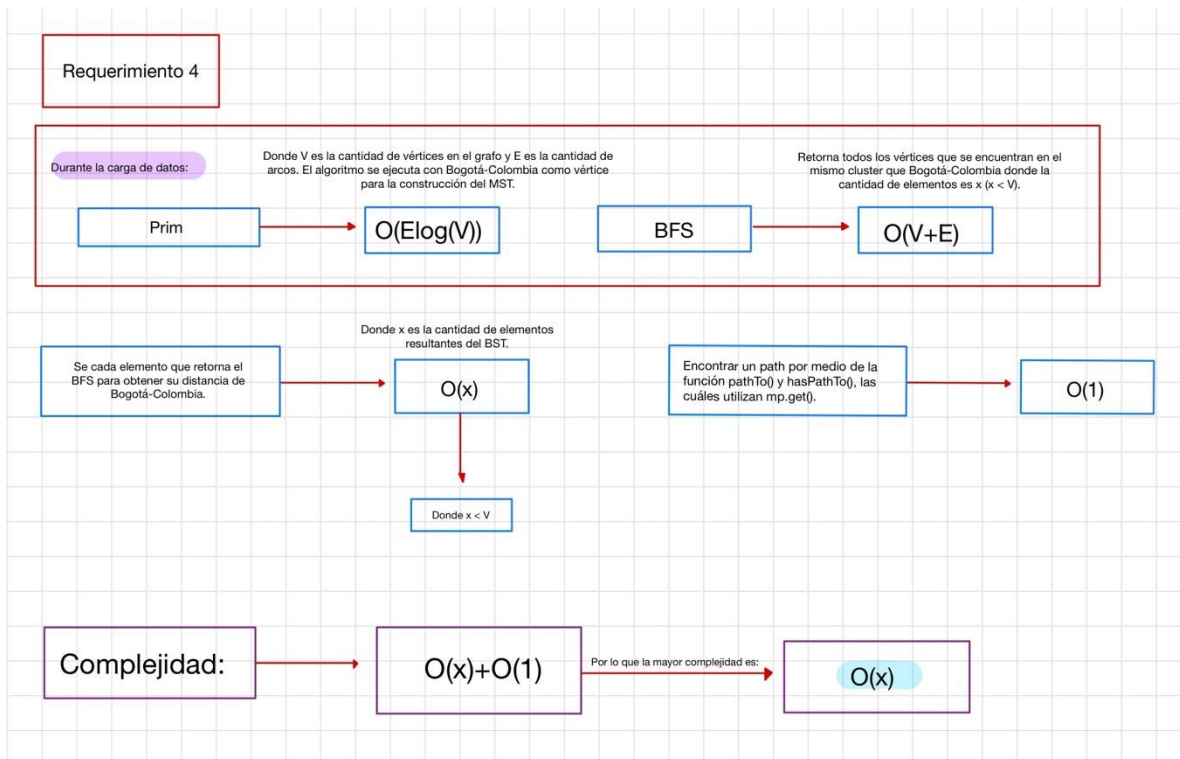
## 2. Requerimiento 2:



## 3. Requerimiento 3:



#### 4. Requerimiento 4:



#### 5. Requerimiento 5:

