

Programación de estructuras de datos y algoritmos fundamentales.

Elmer Osiel Avila Vargas – A00826359

Luis Humberto Gonzáles Guerra

Investigación y reflexión de la importancia y eficiencia del uso de las tablas hash

El uso de la técnica de hashing, es decir, la implementación de tablas hash para la estructuración de datos, permite mapear información en un contenedor asociativo de 2 valores, donde uno representa una llave que conduce al otro. De esta forma, podemos alojar strings como llaves que nos conduzcan hacia estructuras o vectores de datos. El uso de un hash table genera una estructuración de datos eficaz y permite el acceso a ellos de una forma altamente eficiente, pues resulta que para una gran parte de los casos la función de búsqueda tiene una complejidad de orden constante $O(1)$, sin embargo, en el peor de los casos dicha complejidad es lineal $O(n)$.

Dicho esto, es preciso mencionar que, para la presente actividad y el paso final de la resolución del reto se empleó el uso de tablas hash, de forma que la llave consistía en un string cuyo valor era una ip, mientras que el valor al que conducía era un vector de pares de información (la fecha en que la ip había intentado un acceso y la causa del error), esto permitió la lectura de la bitácora entera de manera que al encontrar una ip ya mapeada, a esta simplemente se le agregaran valores en el vector. Finalmente, cuando el usuario intentara consultar la información, se desplegaría un resumen de todos los datos mapeados respecto a la ip de entrada, mientras que la cantidad de accesos se obtendría con el uso de la función `size()` en el vector de valores. Para todo ello se empleó `unordered_map`, una librería ya existente en el lenguaje cuyas funciones resultan ser bastantes eficientes y la cual genera una tabla pertinente a toda la teoría ya redactada. Es así como se logró dar orden a más de noventa y un mil líneas de información en cuestión de segundos de una forma realmente sencilla, mientras que la realización de consultas es casi de manera instantánea.

Por otro lado, si pensamos en otra forma de dar solución a la problemática que se propone para esta actividad, a simple vista podemos darnos cuenta de que otras estructuras como las listas encadenadas o algunos tipos de grafos, que fueron empleados para las actividades anteriores, realmente no corresponden o tienen una buena funcionalidad en este reto. En el caso de los árboles binarios de búsqueda de tipo AVL o Splay Tree, pudiésemos idear alguna forma de implementarlos y que resultaría con una solución bastante eficiente, sin embargo, si lo representamos visualmente podemos observar que el uso de hash table se ajusta mucho mejor al tipo de solución y estructura que requiere la actividad, mientras que la eficiencia de búsqueda también resulta ser mejor por la complejidad que presenta.

Finalmente, cabe destacar que la mayoría de los lenguajes de programación de alto nivel proveen de librerías para el uso de este tipo de tablas, lo cual simplifica en gran magnitud la resolución del problema, mientras que la implementación de otros tipos de estructuras resultaría ser más compleja, tediosa y no la más conveniente.