Strings de Caracteres

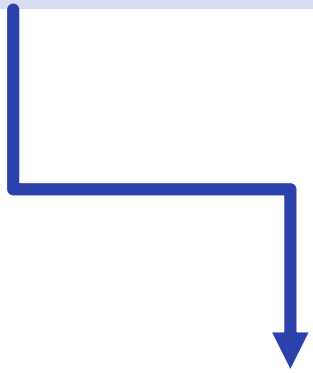
Profa. Elloá B. Guedes

www.elloaguedes.com

UEA
UNIVERSIDADE
DO ESTADO DO
AMAZONAS

Exemplo de String

- **printf(“Adoro programar!!\n”);**



Isso é uma string!

Strings

- Variáveis do tipo char só armazenam um caractere por vez
- Como fazer para armazenar vários caracteres?

- **Solução: Vetores de caracteres (string)**
- **Problemas: A depender do tamanho da palavra, o tamanho do vetor pode variar**

- **Solução 2: Vetores de caracteres (string)**
- **Como fazer: Tamanho do vetor adequado para a palavra a ser armazenada!**

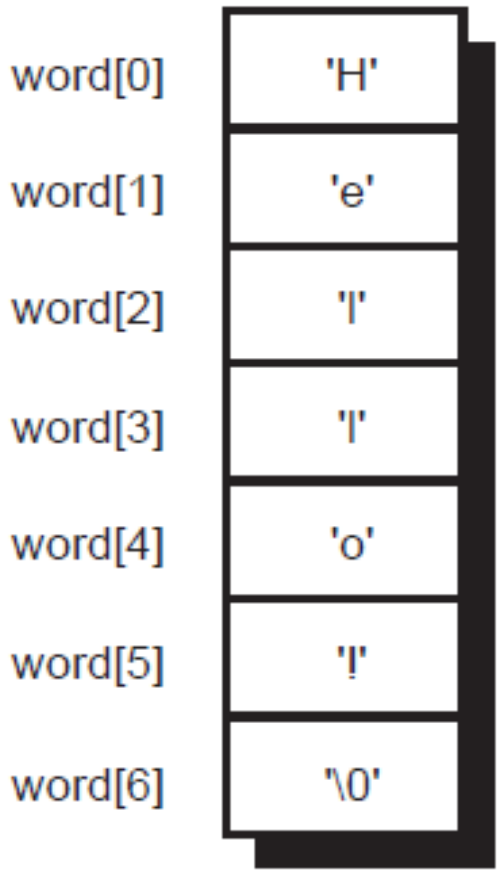
  Marcar o final da entrada

Strings

```
const char  word [] = { 'H', 'e', 'l', 'l', 'o', '!', '\0' };
```

```
char  word[] = { "Hello!" };
```

```
char word[] =  "Hello!";
```



Strings - Exemplo

```c
#include <stdio.h>

void main(){
    char word[] = "Alo Mundo!";
    int i ;

    i = 0;
    while (word[i] != '\0'){
        printf("%c", word[i]);
        i++;
    }
}
```

Strings

- Caractere especial de final da palavra: \0 (contra barra zero)
  - Utilizado para diversos algoritmos com strings
- Impressão com printf e leitura com scanf: %s

Strings - Exemplo

```c
#include <stdio.h>

void main(){
    char word[] = "Alo Mundo!";
    int i ;

    printf("%s\n",word);
    scanf("%s",&word);
    printf("%s",word);
```

String - Comprimento

- O comprimento de uma string é a quantidade de caracteres que ela possui até o \0

- Escreva uma função em C para determinar o comprimento de uma string

String – Exemplo

```
int comprimento(char string[]){
    int i = 0;

    while (string[i] != '\0'){
        i++;
    }

    return i;
}
```

Strings - Concatenação

- A operação de concatenação entre duas strings resulta na justaposição da primeira string com a segunda

- Exemplo: “Hello” concatenado com “World” resulta em “HelloWorld”

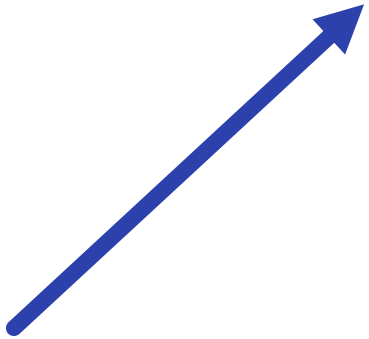
- Faca uma função em C que implemente a concatenação entre duas strings

Strings - Exemplo

```
void concatenacao(char string1[], char string2[],
char string3[]){
    int i = 0, j = 0;

    while (string1[i] != '\0')
        string3[i] = string1[i++];

    while(string2[j++] != '\0')
        string3[i + j] = string2[j];

}
```

Strings - Exemplo

```
void main(){

    char p1[] = "Hello", p2[] = " World!", p3[20];
    concatenacao(p1,p2,p3);

    printf("%s",p3);
}
```



Atenção!!

Strings – Igualdade

- Não é possível verificar a igualdade de duas strings diretamente!!
  - **s1 == s2 é errado!! Não faça isso**!

- Necessário implementar método de comparação de caractere a caractere!

Strings – Exemplo

```
_Bool ehIgual(char s1[],char s2[]){
    int i = 0;

    while (s1[i] != '\0'){
        if (s1[i] != s2[i]) {
            return false;
        }
        i++;
    }

    if (s2[i] != '\0'){
        return false;
    }
    return true;
}
```

Strings – Exemplo

```
void main(){
    char s1[] = "Alo!", s2[] = "Alo!",s3[] = "Alo!.",s4[] = "Festa!";
    printf("%d",ehIgual(s1,s2));
    printf("%d",ehIgual(s1,s3));
    printf("%d",ehIgual(s4,s2));
}
```

Exercícios

- Escreva uma função em C que leia uma linha de caracteres da entrada, com espaços inclusos.

- “Crie uma variável string que armazene tudo isso. Parece fácil, não é?!”

Solução

```
void  readLine (char  buffer[])
{
    char  character;
    int   i = 0;

    do
    {
        character = getchar ();
        buffer[i] = character;
        ++i;
    }
    while ( character != '\n' );

    buffer[i - 1] = '\0';
}
```

Exercícios

- Dada uma string, quantas vogais ela possui?
- Quantos caracteres do tipo ‘espaço em branco’ essa string possui?