



INSTITUTO FEDERAL
Catarinense
Campus São Bento do Sul

LÓGICA DE PROGRAMAÇÃO

EMI Técnico em Informática

Prof. Fernando José Muchalski
Prof. Lucas Dominguez Cordeiro



INSTITUTO FEDERAL
Catarinense
Campus São Bento do Sul

Roteiro

Vetores e Strings



INSTITUTO FEDERAL
Catarinense
Campus Araquari



Conceitos Iniciais

Variáveis do tipo “string”

- Na linguagem C uma cadeia de caracteres (*string*) é implementada como um **vetor** do tipo **char**.
- Variáveis do tipo char são usadas para armazenar um caractere.
 - `char letra = 'a';`





Conceitos Iniciais

Problema:

Muitas vezes precisamos trabalhar com palavras, ou frases; ou seja uma cadeia de caracteres.

- Para resolver isso, usamos uma estrutura de dados especial chamada **vetor de caracteres**.

- `char string[20] = "Aula de hoje";`





Atenção para String.h

```
#include <stdio.h>  
#include <stdlib.h>  
#include <string.h>
```

```
int main () {
```

```
    char string [20]= "Aula de hoje.";
```

```
    printf ("%s", string);
```

```
    return 0;
```

```
}
```



Vetores e Strings

- Na declaração é indicada a quantidade máxima de caracteres que poderá ser armazenada na string.
- Precisamos também saber a real quantidade de caracteres que foi armazenada na string, para isso é preciso uma forma de sinalizar o fim da string.
- Para isso um caractere especial, o '\0' (nulo) é inserido ao final da string. O caractere nulo sucede o último caractere válido da string em questão.

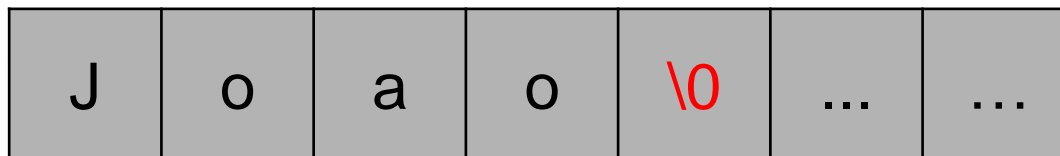


Vetores e strings

Exemplo:

Se inicializarmos a string de 7 posições declarada acima colocando nela a palavra “Joao”, da seguinte forma:

char n [7]=“Joao”;





Vetores e string

Formas de inicialização:

```
char str[7]="Joao";
```

```
char str[]="Joao";
```

```
char str[]={ 'J', 'o', 'a', 'o', '\0' };
```

```
char str[7];  
str[0]='J';  
str[1]='o';  
str[2]='a';  
str[3]='o';  
str[4]='\0';
```

NUNCA

```
str = "Joao"  
str1 = str2;
```




Manipulação de strings

A função **gets** permite a leitura de strings com espaços.

Sintaxe:

```
gets (string);
```

Exemplo:

```
#include <stdio.h>
#include <string.h>
int main () {
    char string[100];
    printf ("Digite o seu nome: ");
    gets (string);
    printf ("\n\n Ola %s\n\n", string);
    return(0);
}
```



Manipulação de strings

A função **puts** permite a impressão de strings na tela.

Sintaxe:

```
puts (string);
```

Exemplo:

```
char string [20] = "Palavra";  
puts (string); // Palavra
```



String.h

A função **strcpy** permite copiar strings (copia a string-origem para a string- destino.)

Sintaxe: `strcpy (destino, origem);`

Exemplo:

```
#include <stdio.h>
#include <string.h>
```

```
int main () {
```

```
    char str1[100],str2[100];
    printf ("Entre com uma string [str1]: ");
    gets (str1);
```

```
    strcpy (str2,str1); /* Copia str1 em str2 */
    printf ("Imprima copia [str2]: %s ",str2);
    return(0);
```

```
}
```



String.h

A função **strcat** permite a concatenação de duas strings.

Sintaxe: `strcat (destino, origem);`

Exemplo:

```
#include <stdio.h>
#include <string.h>
int main () {

    char origem[10]="Inicio", destino[10]="Fim";

    strcat (origem,destino);
    printf ("\n%s",origem);

    return(0);
}
```



String.h

A função **strcmp** permite a comparação de duas strings.

Sintaxe: `strcmp (string1, string2);`

Exemplo:

```
#include <stdio.h>
#include <string.h>
int main (){
    char str1[100],str2[100];
    printf ("Entre com uma string: ");
    gets (str1);
    printf ("\n\n Entre com outra string: ");
    gets (str2);

    if (strcmp(str1,str2))
        printf ("\n\n As duas strings são diferentes.");
    else
        printf ("\n\n As duas strings são iguais."); return(0); }
```



String.h

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**.

Sintaxe: `strlen (string);`

Exemplo:

```
#include <stdio.h>
#include <string.h>
int main () {

    int size;
    char str[100];
    printf ("Entre com uma string: ");
    gets (str);

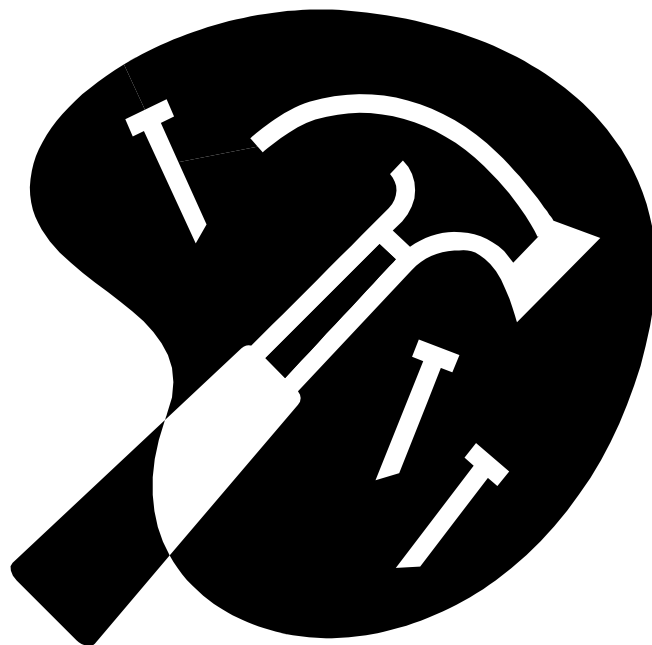
    size=strlen (str);
    printf ("\n\n A string que voce digitou tem tamanho %d",size);

    return(0); }
```



INSTITUTO FEDERAL
Catarinense
Campus São Bento do Sul

E Agora???



Exercícios!!!



INSTITUTO FEDERAL
Catarinense
Campus Araquari



Exercícios

1. Escreva um programa que calcule e escreva a quantidade de vezes que a letra "a" aparece em uma frase fornecida pelo usuário.
2. Escreva um programa para contar quantos espaço em branco existem em uma frase fornecida pelo usuário.
3. Declare uma variável do tipo string em seguida atribua as letras maiúsculas do alfabeto as primeiras vinte e seis posições da variável.
4. Escreva um programa que receba uma frase, conte e imprima o numero de palavras desta frase.
5. Escreva um programa que aceita e valida uma senha com 7 dígitos. Obs: para ser válida a senha de acesso deve ter o valor igual a "tin2019".
6. Escreva um programa que leia o nome de dois alunos e mostre-os em ordem crescente.