

INF1001: Introdução à Ciência da Computação	P2 – 25/05/08
Nome:	
Matrícula:	Turma:

*A prova é sem consulta. Todos os códigos devem ser feitos usando a linguagem C.
Os cabeçalhos das funções que temos usado encontram-se na última folha da prova.
O aluno pode usar o verso da folha para responder a respectiva questão. Nas questões subdivididas em itens, podem-se resolver os itens de forma independente (por exemplo, pode-se resolver o segundo item assumindo que o primeiro tem solução conhecida).*

1ª QUESTÃO

Considere a existência de um arquivo texto denominado “*conta.txt*”, representando a movimentação financeira de uma conta bancária. Neste arquivo, o número existente na primeira linha representa o saldo inicial, os números das demais linhas representam as movimentações (números positivo e negativos simbolizam depósitos e débitos, respectivamente). Um exemplo deste arquivo é apresentado a seguir:

conta.txt

800.00
-40.50
-60.00
100.00
-70.50

Escreva um programa completo que leia o conteúdo deste arquivo e imprima na tela o saldo após as movimentações. Para o exemplo acima seria impresso o valor 729.

Pode-se considerar que sempre será possível abrir o arquivo.

2ª QUESTÃO

Considerando um vetor de números reais, escreva uma função que, dado o vetor e um número real x , retorne o *índice* do elemento igual a x , ou -1 caso não exista um elemento igual a x .

Por exemplo, seja o vetor {3.5, 7.0, 10.5, 14.3, 16.76} e x o valor 10.5, sua função deverá retornar o valor 2.

Sua função deve ter o seguinte cabeçalho:

```
int busca_indice (int n, float v[], float x)
```

3ª QUESTÃO

Implemente uma função que, dado um vetor de inteiros, inverta as ordens do elementos, isto é, o primeiro passa a ser o último, o segundo passa a ser o penúltimo, e assim por diante. Por exemplo, se for fornecido o vetor {3, 7, 10, 14, 16}, após a execução da função este vetor terá seguinte configuração {16, 14, 10, 7, 3}.

Sua função deve ter o seguinte cabeçalho:

```
void inverta (int n, int v[])
```

4ª QUESTÃO

Considere uma matriz $M \times N$ (M linhas e N colunas) que armazena apenas números inteiros não-negativos, e o seguinte arquivo de código “prog.c”:

```
#include "iolib.h"
#define M 5
#define N 4

/* Aqui será inserido as funções implementadas no item a) e b)*/
...

int main (void)
{
    int i, j,max ;
    int Mat[M][N];
    for ( i=0; i<M; i++) {
        for ( j=0; j<N; i++) {
            Mat[i][j] = readint( );
        }
    }
    max = maior_somatorio(Mat);
    writestring("O maior somatorio dos elementos de uma linha e': " ) ;
    writeint (max ) ;
    return 0;
}
```

a) Escreva uma função que receba como parâmetros a matriz (Mat) e o índice de uma determinada linha (i) e retorne o somatório da linha. Sua função deve ter o seguinte cabeçalho:

int somatorio_linha(int Mat[][N], int i)

b) Implemente a função `maior_somatorio`, usada no código “prog.c”, que retorna o maior valor entre os somatórios dos elementos de cada linha. Esta função deve utilizar a função do item anterior.

Para a matriz:

$$\begin{bmatrix} 3 & 0 & 1 \\ 12 & 4 & 11 \\ 1 & 17 & 5 \end{bmatrix}$$

sua função deverá retornar 27, que representa o somatório da segunda linha da matriz. Assim, o programa codificado em “prog.c” exibiria na tela este valor.

Cabeçalho das funções que podem ser úteis:

utlib.h

```
int giventime (int hour, int min, int sec);
int currtime (void);
int timehour (int t);
int timemin (int t);
int timesec (int t);
int givendate (int day, int month, int year);
int currdate (void);
int dateday (int d);
int datemonth (int d);
int dateyear (int d);
void randseed (int seed);
int randint (int min, int max);
float randfloat (float min, float max);
double raddouble (double min, double max);
void maze (int mx, int my, void* matrix);
```

iolib.h

```
int readint (void);
float readfloat (void);
double readdouble (void);
char readchar (void);
char readanychar (void);
void readword (char s[], int nmax);
void readline (char s[], int nmax);
void writeint (int x);
void writefloat (float x);
void writedouble (double x);
void writechar (char x);
void writestring (char s[]);
void writeline (char s[]);
void newline (void);
```

math.h

```
double exp(double x);
double fabs (double x);
double pow(double x, double y);
double sqrt(double x);
double sin (double x);
double cos (double x);
```

stdio.h (para os alunos que não quiserem usar a\iolib.h')

```
int printf (char* format, ...);
int scanf (char* format, ...);
FILE* fopen (char* _lename, char* mode);
int fclose (FILE* fp);
int fprintf (FILE* fp, char* format, ...);
int fscanf (FILE* fp, char* format, ...);
int feof (FILE* fp);
```