

Linguagem de Programação

Linguagem de Programação C
String

Sumário

1 Definição

2 Strings

3 Manipulação

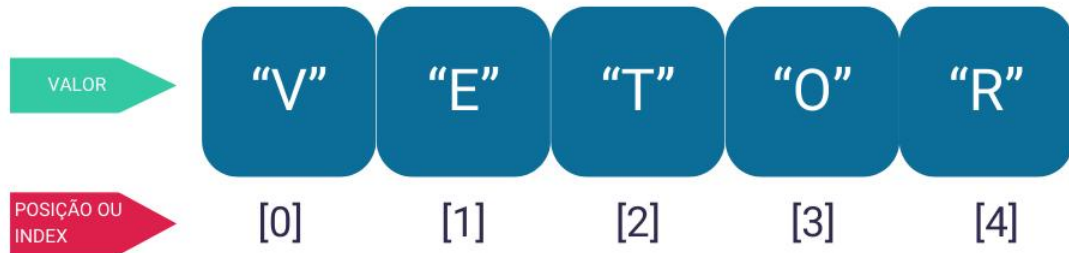
4 Biblioteca

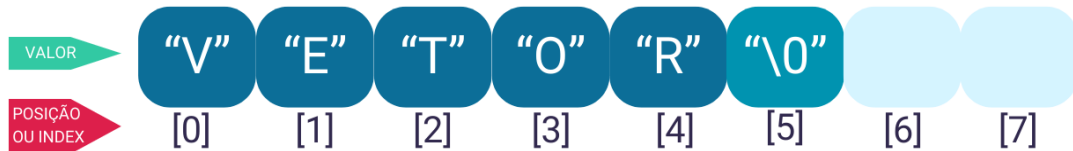
5 Desafio

Definição

O que é uma String?

- › Um vetor de caracteres é uma string quando armazenamos uma **sequência de caracteres** em posições contíguas na memória, terminada com o caractere nulo `\0` para indicar o final da sequência.





Strings

```
1 #include <stdio.h>
2 int main(){
3     char nome_animal[10];
4     printf("Digite o nome de um animal:\n");
5     for (int i = 0; i < 10; i++){
6         scanf(" %c", &nome_animal[i]);
7     }
8     return 0;
9 }
```

Código 1: Exemplo de entrada inadequada de vetor de char.


```
1 #include <stdio.h>
2 int main(){
3     char nome_animal[10];
4     printf("Digite o nome de um animal:\n");
5     scanf(" %c", &nome_animal[i]);
6     for (int i = 0; i < 10; i++){
7         printf("%c", nome_animal[i]);
8     }
9     return 0;
10 }
```

Código 2: Exemplo de saída inadequada de vetor de char.

```
1 #include <stdio.h>
2 int main(){
3     char nome_animal[10];
4     printf("Digite o nome de um animal:\n");
5     scanf("%s", nome_animal);
6     return 0;
7 }
```

Código 3: Exemplo de entrada de string.

```
1 #include <stdio.h>
2 int main(){
3     char palavra[50];
4     scanf("%s", palavra);
5     for(int i=0; palavra[i] != '\0';i++){
6         printf("%c", palavra[i]);
7     }
8     return 0;
9 }
```

Código 4: Exemplo de saída elementos individuais usando \0.

```
1 #include <stdio.h>
2 int main()
3 {
4     char nome_animal[10];
5     printf("Digite o nome de um animal:\n");
6     scanf("%s", nome_animal);
7     printf("%s\n", nome_animal);
8     return 0;
9 }
```

Código 5: Exemplo de saída de string.

Manipulação

```
1 #include <stdio.h>
2 int contar(char *palavra){
3     int i = 0;
4     while(palavra[i] != '\0'){
5         i++;
6     }
7     return i;
8 }
9 int main(){
10     char palavra[100];
11     printf("Digite uma palavra:\n");
12     scanf("%s", palavra);
13     printf("Total de caracteres: %d\n", contar(palavra));
14     return 0;
15 }
```

Código 6: Exemplo de contagem de caracteres.

```
1 #include <stdio.h>
2 int comparar(char *palavra_a, char *palavra_b){
3     int i = 0;
4     while(palavra_a[i] != '\0' && palavra_b[i] != '\0'){
5         if(palavra_a[i] != palavra_b[i])
6             return 0;
7         i++;
8     }
9     return palavra_a[i] == '\0' && palavra_b[i] == '\0';
10 }
11 int main(){
12     char palavra_a [100], palavra_b[100];
13     scanf("%s %s", palavra_a, palavra_b);
14     if (comparar(palavra_a,palavra_b)){
15         printf("Eh igual!\n");
16         return 0;
17     }
18     printf("Nao eh igual!\n");
19     return 0;
20 }
```

Código 7: Exemplo de comparação.

```
1 #include <stdio.h>
2 void copiar(char *origem, char *destino){
3     int i = 0;
4     while(origem[i] != '\0'){
5         destino[i] = origem[i];
6         i ++;
7     }
8     destino[i] = '\0';
9 }
10 int main(){
11     char origem [10], destino[10];
12     scanf("%s %s", origem, destino);
13     copiar(origem , destino);
14     return 0;
15 }
```

Código 8: Exemplo de cópia.

Biblioteca

- › Existem funções que facilitam a manipulação de strings.
- › A biblioteca `string.h` fornece funções para contar caracteres, copiar, concatenar strings, entre outras operações.

```
1 #include <stdio.h>
2 #include<string.h>
3 int main(){
4     char palavra[100];
5     printf("Digite uma palavra:\n");
6     scanf("%s", palavra);
7     int quant_caracteres = strlen(palavra);
8     printf("Total de caracteres: %d\n", quant_caracteres);
9     return 0;
10 }
```

Código 9: Exemplo de uso de strlen().

```
1 #include <stdio.h>
2 #include <string.h>
3 int main(){
4     char palavra_a [10], palavra_b[10];
5     scanf("%s %s", palavra_a, palavra_b);
6     if (strcmp(palavra_a,palavra_b)){
7         printf("Nao eh igual!\n");
8     }else{
9         printf("Eh igual!\n");
10    }
11    return 0;
12 }
```

Código 10: Exemplo de uso strcmp().

```
1 #include <stdio.h>
2 #include <string.h>
3 int main(){
4     char origem [10], destino[10];
5     scanf("%s %s", origem, destino);
6     strcpy(destino,origem);
7     return 0;
8 }
```

Código 11: Exemplo de uso strcpy().

```
1 #include <stdio.h>
2 #include <string.h>
3 int main() {
4     char origem[50] = "Olá, ";
5     char destino[50] = "mundo!";
6     strcat(origem, destino);
7     printf("%s\n", destino);
8     return 0;
9 }
```

Código 12: Exemplo de uso strcat().

Função	Descrição
<code>strcpy(dest, src)</code> <code>strlen(str)</code>	Copia a string <code>src</code> para <code>dest</code> . Retorna o tamanho da string <code>str</code> , sem contar o caractere nulo <code>\0</code> .
<code>strcmp(str1, str2)</code>	Compara duas strings lexicograficamente. Retorna 0 se forem iguais, um valor negativo se <code>str1</code> for menor e um valor positivo se <code>str1</code> for maior.
<code>strcat(dest, src)</code>	Concatena (adiciona) a string <code>src</code> ao final de <code>dest</code> .

Tabela 1: Funções de Manipulação de Strings.

Desafio

- Crie um programa que recebe como entrada uma palavra e, em seguida verifica se essa palavra é **palíndromo**.
- A verificação deve ser feita em uma função.

(Deitel; Deitel, 2011) - Capítulo 8





DE OLIVEIRA, J.F.; MANZANO, J.A.N.G. **Algoritmos: Lógica para Desenvolvimento de Programação de Computadores**. 16. ed. São Paulo: Editora Érica, 2004.

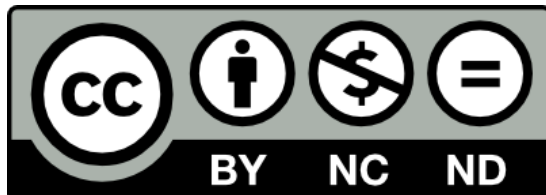


DEITEL, Paul; DEITEL, Harvey. **C: Como Programar**. 6. ed. São Paulo: Pearson Universidades, 2011.



SCHILDT, Herbert. **C Completo e Total**. 4ª. São Paulo: McGraw-Hill Brasil, 2015.

Estes slides estão protegidos por uma licença Creative Commons



Este modelo foi adaptado de Maxime Chupin.

Linguagem de Programação

Linguagem de Programação C
String