



Exercícios – Strings

1. Implemente as funções contidas na biblioteca `cstring`:

- (a) `int strlen(char s[])`: retorna o tamanho da string `s`
- (b) `void strcpy(char dest[], char orig[])`: copia o conteúdo da string `orig` para a string `dest`
- (c) `void strcat(char s1[], char s2[])`: concatena o conteúdo da string `s2` na string `s1`
- (d) `int strcmp(char s1[], char s2[])`: compara a string `s1` com `s2`. Retorna 0 se elas forem iguais; Retorna um número negativo se `s1` for menor do que `s2`; Retorna um número positivo se `s1` for maior do que `s2`.

A função `main` deve ler duas strings e imprimir na tela o resultado das funções.

2. Implemente uma função que receba como parâmetro de entrada uma string e como parâmetro de saída uma outra string. A função a ser implementada deve armazenar na string de saída a string de entrada na ordem inversa. A função `main` deve ler uma string e exibir na tela a string computada pela função. Exemplo:

```
Informe uma frase:
Esta e uma frase
Frase invertida:
esarf amu e atsE
```

3. Implemente uma função que receba como parâmetro de entrada uma string e como parâmetro de saída uma outra string. A função a ser implementada deve armazenar na string de saída a string de entrada com todas as letras maiúsculas convertidas em minúsculas e vice-versa. A função `main` deve ler uma string e exibir na tela a string computada pela função. Exemplo:

```
Informe uma frase:
Estudos de linguagem de programacao
String resultante:
eSTUDOS DE LINGUAGEM DE PROGRAMACAO
```

4. Um palíndromo é uma palavra/frase que pode ser lida tanto da esquerda para a direita quanto da direita para a esquerda. Implemente uma função que receba como parâmetro de entrada uma string e retorne verdadeiro caso ela seja um palíndromo ou falso caso contrário. Considere as duas versões do problema:

- (a) Espaços em branco são considerados como parte da string:
`osso`: é palíndromo
`subi_no_onibus`: não é palíndromo
- (b) Espaços em branco não são considerados como parte da string:
`subi_no_onibus`: é palíndromo
`subi__no__onibus`: é palíndromo

A função `main` deve ler uma string e exibir na tela uma mensagem informando se a string é um palíndromo ou não utilizando a função implementada.

5. Implemente uma função que receba como parâmetro de entrada uma string e como parâmetro de saída um vetor de inteiros de 26 posições. A função a ser implementada deve armazenar no vetor a contagem de cada caractere minúsculo que aparece na string: na posição 0 deve ser armazenada a quantidade de 'a', na posição 1 a quantidade de 'b' e assim por diante até a posição 25, que deve armazenar a quantidade de 'z'. A função `main` deve ler uma string e exibir na tela quantas vezes aparece cada caractere na frase utilizando a função implementada. Exemplo:

```
Informe uma frase:
estudos de linguagem de programacao
Contagem de caracteres:
a: 4
c: 1
d: 3
e: 4
g: 3
i: 1
l: 1
m: 2
n: 1
o: 3
p: 1
r: 2
s: 2
t: 1
u: 2
```

6. Implemente uma função que receba como parâmetro de entrada uma string e como parâmetro de saída uma outra string. Assumindo que na string de entrada estará o nome completo de uma pessoa (separado por um único espaço entre cada nome e sem espaços antes do primeiro ou depois do último nome), a função a ser implementada deve armazenar na string de saída as iniciais do nome seguidas de ponto e espaço. A função `main` deve ler uma string contendo um nome completo e exibir na tela a string computada pela função. Exemplo:

```
Informe um nome:
Joao Francisco da Silva
String resultante:
J. F. d. S.
```

7. Uma string `s2` é considerada uma substring de uma string `s1` se `s2` fizer parte de `s1`. Implemente uma função que receba como parâmetros de entrada duas strings de entrada, `s1` e `s2`. A função a ser implementada deve retornar verdadeiro se `s2` for uma substring de `s1` e falso caso contrário. A função `main` deve ler duas strings e exibir na tela uma mensagem informando se a segunda string lida é uma substring da primeira utilizando a função implementada. Exemplos:

```
-- Exemplo 1:
Informe a primeira string:
Estudos de linguagem de programacao
```

```
Informe a segunda string:
ling
A segunda string faz parte da primeira
-- Exemplo 2:
Informe a primeira string:
Estudos de linguagem de programacao
Informe a segunda string:
ACAO
A segunda string nao faz parte da primeira
-- Exemplo 3:
Informe a primeira string:
Estudos de linguagem de programacao
Informe a segunda string:
std
A segunda string nao faz parte da primeira
```