#include <iostream>

using namespace std;

class Node {

public:

    string page;

    Node\* next;

    Node(string value) : page(value), next(nullptr) {}

};

class LinkedListStack {

private:

    Node\* top;

public:

    LinkedListStack() : top(nullptr) {}

    void push(string page) {

        Node\* newNode = new Node(page);

        newNode->next = top;

        top = newNode;

        cout << "Visited: " << page << endl;

    }

    void pop() {

        if (top == nullptr) {

            cout << "No history to navigate back to." << endl;

            return;

        }

        Node\* temp = top;

        top = top->next;

        cout << "Navigated back to: " << temp->page << endl;

        delete temp;

    }

    string peek() {

        if (top == nullptr) {

            return "";

        }

        return top->page;

    }

    int isEmpty() {

        return top == nullptr ? 1 : 0;

    }

    ~LinkedListStack() {

        while (top != nullptr) {

            pop();

        }

    }

};

class BrowserHistory {

private:

    LinkedListStack historyStack;

public:

    void addPage(string page) {

        historyStack.push(page);

    }

    void navigateBack() {

        historyStack.pop();

    }

    void viewCurrentPage() {

        string currentPage = historyStack.peek();

        if (!currentPage.empty()) {

            cout << "Current page: " << currentPage << endl;

        } else {

            cout << "No current page." << endl;

        }

    }

    int isHistoryEmpty() {

        return historyStack.isEmpty();

    }

};

int main() {

    BrowserHistory browserHistory;

    int choice;

    string page;

    do {

        cout << "\nMenu:\n";

        cout << "1. Visit Page\n";

        cout << "2. Navigate Back\n";

        cout << "3. View Current Page\n";

        cout << "4. Check if History is Empty\n";

        cout << "5. Exit\n";

        cout << "Enter your choice: ";

        cin >> choice;

        switch (choice) {

            case 1:

                cout << "Enter the page URL: ";

                cin >> page;

                browserHistory.addPage(page);

                break;

            case 2:

                browserHistory.navigateBack();

                break;

            case 3:

                browserHistory.viewCurrentPage();

                break;

            case 4:

                cout << "Is history empty? " << (browserHistory.isHistoryEmpty() == 1 ? "Yes" : "No") <<

endl;

                break;

            case 5:

                cout << "Exiting the program." << endl;

                break;

            default:

                cout << "Invalid choice. Please try again." << endl;

        }

    } while (choice != 5);

return 0;

}