Laboratorio 1 E.D.A. Grupo 7 Sección 7

3.1.5

¿Cómo implementaría la función?

Rta. = Se realizan dos ciclos For, uno que busca al director y otro que busca las películas con más de 6 puntos de valoració para posteriormente realizar el conteo y sacar el promedio.

¿Qué orden de complejidad tendría la función implementada?

Rta. = Tendría una complejidad de O(n^2) ya que la función se basa en realizar búsquedas en los dos archivos por medio de ciclos For in, por lo que se debe buscar en cada posición de los archivos.

3.2.2

¿Cuál es el ciclo regular para actualizar código en un repositorio GIT?

Rta. = Lo primero es clonar/fork el repositorio sobre el que vamos a trabajar. Después se debe ingresar el comando “Git: pull” para actualizar de forma local los cambios que se hayan hecho en el repositorio de la organización. Luego se debe ingresar el comando “Git: Stage All Changes” para agregar los cambios en el Commit que se hará posteriormente. Se debe ingresar el comando “Git: Commit”, ingresar algún mensaje especificando los cambios realizados y luego colocar el comando “Git: push” para actualizar correctamente los archivos del repositorio.

¿Qué ventajas y limitantes tiene el uso de Ramas/Branches?

Rta. = Una gran ventaja es que permite dividir el trabajo entre los miembros del equipo para equilibrar las cargas del trabajo. Una desventaja es que al cargar el código pueden generarse conflictos si un integrante realizó cambios en el repositorio.

¿Cuáles serían las buenas prácticas para solucionar conflictos?

Rta. = Principalmente dialogar con los compañeros para llevar un orden del código que se vaya desarrollando. Después se debe tener siempre en cuenta “importar” los cambios que han hecho los demás compañeros del grupo al repositorio de la organización. Cuando vayamos a subir nuestro cambio es importante informar a nuestros compañeros los cambios que realizamos y la parte en la que lo hicimos, con eso ellos también importarán esos cambios a sus computadores.

¿Qué orden de complejidad tendría las funciones (consulta y lectura de archivo)?

Rta. = La complejidad de las funciones se dará de una forma lineal, es es poque entre más datos tengamos, irá aumentando gradualmente el tiempo.

¿Cómo podría reducir o aumentar la complejidad de la consulta?

Rta. =Debemos tener en cuenta que una complejidad constante no se puede reducir. Pero una forma en la que demora menos tiempo podría ser dando más especificaciones, de esa forma se tendrán menos datos para revisar.

¿Cómo afecta un TAD en la complejidad?, ¿Qué alternativas existen?

Rta. = Con un TAD podemos agrupar los problemas lógicos y así la complejidad puede disminuir porque los datos se revisan más rápido, ahorrando más tiempo. Alternativas que podemos ver los de conjuntos, los arboles Binarios, pilas y colas.

Link del repositorio: <https://github.com/EDA-SEC-07-EQUIPO-07/Lab1_202020>

Nota: Nuestra compañera Shadith Perez tuvo problemas técnicos y únicamente pudo colaborarnos con la parte escrita del laboratorio.