

## Aula 21 – 5 Estudo Dirigido – Matrizes

### Exemplos usando funções

**Problema: Jogo Campo Minado. Definir o tamanho do campo de combate em linhas e colunas. O jogador tem 5 chances para não acertar uma bomba no campo. Se acertar uma bomba o jogador é detonado.**

```
#include <iostream>
#include <ctime>
#include <cstdlib>

#define tam 100 // define o tamanho máximo do
                // campo em linhas e colunas
using namespace std;

//Inicializa aleatoriamente o campo minado com
// 0 (zero - sem bomba) e 1 (um - Bomba)
void inicializa_campo(int campo[][tam], int L, int C)
{
    srand((unsigned)time(0)); //para iniciar semente do
    rand.
    int maior = 1, menor = 0;
    rand()%(maior-menor+1) + menor;
    for(int i=0; i<L; i++)
    {
        for(int j=0; j<C; j++)
            //define valor aleatório para cada elem da matriz
            campo[i][j] = rand()%(maior-menor+1) + menor;
    }
}

//Mostra o campo minado com bombas ocultas (O)
// e escolhas do jogador (X)
void mostra_campo(int campo[][tam], int L, int C)
{
    cout << "visao do campo: " << endl;
    for(int i=0; i<L; i++)
    {
        for(int j=0; j<C; j++)
        {
            if (campo[i][j] == 1 || campo[i][j] == 0)
                cout << "O "; //célula oculta não escolhida
            else
                cout << "X "; //célula escolhida sem bomba
        }
        cout << endl;
    }
    cout << endl;
}

int main()
{
    int campo[tam][tam];
    int linhas, colunas, coordL, coordC, qtd;
    int ganhou = 1; //variável que controla se o jogador
                    // ganhou - começa com 1 (TRUE)
    cout << "campo minado! " << endl;
    cout << "Voce tem cinco chances de nao ser detonado! "
            << "Hahahahaha" << endl;
    cout << "Digite a quantidade de linhas do campo (entre 1 e 99): ";
    cin >> linhas;
    cout << "Digite a quantidade de colunas do campo (entre 1 e 99): ";
    cin >> colunas;
    qtd = 5; //define a quantidade de chances do jogador
    inicializa_campo(campo, linhas, colunas);

    for (int chances=0; chances<qtd; chances++)
    {
        mostra_campo(campo, linhas, colunas);
        cout << "Chance " << chances + 1 << endl;
        //le as coordenadas do jogador
        do
        {
            cout << "Digite LINHA (entre 0 e "<<linhas-1 <<"): ";
            cin >> coordL;
        } while(coordL < 0 || coordL > linhas - 1);
        do
        {
            cout << "Digite coluna (entre 0 e "<<colunas-1 <<"): ";
            cin >> coordC;
        } while(coordC < 0 || coordC > colunas - 1);
        // Verifica se uma bomba foi encontrada
        if (campo[coordL][coordC] == 1) //bomba encontrada
        {
            cout << "Booooo!!!! Voce foi detonado" << endl;
            ganhou = 0; //jogador perdeu define variável com 0 (FALSE)
            break;
        }
        else
        { //se não for bomba define valor -1 para a posição
          campo [coordL][coordC] = -1;
          cout << "Voce teve sorte desta vez!!!" << endl;
        }
    }
    if (ganhou == 1)
        cout << "Parabens voce ganhou o jogo!!!!" << endl;
    else
        cout << "Eu falei que voce ia perder!!! Hahahahaha!" << endl;
    return 0;
}
```

**Atividade avaliativa:** Desenvolva os programas em C++ abaixo. A atividade deve ser realizada em duplas que devem apresentar até o final da aula os programas funcionando para o professor. Se a dupla não conseguir realizar o trabalho em sala deverá enviar um e-mail para [anilton.ufu@outlook.com](mailto:anilton.ufu@outlook.com), com o assunto: **FACOM39017 estudo dirigido 5**, e com o nome dos integrantes da dupla e o código fonte desenvolvido (este procedimento terá desconto de 50 % do valor da avaliação).

**1ª Questão** – Para ler uma matriz **a**, quadrada de ordem **n**. Teste cada elemento da matriz **a**, gerando e em seguida imprimindo, uma nova matriz **b** da seguinte forma:

- se o elemento da matriz testada for positivo, o respectivo elemento da nova matriz deverá ser **+10**.
- se o elemento da matriz testada for nulo, o respectivo elemento da nova matriz deverá ser **5**.
- se o elemento da matriz testada for negativo, o respectivo elemento da nova matriz deverá ser **0**.

**2ª Questão** – Para ler uma matriz **a**, **nxm**, de valores reais. Construa um vetor **b** de **m** elementos, tal que cada elemento **b[ j ]** seja igual ao maior elemento da coluna **j** da matriz **a**. Imprima o vetor construído.

Obs. Nas questões acima usar: uma função que lê a matriz **a**; outra função obtém a matriz (vetor) **b**; outra função imprime a matriz (vetor) **b**; a função main lê os valores de **n** e **m**, e **ativa** a execução das funções auxiliares acima. Considere  $n$  e  $m \leq 10$ .