



STRINGS

TÓPICO 6

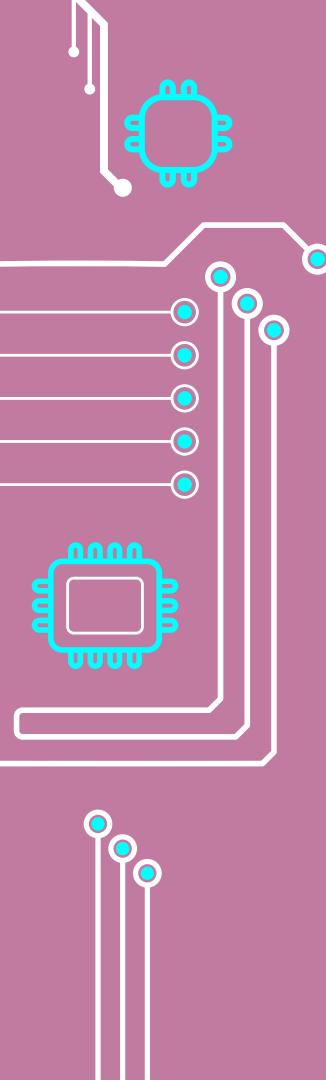


@projetotouuberlandia



Universidade
Federal de
Uberlândia

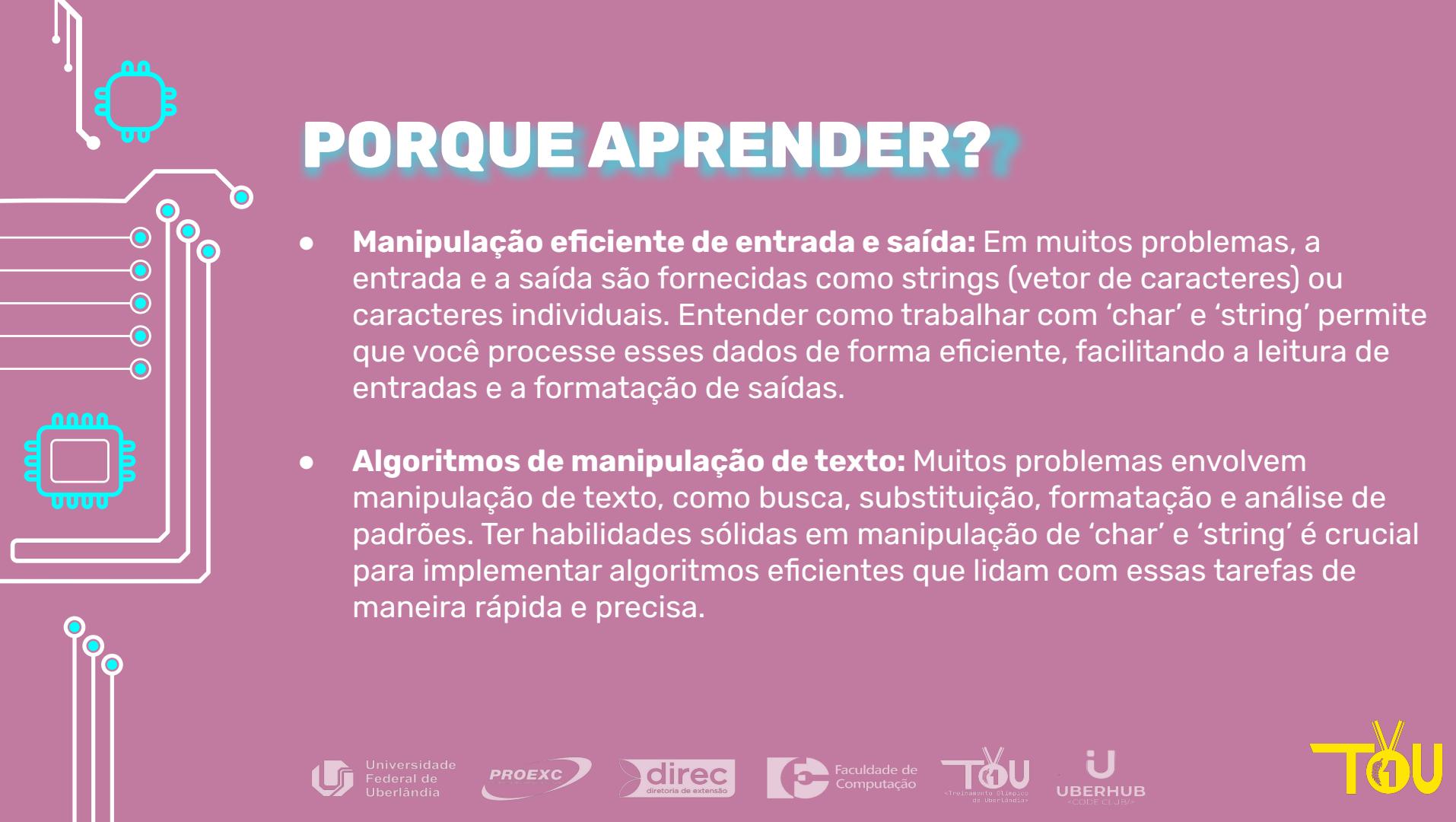




O QUE VAMOS APRENDER?

Neste tópico, vamos aprender sobre **Caracteres** e **Strings**.

- **Caracteres** são unidades individuais de texto.
- **Strings** são sequências de caracteres e são amplamente utilizadas para representar texto em programas.
- Ambos são fundamentais para a manipulação de texto em C++.



PORQUE APRENDER?

- **Manipulação eficiente de entrada e saída:** Em muitos problemas, a entrada e a saída são fornecidas como strings (vetor de caracteres) ou caracteres individuais. Entender como trabalhar com 'char' e 'string' permite que você processe esses dados de forma eficiente, facilitando a leitura de entradas e a formatação de saídas.
- **Algoritmos de manipulação de texto:** Muitos problemas envolvem manipulação de texto, como busca, substituição, formatação e análise de padrões. Ter habilidades sólidas em manipulação de 'char' e 'string' é crucial para implementar algoritmos eficientes que lidam com essas tarefas de maneira rápida e precisa.



APROFUNDANDO...

Caracteres:

- São elementos individuais que compõem o texto. Eles podem ser letras, números, sinais de pontuação, espaços em branco e outros símbolos. Em termos de representação computacional, os caracteres são associados a códigos numéricos e, em C++, eles ficam entre aspas simples “.
- Os caracteres são representados pelo tipo de dado 'char'. Cada caractere é armazenado em um único byte na memória do computador e é associado a um valor numérico de acordo com um conjunto de caracteres específico, como ASCII ou Unicode.



Universidade
Federal de
Uberlândia





ESTRUTURA BÁSICA

```
1 #include <bits/stdc++.h>
2 using namespace std;
3
4 int main(){
5
6     //declaração de um char
7     char letra;
8     //atribuição de valor
9     letra = 'A';
10    //imprimindo na tela
11    cout << "O valor de 'letra' eh: " << letra << endl;
12 }
```



Universidade
Federal de
Uberlândia



direc
diretoria de extensão



TOU
Treinamento Olímpico
de Uberlândia



UBERHUB
(CODE C.I.J.B.)





APROFUNDANDO...

Strings:

- Strings são mais um tipo de dado, como int e double, mas armazenam palavras ou frases.
- Strings possuem diferentes funções próprias desse tipo de dado, sendo capaz de inverter uma palavra ou frase, ordenar uma palavra ou frase de forma crescente ou decrescente, etc.



Universidade
Federal de
Uberlândia





ESTRUTURA BÁSICA

```
#include <bits/stdc++.h>
using namespace std;

int main(){
    string minhaString = "Olá, mundo!";
    cout << minhaString << endl;
}
```



Universidade
Federal de
Uberlândia



<Treinamento Olímpico
de Uberlândia>





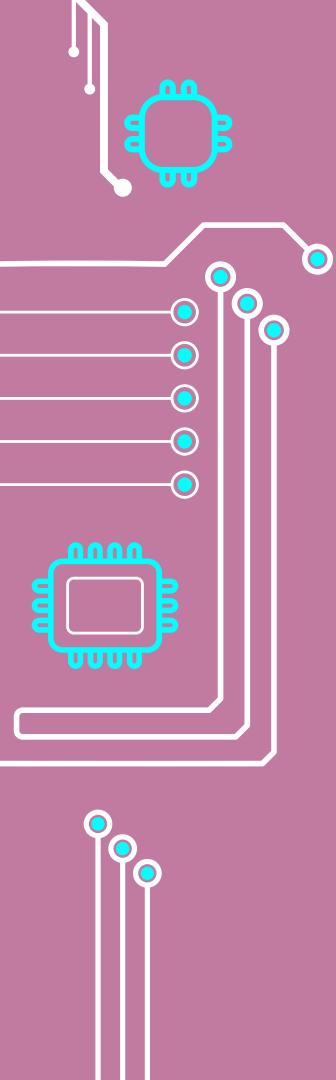
COMANDOS

- `getline()`: lê uma linha completa da entrada padrão. Caso a palavra lida não possua espaços, usamos o `cin` normal. Caso contrário, usamos o `getline`:

```
1 #include <bits/stdc++.h>
2 using namespace std;
3
4 int main(){
5     string linha;
6     getline(cin, linha);
7     cout << "Você digitou: " << linha << endl;
8 }
```

Cuidado: caso você tenha feito a leitura de alguma variável antes do `getline` (mesmo que não seja do tipo `string`), é necessário limpar o buffer de memória. Você pode usar o seguinte comando:

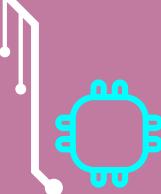
```
cin.ignore();
```



COMANDOS

- `size()`: retorna quantos caracteres tem uma determinada string.

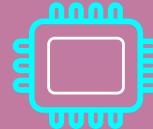
```
1 #include <bits/stdc++.h>
2 using namespace std;
3
4 int main(){
5
6     string minhaString = "Hello, world!";
7     int tamanho = minhaString.size();
8     cout << "O tamanho da string é: " << tamanho << endl;
9 }
```



COMANDOS

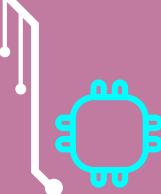
- `+=` : adiciona uma frase (ou outra string) ao final de uma string.

```
1 #include <bits/stdc++.h>
2 using namespace std;
3
4 int main() {
5
6     string A = "Esta eh uma string.";
7     A += " Adicionando outra frase.";
8     cout << A << endl;
9 }
```



Universidade
Federal de
Uberlândia





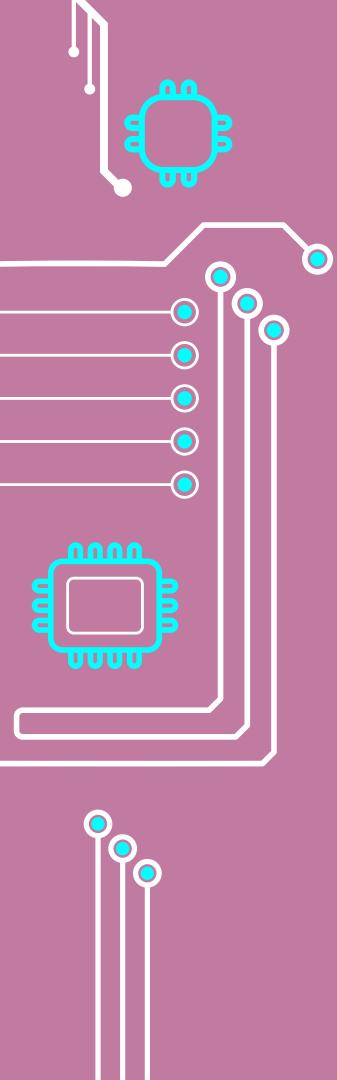
COMANDOS

- `reverse()` : inverte os caracteres da string.

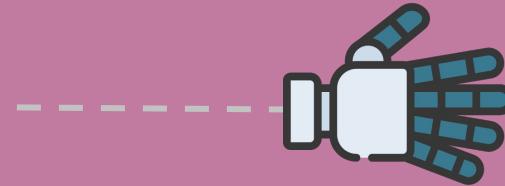
```
1 #include <bits/stdc++.h>
2 using namespace std;
3
4 int main(){
5     string A = "leandro";
6     reverse(A.begin(),A.end());
7     cout << A << endl; // a saída será: ordnael
8 }
```

`begin`: ponteiro para o início da string.

`end`: ponteiro para o fim da string.



MÃO NA MASSA



1 - NepsAcademy Vestibular

<https://neps.academy/br/exercise/160>

2 - NepsAcademy Huaauhahhuahau

<https://neps.academy/br/exercise/118>

3 - NepsAcademy Anagrama

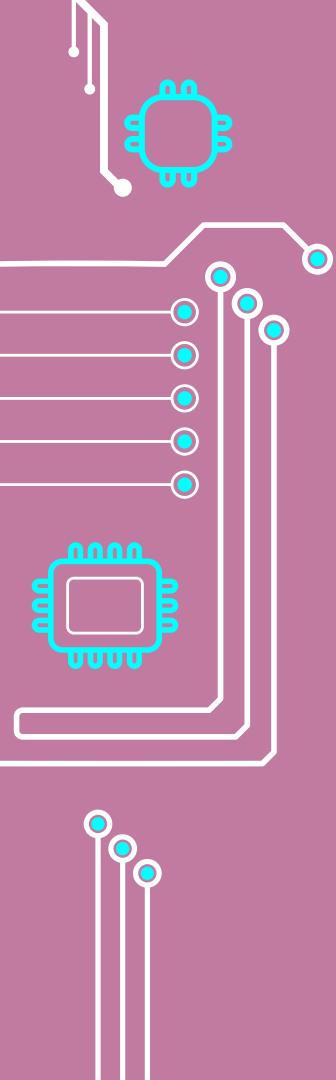
<https://neps.academy/br/exercise/1721>

VAMOS PRATICAR

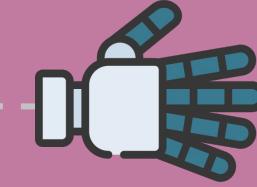


Universidade
Federal de
Uberlândia





MÃO NA MASSA



- Tempo para descomprimir :: <https://neps.academy/br/exercise/1734>
- Subi no Ônibus :: <https://neps.academy/br/exercise/1394>
- Letras (OBI2014) :: <https://neps.academy/br/exercise/388>
- Baralho :: <https://neps.academy/br/exercise/1489>
- Língua do P :: <https://neps.academy/br/exercise/389>
- Palavras Cruzadas :: <https://neps.academy/br/exercise/1013>
- Enigma :: <https://neps.academy/br/exercise/408>
- Próximo Caractere :: <https://neps.academy/br/exercise/2011>
- Flipper :: <https://neps.academy/br/exercise/1736>



Universidade
Federal de
Uberlândia





OBRIGADO



@projetotouuberlandia



Universidade
Federal de
Uberlândia





CREDITS: This presentation template was created by [Slidesgo](#), and includes icons by [Flaticon](#), and infographics & images by [Freepik](#)

