Después de practicar la integración de ramas y conflictos del laboratorio por favor considere y discuta las siguientes preguntas:

* ¿Cuál es el ciclo regular para actualizar código en un repositorio GIT?
  + GitHub Pool: Crear petición

Seleccionar el repositorio

Seleccionar la rama a la que se le hará el cambio

* ¿Qué ventajas y limitantes tiene el uso de Ramas/Branches?
  + El uso de ramas permite tener distintas versiones de un código original por si este necesita cambios con el fin de optimizar el tiempo de ejecución, espacio que ocupan los procesos o simplemente para cambiar alguna cosa. Al tener varias ramas y actualizarlas, se pueden generar varios conflictos que pueden confundir y ser tediosos.

Ejemplo:

Si una rama contiene trabajos sin fusionar, al intentar borrarla on git Branch -d, habrá error.

* ¿Cuáles serían las buenas prácticas para solucionar conflictos?
  + Para solucionar conflictos es necesario saber cuál es el problema y lo que la población (usuario) afectada(o) necesita o quiere, comprender el ámbito en el que se desarrollan y ver cómo diferentes soluciones pueden afectar sus prácticas del día a día.

Después de practicar y variar los parámetros de consultas de estructuras de datos considere y discuta las siguientes preguntas:

* ¿Qué orden de complejidad tendría las funciones (consulta y lectura de archivo)?
  + El orden de complejidad sería lineal debido a que, en el peor de los casos, el objecto a buscar sería el último o puede que no se encuentre, por lo que habrá que recorrer todo el contenido. Su tiempo de ejecución termina siendo proporcional a la cantidad de datos.
* ¿Cómo podría reducir o aumentar la complejidad de la consulta?
  + Cambiando el código, pues se debe mejorar una parte del mismo para poder hacerlo más efectivo en términos de tiempo. Por ejemplo, al tener:
    - For x in A

For y in B

For z in C

Tiene una complejidad aproximada de O(n^3)

* + - x=0

y=0

z=0

while:

x+=1

y+=1

z+=1

Mientras que hacer recorridos con contadores seria menos complejo

* ¿Cómo afecta un TAD en la complejidad?, ¿Qué alternativas existen?
  + Un TAD permite tener diferentes formas de hacer un proceso o una secuencia de los mismos. Dichos procesos se pueden optimizar, ya sea con base en mejorar la cantidad de memoria que se ocupa o, el tiempo que necesita en hacer dicho proceso, teniendo en cuenta en ambas opciones, la cantidad de datos X, que, puede cambiar en el tiempo.