


UNIFEI	Universidade Federal de Itajubá Instituto de Engenharia de Sistemas e Tecnologias da Informação – IESTI
10º Laboratório	Prof. Enzo Seraphim
Parte 1	Simulador de Criação de Peixe

1) Abra o code::blocks e crie novo arquivo usando o menu File| New | Empty File

2) Escreva o algoritmo abaixo, deixando os comentários de (a) até (h) que serão substituídos por novas linhas de algoritmos. O algoritmo vai simular a criação de peixe inspirado em um jogo famoso sendo baseado na manipulação de uma matriz 25x50 chamada tanque.

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<time.h>
int larva=0;
int alga=0;
int sais=1;
int jogadas=1;
char tanque[25][50];
int main( int argc, char * arv[]){
    srand(time(NULL));
    //b
    char op=' ';
    while(op != 's'){
        //a
        printf("larva=%d alga=%d sais=%d\n", larva, alga, sais);
        printf("(c)ultivar / (f)ertilizar / (m)over / (q)uebrar / (p)escar / (n)oite / (s)air"
        printf("[Jogada %d] opcao:", jogadas);
        op = getch();
        printf("%c\n", op);
        switch(op){
            case 'c' : //c
                break;
            case 'f' : //d
                break;
            case 'm' : //e
                break;
            case 'q' : //f
                break;
            case 'p' : //g
                break;
            case 'n' : //h
                break;
            case 's' :
                break;
            default:
                printf("[opcao invalida]");
        }
        if(jogadas > 7){
            //b+h
        }
    }
    return 0;
}
```


3) Use os menus File | File Saver (Ctrl+s), salvando o programa em uma pasta desejada com o nome peixe.c.

4) Execute o algoritmo (tecla F9 ou clique no botão ) e observe que o programa imprime algumas informações sobre recursos (larva, alga e sais) fica solicitando opção indefinidamente.

UNIFEI	Universidade Federal de Itajubá Instituto de Engenharia de Sistemas e Tecnologias da Informação – IESTI
10º Laboratório	Prof. Enzo Seraphim
Parte 1	Simulador de Criação de Peixe


5) Substitua o comentário `//a` para invocar (chamar) uma função chamada `imprime()`. A função `imprime` que não tem parâmetros e não retorna nada deve ser declarada e implementada antes do código `main`. A função deve:

- Usar dois laços de repetição **for**, um dentro do outro, para percorrer os 2 índices da matriz `tab`.
- Dentro desses dois laços de repetição **for** imprima o valor da célula `i, j` da matriz `tanque` sem pular linha. Após o termino do laço **for** mais interno imprima na tela o pular de linha (`printf("\n")`).

6) Para testar execute o algoritmo (tecla F9 ou clique no botão )
Feche a janela para finalizar a execução do programa.


7) Substitua o comentário `//b` para invocar (chamar) uma função chamada `inicializa()`. A função `inicializa` que não tem parâmetros e não retorna nada deve ser declarada e implementada antes do código `main`. A função deve inicializar a matriz `tanque` da seguinte maneira:

- Inicialmente matriz deve estar inicializada com espaço. Laços de repetição para inicializar com espaço ' '.
- Deve conter 3 regiões sorteadas aleatoriamente formando um triângulo de duas linhas definido por 1-3 células com areia ':'. Laço de repetição de 3 sorteios de linha e coluna do canto inferior esquerdo que formam o triângulo, sendo 4 células recebendo areia ':':
- Deve conter 2 regiões sorteadas aleatoriamente formando um triângulo de três linhas definido por 1-3-5 células com areia ':'. Laço de repetição de 2 sorteios de linha e coluna do canto inferior esquerdo que formam o triângulo, sendo 9 células recebendo areia ':':
- Deve conter 10 conchas '@' sorteadas aleatoriamente em posições que tem espaço. Laço de repetição de 10 sorteios de linha e coluna para receber concha '@':
- Deve conter 5 brotos de algas 'y' sorteados aleatoriamente em posições que tem espaço. Laço de repetição de 5 sorteios de linha e coluna para receber broto alga 'y':
- Deve conter 3 peixes jovens 'x' sorteados aleatoriamente em posições que tem espaço. Laço de repetição de 3 sorteios de linha e coluna para receber peixe jovem 'x':

8) Para testar execute o algoritmo (tecla F9 ou clique no botão )
Feche a janela para finalizar a execução do programa.

9) Substitua o comentário `//c` para invocar (chamar) uma função chamada `cultivar()`. A função `cultivar` que não tem parâmetros e não retorna nada deve ser declarada e implementada antes do código `main`. A função deve:


- Verificar se existe saldo em alga.
- Fazer a leitura pelo teclado de linha e coluna que o usuário deseja cultivar alga.
- Verificar se a posição de linha e coluna são posições válidas na matriz.
- Verificar se a posição de linha e coluna na matriz está vazia.
- Atribuir broto de alga 'y' na posição de linha e coluna da matriz e decrementar o recurso alga. Incrementa-se variável jogada.

10) Para testar execute o algoritmo (tecla F9 ou clique no botão ) e escolha a opção 'c'.
Feche a janela para finalizar a execução do programa.

11) Substitua o comentário `//d` para invocar (chamar) uma função chamada `fertilizar()`. A função `fertilizar` que não tem parâmetros e não retorna nada deve ser declarada e implementada antes do código `main`. A função deve:

UNIFEI	Universidade Federal de Itajubá Instituto de Engenharia de Sistemas e Tecnologias da Informação – IESTI
10º Laboratório	Prof. Enzo Seraphim
Parte 1	Simulador de Criação de Peixe

- Verificar se existe saldo em sais.
- Fazer a leitura pelo teclado de linha e coluna que o usuário deseja fertilizar alga.
- Verificar se a posição de linha e coluna são posições válidas na matriz.
- Verificar se a posição de linha e coluna na matriz tem broto de alga 'y'.
- Atribuir alga adulta 'Y' na posição de linha e coluna da matriz e decrementar o recurso sais. Incrementa-se variável jogada.

12) Para testar execute o algoritmo (tecla F9 ou clique no botão ) e escolha a opção 'f'.
Feche a janela para finalizar a execução do programa.