

1. By iterating over each student's information sequentially, linear search might be used in a school's student database to locate a record for that kid. When the data is unsorted or for small-scale searches, this could be appropriate. On the other hand, if the student database is arranged according to something other than student ID numbers, binary search could be utilised. This would make it possible to get student data more quickly, particularly in big databases where binary search has an  $O(\log n)$  time complexity as opposed to an  $O(n)$  time complexity for linear search.
2. Sorting algorithms are essential for organising student exam scores in an educational context. This may be accomplished via bubble sorting and insertion sorting. Due to its ease of use, bubble sort may be applied to instructional activities or tiny datasets. For the purpose of keeping a sorted list updated with new scores, insertion sort may be helpful. Sorting efficiency is important since it directly affects how quickly tasks like class rankings or identifying high-performing students can be completed, especially when dealing with bigger datasets.
3. In order to improve the functionality of search and sort algorithms in school administration systems, optimisation approaches are essential. For example, early exit conditions and fewer pointless comparisons can increase the effectiveness of binary search. Similar methods may be used to improve the efficiency of merge sort, such as bottom-up merging or insertion sort for tiny subarrays. By cutting down on the number of comparisons and iterations needed to finish activities like retrieving student records or computing grades, these optimisation approaches increase efficiency. School management systems may offer better user experiences and quicker reaction times to administrators, teachers, and students by putting these strategies into practice.