

Taller 6: Colas de Prioridad

1. Pruebas y Resultados.

Para obtener resultados más precisos, se realizó el procedimiento para las tablas con diferentes tamaños iniciales. Específicamente, se realizó el procedimiento inicializando la tabla con los tamaños: 1, 10, 100, 1000 y 10000. A continuación se muestran los resultados numéricos:

	Tabla de Hash Linear Probing	Tabla de Hash Separate Chaining
Número de duplas (K,V) en la tabla	4977	4977
Tamaño Inicial del Arreglo de la Tabla	1	1
Tamaño Final del Arreglo de la Tabla	10949	1361
Factor de Carga Final	0,45456	3,6568
Número de Rehashes que Tuvo la Tabla	12	9
Tiempo promedio de consultas get(..) (nanosegundos)	1759924	1678494

	Tabla de Hash Linear Probing	Tabla de Hash Separate Chaining
--	---------------------------------	------------------------------------

Número de duplas (K,V) en la tabla	4977	4977
Tamaño Inicial del Arreglo de la Tabla	10	10
Tamaño Final del Arreglo de la Tabla	12853	1597
Factor de Carga Final	0,38722	3,1164
Número de Rehashes que Tuvo la Tabla	10	7
Tiempo promedio de consultas get(..) (nanosegundos)	21760619	1602286

	Tabla de Hash Linear Probing	Tabla de Hash Separate Chaining
Número de duplas (K,V) en la tabla	4977	4977
Tamaño Inicial del Arreglo de la Tabla	100	100
Tamaño Final del Arreglo de la Tabla	6947	1733
Factor de Carga Final	0,7164	2,8718
Número de Rehashes que Tuvo la Tabla	6	4
Tiempo promedio de consultas get(..) (nanosegundos)	2487409	1635565

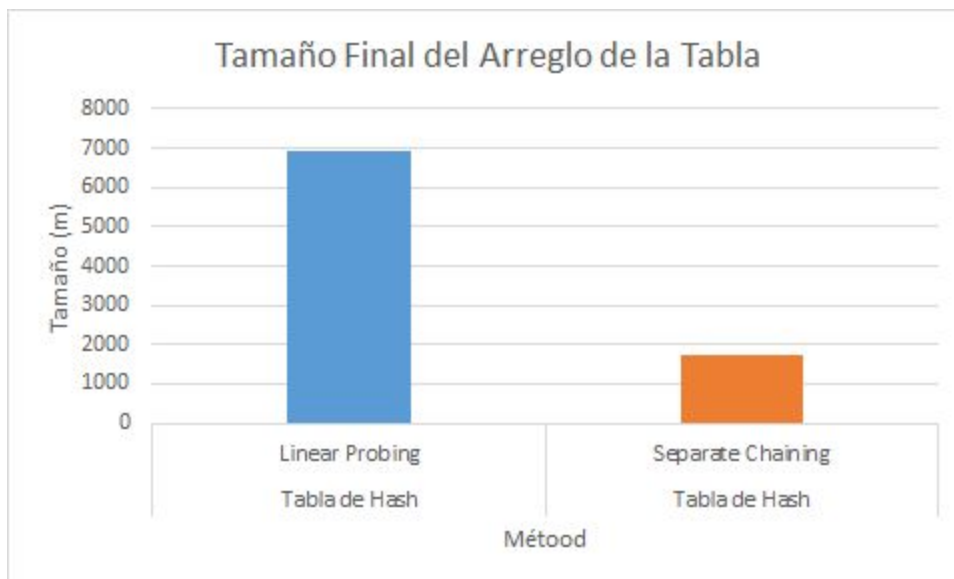
	Tabla de Hash Linear Probing	Tabla de Hash Separate Chaining
Número de duplas (K,V) en la tabla	4977	4977
Tamaño Inicial del Arreglo de la Tabla	1000	1000
Tamaño Final del Arreglo de la Tabla	8017	1000
Factor de Carga Final	0,6208	4,977
Número de Rehashes que Tuvo la Tabla	3	0
Tiempo promedio de consultas get(..) (nanosegundos)	1881935	22466328

	Tabla de Hash Linear Probing	Tabla de Hash Separate Chaining
Número de duplas (K,V) en la tabla	4977	4977
Tamaño Inicial del Arreglo de la Tabla	10000	10000
Tamaño Final del Arreglo de la Tabla	10000	10000
Factor de Carga Final	0,4977	0,4977
Número de Rehashes que Tuvo la Tabla	0	0
Tiempo promedio de consultas get(..) (nanosegundos)	1525542	1426576

Es importante mencionar que, arbitrariamente y para garantizar la uniformidad en el hash code, la tabla sólo utiliza tamaños de números primos. Es decir, al momento de

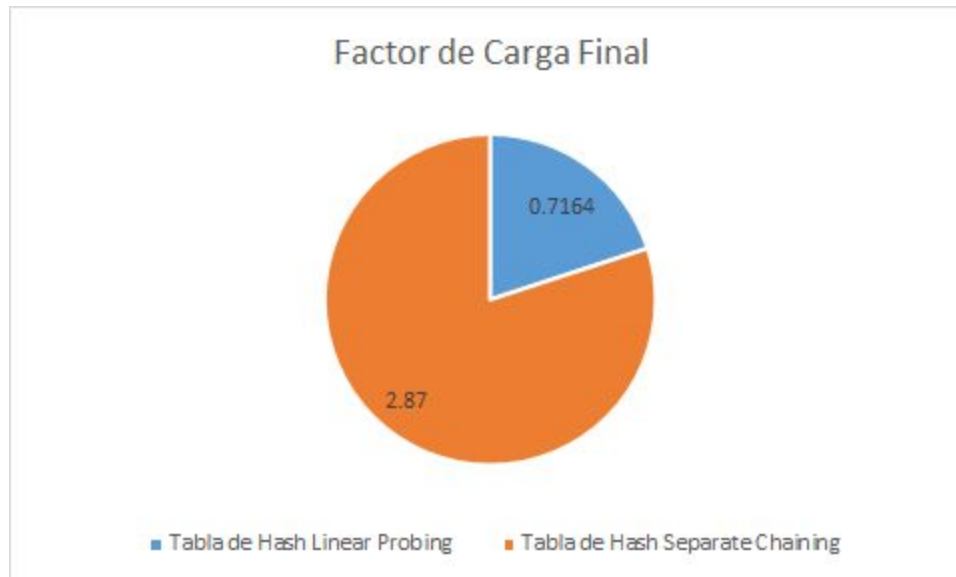
llenarse la tabla, el nuevo tamaño será el siguiente número primo con respecto al tamaño actual. Con propósito de análisis, a continuación se realiza una comparación gráfica cuando se tiene un tamaño inicial de 100:

- 1) Número de Duplas: En estas tablas, puesto que varias infracciones pueden tener el mismo ADDRESS_ID, se tiene que la llave es un entero y el valor que guarda es un arreglo dinámico con las infracciones que tienen ese ADDRESS_ID. Así, siempre se tendrán el mismo número de duplas (4977), que corresponde al número de ADDRESS_ID diferentes. Por esto, no vale la pena realizar un análisis gráfico.
- 2) Tamaño Final del Arreglo: A continuación se muestra gráficamente los resultados:



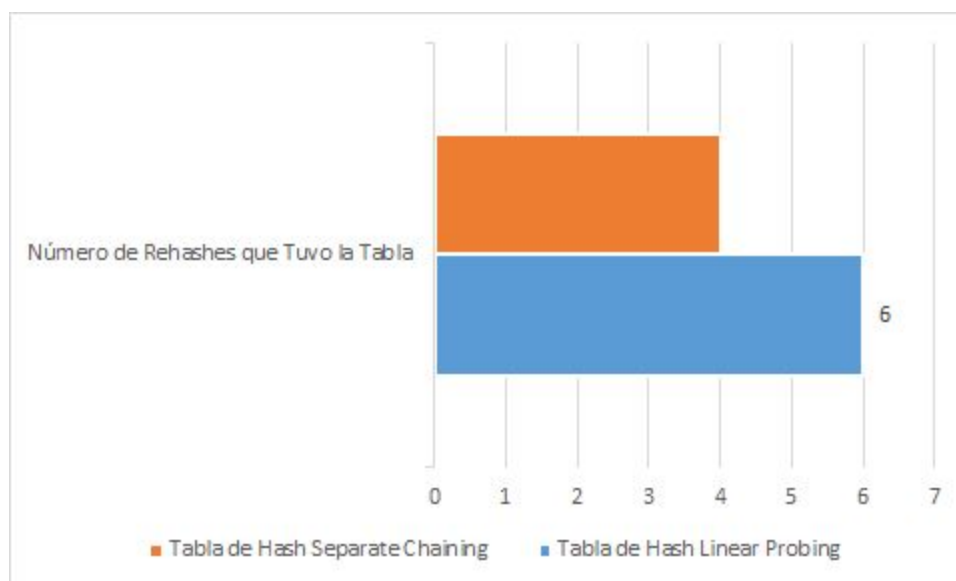
Se tiene que evidentemente el tamaño del método Linear Probing es mucho mayor. Esto se debe al factor de carga y a la implementación de cada uno de los métodos. Puesto que con Separate Chaining se tiene una lista encadenada en cada una de las posiciones del arreglo, se pueden almacenar muchas más llaves sin necesidad de hacer rehash. Además, para el método Separate Chaining se tiene un factor de carga de 5, mientras que para Linear Probing se tiene un factor de carga 0,75.

- 3) Factor de Carga Final: Gráficamente, obtuvimos el siguiente resultado:



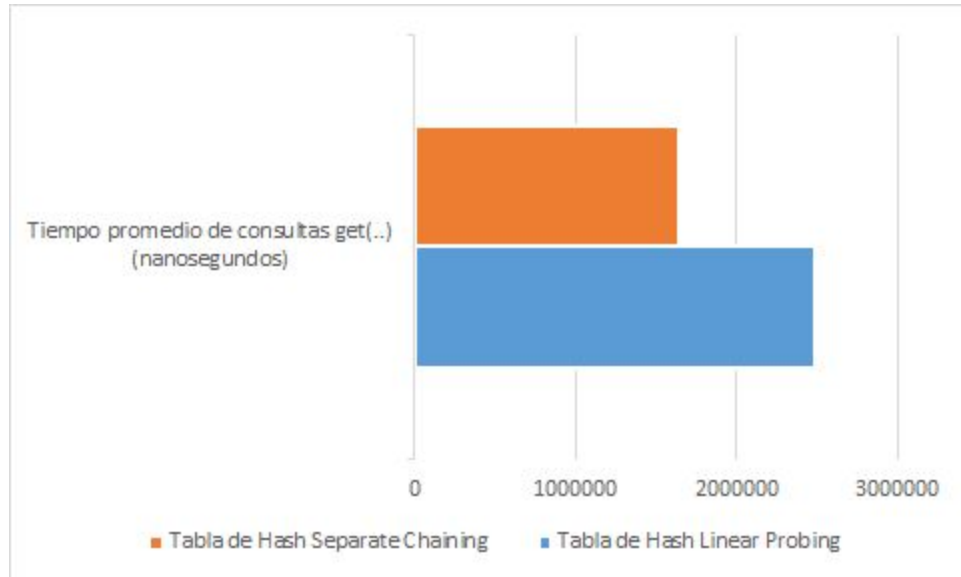
Donde se observa que el método Separate Chaining tiene un factor mucho mayor. Esto se debe, de nuevo, a las condiciones iniciales y a los factores de carga máximo seleccionados. Específicamente, el factor de carga del método Linear Probing siempre será menor que 1, mientras que el método de Separate Chaining, puede tomar valores mayores a 1.

4) Número de Rehashes: A continuación se muestra gráficamente los resultados:



Nótese que el método Linear Probing requiere de más rehashes que el método Separate Chaining. De nuevo, se debe a la estructura de los métodos y a los factores de carga seleccionados. Al tener un factor de carga menor, el método Linear Probing deberá hacer rehash a la tabla con mayor frecuencia.

- 5) Tiempo Promedio de Consultas get(): El resultado en nanosegundos se muestra a continuación:



Teniendo en cuenta que se tiene un tiempo en nanosegundos, se puede concluir que los tiempos de ambos métodos son similares. En el punto 2 se realiza en detalle un análisis de este resultado.

2. Comparación Implementaciones:

En primer lugar, se realiza un análisis teórico sobre la complejidad de las dos implementaciones.

Proposición: Si se tiene una tabla hash, con:

$$N = \alpha M$$

Donde:

N : número de llaves

M : capacidad total de la tabla

Entonces:

α : factor de carga

Tenemos que el orden de las operaciones en promedio son:

- Insertar:

$$\sim \frac{1}{2} \left(1 + \frac{1}{1 - \alpha} \right)$$

- Buscar y Eliminar:

$$\sim \frac{1}{2} \left(1 + \frac{1}{(1 - \alpha)^2} \right)$$

Bajo el supuesto de uniformidad del hash. Es decir, bajo el supuesto de que cada llave tiene la misma probabilidad de tener un hash a un entero entre 0 y M-1. A partir de la proposición planteada, construimos la siguiente tabla:

Caso/Método	Linear Probing	Separate Chaining
Mejor Caso	$O(1)$	$O(1)$
Peor Caso	$O(\lg(N))$	$O(\lg(N))$
Caso Promedio	$O\left(\frac{N}{M}\right)$	$O\left(\frac{N}{M}\right)$

La cual aplica para cualquiera de las operaciones principales: buscar, insertar o eliminar. Es importante mencionar que el mejor caso ocurre cuando el elemento en cuestión es el primero en el arreglo que representa la tabla. Por otro lado, el peor caso, ocurre cuando el factor de carga es muy elevado por lo que es difícil encontrar el elemento en cuestión. Teniendo en cuenta que para el método Linear Probing se tiene un factor de carga de 0.5, mientras que el factor de carga de Separate Chaining es de 5. Por esto, las operaciones en Linear Probing serán mucho más eficientes. Sin embargo, teóricamente, se tienen dos métodos con un rendimiento muy similar cuya ventaja es que se tiene un tiempo de operación constante e igual al factor de carga en el caso promedio.

Por último, se discuten las ventajas de cada uno de los métodos:

- Linear Probing: La principal venta de este método con respecto al método Separate Chaining es que ocupa menos espacio. Esto se debe a que solamente se tiene un arreglo, mientras que el método Separate Chaining requiere de un arreglo de listas encadenadas.
- Separate Chaining: La principal ventaja del método es que tiene una implementación más sencilla. Además, el método es menos sensible ante funciones mal diseñadas de hash. Por lo que podemos decir que tiene un funcionamiento más estable.