**Exercise 6: Library Management System**

Linear search- It searches the library catalog book by book until the specified author or title is discovered. For big datasets, this approach is straightforward yet ineffective.

Binary search- It is more faster, but it needs the data to be sorted first. The search range is halved periodically until the target book is located or confirmed to be missing. For greater search efficiency in larger catalogs, binary search is recommended over linear search, which is appropriate for smaller libraries.

The temporal complexity of linear search is O(n), which means that as the number of entries in the list increases, so does the time required to find an entry. It is hence ineffective for big datasets.

Binary search has an O(log n) time complexity, which makes it much faster. It operates by halving the search interval several times, which makes it incredibly effective for big sorted datasets. Although sorting the data is necessary, the improvement in search time frequently surpasses the sorting overhead.

It is easy to implement, linear search might be sufficient for tiny libraries with a few hundred books. But when the collection expands, binary search becomes much more effective, particularly when book data is arranged according to author or title. Binary search performs best with a sorted dataset, but linear search can be applied to unsorted data as well. The burden of keeping a sorted order for binary search could arise if the library adds or removes books on a regular basis. For quicker lookups in these situations, a hybrid strategy or data structures like hash tables may be taken into consideration.