



Tiempo de Corrida para cada una de las implementaciones:


### HashMap:

Name	Total Time	Total Time (CPU)	Invocations
 main	8.43 ms (100%)	15.6 ms (100%)	4

### TreeMap:

Name	Total Time	Total Time (CPU)	Invocations
 main	7.81 ms (100%)	15.6 ms (100%)	4

### LinkedHashMap:

Name	Total Time	Total Time (CPU)	Invocations
 main	11.8 ms (100%)	15.6 ms (100%)	4

Para las tres implementaciones solamente se creo el respectivo MAP y se mostraron todos los datos de la siguiente manera:

Categoría Mueble de terraza : Producto [Mesas de jardin, Sillas de jardin, Conjuntos mesas y sillas de jardin, Mesas de Ping Pong exteriores] Cantidad de productos 4  
Categoría Sillones de masaje : Producto [Cojines y colchonetas de masaje, Sillones relax y sofás de masajes, Sillones de masajes avanzados, Sofas camas] Cantidad de productos 4  
Categoría Bebidas : Producto [Cerveza tibetana Barley, Te frios, Coca cola 1 litro, Coca cola 2 litros] Cantidad de productos 4  
Categoría Condimentos : Producto [Siropo de regaliz, Especies Cajun del chef, Mezcla Gumbo del chef] Cantidad de productos 3  
Categoría Frutas : Producto [Pera secas, Pasas, Manzana roja, Manzana verde] Cantidad de productos 4  
Categoría Carnes : Producto [Res, Pollo, Cerdo, Camarones, Pescados] Cantidad de productos 5  
Categoría Lacteos : Producto [Queso de cabra, Queso Manchego , Leche descremada, Leche deslactosada, Leche entera] Cantidad de productos 5


Con esto se puede concluir que la manera más rápida es usando el TreeMap, seguida del HashMap, y la más lenta es el LinkedHashMap.

Complejidad de tiempo para la implementación HashMap:


Utilizando solo la categoría mueble:

Name	Total Time	Total Time (CPU)	Invocations
 main	1.32 ms (100%)	0.0 ms (-%)	3


Utilizando solo la categoría sillones, y la anterior:

Name	Total Time	Total Time (CPU)	Invocations
 main	1.35 ms (100%)	15.6 ms (100%)	3


Utilizando solo la categoría bebida, y las anteriores:

Name	Total Time	Total Time (CPU)	Invocations
 main	1.47 ms (100%)	0.0 ms (-%)	3

Utilizando solo las categorías condimentos y frutas, y las anteriores:

Name	Total Time	Total Time (CPU)	Invocations
 main	1.76 ms (100%)	0.0 ms (-%)	3

Utilizando todas las categorías:

Name	Total Time	Total Time (CPU)	Invocations
  <b>main</b>	 1.76 ms (100%)	0.0 ms (-%)	3

Como podemos observar el tiempo de ejecución no varía significativamente en relación a la carga de datos, y la pequeña variación se debe al tipo de implementación de mi diseño que conlleva un String y un ArrayList, este último es el responsable de aumentar el tiempo, por lo que se puede concluir que la complejidad de implementación del HashMap es de  $O(1)$ , ya tardará lo mismo sin importar la cantidad de datos.