Algorithm

Phase 1: Initialization

1. **Define Structures:** Create a Book object containing id, title, author, price, and quantity.
2. **Global Memory:** Initialize an array of 100 Books and an empty array for the cart.
3. **Pre-load Data:** Seed the system with 3 sample books and set bookCount = 3.

Phase 2: Main Control Loop

1. Display the Main Menu (Visitor, Admin, Exit).
2. **If choice is 1:** Enter **Visitor Panel**.
3. **If choice is 2:** Request credentials. If valid, enter **Admin Panel**.
4. **If choice is 0:** Terminate the program.

Phase 3: Visitor Operations

1. **Search:** Compare user input string against title and author using string comparison.
2. **Add to Cart:**
    - Find Book by ID.
    - Verify quantity > 0.
    - Subtract 1 from books[i].quantity.
    - Append ID to cart[].
3. **Remove from Cart:**
    - Find ID in cart[].
    - Add 1 back to the corresponding books[i].quantity.
    - Shift all subsequent items in the cart array one position to the left.
4. **Purchase:** Display total price and reset cartCount to zero.

Phase 4: Admin Operations

1. **Add Book:** Create a new Book object at index bookCount and increment the count.
2. **Delete Book:**
    - Search for Book ID.
    - If found, loop from that index to the end of the array, moving each book at j+1 into position j.
    - Decrement bookCount.

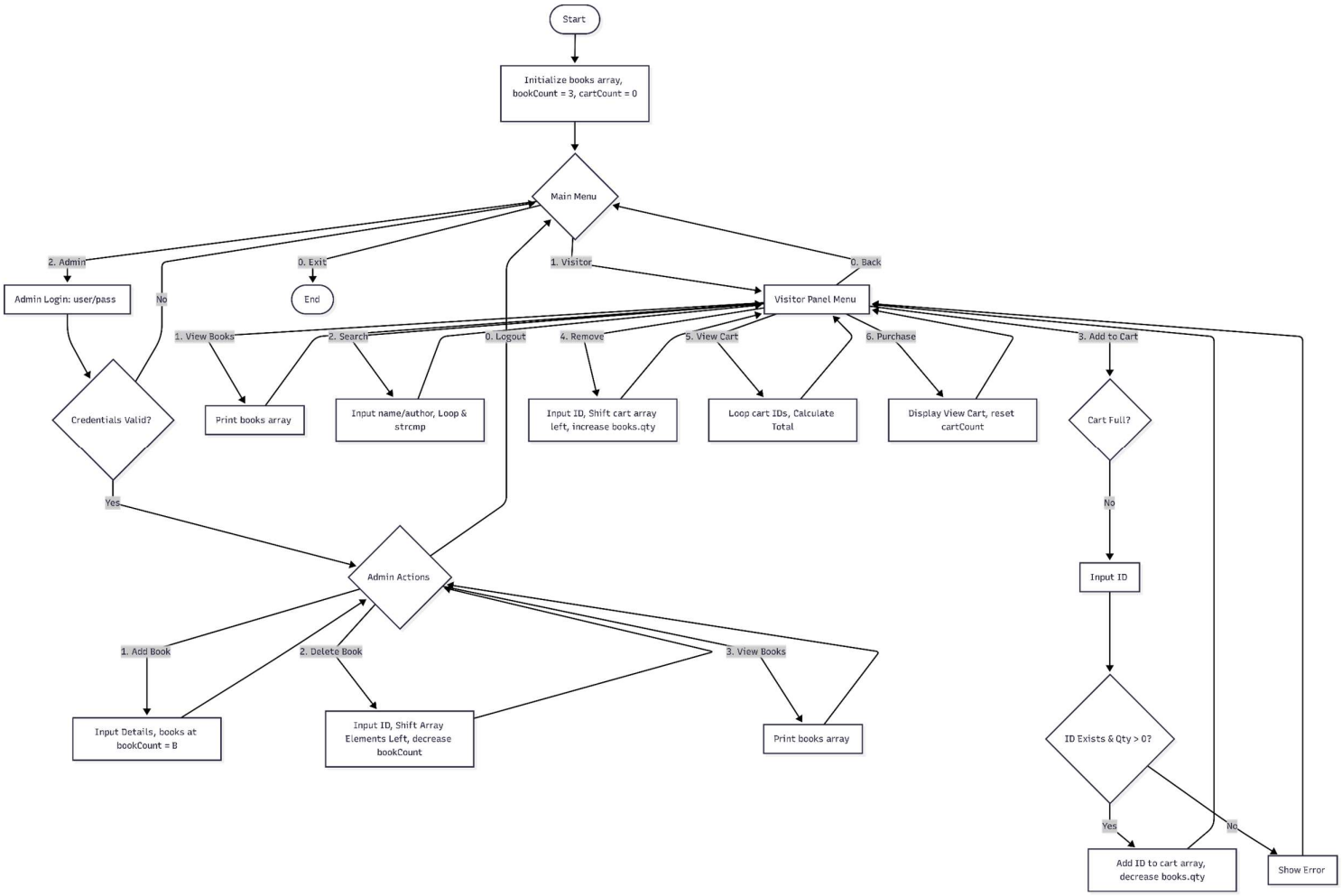
PSEUDO CODE

```
STRUCT Book:
    INTEGER id, quantity
    STRING title, author
    FLOAT price

// GLOBAL VARIABLES
DECLARE books[100] AS Book
DECLARE cart[50] AS INTEGER
DECLARE bookCount = 3, cartCount = 0

// MAIN PROGRAM
BEGIN
    WHILE (TRUE):
        DISPLAY "1. Visitor, 2. Admin, 0. Exit"
        INPUT mainChoice

        IF mainChoice == 1:
            CALL VisitorPanel()
        ELSE IF mainChoice == 2:
            IF AdminLogin() == SUCCESS:
                CALL AdminPanel()
        ELSE IF mainChoice == 0:
            EXIT PROGRAM
    END WHILE
END

// VISITOR LOGIC
FUNCTION VisitorPanel():
    WHILE (TRUE):
        DISPLAY Visitor Options (View, Search, Add, Remove, Cart, Confirm, Back)
        INPUT vChoice

        IF vChoice == 3 (Add to Cart):
            INPUT targetID
            FOR i = 0 TO bookCount - 1:
                IF books[i].id == targetID AND books[i].quantity > 0:
                    cart[cartCount] = targetID
                    cartCount = cartCount + 1
                    books[i].quantity = books[i].quantity - 1
                    PRINT "Added"
                    BREAK

        ELSE IF vChoice == 4 (Remove from Cart):
            INPUT remID
            FOR i = 0 TO cartCount - 1:
                IF cart[i] == remID:
```

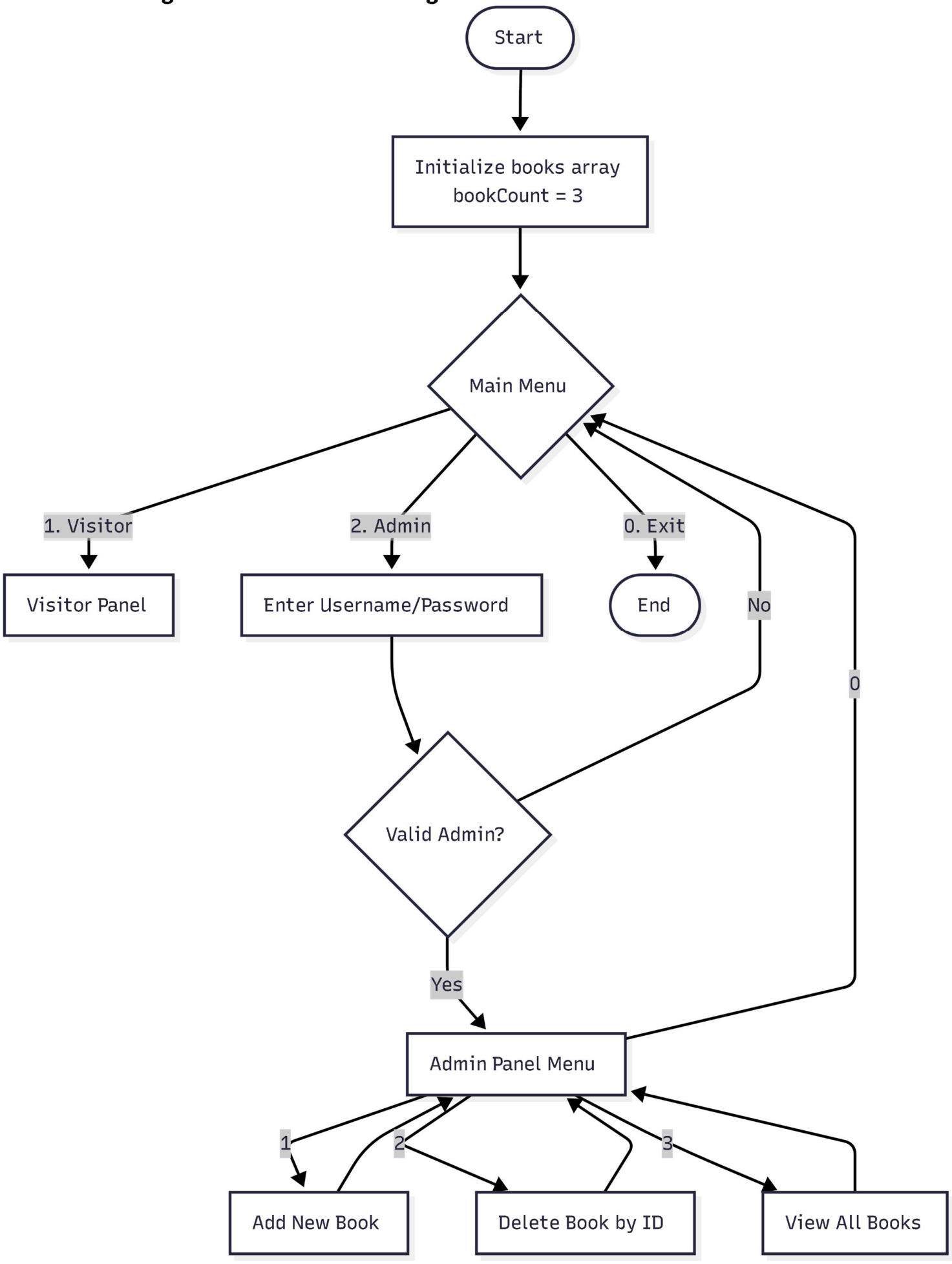
```
                // Restore stock
                FIND book in books[] where id == remID and INCREMENT quantity
                // Shift cart array left
                FOR j = i TO cartCount - 2:
                    cart[j] = cart[j+1]
                cartCount = cartCount - 1
                BREAK

        ELSE IF vChoice == 0:
            RETURN // To Main Menu
    END WHILE

// ADMIN LOGIC
FUNCTION AdminPanel():
    WHILE (TRUE):
        DISPLAY Admin Options (Add, Delete, View, Logout)
        INPUT aChoice

        IF aChoice == 2 (Delete Book):
            INPUT delID
            FOR i = 0 TO bookCount - 1:
                IF books[i].id == delID:
                // Shift books array left to fill the gap
                FOR j = i TO bookCount - 2:
                    books[j] = books[j+1]
                bookCount = bookCount - 1
                PRINT "Deleted"
                BREAK
        ELSE IF aChoice == 0:
            RETURN // Logout
    END WHILE
```

FLOW CHARTS

Master Flowchart



Main Navigation Panel & Admin Logic Flowchart



Visitor Panel & Cart Logic