



Universidad Tecnológica de Durango

Tecnologías de la Información

Programación estructurada

Prácticas

“Evidencias de Prácticas”

Alumnos:

- Gómez Aguirre Eli Gabriel

2°B BIS

Docente:

- Ing. Dagoberto Fiscal Gurrola, M.T.I.

Mayo 2025

Tabla de Ilustraciones

Ilustración 1Evidencia del repositorio de github	3
--	---

Actividad 1

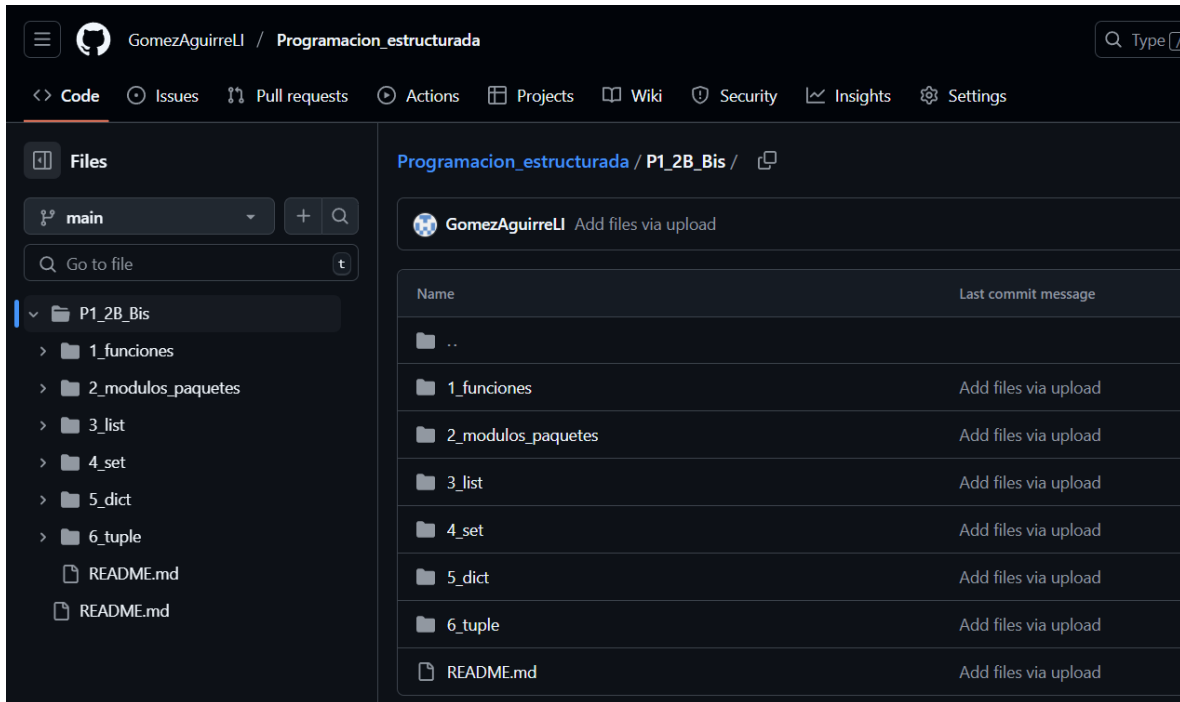


Ilustración 1 Evidencia del repositorio de github

Da clic: [ENLACE PARA ACCEDER AL DRIVE](#)

En la imagen anterior se muestra la evidencia del link al repositorio de “github” que se creó para entregar de manera satisfactoria todas las prácticas de la primera unidad de la materia.

Retroalimentación

La programación estructurada, con sus pilares fundamentales en los paradigmas imperativo y procedimental, se consolida como una disciplina esencial para el desarrollo de soluciones computacionales robustas y eficientes. La clave reside en la determinación y aplicación de buenas prácticas, que se entrelazan con el uso inteligente de herramientas y técnicas.

El paradigma de la programación estructurada se apoya en tres estructuras de control básicas: la secuencia, la selección (condicionales) y la iteración (bucles). Estas estructuras, al ser empleadas de manera disciplinada y anidada, permiten la construcción de algoritmos claros, modulares y de fácil mantenimiento.

Las buenas prácticas de programación van más allá del simple uso de estas estructuras; abarcan la legibilidad del código (nombres significativos, comentarios cuando sean necesarios, formateo consistente), la modularidad (dividir el problema en funciones y procedimientos pequeños), la simplificación del código (evitar la complejidad innecesaria), y la encapsulación para reutilizar fragmentos de código. Herramientas como los sistemas de control de versiones y las plataformas de prueba automatizada son técnicas indispensables que complementan estas prácticas, asegurando la calidad y el seguimiento del desarrollo.

En conclusión, la programación estructurada no es solo un conjunto de reglas, sino una filosofía que promueve la claridad, la eficiencia y el mantenimiento del software. Al combinar la comprensión de los paradigmas de programación, el dominio de las estructuras de control y la adhesión rigurosa a las buenas prácticas, los desarrolladores pueden crear soluciones computacionales que no solo resuelvan problemas, sino que también sean comprensibles, escalables y adaptables a los desafíos futuros.