

Verifica palíndromo

1. O código implementa uma função para verificar se uma palavra ou frase é um palíndromo, ou seja, se pode ser lida da mesma forma de trás para frente. O processo começa com a limpeza da entrada, removendo qualquer caractere não alfabético e convertendo as letras para minúsculas. Em seguida, o código utiliza uma pilha para armazenar os caracteres da primeira metade do texto. A segunda metade é então comparada com os caracteres desempilhados, um a um. Se todos os caracteres coincidirem, a entrada é um palíndromo; caso contrário, não é. A abordagem de usar uma pilha permite verificar a simetria de maneira eficiente e clara.

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

using namespace std;

bool verificaPalindromo(const string& texto){
    stack<char> pilha;
    string textoLimpo;

    // Faz uma limpa no texto, mantendo apenas letras e colocando todas em minúsculo
    for(char c : texto){
        if(isalpha(c)){
            textoLimpo += tolower(c);
        }
    }

    // Empilhando a primeira metade do texto
    int tamanho = textoLimpo.size();
    for (int i = 0; i < tamanho / 2; i++){
        pilha.push(textoLimpo[i]);
    }

    // Determinando o ponto de início da comparação
    int pontoDeInicioComparacao = (tamanho % 2 == 0) ? tamanho / 2 : tamanho / 2 + 1;

    // Comparando a segunda metade do texto com os valores da pilha
    for (int i = pontoDeInicioComparacao; i < tamanho; i++){
        if(pilha.top() != textoLimpo[i]){
            return false;
        }
        pilha.pop();
    }
    return true;
}

int main() {
    string texto;
    cout << "Insira uma palavra ou frase: " << endl;
    getline(cin, texto);

    if(verificaPalindromo(texto)){
        cout<<"----- É UM PALÍNDROMO! :D -----" << endl;
    }else{
        cout<<"----- NÃO É UM PALÍNDROMO :C -----" << endl;
    }

    return 0;
}
```