Logotipo

Descripción generada automáticamente con confianza media

**Instituto Tecnológico y de Estudios Superiores de Monterrey**

**TC1031.701 Programación de Estructura de Datos y Algoritmos Fundamentales**

**Actividad: Reflexión Final**

Yahir Rivera Huerta A00572029

Profesor: Eduardo Arturo Rodríguez Tello

***Reflexión Individual e Investigación***

Fecha de Entrega: 19 de Junio del 2022

**Introducción**

Dentro de este semestre nos encontramos con el un nuevo reto en la materia TC1031, con el cual se tienen que evaluar tres subcompetencias de las cuales, a mi parecer, son más importantes las dos últimas (SICT0302 y SICT0303), ya que en opinión yo encontré un desarrolló más notorio en mi equipo y en mi sobre estas dos competencias.

Al comienzo de la materia como equipo no teníamos el suficiente conocimiento para poder integrar y solucionar los problemas que nos pedía el reto de una manera exitosa, por lo que buscábamos ayuda para casi cada entregable, pero con forme fue avanzando el semestre empezamos a comprender mucho más sobre cada entrega, cada estructura y cada algoritmo que estábamos empleando, hasta que llegamos al punto de resolver el último entregable de una manera optima y con una ayuda mínima por parte del profesor.

**Reto**

Para este reto se nos comentó que habría una bitácora con una increíble cantidad de datos, para la cual teníamos que implementar las estructuras y algoritmos correspondientes a cada entregable, con el fin de poder organizar los datos de diferentes formas y en diferentes estructuras, para poder encontrar una dirección IP, con la cual podíamos intuir varias cosas, a nosotros se nos dio a entender que esta era una IP de un carácter malicioso la cual llamamos “Bot Master” y tenía características muy especificas con una interconexión con muchas IP’S.

Siendo que ese era el planteamiento de nuestra situación problema, nosotros podemos reflexionar sobre los usos y los problemas con los que se enfrentan las empresas día con día y sobre como son cada vez más recurrentes este tipo de problemas con la vulnerabilidad de la información y datos personales y como todos estamos siempre expuestos a un ciberataque.

**Reflexiones Pasadas**

***Actividad 1.3***

Los algoritmos de búsqueda son aplicaciones de la programación las cuales nos permiten hacer búsquedas en millones de datos para ofrecer las respuestas más acertadas a la búsqueda que se desea realizar. Por otra parte, los algoritmos de ordenamiento son aquellos que buscan darle un orden a una lista de datos.

Ahora bien, reflexionando yo creo que estos algoritmos son de lo más importante al momento en que tenemos una lista enorme de datos y buscamos algo en específico y el más claro ejemplo es una plataforma para navegar en internet (Google, Firefox, etc..) ya que al momento en que realizamos una búsqueda de una página web estamos empleando estos algoritmos, por lo que podemos concluir que sin ellos el internet no sería lo que conocemos.

***Actividad 2.3***

Durante la realización de esta actividad, junto con la actividad 1.3 pude entender un poco más sobre el lenguaje c++, ya que entendí un poco más sobre las funciones template, node, stack. Siento que estas funciones son realmente útiles y facilitan mucho a la hora de hacer programas multiusos y multifuncionales en cuestión a empresas.

Además de mencionar que el comprender la funcionalidad del iterative quick sort es de una gran ayuda, ya que son códigos que realmente se utilizan en el día a día y si no se acaban de entender bien de nada sirve aprenderlos. Creo que todo lo que estamos viendo en esta materia es de real ayuda para nuestro futuro profesional y quiero seguir aprendiendo más y más sobre las formas de programar.

***Actividad 3.4***

Dentro de la actividad 3.4 utilizamos varias funcionalidades de c++, una de ellas es los templates, los cuales nos permiten utilizar clases enteras sin especificar el tipo de dato con el cual se utilizará, gracias a esto podemos implementar de una manera sencilla la actividad de los priority heap y gracias a esto podemos hacer un BST (binary search tree) con una combinación de datos entre enteros y strings.

Además de que aprendí que para poder usar muchas funciones simples con objetos se tiene que usar la sobrecarga de operadores casi siempre y aprendí que cada operador tiene su propia forma de programar para hacer una sobrecarga ya que no todos funcionan de la misma manera y depende de para que lo quieras utilizar es el cómo se tiene que programar. Me gustaría recalcar que gracias a estas evidencias he podido darme cuenta de que realmente no se mucho sobre el lenguaje c++, ya que en cada paso que hemos implementado hemos estado usando nuevas funciones.

Ahora bien, si hablamos de nuestro resultado que es un BST ordenado de mayor a menor, de inmediato nos podemos dar cuenta de que IP se está repitiendo de manera constante y si hacemos una búsqueda por fecha sobre esa IP podríamos saber si es una red infectada por el número de accesos que ha tenido en cierto periodo de tiempo, ya que lo normal sería ver a lo mucho unas cuantas veces el acceso al día o incluso a la semana, pero si nos encontramos con una IP que acceso 50 o 40 veces al día, podemos intuir que probablemente es un bot tratando de infectar la red o en este caso el BST.

***Actividad 4.3***

Para la realización de esta entrega se nos pidió que usaremos las IP’S y los grafos para la correcta realización de esta, siendo que se tenía que relacionar una IP de entrada con una IP de salida, por lo que obviamente una de las mejores formas de hacerlo era con los grafos y teníamos que encontrar al bot master, el cual por lógica podemos deducir que es quien tiene cierta cantidad de accesos a otras IP’S de una manera exagerada y no acorde a las demás IP’S.

Además, creo que personalmente para mí lo que más representa un reto es siempre el saber qué tipo de variable vamos a crear para hacer nuestra lista, ya que uno como programador tiene que interpretar cada uno de los puntos que nos está pidiendo la actividad y en base a eso saber qué tipo de objeto o que procesos tenemos que implementar dentro de nuestras clases para generar un óptimo funcionamiento de nuestro programa.

También reflexionar sobre el uso adecuado de los grafos y el cómo en problemas de esta naturaleza nos ayuda a encontrar de una manera óptima nuestro resultado ya que por medio de las complejidades hemos estado viendo que algoritmos de ordenamiento y búsqueda son los mejores para cada tipo de situación y también remarcar la implementación del algoritmo de Dijkstra para realizar la búsqueda del camino más corto y el cómo esa implementación nos facilitó de una manera muy rápida esa búsqueda.

Sin lugar a duda creo que el tema de grafos es uno de los temas que más me ha gustado por todo lo que implica en el desarrollo del algoritmo y también la increíble cantidad de métodos que se le pueden complementar para gestionar las búsquedas de manera fácil y rápida.

***Actividad 5.2***

Dentro de esta actividad utilizamos las tablas hash para tener el acceso a los datos de una manera mucho más directa y sencilla. Para mi creo que las tablas hash junto con los grafos son una excelente herramienta para esta actividad en específico y más si estuviéramos en un ambiente laboral se facilita de una manera increíble ya que si usamos el hash podemos acceder por medio de las key’s a las IP’S infectadas para poder generar un camino sobre cual ha infectado a cuál y ahí generar una red la cual incluso pudimos haber generado otro grafo y así podemos ir escalando poco a poco hasta llegar a todas las IP’S.

Además de que los hash son métodos super eficientes para la búsqueda de datos em comparación a otras estructuras de datos, además de que la técnica de hashing no se le puede definir una velocidad en específico, ya que se tiene que mantener un balance por medio del control de la velocidad gracias a que se selecciona al espacio en memoria y la otra forma de mantener el balance es controlar el espacio gracias a la selección de la velocidad.

Otros ejemplos para el uso de hashing es en la verificación de contraseñas, resumen de mensajes, estructuras de datos, operaciones de compilador.(información recuperada de: Importance of Hashing GeeksForGeeks)

**Conclusión**

Durante estos meses que hemos estado viendo nuevas estructuras y algoritmos creo que si he tenido una mejora al momento de programar y personalmente creo que esta clase me ayudó mucho a crecer como programador, ya que yo llegué sin tener mucho conocimiento en el área y con muchas dificultades al momento de entender el lenguaje, pero con forme fue pasando el tiempo me fui sintiendo más y más cómodo con la forma de programar y con el conocimiento adecuado para poder implementar cada una de las estructuras y saber diferenciar entre las complejidades y poder usar el método adecuado para los problemas que se me presenten.

Adicionalmente creo que no hubiera podido salir de esta materia de la manera que me siento sin la ayuda del profesor Eduardo, ya que él fue un gran pilar para mi desarrollo en esta materia y si hubiera sido otro profesor, no hubiera podido desarrollarme correctamente como programador.