**LIBRARY MANAGEMENT SYSTEM**

**SEARCH ALGORITHMS:**

**1. Linear Search:**

- Scans each element one by one until a match is found.

- Time Complexity: O(n)

- Does not require the list to be sorted.

**2. Binary Search:**

- Divides the list in half repeatedly to locate the target.

- Time Complexity: O(log n)

- Requires the list to be sorted beforehand**.**

**PROGRAM:**

import java.util.\*;

public class LibraryManagementSystem {

static class Book {

int bookId;

String title;

String author;

public Book(int bookId, String title, String author) {

this.bookId = bookId;

this.title = title;

this.author = author;

}

@Override

public String toString() {

return "Book(" + bookId + ", '" + title + "', '" + author + "')";

}

}

public static Book linearSearchByTitle(List<Book> books, String targetTitle) {

for (Book book : books) {

if (book.title.equalsIgnoreCase(targetTitle)) {

return book;

}

}

return null;

}

public static Book binarySearchByTitle(List<Book> books, String targetTitle) {

int low = 0;

int high = books.size() - 1;

targetTitle = targetTitle.toLowerCase();

while (low <= high) {

int mid = (low + high) / 2;

String midTitle = books.get(mid).title.toLowerCase();

if (midTitle.equals(targetTitle)) {

return books.get(mid);

} else if (midTitle.compareTo(targetTitle) < 0) {

low = mid + 1;

} else {

high = mid - 1;

}

}

return null;

}

public static void main(String[] args) {

List<Book> books = new ArrayList<>();

books.add(new Book(1, "The Great Gatsby", "F. Scott Fitzgerald"));

books.add(new Book(2, "To Kill a Mockingbird", "Harper Lee"));

books.add(new Book(3, "1984", "George Orwell"));

books.add(new Book(4, "Pride and Prejudice", "Jane Austen"));

books.add(new Book(5, "Moby Dick", "Herman Melville"));

System.out.println("Linear Search for '1984':");

Book resultLinear = linearSearchByTitle(books, "1984");

System.out.println(resultLinear);

books.sort(Comparator.comparing(book -> book.title.toLowerCase()));

System.out.println("\nBinary Search for '1984':");

Book resultBinary = binarySearchByTitle(books, "1984");

System.out.println(resultBinary);

}

}



**TIME COMPLEXITY:**

- Linear Search: O(n)

Best for small or unsorted datasets.

- Binary Search: O(log n)

Best for large, sorted datasets.

**WHEN TO USE:**

- Use Linear Search:

- When the list is unsorted or very small.

- When simplicity is more important than speed.

- Use Binary Search:

- When the list is large and sorted.

- When search performance is critical.