

PROGRAMAÇÃO DE COMPUTADORES em C

Prof. Geraldo Pereira de Souza (geraldo@mlink.com.br)

Roteiro Prático

Parte I – Tratamento de texto (string) na linguagem C:

1) Digite, compile, rode e interprete o código fonte abaixo no editor:

```
#include <stdio.h>
#include <stdlib.h>
int main (){
    char ch;
    printf ("\nDigite o caracter:");
    scanf("%c", &ch);
    printf ("\nCaracter digitado: %c", ch);
    printf ("\nFim do programa!!!");
    system("pause");
    return(0);
}
```

2) Digite, compile, rode e interprete o código fonte abaixo no editor:

```
#include <stdio.h>
#include <stdlib.h>
int main (){
    char nome[40];
    printf ("\nDigite seu nome:");
    gets(nome); // Lê uma string
    printf ("\nBoa noite:");
    puts(nome); // Escreve uma string na tela
    printf ("\nFim do programa!!!");
    system("pause");
    return(0);
}
```

3) Digite, compile, rode e interprete o código fonte abaixo no editor:

```
#include <stdio.h>
#include <stdlib.h>
int main (){
    char nome[40];
    printf ("\nDigite seu nome:");
    scanf("%s", nome);
    printf ("\nBoa noite: %s ", nome);
    printf ("\nFim do programa!!!");
    system("pause");
    return(0);
}
```

a) Qual a diferença desse exercício com o proposto no exercício 2? Explique a diferença no resultado da execução e por quê isso acontece?

- 4) Digite, compile, rode e interprete o código fonte abaixo no editor:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main (){
    char nome[40];
    printf ("\nDigite seu nome:");
    gets(nome);
printf ("\nSeu nome tem %d letras", strlen(nome));
    system("pause");
    return(0);
}
```

- 5) Digite, compile, rode e interprete o código fonte abaixo no editor:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main (){
    char nome[40];
    printf ("\nDigite seu nome:");
    gets(nome);
    printf ("\nNome Invertido:");
for (int i=(strlen(nome)-1); i>=0; i--){
        printf("%c", nome[i]);
}
    system("pause");
    return(0);
}
```

- 6) Digite, compile, rode e interprete o código fonte abaixo no editor:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main (){
    char nome[40];
char frase[80] = "Bom dia: ";
    printf ("\nDigite seu nome:");
    gets(nome);
strcat(frase, nome);
    printf ("\n%s", frase);
    system("pause");
    return(0);
}
```

- 7) Digite, compile, rode e interprete o código fonte abaixo no editor:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main (){
    char nome[40];
    char nome2[40];
    printf ("\nDigite seu nome:");
```

```

    gets(nome);
    strcpy(nome2, nome); // copia nome para nome2
    printf ("\nNome: %s", nome);
    printf ("\nCópia de nome: %s", nome2);
    system("pause");
    return(0);
}

```

8) Digite, compile, rode e interprete o código fonte abaixo no editor:

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main (){
    char nome1[40];
    char nome2[40];
    printf ("\nDigite nome 1: ");
    gets(nome1);
    printf ("\nDigite nome 2: ");
    gets(nome2);
    int diferenca = strcmp(nome1, nome2);
    printf ("\nDiferença entre %s e %s = %d",
        nome1, nome2, diferenca);
    if(diferenca==0){
        printf ("\n%s é igual a %s", nome1, nome2);
    }
    else if (diferenca > 0){
        printf ("\n%s precede %s !!!", nome2, nome1);
    }
    else {
        printf ("\n%s é precede %s !!!", nome1, nome2);
    }
    system("pause");
    return(0);
}

```

Parte II - Prática:

1) Faça um programa em C que solicite o nome do usuário. O programa deve então imprimir:

- a) O nome do usuário
- b) O nome do usuário invertido
- c) O número de caracteres contidos no nome do usuário
- d) O número de vogais do nome do usuário
- e) O número de consoantes do nome do usuário

2) Faça um programa em C que faça o seguinte:

- a) Solicite ao usuário que digite uma string.
- b) Logo em seguida o programa deve verificar se a palavra é um palíndromo. Se sim deve exibir uma mensagem para o usuário: PALÍNDROMO!!!

Obs: exemplos de palíndromo:

RIR
RAIAR

O palíndromo é verificado quando a string na ordem inversa e direta são iguais.

3) Faça um programa que pede para o usuário informar o nome para uma variável. A regra para dar nome a uma variável em C é a seguinte:

- a) O primeiro caracter pode ser letra, _ ou \$;
- b) O segundo caracter pode ser qualquer um do grupo a concatenado com número;
- c) Repete-se os caracteres do grupo a ou b;

Caso o nome da variável na esteja correto o programa deve imprimir: nome de variável inválido!

4) Faça um programa que solicite ao usuário para digitar uma frase. Logo em seguida o programa deve imprimir:

- a) Para cada vogal ('a', 'e', 'i', 'o' e 'u') quantas vezes cada vogal aparece na frase;
- b) O número total de vogais
- c) O número total de dígitos (0,1, .., 9): pode haver repetição nesse caso será computado quantas vezes o dígito aparece;
- d) O número total da letras;
- e) Quantos caracteres estão em maiúsculas;
- f) Quantos caracteres estão em minúsculas;

Dica: pesquise sobre as funções: isdigit, isalpha, islower, isupper, etc;