Nombres: Vincenzo Angelone – Rafael Ávila – Eddy tapia

**HashMap Vs ArrayList**

|  |  |
| --- | --- |
| HashMap | ArrayList |
| * No mantiene el orden de inserción de los elementos. | * Siempre mantiene el orden de inserción de los elementos. |
| * HashMap almacena en pares (clave, valor) es decir dos objetos, consume más memoria que ArrayList. | * ArrayList almacena los elementos como valor (consumo mínimo de memoria.) |
| * No puede tener claves duplicadas, pero valores sí, es decir, que una clave no puede tener más de un valor. | * Puede tener elementos duplicados. |
| * Puede tener una sola clave nula, y varios valores nulos. | * Puede contener elementos nulos. |
| * Necesita utilizar una colección auxiliar distinta a map y así implementar Comparator.compairing() para poder ordenar los elementos. | * Puede implementar Comparator.compairing() para ordenar los elementos. |
| * Implementa interfaz de map | * Implementa interfaz de list |

Complejidad en ambas colecciones:

**HashMap:**

Métodos:

Put() y get():

Depende de muchas cosas, por lo general es O (1), es decir, lleva un tiempo constante, pero podría tener un hash que demore mucho en calcular, y si hay varios elementos en el mapa hash que devuelven el mismo código hash, get tendrá que iterar sobre ellos llamando equals a cada uno de ellos para encontrar una coincidencia.

Contains():

Internamente HashMap busca el código hash generado para cada elemento del mapa, para así verificar si está en el mapa.

**ArrayList:**

Métodos:

* Add(): O(1) significa que lleva un tiempo constante.
* Get(): O(1) Operación de tiempo constante.
* Remove():se ejecuta en tiempo lineal O (n). Se tiene que iterar toda la matriz para encontrar el elemento que se quiere eliminar.
* IndexOf():también se ejecuta en tiempo lineal. Recorre la matriz interna y comprueba cada elemento uno por uno. Por lo que la complejidad del tiempo para esta operación siempre requiere O (n) tiempo.
* Contains():la implementación se basa en indexOf (). Por lo que también se ejecutará en O (n) tiempo