

## **Tarea 1 – Microprocesadores y Microcontroladores**

José Pablo Álvarez Preinfalk y Marcela Menjívar Sánchez

### Preguntas Teóricas

#### **1) ¿Explique que es git y su relación con github?**

Un Git es un sistema de control de versiones de archivos. Es una herramienta útil para trabajos grupales, ya que este se encarga de mostrar los cambios realizados en tiempo real. GitHub es una plataforma basada en el almacenamiento en la nube, donde se pueden subir, compartir y trabajar archivos enfocados en código con otros usuarios. GitHub se basa entonces en el software de código abierto Git para sus funciones fundamentales.

#### **2) ¿Qué es un branch? ¿Qué es un fork?**

Una rama (branch) es una línea de desarrollo independiente que se encuentra dentro de un repositorio en GitHub. Esta sirve principalmente para trabajar en cambios sin alterar la rama principal, llamada main o master.

Por su parte, un fork es una copia entera de un repositorio en una cuenta particular de GitHub. Se usa para poder trabajar de manera aislada e independiente el repositorio original, pero sin afectar el proyecto fuente original.

#### **3) En el contexto de github. ¿Qué es un Pull Request?**

En GitHub, un Pull Request es una solicitud para que los cambios realizados en una rama o en un fork se revisen y aprueben. Esto para que se integren posteriormente en otra rama del repositorio.

#### **4) ¿Qué es un commit?**

En Git y GitHub, un commit es un historial de una serie de datos y cambios realizados en el repositorio. Este incluye qué se cambió, como archivos editados, agregados y eliminados. Incluye además el autor de los cambios, fecha, hora y razón (mensaje del commit).

**5) Explique que es un “merge conflict” o “rebase conflict” en el contexto de tratar de hacer merge a un Pull Request o de completar una operación git rebase.**

Un merge conflicto ocurre cuando se intentan combinar 2 ramas y la plataforma no puede decidir automáticamente cuáles cambios mantener ya que la misma parte del archivo fue modificada en forma diferente en ambas. Un rebase conflicto es similar, solo que este ocurre cuando la plataforma reaplica los commits sobre otra base y encuentra diferencias que no puede resolver solo.

**6) ¿Qué es una Prueba Unitaria o Unittest en el contexto de desarrollo de software?**

Una prueba unitaria es un tipo de prueba de software que verifica el funcionamiento de una parte específica del código. Por ejemplo, se puede utilizar en una función o método para asegurarse de que produzca el resultado esperado dadas ciertas entradas.

**7) Bajo el contexto de pytest. ¿Cuál es la utilidad de un “assert”?**

Un assert se utiliza para verificar que una condición sea verdadera durante una prueba. Si la condición es verdadera se pasa la prueba, en cambio, si es falsa, pytest marca la prueba como fallida.

**8) ¿Explique que son github-actions y su utilidad para el desarrollo continuo de código?**

Las GitHub Actions son automatizaciones dentro de GitHub que permiten ejecutar tareas automáticamente cuando ocurre un evento en un repositorio. Algunas de estas pueden incluir detectar errores antes de fusionar cambios, ejecutar pruebas cada vez que se sube un código, hacer compilaciones, entre otras.

**9) ¿Qué es Flake 8?**

Es una herramienta de análisis de código en Python que ayuda a mantener el código en formato adecuado y consistente. Este detecta errores de sintaxis y problemas de estilo. Además, Flake 8 indica detalles como espacios de más, nombres de variables inconsistentes o imports sin usar.

## **10) Explique la funcionalidad de parametrización de pytest.**

La parametrización en pytest permite ejecutar una misma función de prueba varias veces con diferentes conjuntos de datos. Esto facilita enormemente la prueba de diferentes escenarios y casos sin tener que alterar de manera importante el código de prueba. Para utilizar la parametrización se usa la sintaxis: `@pytest.mark.parametrize`

### **Bibliografía**

<https://aprendegit.com/que-es-un-pull-request/>

<https://docs.github.com/es/pull-requests/collaborating-with-pull-requests/proposing-changes-to-your-work-with-pull-requests/about-branches>

<https://graphite.dev/guides/git-fork-vs-branch>

<https://keepcoding.io/blog/que-es-unittest-y-para-que-sirve/>

<https://docs.pytest.org/en/stable/how-to/assert.html>

<https://trunk.io/linters/python/flake8>

<https://docs.pytest.org/en/stable/example/parametrize.html>