

EXERCÍCIOS DE FIXAÇÃO Nº 05

Strings

- 1) Dados uma *string* *s* e um caractere *c* faça um programa que verifique se *s* contém *c*. Dica: a implementação dentro de uma função facilita a definição da solução.
- 2) Modifique o código da questão anterior fazendo com que o programa determine quantas vezes o caractere ocorre na *string*.
- 3) Escreva uma função que implementa o comportamento da função *strcat()*, ou seja, dadas duas strings *str1* e *str2*, a função deve concatenar as duas *strings* e o conteúdo deve ficar em *str1*. Não utilize funções predefinidas. Protótipo:

```
void concatena( char str1[], char str2[] );
```

- 4) Escreva um programa que leia uma *string* e determina se a mesma é palíndrome, ou seja, se forma a mesma sequência de caracteres quando lida de trás para frente. Ex.: ARARA.
- 5) Faça um programa que leia uma cadeia de caracteres e converta todos os caracteres que forem *letras minúsculas* para *letras maiúsculas*. Dica: é preciso fazer uma subtração no código do caractere. Verifique na tabela ASCII e veja qual valor deve ser usado.
- 6) Faça um programa que verifica se uma *string* contém somente dígitos decimais (0 a 9).
- 7) Escreva um programa que converta uma *string* que contém somente dígitos em um valor inteiro (variável *int*).
- 8) Escreva um programa que leia uma cadeia de caracteres no formato "DD/MM/AAAA" e copie o dia, mês e ano para 3 variáveis inteiras. Antes disso, o programa deve verificar se o formato está correto, ou seja, se as barras estão no lugar certo, e se D, M e A são dígitos.
- 9) Escreva uma função remove todos os espaços no início e no final de uma *string* (processo é chamado de *trimming*). Protótipo:

```
void trim( char str[] );
```

Exemplo:

antes:															
			A	l	ô		m	u	n	d	o	!		\0	
depois:															
A	l	ô		m	u	n	d	o	!	\0					

- 10) Escreva uma função que inverte a ordem dos caracteres de uma *string*.
- 11) Crie uma função que recebe uma *string* *s* e um caractere *c*, e apague todas as ocorrências de *c* em *s*.
Exemplo:

Entrada: *s* = "ManhattanConnection" e *c* = 'n'

Saída: *s* = "MahattaCoectio"

- 12) Escreva uma função que implementa o comportamento da função *strcmp()*, ou seja, dadas duas strings *str1* e *str2*, a função deve comparar os conteúdos considerando a *ordem alfabética*. Protótipo:

```
int compara( char str1[], char str2[] );
```

- 13) Escreva uma função que, a partir de uma *string* *str*, copia para o parâmetro **sub** a *substring* a partir do índice inicial **ini** e que contém a quantidade de caracteres **n**. Alguns casos particulares devem ser considerados, conforme os exemplos a seguir. Protótipo:

```
void substring( char str[], int ini, int n, char sub[] );
```

Exemplos: considere **str** = "Alguma Coisa 123"

- | | | |
|-----------------------------------|---|--|
| (1) substring(str, 7, 5, sub) | → | sub = "Coisa" |
| (2) substring(str, 13, 3, sub) | → | sub = "123" |
| (3) substring(str, 13, 10, sub) | → | sub = "123" // No máximo vai até o final de str |
| (4) substring(str, 20, 5, sub) | → | sub = "" // <i>string vazia</i> , contém somente o '\0' |