```
#include <iostream>
#include <string>
#include <vector>
#include <algorithm>

using namespace std;

class Book {
private:
    string title;
    string author;
    int publicationYear;
    bool isAvailable;

public:
    Book(string title, string author, int publicationYear)
        : title(title), author(author), publicationYear(publicationYear), isAvailable(true) {}

    string getTitle() const { return title; }
    string getAuthor() const { return author; }
    int getPublicationYear() const { return publicationYear; }
    bool getAvailability() const { return isAvailable; }

    void setAvailability(bool availability) { isAvailable = availability; }
};

class User {
private:
    string name;
    vector<Book*> borrowedBooks;

public:
    User(string name) : name(name) {}

    string getName() const { return name; }
    vector<Book*> getBorrowedBooks() const { return borrowedBooks; }

    void borrowBook(Book* book) {
        if (book->getAvailability()) {
            book->setAvailability(false);
            borrowedBooks.push_back(book);
        }
    }
```

```
    void returnBook(Book* book) {
        for (auto it = borrowedBooks.begin(); it != borrowedBooks.end(); ++it) {
            if (*it == book) {
                book->setAvailability(true);
                borrowedBooks.erase(it);
                break;
            }
        }
    }
};

class Library {
private:
    vector<Book*> books;
    vector<User*> users;

public:
    void addBook(Book* book) { books.push_back(book); }
    void removeBook(Book* book) {
        for (auto it = books.begin(); it != books.end(); ++it) {
            if (*it == book) {
                books.erase(it);
                break;
            }
        }
    }

    void addUser(User* user) { users.push_back(user); }
    void removeUser(User* user) {
        for (auto it = users.begin(); it != users.end(); ++it) {
            if (*it == user) {
                users.erase(it);
                break;
            }
        }
    }

    vector<Book*> searchByTitle(const string& title) {
        vector<Book*> result;
        for (Book* book : books) {
            if (book->getTitle() == title) {
                result.push_back(book);
            }
        }
```

```
        return result;
    }

    vector<Book*> searchByAuthor(const string& author) {
        vector<Book*> result;
        for (Book* book : books) {
            if (book->getAuthor() == author) {
                result.push_back(book);
            }
        }
        return result;
    }

    vector<Book*> searchByPublicationYear(int year) {
        vector<Book*> result;
        for (Book* book : books) {
            if (book->getPublicationYear() == year) {
                result.push_back(book);
            }
        }
        return result;
    }
};

int main() {
    Library library;

    Book* book1 = new Book("Book1", "Author1", 2020);
    Book* book2 = new Book("Book2", "Author2", 2021);
    Book* book3 = new Book("Book3", "Author3", 2022);

    library.addBook(book1);
    library.addBook(book2);
    library.addBook(book3);

    User* user = new User("User1");
    library.addUser(user);

    user->borrowBook(book1);
    user->borrowBook(book2);

    cout << "Borrowed books by " << user->getName() << ":" << endl;
    for (Book* book : user->getBorrowedBooks()) {
        cout << book->getTitle() << endl;
```

```
    }

    user->returnBook(book1);

    cout << "Available books:" << endl;
    for (Book* book : library.books) {
        if (book->getAvailability()) {
            cout << book->getTitle() << endl;
        }
    }

    delete book1;
    delete book2;
    delete book3;
    delete user;

    return 0;
}
```