## Cadeia de caracteres: Strings Algoritmos e Estrutura de Dados I

Instituto de Engenharia – UFMT

#### Roteiro

- Objetivos
- 2 Introdução
- Cadeia de caracteres (strings)
- 4 Exemplos



## **Objetivos**

#### Esta aula tem como objetivos:

- Apresentar os conceitos básicos sobre cadeias de caracteres;
- Explicitar os exemplos de funções que manipulam cadeias de caracteres;
- 3 Exemplificar a execução desses algoritmos.

- Um caracter é considerado um tipo de dado primitivo.
- Um tipo de dado é primitivo se o computador possui instruções em linguagem de máquina que permite a manipulação deste tipo.
- Desde que uma cadeia é uma sequência de caracteres, o caracter é a entidade fundamental de manipulação de uma cadeia.

- Um caracter pertence a um conjunto finito de caracteres: um alfabeto.
- Um exemplo de alfabeto é o conjunto de letras da língua portuguesa.
- Outro alfabeto comum é o conjunto de dígitos decimais.
- Ao longo dos anos, vários alfabetos foram desenvolvidos para serem utilizados em computadores.

- Caracteres literais são representados por aspas simples, como em 'A' ou 'a'.
- Variáveis do tipo char podem receber valores literais do tipo caracter ou também valores inteiros (que nesse caso representam o caracter correspondente, conforme o sistema de condificação adotado)
- Variáveis do tipo char podem também ter o seu valor comparado com inteiros.

- Entre os vários métodos de codificação, os mais populares são:
  - Código ASCII (7 bits) American Standard Code for Information Interchange.
  - ▶ Código EBCDIC (8 bits) Extended Binary Coded Decimal Interchange Code.
  - Código UNICODE (8 bits).
  - ► Código UTF-8, nos dias de hoje substitui o sistema ASCII.

## Tabela ASCII

	0	1	2	3	4	5	6	7	8	9
30			sp	!	"	#	\$	용	&	′
40	(	)	*	+	,	-		/	0	1
50	2	3	4	5	6	7	8	9	:	;
60	<	=	>	?	9	А	В	С	D	E
70	F	G	Н	I	J	K	L	М	N	0
80	P	Q	R	S	Т	IJ	V	W	Х	Y
90	Z	[	\	]	^		`	а	b	С
100	d	Ф	f	g	h	i	j	k	1	m
110	n	0	р	q	r	S	t	u	V	W
120	х	У	Z	{		}	~			

- Cadeia de caracteres (string): são sequências de letras, números ou símbolos onde o último caracter é o caracter nulo (\0).
- Na linguagem C utilizamos vetores do tipo char para armazenar cadeia de caracteres (strings).
- Obrigatoriamente o último caracter é o caracter nulo.
- O caracter nulo serve para manipular a string sem correr o risco de acessar outras áreas de memória.

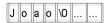
• Por exemplo, para declarar um espaço de memória que contenha 20 caracteres, fazemos:

```
char nome[20];
```

• Inicializamos o vetor de caracteres da seguinte forma:

```
char nome[20] = {'J','o','a','o','\0'};
char nome[20] = "Joao";
char nome[] = "Joao";
```

• Dessa forma, teremos:



• Ao utilizar a função scanf() utilizando cadeia de caracteres, utiliza-se o código "%s" para leitura, omitindo o &, que é utilizado em outros tipos. Por exemplo:

```
printf("Digite o seu nome: ");
scanf("%s", nome);
printf("Seu nome eh %s.",nome);
```

## Exemplo

Escreva um programa que leia uma palavra da entrada e imprima o número de caracteres desta palavra.

```
#include<stdio.h>
int main() {
 char vetor[100];
 int i, n;
 printf("Digite uma palavra:");
 scanf("%s",vetor);
 i=0;
 while (vetor[i]!='\0') {
   i++:
 printf("A quantidade de caracteres eh %d", i);
 return 0;
```

#### Exemplo

Faça uma função que receba uma string e um caracter e verifica se o caracter esta contido na string. Caso o caracter seja encontrado, a funcao retorna a posicao em que o caracter se encontra na string. Caso contrario, retorna -1.

```
int buscaCaracter(char string1[], char caracter) {
 int i = 0;
 while (string1[i]!='\0' && string1[i]!=caracter) {
     i++:
 // Verifica se o contador i chegou no fim da string
 if (i==tamanhoString(string1)) {
   return -1;
 // Retorna posicao da string
 return i:
```

### Exemplo

Faça uma função que receba uma string imprima cada caracter dessa string separado por um hífen.

```
void imprimeString(char string[]) {
  int i = 0;
  while (string[i]!='\0') {
    printf("%c",string[i]);
    if (string[i+1]!='\0')
       printf("-");
    i++;
  }
  printf("\n");
}
```

### Exemplo

Faça uma função que receba duas strings e troque a primeira letra da primeira string com a última letra da segunda string.

```
void trocaCaracteres(char string1[], char string2[]) {
  int i = 0, aux = 0;
  // Calcula o tamanho da segunda string
  while (string2[i]!='\0')
    i++;
  aux = string1[0];
  string1[0] = string2[i-1];
  string2[i-1] = aux;
}
```

## Dúvidas

Fim

# Fim

