# Lucas Astor Análisis

Para realizar esta solución se tuvieron en cuenta 2 algoritmos de ordenación.

**BubbleSort y HeapSort**

La gran diferencia de estos dos algoritmos es su complejidad. Mientras que **BubbleSort** es un algoritmo muy sencillo que para esta solución podría resolver el problema con muy pocas líneas de código, su simplicidad acarrea consecuencias y es que su orden de tiempo de ejecución siempre es **O(N2)** esto hace que si bien para la muestra de datos que poseemos en este problema el algoritmo podría manejarlo perfectamente, si el problema crece podríamos encontrar un impedimento en cuanto a performance.

Por esto se optó por seleccionar el algoritmo de **HeapSort** que nos asegura un orden de tiempo de ejecución de **O(N Log N)** en todos los casos, si bien la complejidad de su programación es mucho mayor, para la realidad del problema presentado va a ser el que mejor la solucione dado que como se muestra en la letra, pueden llegar a haber millones de entradas, esto hace que si utilizamos un algoritmo como **BubbleSort** puede ser que no logremos correr la ordenación, mientras que si bien **HeapSort** no es el mejor algoritmo, nos asegura un tiempo medianamente razonable, capaz de manejar a la perfección una cantidad considerable de datos