



Programação Procedimental

GBC014 – 2015/2

Prof. Renan Cattelan – [www.facom.ufu.br/~renan](http://www.facom.ufu.br/~renan)

# Prática 6

## Strings

# Strings

- ❑ São arrays de caracteres
- ❑ Declaração

```
char <nome_da_string> [ <tamanho> ] ;
```

Exemplo:

```
char str[6] = "teste";
```



# Strings

- Como é um array, permite acesso individual dos elementos (no caso, caracteres individuais)

Ex: `char str[6] = "Teste";`

T	e	s	t	e	\0
---	---	---	---	---	----

`str[0] = 'L';`

L	e	s	t	e	\0
---	---	---	---	---	----

# Strings

## ❑ Leitura

- ❑ `gets(str);`
- ❑ `fgets(str, tamanho, stdin);`
- ❑ `scanf("%s", &str);`
- ❑ `scanf(" %50[^\n]%*c", &str);`

## ❑ Escrita

- ❑ `printf("%s", str);`
- ❑ `fputs (str, stdout);`

# <string.h>

- ❑ `strlen(str);` // retorna o tamanho da string
- ❑ `strcpy(destino, fonte);` // copia fonte para destino
- ❑ `strcmp(str1, str2);` // retorna 0 se strings iguais
- ❑ `strcat(str1, str2);` // concatena str2 em str1

# Exercício 1

- ❑ Digite um nome, calcule e retorne quantas letras tem esse nome.

```
1  #include<stdio.h>
2
3  int main(void)
4  {
5      char nome[50];
6
7      printf("\nDigite o nome: ");
8      scanf("%s", &nome);
9
10     int i = 0;
11     while (i < 50 && nome[i] != '\0') {
12         i++;
13     }
14
15     printf("O tamanho da string eh: %d\n", i);
16
17     system("pause");
18     return 0;
19 }
20
```

# Extra

- E com strlen()?



```
1  #include<stdio.h>
2  #include<string.h>
3
4  int main(void)
5  {
6      char nome[50];
7
8      printf("\nDigite o nome: ");
9      scanf("%s", &nome);
10
11     printf("O tamanho da string eh: %d\n", strlen(nome));
12
13     system("pause");
14     return 0;
15 }
16
```

## Exercício 2

- ❑ Entre com um nome e imprima o nome somente se a primeira letra do nome for 'a' (maiúscula ou minúscula).

```
1      #include<stdio.h>
2
3      int main(void)
4      {
5          char nome[50];
6
7          printf("\nDigite o nome: ");
8          scanf("%s", &nome);
9
10         if (nome[0] == 'a' || nome[0] == 'A') {
11             printf("O nome eh: %s\n", nome);
12         }
13
14         system("pause");
15         return 0;
16     }
17
```

## Exercício 3

- Receba uma palavra e imprima de trás-para-frente.

```
1  #include<stdio.h>
2  #include<string.h>
3
4  int main(void)
5  {
6      char nome[50];
7
8      printf("\nDigite o nome: ");
9      scanf("%s", &nome);
10
11     int i;
12     for (i = strlen(nome)-1; i >=0; i--) {
13         printf("%c", nome[i]);
14     }
15
16     printf("\n");
17     system("pause");
18     return 0;
19 }
20
```

# Exercício 4

- ❑ Dadas duas strings (uma contendo uma frase e outra contendo uma palavra), determine o número de vezes que a palavra ocorre na frase
- ❑ Exemplo:
  - ❑ Para a palavra
    - ❑ ANA
  - ❑ e a frase
    - ❑ ANA E MARIANA GOSTAM DE BANANA
  - ❑ Temos que a palavra ocorre 4 vezes na frase

```
1 #include<stdio.h>
2 #include<stdlib.h>
3 #include<string.h>
4
5 int main(void) {
6     char palavra[5], frase[50];
7
8     printf("\nDigite a palavra: ");
9     gets(palavra);
10    printf("\nDigite a frase: ");
11    gets(frase);
12
13    int i, j, ocorre, numOcorrencias = 0;
14    for (i=0; i<strlen(frase)-strlen(palavra)+1; i++) {
15        if (frase[i]==palavra[0]) {
16            j = 1;
17            ocorre = 1;
18            while (j<strlen(palavra) && ocorre) {
19                if (palavra[j]==frase[i+j])
20                    j++;
21                else
22                    ocorre = 0;
23            }
24            if (ocorre)
25                numOcorrencias++;
26        }
27    }
28    printf("\nA palavra ocorre %d vezes na frase\n", numOcorrencias);
29
30    system("pause");
31    return 0;
32 }
```