

Chương 5: Đồ án tổng hợp

1. Dự Án 1: Quản Lý Thư Viện Sách

Yêu Cầu:

Quản lý sách (thêm, xóa, tìm kiếm, hiển thị).

Sử dụng lớp `Book` và STL containers (vector, map).

Cách Triển Khai:

Bước 1: Xây dựng lớp `Book`:

```
#include <string>using namespace std;
class Book {private:
    string id;    // Mã sách
    string title; // Tên sách
    string author; // Tác giả
    int year;    // Năm xuất bảnpublic:
    Book(string id, string title, string author, int year)
        : id(id), title(title), author(author), year(year) {}
```

```
// Getter methods
string getId() const { return id; }
string getTitle() const { return title; }
string getAuthor() const { return author; }
int getYear() const { return year; }
```

```
void display() const {
    cout << "ID: " << id
        << ", Ten sach: " << title
        << ", Tac gia: " << author
        << ", Nam: " << year << endl;
};
```

Bước 2: Quản lý danh sách sách bằng `vector`:

```
#include <vector>#include <algorithm>
class Library {private:
    vector<Book> books;public:
    void addBook(const Book &book) {
        books.push_back(book);
    }

    void removeBook(const string &id) {
        books.erase(remove_if(books.begin(), books.end(),
            [&id](const Book &b) { return b.getId() == id; }),
            books.end());
    }

    void displayAll() const {
        for (const Book &b : books) {
            b.display();
        }
    }
};
```

```

    }
}

// Tìm sách theo tiêu đề (sử dụng lambda và algorithm)
Book* findBookByTitle(const string &title) {
    auto it = find_if(books.begin(), books.end(),
        [&title](const Book &b) { return b.getTitle() == title; });
    return (it != books.end()) ? &(*it) : nullptr;
};

Bước 3: Chương trình chính:
int main() {
    Library lib;
    lib.addBook(Book("B001", "Dac Nhan Tam", "Dale Carnegie", 1936));
    lib.addBook(Book("B002", "Nha Gia Kim", "Paulo Coelho", 1988));

    lib.displayAll();
    lib.removeBook("B001");
    Book* found = lib.findBookByTitle("Nha Gia Kim");
    if (found) found->display();
    return 0;}

```

2. Dự Án 2: Game Tic-Tac-Toe

Yêu Cầu:

2 người chơi (X và O), nhập tọa độ từ bàn phím.

Kiểm tra thắng thua, hiển thị bảng.

Cách Triển Khai:

Bước 1: Khởi tạo bảng và logic:

```
#include <iostream>using namespace std;
```

```
class TicTacToe {private:
    char board[3][3]; // Bảng 3x3
    char currentPlayer; // X hoặc Opublic:
    TicTacToe() : currentPlayer('X') {
        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {
                board[i][j] = '-';
            }
        }
    }
}
```

```
void printBoard() {
    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            cout << board[i][j] << " ";
        }
        cout << endl;
    }
}
```

```

    }

    bool makeMove(int row, int col) {
        if (row < 0 || row >= 3 || col < 0 || col >= 3 || board[row][col] != '-') {
            return false; // Nước đi không hợp lệ
        }
        board[row][col] = currentPlayer;
        currentPlayer = (currentPlayer == 'X') ? 'O' : 'X';
        return true;
    }

    // Kiểm tra thắng thua
    bool checkWin() {
        // Kiểm tra hàng ngang và dọc
        for (int i = 0; i < 3; i++) {
            if (board[i][0] != '-' && board[i][0] == board[i][1] && board[i][0] == board[i][2]) return true;
            if (board[0][i] != '-' && board[0][i] == board[1][i] && board[0][i] == board[2][i]) return true;
        }
        // Kiểm tra đường chéo
        if (board[0][0] != '-' && board[0][0] == board[1][1] && board[0][0] == board[2][2]) return true;
        if (board[0][2] != '-' && board[0][2] == board[1][1] && board[0][2] == board[2][0]) return true;
        return false;
    }
};

Bước 2: Vòng lặp game:
int main() {
    TicTacToe game;
    int row, col;
    while (true) {
        game.printBoard();
        cout << "Luot cua " << game.getCurrentPlayer() << ":Nhap hang (0-2) va cot (0-2): ";
        cin >> row >> col;
        if (game.makeMove(row, col)) {
            if (game.checkWin()) {
                game.printBoard();
                cout << "Nguoi choi " << game.getCurrentPlayer() << " thang!" << endl;
                break;
            }
        } else {
            cout << "Nuoc di khong hop le!" << endl;
        }
    }
}

```

```
}  
return 0;}
```