
Algoritmos e Programação II

Cadeias de caracteres (Strings)

Prof. Viviane Bonadia dos Santos

“Um vetor é uma coleção de variáveis do **mesmo tipo** que compartilham um **mesmo nome**.”

- Cada variável é diferenciada por meio de um número chamado **índice**.
- Os elementos do vetor são alocados sequencialmente na memória.

Vetores

```
1 #include <stdio.h>
2 void main(){
3
4     //<tipo><nome da variavel><tamanho>
5     int vetorInt[5] = {1, 2, 3, 4, 5};
6
7     float vetorFloat[3] = {1.1, 2.2, 3.3};
8 }
```

vetorInt	1	2	3	4	5
índice	0	1	2	3	4

vetorFloat	1.1	2.2	3.3
índice	0	1	2

vetorInt[3] = 4

vetorFloat[2] = 3.3

Cadeias de caracteres (Strings)

- *String* é um vetor do tipo **char**, que armazena um texto formado por caracteres;
- Cada caractere ocupa um byte de memória (armazenados em sequência);
- Uma *string* sempre termina com o caractere **\0**;
- O tamanho mínimo deste vetor é o comprimento do texto + 1.

vetorChar[12]

índice

O	L	A		M	U	N	D	O	\0		
0	1	2	3	4	5	6	7	8	9	10	11

Declaração

```
1 void main(){  
2  
3     char string1[10] = "oi mundo";  
4     char string2[] = {'o','i',' ','m','u','n','d','o'};  
5     char string3[] = "oi mundo";  
6 }
```

- O caractere `\0` é colocado automaticamente no final da *string*.

MANIPULAÇÃO DE STRINGS

A função `printf()`

```
1 #include <stdio.h>
2
3 void main(){
4
5     char disciplina[15] = "Algoritmos II";
6
7     printf("%s\n", disciplina);
8     printf("%c\n", disciplina[0]);
9 }
```

```
$ gcc strings.c -o strings
```

```
$ ./strings
```

```
Algoritmos II
```

```
A
```

Leitura

A função `scanf()`

- Entende um espaço em branco como término da palavra.

```
1 #include <stdio.h>
2
3 void main(){
4
5     char nome[30];
6     printf("Digite seu nome: ");
7     scanf("%s", nome); //nao usa o operador &
8     printf("Ola %s\n", nome);
9 }
```

```
$ gcc strings.c -o strings
```

```
$ ./strings
```

```
Digite seu nome: Viviane Bonadia
```

```
Ola Viviane
```


Leitura

A função `scanf()`

- Usando `%[^\n]s` podemos ler uma cadeia de caracteres até a quebra de linha (`\n`).

```
1 #include <stdio.h>
2 void main(){
3     char nome[30];
4     printf("Digite seu nome: ");
5     scanf("%[^\n]s", nome); //atencao ao tamanho do
    vetor!
6     printf("Ola %s\n", nome);
7 }
```

```
$ gcc strings.c -o strings
```

```
$ ./strings
```

```
Digite seu nome: Viviane Bonadia
```

```
Ola Viviane Bonadia
```

Atribuição

```
1 #include <stdio.h>
2 void main(){
3     char string[15];
4     string = "Atribuicao incorreta!";
5     printf("%s", string);
6 }
```

```
$ gcc strings.c -o strings
```

```
$ ./strings
```

```
error: incompatible types when assigning to type 'char[15]' from type  
'char *'
```

Atribuição

A função `strcpy()`

- É preciso adicionar a biblioteca `string.h`

```
1 #include <stdio.h>
2 #include <string.h>
3 void main(){
4     char string[15];
5     strcpy(string, "Atribuicao correta!");
6     printf("%s\n", string);
7 }
```

```
$ gcc strings.c -o strings
```

```
$ ./strings
```

```
Atribuicao correta!
```

OUTRAS FUNÇÕES DE MANIPULAÇÃO DE STRINGS...

Número de caracteres de uma string

A função `strlen()`

- Definida na biblioteca `string.h`

```
1 #include <stdio.h>
2 #include <string.h>
3 void main(){
4     int num;
5     char string[] = "Queremos intervalo!";
6     num = strlen(string); //retorna a quantidade de
    caracteres
7     printf("%d\n", num);
8 }
```

```
$ gcc strings.c -o strings
```

```
$ ./strings
```

```
19
```

Número de caracteres de uma string

Como poderia ser a implementação de uma função que conta o número de caracteres de uma string?

Número de caracteres de uma string

Como poderia ser a implementação de uma função que conta o número de caracteres de uma string?

```
1 #include <stdio.h>
2 int strlenImplementacao (char string[]) {
3     int k;
4     for (k = 0; string[k] != '\0'; ++k) ;
5     return k;
6 }
7
8 void main(){
9     int num;
10    char string1[] = "Algoritmos";
11
12    num = strlenImplementacao(string1);
13    printf("%d\n", num);
14 }
```

Comparando strings

A função `strcmp()`

- Retorna 0 se as strings forem iguais;
- um número < 0 se `string1` for alfabeticamente menor que `string2`;
- um número > 0 se `string1` for alfabeticamente maior que `string2`;

```
1 #include <stdio.h>
2 void main(){
3     int num;
4     char string1[] = "Algoritmos";
5     char string2[] = "dois";
6
7     num = strcmp(string1, string2);
8     printf("%d\n", num);
9 }
```

```
$ gcc strings.c -o strings
$ ./strings
-35
```


Concatenando strings

A função `strcat()`

- Definida na biblioteca `string.h`

```
1 #include <stdio.h>
2 #include <string.h>
3 void main(){
4     int num;
5     char string1[] = "Algoritmos";
6     char string2[] = "2";
7     strcat(string1, string2);
8     printf("%s\n", string1);
9 }
```

```
$ gcc strings.c -o strings
```

```
$ ./strings
```

```
Algoritmos2
```

EXERCÍCIOS

Exercícios

- 1 Como poderia ser a implementação da função **strcmp()** que compara duas strings?
- 2 Como poderia ser a implementação da função **strcat()** que concatena duas strings?
- 3 Qual a diferença entre **"A"** e **'A'**?
- 4 Escreva uma função que decide se uma string é ou não um palíndromo (ou seja, se o inverso da string é igual a ela).