DEPARTMENT OF COMPUTER ENGINEERING

A. Y. 2025-26 Semester-I

MINI PROJECT REPORT

Subject: Data Structures Laboratory

Group No. : MP 1

Title of the Project: Library Management System

Group Members:

Sr. No.	PRN No.	Name of the Student
01.	B24CE1015	Swarada Deshpande
02.	B24CE1024	Jidnyasa Patil
03.	B24CE1025	Ashutosh Kulkarni

Data Structures Used: Array,Simple queue (using array),Dictionary
Algorithm : Binary Search,Quick sort

Mini-Project Idea:
The Library Management System is designed to manage a digital library where users can search, add, remove, and issue books efficiently.
The system demonstrates how Data Structures and Algorithms can be applied to manage large collections of books quickly and effectively.

Input:

- Book details (Title, Author, ISBN, Genre)
- User requests (Search, Add, Issue, Return)

Output:

- Displays search results for books
- Shows successful addition, issue, or return messages
- Maintains lists of available and issued books

Techniques Used:

- Efficient searching using Binary Search
- Fast sorting using Quick Sort
- Data management using Array, Simple Queue, and Dictionary

Program:

Data Structure	**Purpose**	**Implementation**
Array (List)	To store the collection of books in sorted order for binary search	Python list
Simple Queue (using Array)	To maintain the waitlist for issued books	Enqueue and Dequeue operations implemented using lists

CODE:-

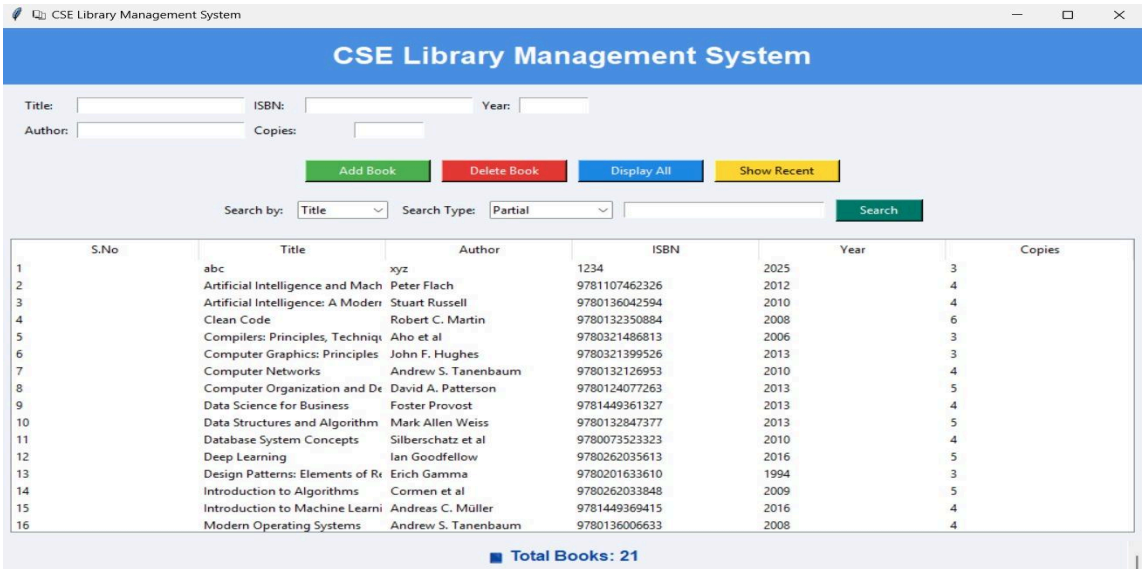
```
# ---------------------- DATA & GLOBALS ----------------------
books = []
recent_books = []
MAX_QUEUE_SIZE = 5
excel_file = "cse_books.xlsx"

# ---------------------- DATA STRUCTURE FUNCTIONS ----------------------
def quick_sort(arr, key="Title"):
    if len(arr) <= 1:
        return arr
    pivot = arr[len(arr)//2][key]
    left = [x for x in arr if x[key].lower() < pivot.lower()]
    middle = [x for x in arr if x[key].lower() == pivot.lower()]
    right = [x for x in arr if x[key].lower() > pivot.lower()]
    return quick_sort(left, key) + middle + quick_sort(right, key)

def binary_search(arr, keyword, key="Title"):
    low, high = 0, len(arr)-1
    keyword = keyword.lower()
    while low <= high:
        mid = (low+high)//2
        mid_val = arr[mid][key].lower()
```

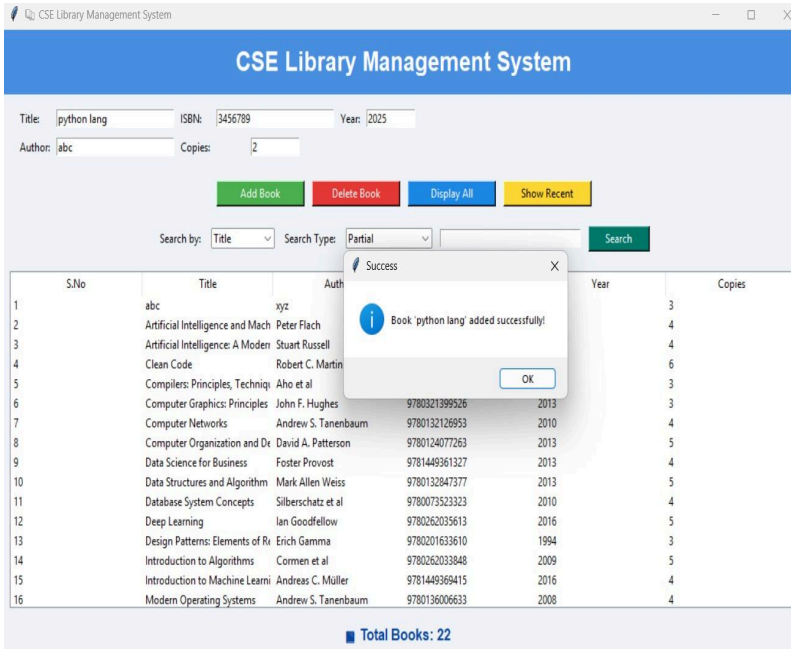
```
        if mid_val == keyword:
            return arr[mid]
        elif mid_val < keyword:
            low = mid+1
        else:
            high = mid-1
    return None

# ---------- LOAD & SAVE ----------
def load_books():
    global books
    if os.path.exists(excel_file):
        df = pd.read_excel(excel_file)
        books = df.to_dict('records')
    else:
        books = []
        save_books()

def save_books():
    df = pd.DataFrame(books)
    df.to_excel(excel_file, index=False)
    update_count_label()

# ---------- BOOK OPERATIONS ----------
def add_book():
    title = title_var.get().strip()
    author = author_var.get().strip()
    isbn = isbn_var.get().strip()
    year = year_var.get().strip()
    copies = copies_var.get().strip()

    if not title or not author or not isbn or not copies:
        messagebox.showwarning("Warning", "Please fill in all fields!")
        return

    try:
        year = int(year)
        copies = int(copies)
    except:
        messagebox.showwarning("Warning", "Year and Copies must be integers!")
        return

    books.append({"Title": title, "Author": author, "ISBN": isbn, "Year": year, "Copies": copies})
    recent_books.append(title)
    if len(recent_books) > MAX_QUEUE_SIZE:
```

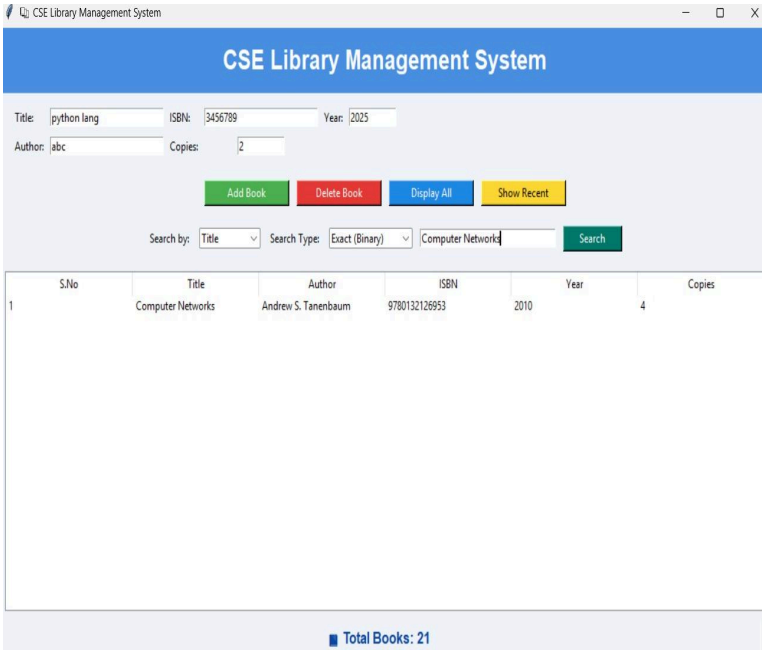
```
        recent_books.pop(0)

    save_books()
    refresh_table()
    messagebox.showinfo("Success", f"Book '{title}' added successfully!")

def delete_book():
    selected = tree.focus()
    if not selected:
        messagebox.showwarning("Warning", "Select a book to delete!")
        return
    values = tree.item(selected, "values")
    title_to_delete = values[1]

    global books
    books = [b for b in books if b["Title"] != title_to_delete]

    save_books()
    refresh_table()
    messagebox.showinfo("Deleted", f"Book '{title_to_delete}' deleted.")

def search_books():
    keyword = search_var.get().strip()
    if not keyword:
        messagebox.showinfo("Info", "Enter a keyword to search.")
        return
    mode = search_mode.get()  # Title, Author, ISBN
    stype = search_type.get()  # Partial or Exact
    for row in tree.get_children():
        tree.delete(row)

    found = False
    if stype == "Exact (Binary)":
        sorted_books = quick_sort(books, key=mode)
        result = binary_search(sorted_books, keyword, key=mode)
        if result:
            tree.insert("", "end", values=(1, result["Title"], result["Author"], result["ISBN"], result["Year"],
result["Copies"]))
            found = True
    else:  # Partial search
        for i, book in enumerate(books):
            if keyword.lower() in str(book[mode]).lower():
                tree.insert("", "end", values=(i+1, book["Title"], book["Author"], book["ISBN"], book["Year"],
book["Copies"]))
                found = True
```

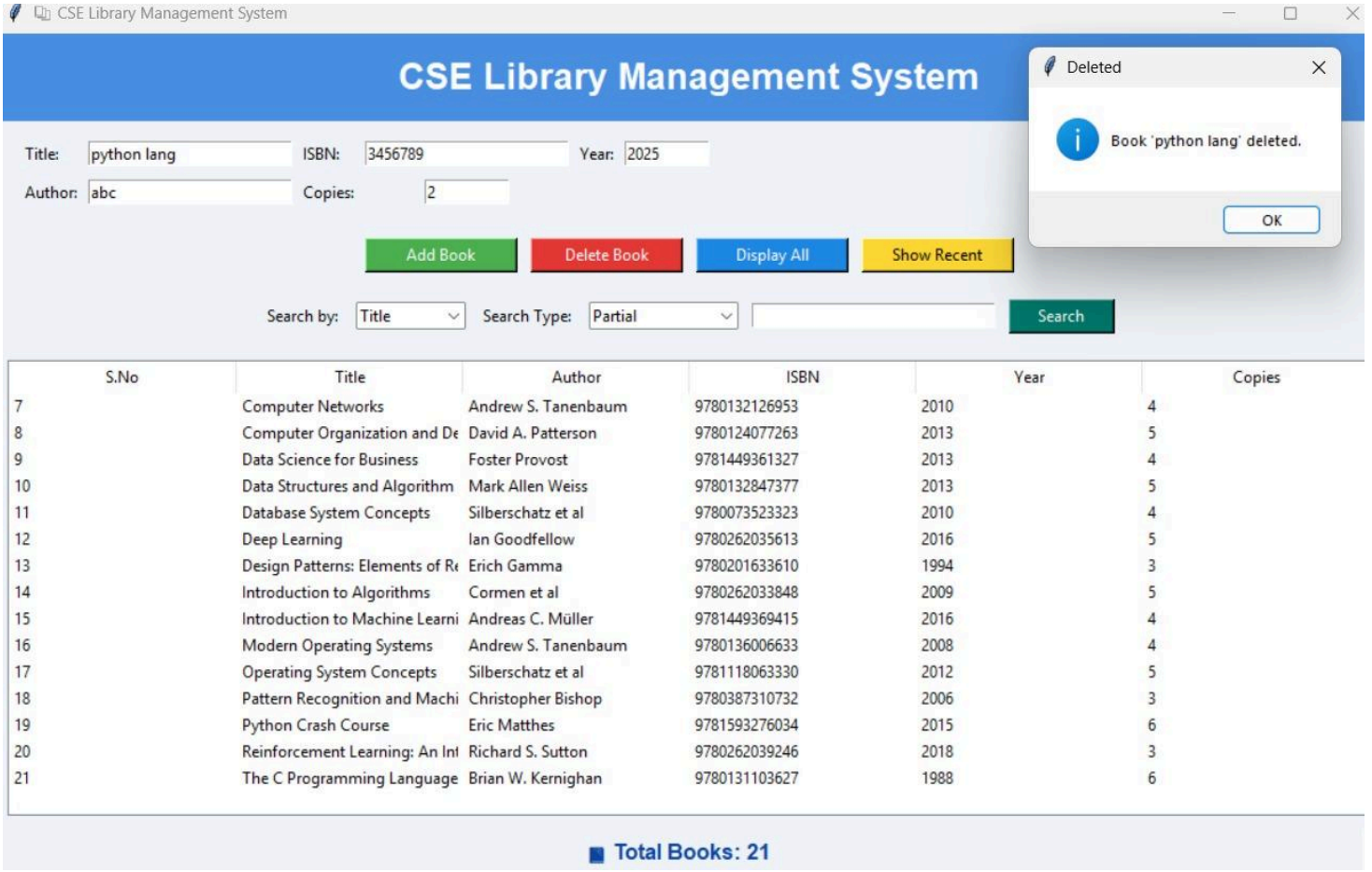
```
    if not found:
        messagebox.showinfo("No Results", "No books found matching your keyword.")

def display_all():
    refresh_table()

def show_recent():
    if not recent_books:
        messagebox.showinfo("Recent Books", "No recent books added yet.")
        return
    msg = "\n".join(recent_books)
    messagebox.showinfo("Recently Added Books", msg)
```

Output:




CSE Library Management System
CSE Library Management System
Title: python lang
ISBN: 3456789
Year: 2025
Author: abc
Copies: 2
Add Book
Delete Book
Display All
Show Recent
Search by: Title
Search Type: Partial
Search
Success
Book 'python lang' added successfully!
OK
S.No
Title
Author
ISBN
Year
Copies
1 abc xyz 3
2 Artificial Intelligence and Mach Peter Flach 4
3 Artificial Intelligence: A Moden Stuart Russell 4
4 Clean Code Robert C. Martin 6
5 Compilers: Principles, Techniq Aho et al 3
6 Computer Graphics: Principles John F. Hughes 9780321399526 2013 3
7 Computer Networks Andrew S. Tanenbaum 9780132126953 2010 4
8 Computer Organization and De David A. Patterson 9780124077263 2013 5
9 Data Science for Business Foster Provost 9781449361327 2013 4
10 Data Structures and Algorithm Mark Allen Weiss 9780132847377 2013 5
11 Database System Concepts Silberschatz et al 9780073523323 2010 4
12 Deep Learning Ian Goodfellow 9780262035613 2016 5
13 Design Patterns: Elements of Re Erich Gamma 9780201633610 1994 3
14 Introduction to Algorithms Cormen et al 9780262033848 2009 5
15 Introduction to Machine Learni Andreas C. Müller 9781449369415 2016 4
16 Modern Operating Systems Andrew S. Tanenbaum 9780136006633 2008 4
Total Books: 22


CSE Library Management System
CSE Library Management System
Title: python lang
ISBN: 3456789
Year: 2025
Author: abc
Copies: 2
Add Book
Delete Book
Display All
Show Recent
Search by: Title
Search Type: Exact (Binary)
Computer Networks
Search
S.No
Title
Author
ISBN
Year
Copies
1 Computer Networks Andrew S. Tanenbaum 9780132126953 2010 4
Total Books: 21


CSE Library Management System
CSE Library Management System
Deleted
Book 'python lang' deleted.
OK
Title: python lang
ISBN: 3456789
Year: 2025
Author: abc
Copies: 2
Add Book
Delete Book
Display All
Show Recent
Search by: Title
Search Type: Partial
Search
S.No
Title
Author
ISBN
Year
Copies
7 Computer Networks Andrew S. Tanenbaum 9780132126953 2010 4
8 Computer Organization and De David A. Patterson 9780124077263 2013 5
9 Data Science for Business Foster Provost 9781449361327 2013 4
10 Data Structures and Algorithm Mark Allen Weiss 9780132847377 2013 5
11 Database System Concepts Silberschatz et al 9780073523323 2010 4
12 Deep Learning Ian Goodfellow 9780262035613 2016 5
13 Design Patterns: Elements of Re Erich Gamma 9780201633610 1994 3
14 Introduction to Algorithms Cormen et al 9780262033848 2009 5
15 Introduction to Machine Learni Andreas C. Müller 9781449369415 2016 4
16 Modern Operating Systems Andrew S. Tanenbaum 9780136006633 2008 4
17 Operating System Concepts Silberschatz et al 9781118063330 2012 5
18 Pattern Recognition and Machi Christopher Bishop 9780387310732 2006 3
19 Python Crash Course Eric Matthes 9781593276034 2015 6
20 Reinforcement Learning: An Int Richard S. Sutton 9780262039246 2018 3
21 The C Programming Language Brian W. Kernighan 9780131103627 1988 6
Total Books: 21

Analysis:

Operation	Best Case	Average Case	Worst Case	Space Complexity
Add Book	O(n log n)	O(n log n)	O(n log n)	O(n)
Delete Book	O(1)	O(n)	O(n)	O(1)
Search Book	O(1)	O(log n)	O(n)	O(1)
Display Books	O(n)	O(n)	O(n)	O(1)
Quick Sort	O(n log n)	O(n log n)	$O(n^2)$	O(log n)

Final Overall Time Complexity:

Best Case: O(log n) — direct hit using binary search

Average Case: O(n log n) — sorting + searching + basic operations

Worst Case: O(n) — keyword search or full list traversal

Final Overall Space Complexity:

Best Case: O(n)

Average Case: O(n)

Worst Case: O(n)