

Questão 1

[Valor: 3,0 pontos] O candidato a um cargo eletivo contratou um instituto de pesquisa de opinião para acompanhar seu desempenho durante a campanha eleitoral. Esse instituto realizou n pesquisas ao longo da campanha, em cada uma das quais verificou o percentual de intenções de voto no candidato e em seu principal concorrente.

Escreva uma função em C que receba como parâmetros o vetor com os percentuais de intenções de voto no candidato (*candidato[]*), o vetor com os percentuais de intenções de voto no seu concorrente (*concorrente[]*) e o tamanho desses vetores (n), e determine qual o valor da maior vantagem percentual que o *candidato* teve sobre seu *concorrente* em uma pesquisa, isto é, a maior diferença entre os percentuais de intenções de voto para um mesmo índice, considerando apenas quando o *candidato* esteve na frente do *concorrente*. Se o *candidato* nunca esteve em vantagem, a função deve retornar 0. O protótipo da função é:

```
float vantagem(float candidato[], float concorrente[], int n);
```

Exemplo

Para os seguintes vetores de entrada:

```
candidato[8]={10.3, 15.3, 17.5, 19.4, 22.5, 22.6, 22.3};  
concorrente[8]={20.4, 18.2, 16.3, 15.8, 14.5, 14.9, 15.6};
```

A função deve retornar:

```
8.0
```

```
float vantagem(float candidato[], float concorrente[], int n)  
{  
    int pos;  
    float vantagem, maior_vantagem = -1;  
  
    for (pos = 0; pos < TAM; pos++)  
    {  
        vantagem = candidato[pos] - concorrente[pos];  
        if (vantagem > 0 && vantagem > maior_vantagem)  
            maior_vantagem = vantagem;  
    }  
    if (maior_vantagem != -1)  
        return maior_vantagem;  
    return 0;  
}
```

Nome:

Matrícula:

Turma:

Questão 2

[Valor: 3,0 pontos] Uma equipe de biólogos acompanha uma população de animais de uma espécie rara, coletando informações específicas que são armazenadas em um arquivo de texto. De cada animal são coletados os seguintes dados: número de identificação (valor inteiro), sexo (valor inteiro, 1 para macho ou 2 para fêmea), peso (valor real), tamanho (valor real), idade (valor inteiro). Para as fêmeas também é armazenado o número de acasalamentos (valor inteiro), número de ovos postos no ninho na última postura (valor inteiro), número de eclosões na última postura, isto é, ovos que deram filhote (valor inteiro).

Como a taxa de crescimento da população está diminuindo vertiginosamente, a equipe deseja monitorar um grupo de interesse composto pelas fêmeas que têm entre 2 e 4 anos, já acasalaram mais de uma vez e na última postura tiveram uma taxa de eclosão (razão entre o número de eclosões e o número de ovos postos no ninho) menor que 0,5.

Escreva um programa em C que leia — do arquivo “dados.txt” — os dados sobre toda a população de animais e salve — no arquivo “preocupantes.txt” — o número de identificação, a idade e a taxa de eclosão, apenas dos animais que pertencem ao grupo de interesse.

Exemplo

dados.txt

```
1 1 12.1 5.1 1
2 2 12.2 5.2 2 2 10 5
3 1 12.3 5.3 3
4 2 12.4 5.4 5 1 10 4
5 1 12.5 5.5 2
6 2 12.6 5.6 3 3 10 2
7 1 12.7 5.7 1
8 2 12.8 5.8 2 2 10 7
```

preocupantes.txt

```
6 3 0.2
```

INF 1005 – P2 – 21/05/10	
Nome:	
Matrícula:	Turma:

Espaço extra para resolução da Questão 2.

```
#include <stdio.h>
#include <stdlib.h>

int main (void)
{
    FILE *in, *out;
    int ident, sexo, idade, n_acasalamentos, n_ovos, n_eclosoes;
    float peso, tamanho, taxa;

    if ((in = fopen ("c:\\temp\\dados.txt", "rt")) == NULL)
    {
        puts ("\nErro na abertura do arquivo DADOS.TXT\n");
        exit (1);
    }
    if ((out = fopen ("c:\\temp\\preocupantes.txt", "wt")) == NULL)
    {
        puts ("\nErro na abertura do arquivo
PREOCUPANTES.TXT\n");
        exit (1);
    }
    while (fscanf(in, "%d%d%f%d", &ident, &sexo, &peso, &tamanho,
&idade) == 5)
    {
        if (sexo == 2)
        {
            fscanf (in, "%d%d", &n_acasalamentos, &n_ovos,
&n_eclosoes);
            if (idade > 2 && idade < 4 && n_acasalamentos > 1)
            {
                taxa = (float) n_eclosoes / n_ovos;
                if (taxa < 0.5)
                    fprintf (out, "%d\t%d\t%.2f\n", ident,
idade, taxa);
            }
        }
    }
    fclose (in);
    fclose (out);
    return 0;
}
```

Questão 3

O gerente de uma locadora de jogos deseja lançar uma promoção na qual os clientes ganharão uma locação grátis para cada 10 locações que realizarem em um mês. Para dar início a essa promoção entre seus 500 clientes, durante o mês de abril ele registrou as operações diárias dos clientes da locadora, escrevendo no arquivo “registros.txt” o código do cliente (valor inteiro entre 1 e 500), dia da retirada (valor inteiro) e quantidade de jogos retirados pelo cliente neste dia (valor inteiro). Um mesmo cliente pode ter realizado várias retiradas ao longo do mês.

a) [Valor: 2,0 pontos] Escreva uma função em C que recebe como parâmetros um vetor de inteiros *vet* e seu tamanho *n*, em que cada elemento contém o valor 0. Essa função deve ler as informações do arquivo “registros.txt”, determinar e armazenar na posição *j-1* do vetor a quantidade de locações gratuitas a que o cliente de código *j* tem direito. O protótipo da função é:

```
void contabiliza(int vet[], int n);
```

b) [Valor: 2,0 pontos] Escreva um programa em C que implemente a promoção. O programa deve chamar a função do item anterior, fornecendo um vetor de 500 elementos (inicializados com 0), para contabilizar as locações grátis a que cada cliente tem direito. Posteriormente, deve exibir na tela o código de cada cliente que faz jus à promoção, com a respectiva quantidade de locações grátis a que tem direito. Não deve ser exibido nenhum dado dos clientes que não têm direito à promoção.

Exemplo

registro.txt

```
1 3 4
11 3 5
1 4 6
21 4 4
11 4 7
1 7 2
1 10 9
12 13 9
```

Vetor antes da chamada da função *contabiliza*:

```
vet[500]={0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ..., 0};
```

Vetor após a chamada da função *contabiliza*:

```
vet[500]={2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, ..., 0};
```

Texto exibido na tela:

```
1 tem 2 locacoes gratis
11 tem 1 locacoes gratis
```

Espaço extra para resolução da Questão 3.

```
#include <stdio.h>
#include <stdlib.h>

void contabiliza(int vet[], int n);
void gera_promocao (int vet[], int n);
void imprime_promocao (int vet[], int n);
void inicializa_vetor (int vet[], int n);

int main (void) {
    int v[500];
    inicializa_vetor (v, 500);
    contabiliza (v, 500);
    imprime_promocao (v, 500);
    return 0;
}

void contabiliza(int vet[], int n) {
    FILE *in;
    int codigo, dia, n_locacoes;

    if ((in = fopen ("c:\\temp\\registros.txt", "rt")) == NULL) {
        puts ("\nErro na abertura do arquivo REGISTROS.TXT\n");
        exit (1);
    }
    while (fscanf (in, "%d%d%d", &codigo, &dia, &n_locacoes)==3) {
        vet[codigo - 1] = vet[codigo - 1] + n_locacoes;
    }
    gera_promocao(vet, n);
    fclose (in);
}

void inicializa_vetor (int vet[], int n) {
    int pos;
    for (pos = 0; pos < n; pos++)
        vet[pos] = 0;
}

void gera_promocao (int vet[], int n) {
    int pos;
    for (pos = 0; pos < n; pos++)
        vet[pos] = vet[pos] / 10;
}

void imprime_promocao (int vet[], int n) {
    int pos;
    for (pos = 0; pos < n; pos++)
        if (vet[pos] > 0)
            printf ("\n%d tem %d locacoes gratis\n", pos + 1,
                vet[pos]);
}
```