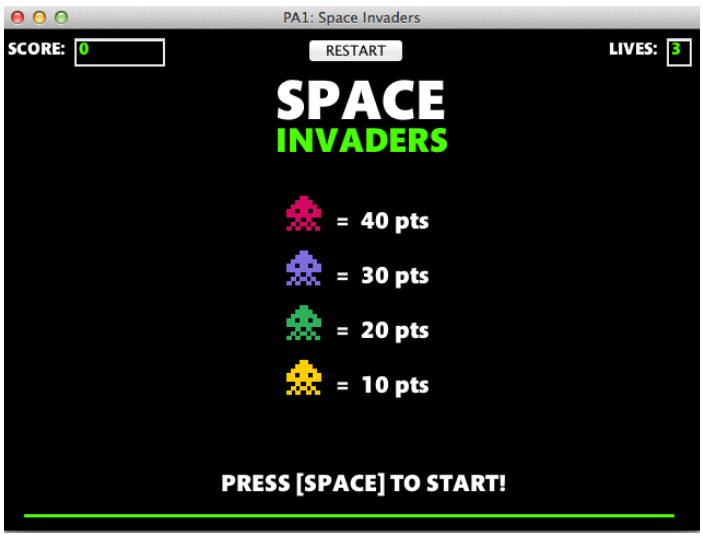
Universidade do Oeste de Santa Catarina
Área de Ciências Exatas e da Terra - ACET
Curso de Engenharia da Computação
Disciplina: Programação de Aplicativos 1

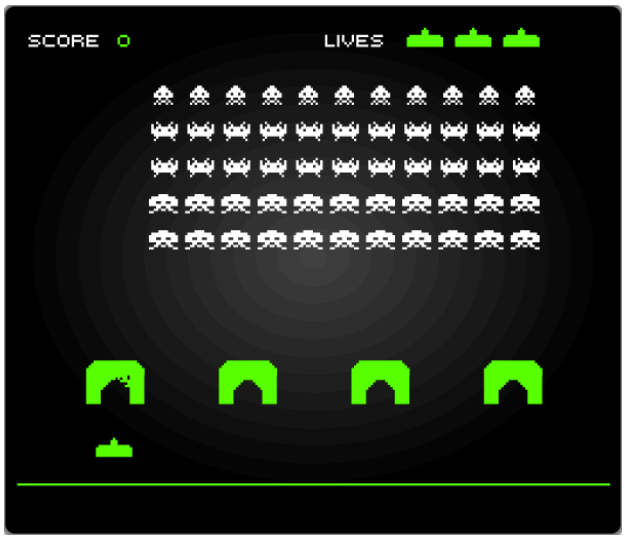
1. O trabalho é individual e sem consulta. A interpretação das informações abaixo fazem parte da avaliação.
2. É permitido tirar dúvidas com o professor, entretanto, se um estudante obter ou conseguir ajuda inapropriada de terceiros para realizar esta atividade ele estará sujeito a ter sua nota zerada.
3. Desonestidade acadêmica: plágio e outras formas de trapaça são ofensas sérias, e o acadêmico estará sujeito as penalidades estabelecidas pela universidade.

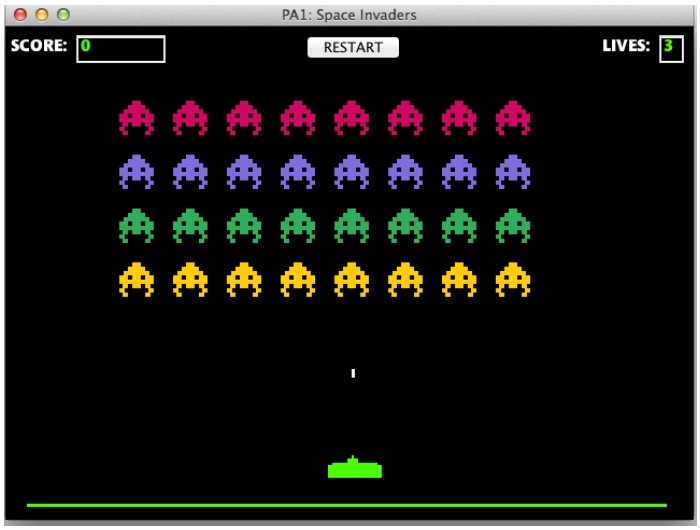
Space Invaders

Neste trabalho você demonstrará domínio básico sobre programação gráfica utilizando **Java**, **Swing**, **Java 2D** e **armazenamento persistente** ao implementar uma versão mínima do jogo *Space Invaders*, exibido nas figuras abaixo:



Este jogo é um clássico de *arcade* desenvolvido pela *Taico Corporation* em 1978. *Space Invaders* é um jogo 2D, de tiro, que permite o jogador controlar um canhão laser através das teclas direcionais do teclado para movimentar o canhão lateralmente na tela, com o intuito de disparar tiros nos alienígenas invasores. O objetivo do jogo é derrotar as linhas e colunas de alienígenas.





Nesta versão simplificada do jogo que você deve implementar, os EBEs (*Extraterrestrial Biological Entities*) se movimentam verticalmente e em grupo, avançando em direção a parte inferior da tela. O jogador destrói um alienígena e ganha pontos quando acerta um tiro de canhão nele. O jogador perde uma vida quando um dos alienígenas alcança a linha verde localizada na parte inferior do tabuleiro.

Estas são as tarefas que você deverá completar:

1) Projetar a interface gráfica de forma que ela apresente uma janela (**JFrame**) com elementos gráficos para:
    a. Exibir a pontuação (*score*) atual do jogador (**JTextField**);
    b. Exibir a quantidade de vidas (lives) que ainda restam (**JTextField**);
    c. Exibir o tabuleiro do jogo (**JPanel**);
    d. Exibir um botão (**JButton**) para reiniciar o jogo, fazendo com que os EBEs, a pontuação e a quantidade de vidas sejam revertidas para suas configurações iniciais.
2) O jogador deve poder movimentar o canhão através das teclas direcionais direita/esquerda. A movimentação deve ser restrita a área visível do tabuleiro.
3) A pintura do tabuleiro deve acontecer de forma automática, ou seja, através de um temporizador (**java.util.Timer**) configurado em aproximadamente 250ms. Desta forma, a interação através das teclas não deve acionar a repintura dos elementos gráficos da janela.
4) O tabuleiro deve apresentar pelo menos 4 linhas e 8 colunas de EBEs para o combate.
5) As seguintes teclas e operações devem ser executadas:
    a. Seta direcional esquerda: move o tanque na horizontal para a esquerda;
    b. Seta direcional direita: move o tanque na horizontal para a direita;
    c. Barra de espaço: dispara o projétil em direção aos EBEs;
6) O sistema do jogo deve suportar a detecção de colisão entre:
    a. O projétil do canhão com os EBEs;
    b. Os EBEs com a linha verde (parte inferior da tela);
7) Os EBEs devem se movimentar na vertical de forma ordenada:
    a. Apenas depois que o jogador marcar alguns pontos é que o grupo de EBEs deve se movimentar algumas unidades na vertical, fazendo com que todo o grupo se desloque em direção a linha verde localizada na parte inferior da tela.
8) Quando os EBEs atingem seu objetivo, o jogador perde uma vida e jogo deve ser reinicializado para seu estado original, com exceção da pontuação do jogador.
9) Quando um jogador perder todas as vidas, o jogo deve apresentar uma outra janela com os seguintes elementos:
    a. Na parte superior da janela, o jogador deve poder inserir seu nome para que sua pontuação seja armazenada de forma persistente no banco de dados (MySQL).
    b. Na parte inferior da janela, deve ser exibido um **JTable** que mostra as 5 maiores pontuações armazenadas no banco e o nome dos respectivos jogadores.

A classe disponibilizada nas próximas páginas deste documento deve ser utilizada, obrigatoriamente, para desenhar os alienígenas do seu jogo. Você tem total liberdade para criar novos personagens, entretanto, a utilização correta desta classe faz parte da avaliação do trabalho.

Ebe.java

```
import java.awt.Color;
import java.awt.Graphics2D;

public class Ebe {
    private int _pos_x;
    private int _pos_y;
    private boolean draw_shape1 = true;

    public void draw(Graphics2D g2d, int x, int y, Color c)     {
        _pos_x = x;
        _pos_y = y;

        g2d.setColor(c);

        /* head */
        g2d.fillRect(x+12,    y,     8,  4);
        g2d.fillRect(x+8,   y+4, (8*2),  4);
        g2d.fillRect(x+4,   y+8, (8*3),  4);
        /* eyes */
        g2d.fillRect(x,    y+12,     8,  4);
        g2d.fillRect(x+12, y+12,     8,  4);
        g2d.fillRect(x+24, y+12,     8,  4);
        /* chin */
        g2d.fillRect(x,    y+16, (8*4),  4);

        if (draw_shape1) {
            draw_shape1 = false;
            /* hips */
            g2d.fillRect(x+8,  y+20,    5,  4);
            g2d.fillRect(x+18, y+20,    5,  4);
            /* legs part 1 */
            g2d.fillRect(x+4,  y+24,    5,  4);
            g2d.fillRect(x+12, y+24,    7,  4);
            g2d.fillRect(x+22, y+24,    5,  4);
            /* legs part 2 */
            g2d.fillRect(x,    y+28,    5,  4);
            g2d.fillRect(x+8,  y+28,    5,  4);
            g2d.fillRect(x+18, y+28,    5,  4);
            g2d.fillRect(x+26, y+28,    5,  4);
        }
        else {
            draw_shape1 = true;
            /* hips */
            g2d.fillRect(x+3,  y+20,    5,  4);
            g2d.fillRect(x+12, y+20,    8,  4);
            g2d.fillRect(x+24, y+20,    5,  4);
            /* legs part 1 */
            g2d.fillRect(x,    y+24,    5,  4);
            g2d.fillRect(x+27, y+24,    5,  4);
            /* legs part 2 */
            g2d.fillRect(x+3,  y+28,    5,  4);
            g2d.fillRect(x+24, y+28,    5,  4);
        }
    }

    public int x() {
        return _pos_x;
    }

    public int y() {
        return _pos_y;
    }
}
```