

<b>INF1001: Introdução à Ciência da Computação</b>	<b>P4 – 30/06/08</b>
<b>Nome:</b>	
<b>Matrícula:</b>	<b>Turma:</b>

*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

O "meio do intervalo" (mid-range) de um vetor é dado pela média do valor mínimo (vmin) com o valor máximo(vmax) armazenados no vetor :

$$\text{mid-range} = (\text{vmin} + \text{vmax})/2$$

Dado um vetor de números reais, escreva uma função que calcule e retorne o meio do intervalo. A função deve receber como parâmetros o número de elementos armazenados no vetor, *n*, e uma referência para o vetor, *v*, seguindo o seguinte cabeçalho:

```
float mid_range( int n, float v[])
```

## 2ª QUESTÃO

Implemente uma função que retorne quantos alunos em um dado quadrimestre superaram a quantidade de faltas do quadrimestre anterior. Esta função recebe como parâmetro uma matriz com 4 colunas, onde as colunas armazenam o total de faltas de um aluno em cada um dos últimos 4 meses. Cada linha da matriz corresponde a um aluno. A função recebe também um vetor com as faltas dos  $n$  alunos no quadrimestre anterior.

Para exemplificar, considere os dados relativos a 6 alunos:

Dado o seguinte vetor as faltas do quadrimestre anterior

23	25	13	10	8	20
----	----	----	----	---	----

e a matriz de faltas do quadrimestre:

7	10	5	10
5	6	11	0
1	0	5	0
6	4	6	6
5	1	3	7
6	6	4	4

o retorno da função, neste caso, será 3, pois o total de faltas do 1º, do 4º e do 5º alunos no quadrimestre é superior ao valor armazenado no vetor com o total de faltas do quadrimestre anterior.

### 3ª QUESTÃO

O Índice de Massa Corporal (**IMC**) é reconhecido como padrão internacional para avaliar o grau de obesidade. O **IMC** é calculado dividindo o peso  $p$  (em kg) pela altura  $h$  ao quadrado (em m).

$$\text{IMC} = p / q^2$$

Escreva um programa que dado o arquivo “medidas.txt”, que contém em cada linha o peso (em kg) e a altura (em m) de uma pessoa, gere um arquivo “saida.txt”, o qual contém os mesmos dados da arquivo “medidas.txt” acrescido do IMC de cada indivíduo.

*medidas.txt*

90	1.67
65	1.73
42	1.20
98	1.60

*saida.txt*

90	1.67	32.2
65	1.73	21.7
42	1.38	22.0
98	1.53	41.8

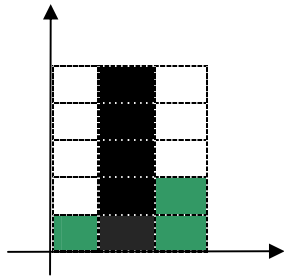
Pode-se considerar que sempre será possível abrir os arquivo, mas o número de linhas existente no arquivo de entrada não é especificado

#### 4ª QUESTÃO

Implemente uma função que faça o desenho de um gráfico de barras, formado por retângulos, de dois tons de cores dispostos de forma alternada. A função recebe como parâmetro o número de barras e um vetor com a altura de cada barra, seguindo o cabeçalho abaixo:

```
void grafico_barras (int n, float h[ ])
```

Exemplo: dado que retângulo tem base  $i$  e  $i+1$  e altura  $h[i]$ , seja  $n$  igual a 3 e  $h$  o vetor  $\{1,5,2\}$ , a função deve desenhar o gráfico ilustrado na figura abaixo:



Considere que o canto inferior esquerdo do gráfico na posição  $(0,0)$  e que a janela sempre terá o tamanho necessário para o desenho.

*Observações:* Os eixos e o grid da figura são apenas ilustrativos, a função deve desenhar apenas os retângulos que compõem gráfico.

## 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);
```