Tarea de Micros

1. Git es un sistema capaz de pautar cambios efectuados en el código de un algoritmo gracias a que se crea una copia entera del historial de desarrollo con los cambios que se transfiere de un repositorio a otro, su objetivo principal es el de aumentar la velocidad de ejecución de código e integridad de la información circulante en el algoritmo. Se ha de denotar que este sistema se utiliza cuando se busca una gran velocidad en códigos no secuenciales, esto mediante la utilización de “branches” livianos, término definido posteriormente, donde cada rama se le asigna solo una acción a ejecutar.
2. Github es el ente encargado de alojar los diseños de software descritos por Git, proporciona todo el historial de escritura de código además de brindar cierto tipo de ayuda para optimizar el mismo, esto mediante “bug tracking”, que demarca los errores en el software mediante un aviso o flag; “task management”, parte que se encarga de dirigir cada acción en el código; creación de documentación con caracteres importantes (tales como los “README files”) y la capacidad de visualizar archivos con formato PDF.
3. El tópico de “branch”; según el propósito de la asignación, tiene dos contextos que se han de pautar. El primero, el cual engloba el tópico de Git, define una instrucción que se encarga de aislar una parte del código, o bien, trabajo sin afectar a los otros branches en el repositorio, cada uno posee un Branch por default, número que puede aumentarse si es necesario. El segundo, referente a las ciencias de computación, describe a una instrucción en un algoritmo encargado de decirle a la unidad de control que empiece a ejecutar instrucciones en un orden distinto, esto lo hace mediante brincos de instrucciones sobre direcciones de memoria donde la acción siguiente se va a ejecutar de tal forma que el acceso a cada celda de memoria no sea en el orden definido.
4. Commit define a la escritura de código tentativo como permanente en un branch, esto en contexto de la computación. El código principal siempre se cambia cuando se le hace commit al archivo, estos cambios se realizan en un branch específico; posibilitando cualquier cambio mediante cualquier usuario.
5. Los archivos de Git pueden estar en 3 estados, el estado committed donde los elementos están almacenados de manera segura en la base de datos, el estado modified donde se los archivos no están guardados, por ende, se pueden sobrescribir y el estado staged, un estado intermedio en el que se checkea que todos los cambios estén avalados para pasar al staging donde se direcciona el archivo para que este se transfiera a committed.
6. El comando git checkout se encarga de cambiar branches mediante la actualización de archivos en el árbol de operaciones, se guardan y se realizan operaciones directamente en el commit del branch.
7. Git stash es una instrucción destinada a salvar de manera “rápida”, lo que hace es tomar el código trabajado con todas sus modificaciones y guardarlo en una pila provisional para usarse posteriormente y tener el espacio de trabajo “limpio”.
8. Git Add refiere a la herramienta utilizada para guardar, en el estado staged, un cambio del directorio; cada estatuto realizado demarca al sistema principal todas las actualizaciones de un archivo específico al siguiente estado de commit, mas no cambios en el principal.
9. Pytest refiere a la librería, en Python, destinada para pequeñas pruebas de código. Esta se encarga de divisar las pruebas impuestas por el usuario, correr las mismas y mostrar toda la información recolectada para realizar los debidos cambios posteriormente.
10. En el contexto de Python, la instrucción assert permite que el programador declare alguna condición como verdadera. Si esta es en realidad verdadera, no pasará nada y el código seguirá ejecutándose; de lo contrario, se dará un Assertion Error. Un ejemplo muy simple es si se deseara probar una función de suma de números a y b, cuyo resultado sería c. Se podría poner la instrucción “assert c==a+b”, para comprobar que la función realmente lo realiza correctamente.
11. Flake8 es una librería en Python encargada de la detección de errores con respecto al estilo de código PEP8, como lo son las variables no definidas, bibliotecas no utilizadas y complejidades ciclomáticas, término referente a todos los caminos independientes que posee un código, esto se puede representar como los condicionales if y las distintas operaciones según la condición.