

```

#include <iostream> // necessario para o cout
// #include <locale.h>
#include <string> // necessario para usar recursos de string
#include <windows.h> // necessario para usar sleep
using namespace std; //usado para não precisar std::

/*Luis Eduardo Ribeiro Guerra
   Mini kit (Exemplo de substituir tag) de Dev C++ v0.1 */
/* feito no Dev-C++ 5.11 para livre uso creativo e distribuição*/
/* Contato: luiseduardoribeiroguerra@gmail.com */
/*Github com códigos fontes e atualizações:
   https://github.com/luisrguerra/devcpp-ptbr-mini-kit-cpp
   */
/* este exemplo é um exemplo que faz parte de um kit de c++, procure também fazer download
do kit principal no GitHub para ter acesso a um conteúdo mais completo */

string substituir_tag (string tag, string variavel_da_tag, string string_leitura, string string_substituta){

    //montar automaticamente as strings de tag inicial e final para serem procuradas depois
    string tag_completa = "<" + tag + ">" + variavel_da_tag + "</" + tag + ">";
    //cout <<"tag completa: " << tag_completa << endl;

    // Procurar a posição inicial da tag completa na string que está sendo lida
    size_t posicao_tag_completa = /*string a ser analisada:*/string_leitura .find(/*procurar:*/tag_completa);
    //cout << posicao_tag_completa << endl;

    //checar se há erro
    // Se a tag completa não for encontrada retorna erro
    if ( posicao_tag_completa == string::npos ){
        return "$#erro";
    }

    //posição inicial da tag
    size_t inicio = posicao_tag_completa;
    //posição final dda tag
    size_t final = tag_completa.size();

    //Retornar a string_leitura modificada com a tag substituída
    return string_leitura .replace( inicio, final, string_substituta);

}

string ler_tag(string tag,string string_leitura ){

    //montar automaticamente as strings de tag inicial e final para serem procuradas depois
    string tag_inicial = "<" + tag + ">";
    string tag_final = "</" + tag + ">";
    //cout << tag_inicial << endl;
    //cout << tag_final << endl;

    // Procurar a posição inicial da tag inicial na string que está sendo lida
    size_t posicao_tag_inicial = /*string a ser analisada:*/string_leitura .find(/*procurar:*/tag_inicial);
    //cout << posicao_tag_inicial << endl;

    // Procurar a posição inicial da tag final na string que está sendo lida
    size_t posicao_tag_final = /*string a ser analisada:*/string_leitura .find(/*procurar:*/tag_final);
    //cout << posicao_tag_final << endl;

    //checar se há erro
    // Se a tag inicial ou final não forem encontradas retorna erro
    if ( posicao_tag_inicial == string::npos || posicao_tag_final == string::npos ){
        return "%#erro";
    }

    //posição inicial do texto contido na tag
    size_t inicio = posicao_tag_inicial + tag_inicial.size();
    //posição final do texto contido na tag
    size_t final = posicao_tag_final - tag_inicial.size();

    //Retornar o texto contido dentro tag buscada
    return string_leitura .substr( inicio , final);
}

int main(){
    setlocale(LC_ALL,"Portuguese");

    cout << "Abrindo GitHub..." << endl;
    //Sleep(1400); //dormindo por 1400 milissegundos
    /// Abrir no browser o link do GitHub do Mini Kit de c++
    //system("start https://github.com/luisrguerra/devcpp-ptbr-mini-kit-cpp");

    string string_tag_exemplo = "<formato>txt</formato>";
    string string_tag_exemplo2 = "<texto>olá</texto>";

    cout << "Informações contidas na tag formato: ";

    cout << ler_tag( /*Tag:*/"formato", /*String a ser lida:*/string_tag_exemplo ) << endl;

    cout << "Informações contidas na tag texto: ";
    cout << ler_tag( /*Tag:*/"texto", /*String a ser lida:*/string_tag_exemplo2 ) << endl;

    cout << substituir_tag(/*tag:*/"texto", /*variavel:*/"variavel", /*string a ser lida:*/"olá,<texto>variavel</texto>,bonjour",/*texto substituto:*/

    system("PAUSE"); //Pausar antes de fechar o programa
    return 0;
}

```