Janet Alejandra Espronceda Cruz

A01197814

**Act 5.2 - Actividad Integral sobre el uso de códigos hash (Evidencia Competencia)**

**REFLEXIÓN**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Link de Replit:** [**https://replit.com/join/opyrzolzuq-a01197814**](https://replit.com/join/opyrzolzuq-a01197814)

En esta última actividad de la materia Programación de estructuras de datos y algoritmos fundamentales se nos pide realizar un programa de c++ que construya una tabla hash. Este concepto, al contrario con las estructuras lineales y no lineales que utilizamos en proyectos pasados nos ayuda a hacer un programa más rápido y eficiente ya que cada elemento cuenta con una clave de identificación que funciona para buscar su determinado valor. En este caso la clave que utilizamos para los datos son las direcciones IP que se nos proporcionaron en el archivo txt y el valor son el resultado de este dominio.

Este programa nos sirve ya que el archivo “bitacora-5\_2.txt” de la actividad cuenta con 91,910 datos. Y al utilizar conceptos comprendidos en diversas clases de actividades anteriores como lo son node, listas enlazadas ayuda en la implementación de tabla hash estos datos se pueden reducir hasta llegar a solo tener la cantidad de diferentes IP que aparecen en la lista.

Una vez teniendo las claves IP, el main le debe de pedir al usuario que de una dirección está al ya saber cual es la dirección que se desea buscar proporciona los datos resumen la información sobre ese IP que se mencionó anteriormente.

Para esta actividad yo cree el replit compartido con mis compañeros y me encargue de poner el orden del archivo agregando todas las clases que podríamos usar para la actividad que anteriormente habíamos usado como lo son linklist, nodos, entre otras. Después en zoom fuimos agregando y/o seleccionando las funciones que necesitaríamos para la actividad, estuvimos haciendo diversas modificaciones hasta que se lograra utilizar las tablas hash que vimos en clase.

La similitud que observe en esta actividad es que los datos también están estructurados pero su relación en este caso es clave que apunta a un valor y hace que podamos reducir los datos en la primera estructura donde no se repiten IP. La diferencia que observe con esta estructura es que es mucho más sencillo de implementar ya que al contrario con los árboles no existen tantas conexiones solo las de Id con su valor.

Este concepto de Tablas Hash lo considero sencillo de implementar ya que es fácil comprender la relación entre la clave relacionado con su valor, lo único que lo hacía complicado es que es la primera vez que este concepto lo implementan en forma de código en c++ y no conceptualmente como lo vimos en clase de programación o en otras materias relacionadas a ingeniería computacional.