

Nono Laboratório de ECOP11

UNIFEI - Universidade Federal de Itajubá

Prof. Gabriel Cirac Mendes Souza (gabrielcirac@unifei.edu.br)

Assunto: Funções com arrays e passagem de parâmetro por referência

Iremos desenvolver uma versão simplificada do amado jogo “**River Raid**”, imortalizado no videogame Atari. Basearemos o programa em quatro funções e uma matriz de 20 por 20. A matriz representa o mapa de obstáculos (helicópteros) e o avião estará posicionado sempre na última linha da matriz.

- 1) (10 pontos) Declare a utilização das bibliotecas e constantes abaixo:

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

#define VAZIO ' '
#define PAREDE '|'
#define OBSTACULO 'o'
#define JOGADOR '^'
```

- 2) (20 pontos) Escreva uma função chamada inicializa_mapa que não retorna nada (void) e recebe uma matriz de dimensões 20x20 do tipo char. Esta função deve inicializar todos os elementos com VAZIO. Em seguida, deve-se colocar em todas as colunas 0 e 19 da matriz o valor PAREDE. Note que o caminho que pode ser utilizado pelo avião possuirá um total de 18 colunas. O avião está se deslocando dentro de um cânion, e há paredes altas dos dois lados.
- 3) (20 pontos) Escreva uma função chamada desenha_mapa que não retorna nada (void) e recebe uma matriz de dimensões 20x20 do tipo char e um valor inteiro da coluna do jogador. Esta função deve imprimir na tela os valores da matriz sendo que cada linha deve conter 20 elementos. No entanto, deve-se imprimir o valor JOGADOR na respectiva coluna da matriz quando está sendo impressa a última linha (linha 19). Em seguida, copie a função *main* abaixo para testar as funções:

```
int main()
{
    char mapa[20][20];
    int col_jogador = 10;
    inicializa_mapa(mapa);
    while(mapa[19][col_jogador] != OBSTACULO)
    {
        desenha_mapa(mapa, col_jogador);
    }
    return 0;
}
```

Observação: Para que serve a condição do while? Para cancelar a execução do programa, pressione as teclas *Ctrl+C*.

- 4) (30 pontos) Copie a função *main* abaixo, que verifica quando é digitada a tecla 'a' (avião move-se para esquerda, sem ultrapassar parede). Modifique-o para garantir que, quando for digitada a tecla 'd', o avião mova-se para a direita, sem ultrapassar a parede. Por fim, quando for digitada a letra 'w', o avião fica na mesma coluna, dando a impressão de que apenas se moveu para frente.

```
int main()
{
    char dig, mapa[20][20];
    int col_parede, col_jogador = 10;
    inicializa_mapa(mapa);
    desenha_mapa(mapa, col_jogador);
    while(mapa[19][col_jogador] != OBSTACULO)
    {
        dig = getchar();
        if((dig == 'a' || dig == 'A') && (col_jogador > 5))
        {
            col_jogador--;
            desenha_mapa(mapa, col_jogador);
        }
    }
    return 0;
}
```

- 5) (20 pontos) Escreva uma função chamada atualiza_mapa que não retorna nada (void) e recebe uma matriz de dimensões 20x20 do tipo char. Esta função será inserida na função main antes de todas as funções desenha_mapa e deve copiar todos os elementos uma linha para baixo, para dar impressão de movimento. Inicie a cópia da linha 18 e suba até a linha 0, sendo que a linha x + 1 recebe o conteúdo da linha x. Em seguida, deixe vazia a primeira linha. Adicione as paredes nas colunas 0 e 19 da primeira linha. Por fim, adicione a chance de 50% de aparecer um helicóptero na primeira linha, copiando as linhas abaixo:

```
// chance de 50% em aparecer carro
if ((rand() % 100) < 50) {
    posCarro = 1 + (rand() % 18);
    m[0][posCarro] = OBSTACULO;
}
```

Dica:

```
// copia todas as linhas até a linha 1
for (i = 18; i >= 0; i--)
    for (j = 0; j < 20; j++)
        m[i + 1][j] = m[i][j];
```