

Programação Imperativa – Folha das aulas práticas nº 8

Temas: EXERCÍCIOS DE REVISÃO!

Nome _____ Ncd _____

1. Escreva um programa que leia os caracteres da entrada padrão, escrevendo-os na saída padrão *sem repetir caracteres consecutivos*.
Exemplo

Entrada: 1aa2002aaaa!!
Saída: 1a202a!

Como instruções de entrada/saída pode usar apenas `getchar` e `putchar`.

2. Considere seguinte esquema de codificar frases¹ :

- Todos os caracteres que não são letras são retirados.

```
Os bisontes gritaram Ai! Az!  
OsbisontesgritaramAiAz
```

- As letras maiúsculas são convertidas para letras minúsculas.

```
OsbisontesgritaramAiAz  
osbisontesgritaramaiaz
```

- Há uma palavra secreta, a “senha”, que é usada do modo exemplificado:

```
senha:      by  
mensagem:  osbisontesgritaramaiaz  
código:    pscitootfshrjtbrmbibz
```

A senha neste caso é by; a sua primeira letra é b e, como 'b'-'a'=1, a primeira letra da mensagem “anda 1 para a frente”, transformando-se em p. O mesmo vai ocorrer com a terceira letra, com a quinta, etc. Podemos imaginar a senha repetida indefinidamente:

```
senha:      bybybybybybybybybyby...  
mensagem:  osbisontesgritaramaiaz  
código:    pqcgtmorfqhpjrbpbkbgbx
```

Note-se que o processo é rotativo, estando por exemplo a letra a imediatamente à frente da letra z. De forma análoga, 'y'-'a'=24 e as letras de índice par “andam 24 para a frente” (equivalente a “andarem 2 para trás”).

Complete a função codifica do programa seguinte

```
#define MAIUS(C) (C>='A' && C<='Z')  
#define MINUS(C) (C>='a' && C<='z')  
#define LETRA(C) (MAIUS(C) || MINUS(C))  
  
void codifica(char s[], char senha[],char cod[]){  
    COMPLETE A FUNÇÃO!  
    ...  
}  
  
#define MAX 1000  
#define MAXS 100  
  
main(){  
    char men[MAX], codigo[MAX], senha[MAXS];  
    printf("Mensagem: ");  
    scanf("%s",men);  
    // supõe-se que é de facto dada uma senha só com letras minúsculas!  
    printf("Senha:   ");  
    scanf("%s",senha);  
    codifica(men, senha, codigo);  
    printf("Código:   %s\n",codigo);  
    return 0;  
}
```

Codificando uma mensagem

```
Mensagem: OsbisontesgritaramAiAz  
Senha:      by  
Código:    pqcgtmorfqhpjrbpbkbgbx
```

¹Ressalvam-se os eventuais erros de impressão.

DCCC-FCUP

3. Escreva uma função

```
double media(double v[],int m,double exclu[],int n){
```

que retorna a média dos elementos de v (com m elementos) que não pertencem ao vector exclu (com n elementos).

No exemplo seguinte há 2 elementos não excluídos (2.0 e 1.5) cuja média é 1.75.

```
double media(double v[],int m,double exclu[],int n){  
    EXCREVA ESTA FUNÇÃO!  
}
```

```
int main(){  
    double  
    a[]={2.0,2.2,1.5,2.2},  
    b[]={2.2,3.0};  
    printf("Media = %5.2lf\n",media(a,4,b,2));  
}  
/* Correndo o programa:  
$ a.out  
Media = 1.75  
*/
```

4. Um autómato pode estar em n estados a que chamamos $0, 1, \dots, n-1$. Quando o autómato está no estado q e lê o símbolo s ($s \in \{0, 1\}$), muda para o estado $\text{nov}[q][s]$ onde “ $\text{nov}[][]$ ” é um vector com 2 dimensões. Por exemplo², se o autómato caracterizado por

$$\text{nov}[\text{estado}][\text{símbolo}] = \begin{bmatrix} 1 & 2 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}$$

estiver no estado 0 e ler a palavra 110 fica no estado 2. conforme representamos

símbolo lido	1	1	0	
estado	0	2	0	1

Escreva uma função

```
int estado(int q, novo[][N],char *s)
```

que retorna o estado do autómato caracterizado por nov , sendo q o estado inicial e s a palavra lida.

```
// N: número de símbolos
#define N 2
#define MAX 100

int estado(int q,int novo[][N],char *s){
    ESCREVA ESTA FUNÇÃO!
}

int main(){
    int novo[][N]=
    // 0 1 simb / estado
    //-----
    { {1,2}, // 0
      {2,1}, // 1
      {1,0} // 2
    };
    char s[MAX];
    // supõe-se que é de facto dada uma palavra só com 0's e 1's!
    printf("Palavra (0's e 1's): ");
    scanf("%s",s);
    printf("Estado final: %d\n",estado(0,novo,s));
    return 0;
}
```

Correndo o programa:

```
$ a.out
Palavra (0's e 1's): 110
Estado final: 1
```

²Ressalvam-se os eventuais erros de impressão.