

#### **UNIPAC**

#### Universidade Presidente Antônio Carlos

Bacharelado em Ciência da Computação

# Introdução a Programação Material de Apoio

Parte VII - String

Prof. Nairon Neri Silva naironsilva@unipac.br

 $1^{\circ}$  sem / 2020

## String

 String é uma sequência de caracteres utilizada para o armazenamento de texto.

 Na linguagem C strings são vetores de caracteres que possuem um caracter que indica o término de seu conteúdo, o caracter nulo '\0' (contrabarra zero).

## String

 Declaração de strings: Como a string possui o caracter nulo para delimitar o final do seu conteúdo, o tamanho da string deve ser definido com um caracter a mais do que será efetivamente necessário.

## String

#### • Sintaxe:

```
char <nome_variavel>[tamanho+1];
```

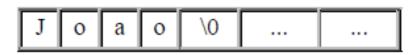
#### Exemplo:

```
char vetc [6];
```

vetc é um vetor de caracteres (string) de tamanho 6. Pode receber uma palavra de no máximo 5 letras.

## Exemplo

 Vamos supor que declaremos uma string de 7 posições e coloquemos a palavra João nela.
 Teremos:



 No caso acima, as duas células não usadas têm valores indeterminados. Isto acontece porque o C não inicializa variáveis, cabendo ao programador esta tarefa.

## Leitura de Strings

- Se quisermos ler uma string fornecida pelo usuário podemos usar a função gets ().
- A função gets () coloca o terminador nulo na string, quando você aperta a tecla "Enter".

```
#include <stdio.h>
void main ()
{
  char string[100];
  printf ("Digite uma string: ");
  gets (string);
  printf ("\n\nVoce digitou %s", string);
}
```

# Leitura de Strings – fflush()

 A função fflush(stdin) serve para limpar o buffer do teclado para que outros caracteres armazenados no buffer do computador sejam liberados. Desta forma a leitura de outra string ou caracter que acontece logo após a sua execução não ficará prejudicada.

# Acesso às posições

 Como as strings são vetores de caracteres, para se acessar um determinado caracter de uma string, basta "indexarmos", ou seja, usarmos um índice para acessarmos o caracter desejado dentro da string.

 Suponha uma string chamada str. Podemos acessar a segunda letra de str da seguinte forma:

## Acesso às posições

 Para isto, basta você lembrar-se que o índice sempre começa em zero. Assim, a primeira letra da string sempre estará na posição 0. A segunda letra sempre estará na posição 1 e assim sucessivamente.

# Manipulação de Strings

 Não se deve atribuir um valor à uma string utilizando o operador de atribuição (=).

```
string1=string2; /* NAO faca isto */
```

- As strings devem ser igualadas elemento a elemento, e para isso existem várias funções já implementadas.
- As funções apresentadas posteriormente estão no arquivo cabeçalho string.h.

#### strcpy

- Sua forma geral é: strcpy (string\_destino, string\_origem);
- A função strcpy() copia a string-origem para a string- destino.

```
#include <stdio.h>
#include <string.h>
void main ()
char str1[100], str2[100], str3[100];
printf ("Entre com uma string: ");
gets (str1);
strcpy (str2, str1);
strcpy (str3, "Voce digitou a string");
printf ("\n\n%s%s", str3, str2);
```

#### strcat

• A função strcat() tem a seguinte forma geral:

```
strcat (string_destino, string_origem);
```

 A string de origem permanecerá inalterada e será anexada ao fim da string de destino. Um exemplo:

```
#include <stdio.h>
#include <string.h>
void main ()
{
   char str1[100], str2[100];
   printf ("Entre com uma string: ");
   gets (str1);
   strcpy (str2, "Voce digitou a string ");
   strcat (str2, str1);
}
```

#### strlen

- Sua forma geral é: strlen (string);
- A função strlen() retorna o comprimento da string fornecida.
   O terminador nulo não é contado. Isto quer dizer que, de fato, o comprimento do vetor da string deve ser um a mais que o inteiro retornado por strlen(). Um exemplo:

```
void main ()
{
int size;
char str[100];
printf ("Entre com uma string: ");
gets (str);
size=strlen (str);
printf ("\n\nA string digitada tem tamanho %d", size);
}
```

#### strcmp

- Sua forma geral é: strcmp (string1, string2);
- A função strcmp() compara a string 1 com a string 2. Se as duas forem idênticas a função retorna zero. Se elas forem diferentes a função retorna não-zero.
- Exemplo no próximo slide.

#### strcmp

```
#include <stdio.h>
#include <string.h>
void main () {
char str1[100], str2[100];
printf ("Entre com uma string: ");
gets (str1);
printf ("Entre com outra string: ");
gets (str2);
  (strcmp(str1, str2) == 0)
    printf ("\n\nAs duas strings sao iguais.");
else
    printf ("\n\nAs duas strings sao diferentes.");
```