

# Programação para Computação

Prova 2 – 26/11/2008

Prof. Marcus Vinícius Midena Ramos

Engenharia de Computação – UNIVASF

## **Questão 1 (3 pontos):**

*Manipulação de agregados:* Construa uma função F que aceite como entrada uma matriz M de 5 linhas e 5 colunas, cujos elementos são do tipo inteiro, e gere como saída um vetor N com 5 elementos do tipo:

```
struct elem {
    int menor;
    int maior;
    int col_menor;
    int col_maior;
}
```

A função F deverá pesquisar as linhas da matriz M e, para cada linha analisada, atribuir na posição correspondente do vetor de saída N um elemento do tipo struct elem, tal que:

- O campo `menor` representa o menor valor encontrado na linha correspondente da matriz M;
- O campo `maior` representa o maior valor encontrado na linha correspondente da matriz M;
- O campo `col_menor` representa o número da coluna onde o menor valor foi encontrado na respectiva linha.
- O campo `col_maior` representa o número da coluna onde o maior valor foi encontrado na respectiva linha.

Caso haja números repetidos na mesma linha, o resultado poderá reportar qualquer um deles. Exemplo de entrada:

1	2	3	4	5
5	4	3	2	1
2	6	3	0	5
6	2	3	7	5
8	2	1	4	5

Saída correspondente (menor,col\_menor # maior,col\_maior):

```
1,0 # 5,4
1,4 # 5,0
0,3 # 6,1
2,1 # 7,3
1,2 # 8,0
```

O programa deverá exibir o conteúdo do vetor N na tela após o retorno da função F.

```
#include <stdio.h>
#define Q 5
struct elem {
    int menor;
    int maior;
    int col_menor;
    int col_maior;};
void F (int M [Q][Q], struct elem N[Q]) {
    int i,j;
    int maior, menor;
    int col_maior, col_menor;
    for (i=0;i<Q;i++) {
```

```

maior=M[i][0];
menor=M[i][0];
col_maior=0;
col_menor=0;
for (j=1;j<Q;j++) {
    if (M[i][j]>maior) {
        maior=M[i][j];
        col_maior=j;
    }
    if (M[i][j]<menor) {
        menor=M[i][j];
        col_menor=j;
    }
}
N[i].maior=maior;
N[i].menor=menor;
N[i].col_maior=col_maior;
N[i].col_menor=col_menor;
}
}
int main () {
int i,j;
int M[Q][Q];
struct elem N[Q];
for (i=0;i<Q;i++)
    for (j=0;j<Q;j++) scanf ("%d",&M[i][j]);
F(M,N);
for (i=0;i<Q;i++)
    printf ("%1d,%1d) - (%1d,
%1d)\n",N[i].menor,N[i].col_menor,N[i].maior,N[i].col_maior);
}

```

**Questão 2 (3 pontos):**

*Numeração das linhas de um arquivo texto:* Construa um programa que aceite três argumentos na linha de entrada, conforme descrito abaixo:

- O primeiro argumento corresponde ao nome de um arquivo de entrada a ser processado;
- O segundo argumento corresponde ao nome do arquivo de saída a ser gerado pelo programa; e
- O terceiro argumento corresponde a uma cadeia de caracteres que deverá ser pesquisada no arquivo de entrada.

O programa deve criar o arquivo de saída a partir do outro fornecido como entrada, de tal forma que o arquivo de saída reproduza o conteúdo do arquivo de entrada, com as seguintes modificações (supor que o arquivo de entrada contém cadeias de caracteres de comprimento máximo 80 e sempre terminadas pelo caracter '\0'):

- As linhas do arquivo de saída devem ser numeradas a partir de 1;
- O campo de numeração das linhas deve ocupar 5 posições e os números das linhas devem ser alinhados à direita;
- As linhas que são copiadas do arquivo de entrada devem iniciar na coluna 9 na linha correspondente no arquivo de saída;
- A coluna 7, no arquivo de saída, deve ser usada para indicar, através do caracter '\*', se a correspondente linha contém a subcadeia fornecida como terceiro argumento na linha de comando; caso não contenha, ela deverá ser deixada em branco.
- As colunas 6 e 8 devem ser deixadas em branco.

Linha de comando:

```
converte entrada.txt saida.txt bcd
```

Exemplo de entrada (arquivo entrada.txt):

```
a
ab
abc
```

```
abcd
abcde
```

Exemplo de saída correspondente (arquivo saida.txt):

```
1  a
2  ab
3  abc
4 * abcd
5 * abcde
```

O programa deverá exibir o conteúdo do arquivo de saída na tela após a sua criação.

```
#include <stdio.h>
int main (int argc, char *argv[]) {
FILE *f,*g;
int i;
char linha [100];
i=1;
if (argc==4) {
    f=fopen(argv[1],"r");
    g=fopen(argv[2],"w");
    if ((f!=NULL) && (g!=NULL)) {
        while (!feof(f)) {
            fgets (linha,100,f);
            fprintf (g,"%5d ",i);
            i++;
            if (strstr (linha,argv[3])) fputc ('*',g);
            else fputc (' ',g);
            fputc (' ',g);
            fputs (linha,g);
        }
        fclose (f);
        fclose (g);
    }
    else printf ("Erro na abertura dos arquivos");
}
else printf ("Erro na quantidade de argumentos");
}
```

### **Questão 3 (4 pontos):**

*Batalha Naval Simplificada:* construa um programa que crie uma representação de 10 posições, numeradas de 1 a 10, para um cenário de batalha naval. O único navio desse jogo ocupa 2 posições contíguas do total de 10 que compõem o cenário. A posição do navio é aleatória, sorteada pelo próprio programa no início da sua execução, e não é informada ao jogador.

O jogador pode fazer até 6 tentativas de tiros para afundar o navio. Para realizar um tiro, o jogador deverá informar um número de 1 a 10 ao programa. Para cada tiro efetuado pelo jogador, o programa deverá informar o mesmo se o navio foi atingido ou não. Caso o navio tenha sido atingido, essa informação deverá ficar registrada no programa.

O jogo é ganho se, em até 6 tentativas, o jogador conseguir atingir as duas posições do cenário onde o navio está posicionado. Caso contrário, ele perde o jogo. O jogo encerra apenas quando o jogador ganha ou perde, e pode terminar após 2, 3, 4, 5 ou 6 tentativas, dependendo dos resultados obtidos em cada uma delas.

Sugestão: use a função abaixo, que gera um número aleatório inteiro entre 1 e n, onde n é um parâmetro passado para a mesma:

```
int aleat (int n) {
srand(time(NULL));
return (rand()%n)+1;
}

#include <stdio.h>
int aleat (int n) {
srand(time(NULL));
```

```

return (rand()%n)+1;
}
int main () {
int mar[10];
int i,cont,afundou;
for (i=0;i<10;i++) mar[i]=0;
i=aleat(9);
mar[i-1]=1;
mar[i]=1;
printf ("\nNavio posicionado.\n");
afundou=0;
for (cont=1;cont<=6;cont++) {
do {
printf ("Digite um numero de 1 a 10: ");
scanf ("%d",&i);
}
while ((i<1) || (i>10));
if (mar[i-1]==1) {
printf ("Voce acertou o navio.\n");
mar[i-1]=0;
if (i==1) if (mar[i]==0) {afundou=1; break;}
if (i==10) if (mar[i-2]==0) {afundou=1; break;}
if ((i>=2) && (i<=9)) if ((mar[i]==0) && (mar[i-2]==0)) {afundou=1;
break;}
}
else printf ("Tiro na agua.\n");
}
if (afundou)
printf ("Voce GANHOU o jogo.\n");
else printf ("Voce PERDEU o jogo.\n");
}

```