



Algoritmos e Programação II

Prof.^a Noeli A. Pimentel Vaz
Prof. Joilson dos Reis Brito

Agenda da Aula

- Correção de Exercícios
- Vetor de strings
- Exemplos

Lista 4 - Questão 1

1 – Faça um programa que receba uma frase (string), substitua todas as vogais pelo caractere * e converte todas as consoantes para maiúsculo.

Exemplo:

Entrada: fazer exercicios faz bem.

Saída: F*Z*R *X*RC*C**S F*Z B*M.

Solução 1

```
#include <stdio.h>
#include <ctype.h>
#include <stdio.h>
#include <string.h>
#define TAM 30
int main()
{
    char Frase[TAM];
    int l, X;
    printf("Entrada:\n");
    scanf(" %[^\\n]s",Frase);
    for(l=0;Frase[l]!='\\0';l++)
    {
        Frase[l] = toupper(Frase[l]);
        if(Frase[l] == 'A' || Frase[l] == 'E' || Frase[l] == 'I' || Frase[l] == 'O' || Frase[l] == 'U')
        {
            Frase[l] = '*';
        }
    }
    printf("Saida:\n");
    printf("%s\\n",Frase);
    return 0;
}
```

Solução 2

```
#include <stdio.h>
#include <ctype.h>
#include <stdio.h>
#include <string.h>
#define TAM 30
int main()
{
    char Frase[TAM], Vogais[5]={'A','E','I','O','U'};
    int l, X;
    //printf("Entre com uma string\n");
    scanf(" %[^\\n]s",Frase);
    for(l=0;Frase[l]!='\\0';l++)
    {
        Frase[l] = toupper(Frase[l]);
        for(X=0;X<5;X++)
        {
            if(Frase[l] == Vogais[X])
            {
                Frase[l] = '*';
            }
        }
    }
    printf("%s\\n",Frase);
    return 0;
}
```

Lista 4 - Questão 4

Faça um programa que leia uma string e mostre quantas palavras existem nesta string.

```
#include <stdio.h>
#include <ctype.h>
#define TAM 30
#include <stdio.h>
int main()
{
    char Frase[TAM];
    int l, ContadorPalavras = 0, Flag = 0;
    printf("Informe um frase\n");
    gets(Frase);
    l=0;
    while (Frase[l]!='\0')
    {
        Flag ++;
        if(Frase[l] == ' ')
        {
            while (Frase[l] == ' ')
                l++;
            if(l!=0 && Frase[l]!='\0' && Flag!=1 )
                ContadorPalavras++;
        }
        else
            l++;
    }
    if( Flag>=1)
        printf(" Existe(m) %d palavras na frase.\n",ContadorPalavras+1);
    else
        printf("Nao foram digitadas palavras na frase");
    return 0;
}
```

Lista 3 - Questão 9

Desenvolva um programa que leia uma string e um caracter. O programa deve retirar da string todas as ocorrências desse caractere. Imprima a cadeia de caracteres modificada ao final.


```
#include<stdio.h>
#include <locale.h>
#include <string.h>
#define T 100
int main()
{
    setlocale(LC_ALL, "Portuguese");
    char Frase[T],Letra;
    int I, J;
    printf("Digite a string:"); scanf(" %[^\n]s",Frase);
    fflush(stdin);
    printf("Digite o caracter:");scanf("%c",&Letra);
    J = 0;
    for(I=0;I<strlen(Frase);I++)
    {
        if(Frase[I]==Letra)
        {
            for(J=I;J<strlen(Frase);J++)
                Frase[J]=Frase[J+1];
        }
    }
    printf("String sem o caracter %c: %s",Letra,Frase);
    return 0;
}
```

Manipulamos até agora, vetor de caracteres, ou seja, o vetor armazena um caracter em cada posição.

E se fosse necessário armazenar uma palavra em cada posição do vetor?

VETOR DE STRING

Segunda-feira	Programação	Estudos	Universidade	Futuro
0	1	2	3	4

Vetor de String

Para armazenar uma palavra ou um conjunto de caracteres em cada posição do vetor é necessário construir um tipo de dado na Linguagem C.

Para isso vamos usar o comando **typedef**

Vetor de String

Para armazenar uma palavra ou um conjunto de caracteres em cada posição do vetor é necessário construir um tipo de dado na Linguagem C.

Para isso vamos usar o comando **typedef**

Ao usar o comando typedef estamos construindo um tipo de dado.

Criação de tipos de dados

Sintaxe:

```
typedef tipo_primitivo nome_tipo_construído;
```

Exemplo:

```
typedef int inteiro;
```

```
inteiro X;
```

Neste exemplo criamos um apelido para o tipo int.

Criação de tipos de dados - String

Sintaxe:

```
typedef tipo_primitivo nome_tipo_construído[Tamanho];
```

Exemplo:

```
typedef char String[100];
```

```
String VetorNomes[5];
```

Neste exemplo criamos um vetor de strings chamado VetorNomes com 5 posições..

Exemplo 1

Programa que declara um vetor de strings com 6 posições, faz a leitura e a impressão do vetor.

```
#include <stdio.h>
#define T 100 // constante para o tamanho de cada string
#define Y 5 // constante para o tamanho do vetor
```


Exemplo 1

Programa que declara um vetor de strings com 6 posições, faz a leitura e a impressão do vetor.

```
#include <stdio.h>
#define T 100 // constante para o tamanho de cada string
#define Y 5 // constante para o tamanho do vetor

typedef char String[T]; //construção do tipo chamado String
```

Exemplo 1

Programa que declara um vetor de strings com 6 posições, faz a leitura e a impressão do vetor.

```
#include <stdio.h>
#define T 100 // constante para o tamanho de cada string
#define Y 5 // constante para o tamanho do vetor

typedef char String[T]; //construção do tipo chamado String

int main()
{
    String VetorNomes[Y]; //declaração do vetor de Strings com 5 posições
```

Exemplo 1

Programa que declara um vetor de strings com 6 posições, faz a leitura e a impressão do vetor.

```
#include <stdio.h>
#define T 100 // constante para o tamanho de cada string
#define Y 5 // constante para o tamanho do vetor

typedef char String[T]; //construção do tipo chamado String

int main()
{
    String VetorNomes[Y]; //declaração do vetor de Strings com 5 posições
    int I;
    printf("Digite %d Nomes: ", Y);
    for(I=0;I<Y;I++)
    {
        gets(VetorNomes[I]);
    }
}
```

Exemplo 1

Programa que declara um vetor de strings com 6 posições, faz a leitura e a impressão do vetor.

```
#include <stdio.h>
#define T 100 // constante para o tamanho de cada string
#define Y 5 // constante para o tamanho do vetor

typedef char String[T]; //construção do tipo chamado String

int main()
{
    String VetorNomes[Y]; //declaração do vetor de Strings com 5 posições
    int I;
    printf("Digite %d Nomes: ", Y);
    for(I=0;I<Y;I++)
    {
        scanf(" %[^\\n]s",VetorNomes[I]);
    }
    printf("Os nomes digitados foram:\\n");
    for(I=0;I<Y;I++)
    {
        printf("%s\\n",VetorNomes[I]);
    }
    return 0;
}
```

Vetor de String

Com esse recurso de construir um tipo de dado, poderemos criar vetores para armazenar strings e aumentar as possibilidades de problemas que podemos resolver.

Exemplo 2

Programa que declara um vetor de strings com 10 posições, faz a leitura e realiza a busca de um nome no vetor. Suponha que todos os nomes são diferentes.

```
#include <stdio.h>
#include <string.h>
#define T 100
#define TAMVET 6
typedef char String[T];
int main()
{
    String VetorNomes[TAMVET], NomeBusca;
    int I, Indice = -1;
    printf("Digite %d Nomes: ", TAMVET);
    for(I=0;I<TAMVET;I++)
    {
        scanf(" %[^\\n]s",VetorNomes[I]);
    }
    printf("Digite um nome para realizar a busca: ");
    fflush(stdin);
    scanf(" %[^\\n]s",NomeBusca);
    for(I=0;I<TAMVET;I++)
    {
        if(stricmp(NomeBusca,VetorNomes[I]) == 0)
            Indice = I;
    }
    if(Indice == -1)
        printf("O nome buscado nao existe no vetor.\\n");
    else
        printf("%s existe no vetor na posicao %d.\\n",NomeBusca,Indice);
    return 0;
}
```