```
#include <iostream>
#include <cstring>

using namespace std;

class pal {
public:
    int top, length;
    string str;
    char stk[100];

public:
    void accept();
    void cpy_stack();
    void check_pal();
    int calculateLength();
}s;

void pal::accept() {
    cout << "Enter the string to check: ";
    getline(cin,str);
    length = calculateLength();
}


int pal::calculateLength() {
    int len = 0;
    while (str[len] != '\0') {
        len++;
    }
    return len;
}

void pal::cpy_stack() {
    top = -1;

    for (int i = 0; i < length; i++) {
        top++;
        stk[top] = str[i];
    }

    cout << "Reversed string: ";
    for (int j = top; j >= 0; j--) {
        cout << stk[j];
```

```
    }
    cout << endl;
}

void pal::check_pal() {
    int f = 1;
    top = length - 1;

    for (int i = 0; i < length; i++) {
        if (str[i] != stk[top]) {  // Compare original string with stack content
            f= 0;  // If characters do not match, it's not a palindrome
            break;
        }
        top--;  // Move down the stack
    }

    if (f == 1) {
        cout << "The string is a palindrome" << endl;
    } else {
        cout << "The string is not a palindrome" << endl;
    }
}

int main() {


    s.accept();  // Accept the input string
    s.cpy_stack();  // Copy to the stack and print reversed string
    s.check_pal();  // Check if it's a palindrome

    return 0;
}
```