

SEGUIMIENTO: COMPUTACIÓN Y ESTRUCTURAS DISCRETAS  
PROFESOR: MARLON GÓMEZ

**Instrucciones** El trabajo debe realizarse en los grupos de la tarea integradora y debe ser entregado el primer día de clase de la próxima semana. Si tienen alguna duda o inquietud, pueden hacérmela saber por correo electrónico. La entrega debe ser un proyecto de IntelliJ que se ejecute por consola. Dentro del proyecto debe haber una carpeta con documentación, en la que se incluya el análisis temporal y espacial de los algoritmos desarrollados. No es necesario incluir pruebas de ningún método.

1. Escribe un algoritmo que tenga como entrada una lista de  $n$  enteros en orden no decreciente y genere la lista de todos los valores que aparecen más de una vez.
  - a) Entrada:  $L = [1, 2, 2, 3, 4, 4, 4, 5, 6, 6]$
  - b) Salida:  $R = [2, 4, 6]$
2. Un palíndromo es una cadena que se lee igual de izquierda a derecha que al revés. Escribe un algoritmo para determinar si una cadena de  $n$  caracteres es un palíndromo.
  - a) Entrada: *cadena* = "reconocer"
  - b) Salida: *verdadero*
  - c) Entrada: *cadena* = "hola"
  - d) Salida: *falso*
3. Escribe un algoritmo que cuente los bits 1 que aparecen en una cadena de bits, examinando cada bit para determinar si es un 1.
  - a) Entrada: *cadenaBits* = "1011001"
  - b) Salida: 4
4. Analiza la complejidad temporal y espacial, en el peor caso, de cada uno de los algoritmos.
  - a) Realiza una tabla y describe línea por línea la cantidad de veces que se ejecuta cada una de ellas.
  - b) Al final, realiza la suma de las líneas para hallar la complejidad del algoritmo y concluye cuál es la cota superior. Una manera formal de encontrar la cota es mediante una constante  $c$  que permita acotar por arriba la función encontrada.