

**Objetivos****Unidad 1: Análisis de Algoritmos**

- OE1.1. Calcular la complejidad temporal de algoritmos iterativos.
- OE1.2. Calcular la complejidad espacial de algoritmos iterativos.
- OE1.3. Caracterizar la entrada de un algoritmo iterativo con el fin de calcular la complejidad para el mejor y peor caso.
- OE1.4. Analizar algoritmos independiente de una implementación concreta (no dependiente del lenguaje de programación).
- OE1.5. Utilizar notación asintótica para describir la complejidad de algoritmos.
- OE1.6. Evaluar varios algoritmos que resuelven el mismo problema en términos de sus complejidades computacionales.
- OE1.7. Comprender la importancia del Modelo RAM en el proceso de análisis de algoritmos.

**Enunciado**

Gran parte de los algoritmos más utilizados hoy en día en diferentes áreas de la computación descansan en algoritmos fundamentales como los algoritmos de ordenamiento. Una empresa de fabricación de microprocesadores está evaluando la posibilidad de implementar varios algoritmos de ordenamiento como instrucciones básicas de su próximo coprocesador matemático.

Un coprocesador es un microprocesador de un ordenador utilizado como suplemento de las funciones del procesador principal (la CPU). Las operaciones ejecutadas por uno de estos coprocesadores pueden ser operaciones de aritmética en coma flotante, procesamiento gráfico, procesamiento de señales, procesamiento de texto, criptografía, etc. Y su función es evitar que el procesador principal tenga que realizar estas tareas de cómputo intensivo, estos coprocesadores pueden acelerar el rendimiento del sistema por el hecho de esta descarga de trabajo en el procesador principal y porque suelen ser procesadores especializados que realizan las tareas para las que están diseñado más eficientemente.<sup>1</sup>

Después de estudiar los costos de implementación del ordenamiento como una operación nativa del coprocesador, la empresa ha decidido implementar tres (3) algoritmos diferentes de ordenamiento que permitan ordenar, muy rápidamente, números enteros de tamaño arbitrariamente grande y números en formato de coma flotante de cualquier tamaño.

Su equipo de desarrollo ha sido elegido para la selección e implementación del prototipo de pruebas en software de los algoritmos que finalmente serán implementados como operaciones nativas en hardware.

El programa esperado por la empresa debe implementar:

- (1) una interfaz gráfica de usuario que le permita ingresar los valores a ordenar
- (2) una interfaz gráfica que le permita generar aleatoriamente los valores (tanto enteros como de coma flotante) permitiendo configurar la cantidad total de números a generar y el intervalo en el cual se generarán los números. Debe permitir indicar si los números a generar deben ser todos diferentes o pueden haber repetidos.

La generación aleatoria debe permitir elegir entre estas posibles configuraciones de los valores en la secuencia:

- (a) que los valores estén ya ordenados
- (b) que los valores estén ordenados inversamente
- (c) que los valores estén en orden completamente aleatorio
- (d) que los valores estén desordenados en un % indicado por el usuario. Este % de desorden se debe generar de la siguiente manera:
  - (i) Se genera ordenada la secuencia
  - (ii) Con base en el tamaño de la secuencia y el % de desorden se obtiene un número k de cuantas posiciones deben estar desordenadas.
  - (iii) Se generan k/2 pares de posiciones diferentes y se intercambian los valores entre cada par de ellas

---

<sup>1</sup> <https://es.wikipedia.org/wiki/Coprocesador>

(3) ordenar los valores ingresados o generados utilizando el algoritmo apropiado entre los tres -3- finalmente seleccionados y presentar el tiempo que toma el ordenamiento. De acuerdo con el tipo de número a ordenar y el intervalo de números generados, el programa debe restringir/permitir los algoritmos para ordenar.

Usted debe utilizar el método de la ingeniería para resolver este problema y dejar evidencia en su informe de los resultados de cada fase. Por ejemplo, en la fase 1 deben identificar claramente el problema, justificarlo y especificar los requerimientos funcionales. Recuerde revisar el [Resumen del Método de la Ingeniería](#) y el [ejemplo del Método de la Ingeniería aplicado a un problema](#).

**Entregables.** 1. Análisis de complejidad temporal de cada uno de sus algoritmos 2. Análisis de complejidad espacial de cada uno de sus algoritmos 3. Especificación de Requerimientos y Diseño. 4. Implementación del programa con todo los requerimientos en su lenguaje de programación favorito.

El laboratorio debe ser desarrollado en grupos de máximo 3 estudiantes. Tenga en cuenta la rúbrica de evaluación para esta actividad: [Rúbrica del Laboratorio 1](#).

**Nota:** Usted debe entregar un archivo comprimido en formato zip de un directorio con únicamente 2 archivos: 1 archivo de informe en formato pdf con toda la documentación (de cada una de las fases del método y el análisis) y otro archivo comprimido de un directorio con los archivos de codificación en sus respectivos paquetes.

El nombre del archivo comprimido debe tener el formato: PRIMERAPELLIDOEST1\_PRIMERAPELLIDOEST2\_PRIMERAPELLIDOEST3.zip (tenga en cuenta que el separador entre cada apellido es un guion al piso).