

Computação Básica – Prática de Laboratório V

Prof. Alexandre Zaghetto

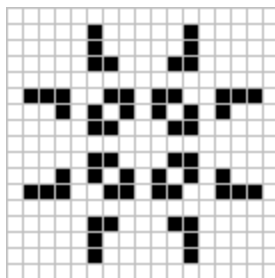
zaghetto@gmail.com

1. The **Game of Life**, invented by John Conway in 1970, is an example of a zero-player “game” known as a cellular automaton. The game consists of a two-dimensional world extending infinitely in all directions, divided into “cells.” Each cell is either “dead” or “alive” at a given “generation.” The game consists of a set of rules that describe how the cells evolve from generation to generation. These rules calculate the state of a cell in the next generation as a function of the states of its neighboring cells in the current generation. In a 2-D world, a cell’s neighbors are those 8 cells vertically, horizontally, or diagonally adjacent to that cell. At each step in time, the following transitions occur:

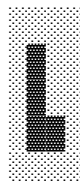
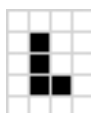
1. Any live cell with fewer than two live neighbours dies, as if caused by under-population.
2. Any live cell with two or three live neighbours lives on to the next generation.
3. Any live cell with more than three live neighbours dies, as if by overcrowding.
4. Any dead cell with exactly three live neighbours becomes a live cell, as if by reproduction.

In this lab, we will be implementing Conway’s Game of Life, with the minor restriction that our 2-D world is finite. The neighbors of a cell on the edge of the world that would be beyond the edge are assumed dead. You can read more about Conway’s Game of Life on Wikipedia at [http://en.wikipedia.org/wiki/Conway%27s Game of Life](http://en.wikipedia.org/wiki/Conway%27s_Game_of_Life).

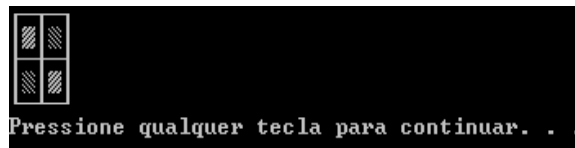
Declare uma matriz, definida conforme a ilustração abaixo, que indica a geração inicial do autômato.



As células brancas (mortas) devem ser apresentadas na tela do computador com o caractere cujo código ASCII é 176 (░). As células pretas (vivas) devem ser apresentadas com o caractere cujo código ASCII é 178 (▒).



A grade também deve ser desenhada. Utilize os caracteres da tabela ASCII cujos códigos estão entre 179 (|) e 218 (┌).



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

int main(int argc, char *argv[])
{

    printf("%c%c%c%c%c\n", 218,196, 194,196, 191);
    printf("%c%c%c%c%c\n", 179,178, 179,176, 179);
    printf("%c%c%c%c%c\n", 195,196, 197,196, 180);
    printf("%c%c%c%c%c\n", 179,176, 179,178, 179);
    printf("%c%c%c%c%c\n", 192,196, 193,196, 217);

    system("PAUSE");
    return 0;
}
```

Proponha uma maneira inteligente de se construir a grade, utilizando repetições, como no exemplo a seguir.



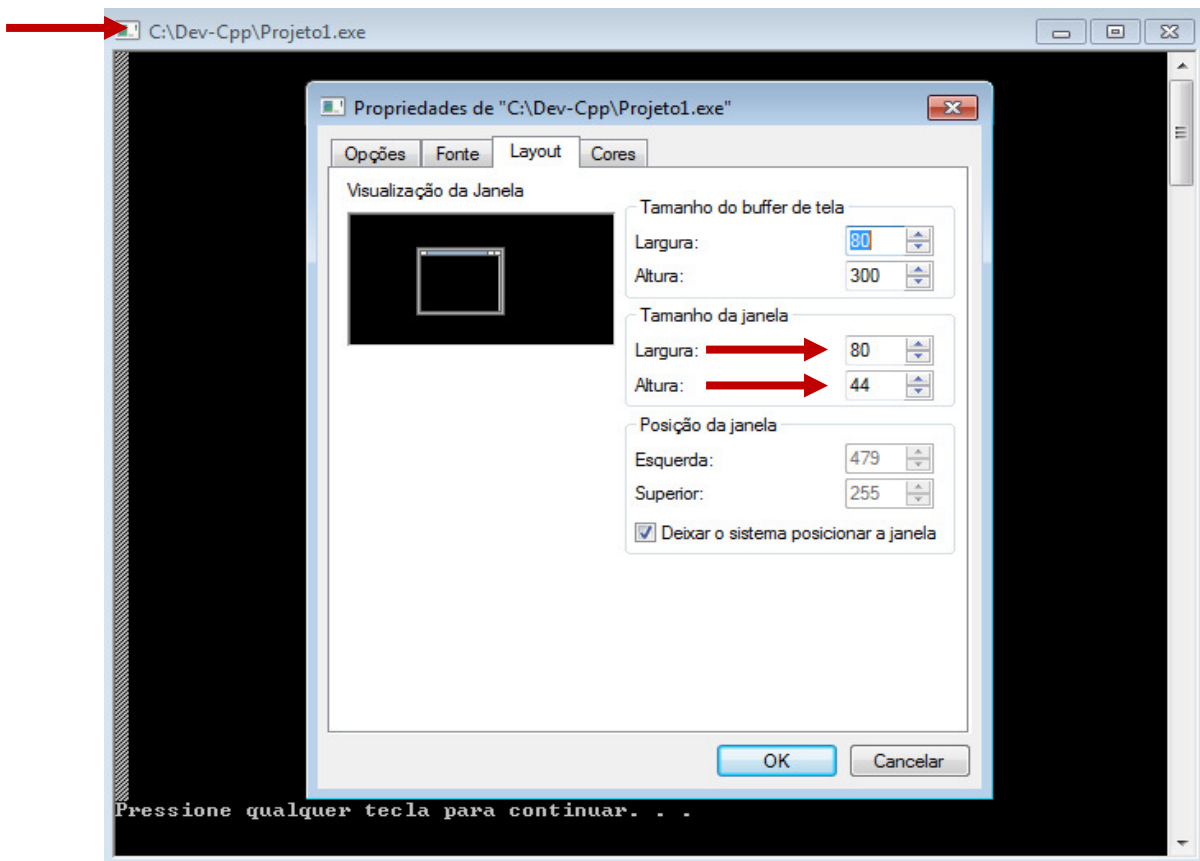
```
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[])
{
    int i;

    printf("%c%c",218,196);
    for(i = 0; i<10; i++)
        printf("%c%c%c",196, 194,196);
    printf("%c%c\n\n",196,191);

    system("PAUSE");
    return 0;
}
```

É possível que você precise ajustar o tamanho da janela do *prompt* de comandos. Para isso basta clicar com o botão direito do *mouse* no canto superior direito da janela e selecionar propriedades.



Escreva um programa em C que dada a geração inicial mostrada no item 1, gera e mostra na tela do computador as gerações seguintes, respeitando as regras 1 a 4 expostas no enunciado sobre o **Game of Life**.

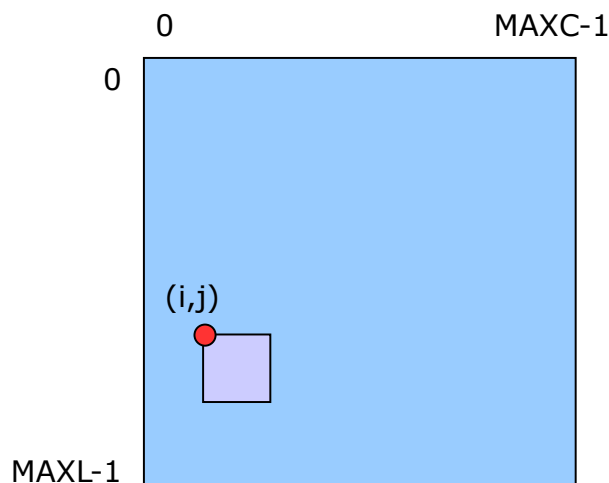
2. Escreva um programa em linguagem C que implementa o jogo de batalha naval:

Imprimir inicialmente uma tela de entrada do jogo:

[illegible]

Definir área de jogo como sendo $MAXL \times MAXC$, com $MAXL = 20$ e $MAXC = 50$.

Em seguida, o programa deverá sortear aleatoriamente dois valores inteiros i e j . Estes dois valores i e j deverão ser adequadamente limitados (em função de $MAXI$ e $MAXC$, respectivamente), para que a área do jogo não seja extrapolada pelo navio. O elemento (i, j) corresponde ao vértice superior esquerdo do navio, conforme ilustrado abaixo:



O navio tem seu formato definido por:



Navio posicionado. O jogo começou!

Digite a linha i (entre 0 e 19):

Digite a coluna j(entre 0 e 49):

[illegible]

[illegible]

Se (i,j) não estiver contida na área do navio, o tabuleiro deve ser reimpresso, marcando com a letra A a parte do tabuleiro atacada pelo usuário.

[illegible]

O exercício 1 foi adaptado do curso *6.087: Practical Programming in C* promovido pelo Massachusetts Institute of Technology (MIT), Department of Electrical Engineering and Computer Science.