

Estruturas de Dados

Lista 1

J. L. Rangel

Teste todo o software escrito, escrevendo programas de teste, se necessário.

1. Escreva um programa em C que lê três números inteiros do teclado, e imprime os três números *em ordem crescente*.
2. Idem, três números reais.

Sugestão: Para descobrir formatos adequados para ler e escrever números reais, ou seja, do tipo `float`, leia a documentação do seu compilador.

3. Escreva um programa que recebe como entrada uma temperatura em °C ou em °F e converte a temperatura para a outra escala. A escala em que está a temperatura de entrada é indicada pela letra C ou F.

4. Escreva uma função com protótipo

```
int conta letra(char *s);
```

que calcula o número de letras (maiúsculas e minúsculas) da cadeia `s`.

5. Escreva uma função que converte todas as letras de uma cadeia para maiúsculas. O protótipo pode ser

```
void converte(char *s);
```

6. Escreva uma função que recebe como entrada uma cadeia, e devolve como saída uma cópia da cadeia. Esta função deve alocar o espaço necessário para a cópia.

O protótipo pode ser

```
char *copia(char *s);
```

7. Idem, convertendo as letras da cópia para maiúsculas.
8. Um código extremamente simples substitui cada letra pela letra seguinte, circularmente (Z é codificado como A). Por exemplo, “Estruturas de Dados” se transformaria em “Ftusvuvst ef Ebpt”. Escreva funções para codificar e para decodificar cadeias segundo este código.
9. Escreva um programa que lê uma frase (várias palavras) do teclado, e escreve as palavras uma por linha. Por exemplo, se a entrada fosse

```
Estruturas de      Dados
```

a saída seria

```
Estruturas
de
Dados
```

Sugestão: O formato `%[^\n]` (qualquer coisa, exceto `\n`) pode ser usado por `scanf` para ler uma cadeia que contém brancos, até o Enter final. Além disso, há outras funções de leitura de cadeias, como `gets` e `fgets`, em `<stdio.h>`.

10. Escreva uma função

```
int igual(char *s1, char *s2);
```

para descobrir se as cadeias `s1` e `s2` são iguais.

Sugestão: comece descobrindo porque a solução abaixo não é aceitável.

```
/* bobagem! */  
int igual(char *s1, char *s2) {  
    return s1==s2;  
}
```

11. Escrever uma função que recebe como argumentos uma cadeia s e um caracter c , e que dá como resultado o número de vezes que c aparece em s .
12. Escrever um programa que lê duas cadeias $s1$ e $s2$, e imprime uma cadeia $s3$ que contém todos os caracteres que aparecem em $s1$, mas não aparecem em $s2$.
13. Escreva um programa que lê duas cadeias $s1$ e $s2$, e um natural n , e imprime a cadeia resultante da inserção de $s2$ na posição n de $s1$. (Inserir $s2$ na posição 0 de $s1$ tem como resultado $s2$ seguida de $s1$.)

(abr 00)