

## El Problema: Analizador de Mutaciones Genéticas

En el campo de la bioinformática, una tarea fundamental es comparar dos secuencias de datos biológicos (como ADN o ARN) para identificar las diferencias, que a menudo son el resultado de mutaciones. Tu desafío es diseñar una herramienta flexible y segura que pueda realizar esta tarea para cualquier tipo de secuencia comparable.

### Objetivos de la Tarea

Tu implementación debe ser capaz de cumplir con los siguientes puntos, utilizando los conceptos de **genéricos**, **restricciones de tipo** y **records**:

#### 1. Modelado de Datos:

- Crea un record llamado ParBase. Este record debe encapsular una base de la secuencia original y su correspondiente base en la secuencia mutada.
- Los campos de este record deben ser genéricos, de modo que puedan contener cualquier tipo de dato, como String para bases ("A", "T") o incluso Character.
- Este record debe ser Comparable. La lógica de la comparación debe ser simple: un ParBase es "igual" a otro si las bases original y mutada son las mismas. Es "diferente" si las bases son distintas, lo que indica una mutación.

#### 2. Lógica del Algoritmo:

- Crea una clase utilitaria (no genérica) llamada AnalizadorSecuencias.
- Dentro de esta clase, implementa un método **estático genérico** llamado analizarMutaciones. Este método es la pieza central de tu solución.
- El método debe aceptar dos listas como entrada: List<T> secuenciaOriginal y List<T> secuenciaMutada.
- El tipo T debe estar restringido de tal forma que solo acepte tipos que sean Comparable.
- El método debe iterar sobre las dos listas en paralelo, creando un ParBase para cada par de elementos y agregándolo a una lista de resultado.
- El método debe devolver una List de tu record ParBase.

#### 3. Prueba y Verificación:

- En tu método main, crea dos secuencias de ADN usando Strings. Asegúrate de que una secuencia tenga al menos una mutación (es decir, un cambio en una de las bases).
- Llama a tu método analizarMutaciones con estas secuencias.
- Recorre la lista de ParBases que el método devuelve.
- Para cada ParBase, usa su capacidad Comparable para determinar si representa una mutación o no. Imprime un mensaje claro en la consola que indique qué pares son mutaciones y cuáles no.