

```
#include <iostream>
#include <stack>
#include <string>

using namespace std;

int main() {
    stack<string> backStack;
    stack<string> forwardStack;
    string current = "";
    string command;

    cout << "Simple Browser History (type 'help' for commands)\n";

    while (true) {
        cout << "\n> ";
        cin >> command;

        if (command == "visit") {
            string url;
            getline(cin, url); // read rest of the line
            if (url.size() > 0 && url[0] == ' ')
                url.erase(0, 1); // remove leading space

            if (url.empty()) {
                cout << "No URL provided.\n";
                continue;
            }
        }

        if (!current.empty())
            backStack.push(current);

        // visiting new page clears forward history
        while (!forwardStack.empty())
            forwardStack.pop();

        current = url;
        cout << "Visited: " << current << "\n";
    }

    else if (command == "back") {
        if (backStack.empty()) {
            cout << "No page to go back to.\n";
            continue;
        }

        forwardStack.push(current);
        current = backStack.top();
    }
}
```

```

backStack.pop();

cout << "Current: " << current << "\n";
}

else if (command == "forward") {
    if (forwardStack.empty()) {
        cout << "No page to go forward to.\n";
        continue;
    }

    backStack.push(current);
    current = forwardStack.top\(\);
    forwardStack.pop();

    cout << "Current: " << current << "\n";
}

else if (command == "show") {
    cout << "\nCurrent page:\n";
    if (current.empty()) cout << "[none]\n";
    else cout << current << "\n";

    // Print back stack
    cout << "\nBack stack (top -> bottom):\n";
    if (backStack.empty()) cout << "[empty]\n";
    else {
        stack<string> temp = backStack;
        while (!temp.empty()) {
            cout << temp.top\(\) << "\n";
            temp.pop();
        }
    }

    // Print forward stack
    cout << "\nForward stack (top -> bottom):\n";
    if (forwardStack.empty()) cout << "[empty]\n";
    else {
        stack<string> temp = forwardStack;
        while (!temp.empty()) {
            cout << temp.top\(\) << "\n";
            temp.pop();
        }
    }
}

else if (command == "help") {
    cout << "Commands:\n";
}

```

```
cout << " visit <url> : Visit a new URL (example: visit https://example.com)\n";
cout << " back      : Go back to the previous page\n";
cout << " forward   : Go forward to the next page\n";
cout << " show      : Show current page and stacks\n";
cout << " exit      : Quit program\n";
}

else if (command == "exit") {
    break;
}

else {
    cout << "Unknown command. Type 'help' for list of commands.\n";
}
}

return 0;
}
```