

Função / Método	Descrição
<code>length()</code> ou <code>size()</code>	Retorna o tamanho da string (número de caracteres).
<code>empty()</code>	Retorna <code>true</code> se a string estiver vazia.
<code>clear()</code>	Remove todo o conteúdo da string.
<code>at(pos)</code>	Acessa o caractere na posição <code>pos</code> com verificação de limites.
<code>operator[]</code>	Acessa o caractere na posição <code>pos</code> <b>sem</b> verificação de limites.
<code>front()</code>	Retorna o primeiro caractere.
<code>back()</code>	Retorna o último caractere.
<code>append(str)</code>	Adiciona a string <code>str</code> ao final da atual.
<code>push_back(c)</code>	Adiciona o caractere <code>c</code> ao final da string.
<code>pop_back()</code>	Remove o último caractere.
<code>insert(pos, str)</code>	Insere <code>str</code> na posição <code>pos</code> .
<code>erase(pos, len)</code>	Remove <code>len</code> caracteres a partir da posição <code>pos</code> .
<code>replace(pos, len, str)</code>	Substitui parte da string por <code>str</code> , começando em <code>pos</code> e indo até <code>len</code> .
<code>substr(pos, len)</code>	Retorna uma substring de <code>len</code> caracteres a partir de <code>pos</code> .
<code>find(str)</code>	Retorna a posição da primeira ocorrência de <code>str</code> . Retorna <code>npos</code> se não encontrar.
<code>rfind(str)</code>	Retorna a posição da última ocorrência de <code>str</code> .
<code>compare(str)</code>	Compara a string com <code>str</code> . Retorna 0 se forem iguais.
<code>c_str()</code>	Retorna um ponteiro para o array de <code>char</code> (útil para funções C).

biblioteca **<cctype>**, que fornece as funções:

- `toupper(char c)` → converte um caractere para maiúsculo.
- `tolower(char c)` → converte um caractere para minúsculo.

`TEST_CASE("Texto explicando o que está sendo testado") { ... }`

`CHECK(expressão lógica do teste)` → se verdade, então passou no teste

```
#define DOCTEST_CONFIG_IMPLEMENT_WITH_MAIN
#include "doctest.h"
#include "utils.cpp"

TEST_CASE("Testando a função que calcula o maior entre 3 valores") {
    CHECK(maior(5, 7, 1) == 7);
    CHECK(maior(-5, 0, -1) == 0);
    CHECK(maior(-5, -7, -1) == -1);
    CHECK(maior(8, 1, 1) == 8);
    CHECK(maior(0, 0, 0) == 0);
}
```