

Retos - 005: El biólogo

Nivel de dificultad aproximado (1 a 5): 3 ★★☆☆☆

Enunciado del problema

Eres un biólogo que examina secuencias de ADN de formas de vida diferentes. Se te darán dos secuencias de ADN, y el objetivo es encontrar el conjunto ordenado de bases adyacentes de mayor tamaño que es común en ambos ADN.

Las secuencias de ADN se darán como conjuntos ordenados de bases de nucleótidos: adenina (abreviado A), citosina (C), guanina (G) y timina (T):

ATGTCTTCCTCGA TGCTTCCTATGAC

Para el ejemplo anterior, el resultado es CTCCTC porque que es el conjunto ordenado de bases adyacentes de mayor tamaño que se encuentra en ambas formas de vida.

(Asegúrate de que has leído las [preguntas frecuentes](#) antes de plantear tu solución)

Preguntas frecuentes

Recuerda que hay ciertas pautas que son frecuentes en los retos de programación:

- Un reto de programación NO es un programa interactivo, no debe avisar al usuario con frases como "Introduce un número" o "Dame un número", ni responder con "Tu resultado es...".
- Por lo mismo, no debe hacer ninguna pausa la final de la ejecución.
- Debes tomar los datos de la entrada estándar y mostrar los resultados en la salida estándar. no debes utilizar ficheros ni parámetros de línea de comandos.
- Debes analizar los datos caso a caso: analizar el primer caso y dar su respuesta, luego analizar el segundo y dar su respuesta, luego el tercero y así sucesivamente. No debes memorizar todas las soluciones en un array (ni ninguna otra estructura) para mostrar todas ellas al final.
- En general, no deberás validar los datos de entrada. Por ejemplo, si se te dice que son números enteros entre 1 y 1.000.000, no debes comprobar si se trata de algún número negativo o por encima de un millón (además, ¿qué debería hacer tu programa en ese caso?). Esa información es sólo para que sepas el tipo de datos que debes usar (por ejemplo, quizá no te sirva un entero corto).
- Prueba si las entradas de ejemplo dan como resultado las salidas de ejemplo. Si no es así, el programa claramente está mal. Si parece comportarse correctamente con esos casos, eso no quiere decir que sea perfecto, quizá falle con otros casos de prueba más complejos, o resulte lento con datos grandes (y muchos concursos de programación incluyen limitaciones de tiempo).