**Taller 5 Estructuras de Datos.**

Juan Felipe Castaño.

Sergio Guillen.

**Análisis Comparativo.**

|  |  |  |
| --- | --- | --- |
|  | **Tabla de Hash LP** | **Tabla de Hash SC** |
| Numero de duplas (K, V) en la tabla (Valor N) | 428432 | 428432 |
| Tamaño Inicial del Arreglo de la tabla (Valor M inicial) | 4 | 4 |
| Tamaño Final del Arreglo de la tabla (Valor M final) | 1048576 | 131072 |
| Factor de Carga final (N/M) | 0.4085846 | 3.0 |
| Numero de rehashes que tuvo la tabla (Desde que se creó) | 18.0 | 15.0 |

|  |  |  |
| --- | --- | --- |
|  | **Tabla de Hash LP** | **Tabla de Hash SC** |
| Tiempo mínimo de get(…) | 21 milisegundos | 58 milisegundos |
| Tiempo Promedio de get(…) | 22 milisegundos | 64 milisegundos |
| Tiempo máximo de get(…) | 25 milisegundos | 69 milisegundos |

A partir de las graficas con los valores obtenidos a partir de las pruebas, podemos concluir que las tablas de Hash por Linear Probing pueden usar una menor cantidad de memoria que Separate Chaining. Ya que Separate Chaining crear varios sub-arreglos donde guarda las llaves y valores por aparte. Linear Probing por otro lado guarda todo en un mismo arreglo, haciendo que sea sustancialmente más rápido, como por ejemplo al realizar un get, esto es porque los datos o memoria están mas cerca uno del otro teniendo un acceso a la información en cualquier posición de nuestro arreglo.