

Considero que la complejidad de todo el programa es $O(n \log n)$, ya que quien posee la mayor complejidad en todo es la funcion de merge sort. Esto es debido a que tiene que estar dividiendo a la mitad todos los datos otorgados a fin de poder encontrar la posición a la que pertenece el dato.

Esta división hace que dicha parte del merge tenga una complejidad de $O(\log n)$, pero en el peor de los casos, esta accion la tendría que estar repitiendo en base a la cantidad de datos otorgados, siendo así que su complejidad sea de $O(n \log n)$.

Cabe resaltar que, aunque el programa tenga la función de busqueda binaria, cuya complejidad es $O(\log n)$ debido a que divide a la mitad la cantidad de datos otorgados al igual que en el merge sort, no es contada dentro de esta debido a que la complejidad de merge sort es superior en comparación a la busqueda binaria.