

Relatório de Computação Gráfica

Trabalho 2

Kelvin Guilherme de Oliveira, 9293286

Leonardo Alves Gomes, 9293178

Lucas Yudi Sugi, 9293251

Maurício Caetano da Silva, 9040996

Enunciado do trabalho: Fazer uma cópia simplificada do jogo Space Invaders. O usuário se move com as teclas direcionais para a esquerda e a direita e atira com a barra de espaço. O objetivo é eliminar todos os aliens antes que eles atinjam o nível da nave ou que o número de vidas do jogador acabe.

Plataforma utilizada: O referido trabalho foi confeccionado no sistema operacional GNU/Linux. Utilizando o Vim e o GCC, com uso do compilador g++ e a linguagem C++.

Tecnologias externas: Para a confecção do referido trabalho foram utilizadas as tecnologias OpenGL e a freeGLUT, especificamente, para escrever textos na tela, as quais foram obtidas com estudos realizados nos fóruns do StackOverflow.

Principais dificuldades: Uma dificuldade foi desenhar os aliens com pontos, mas ao mesmo tempo gerou um resultado interessante.

Divisão de tarefas: O Maurício foi responsável pela movimentação do jogador principal, enquanto que o Lucas fez as naves inimigas e a movimentação em grupo delas. O Kelvin fez a bala e sua movimentação, enquanto o Leonardo fez a união de todas estas partes somadas a detecção de colisões.

Tempo: Para a realização deste trabalho foi preciso trabalhar por 7 dias.

Trecho de código que o grupo julga mais importante: O grupo acredita que o código a seguir foi a peça principal no referido trabalho. Os trechos de código a seguir se referem ao *update* das balas e *drawing* da tela, considerando os estados possíveis do jogo.

```

// Moves the bullets
for (vector<BULLET>::iterator it = bullet.begin(); it != bullet.end(); it++) {
    if (it->direction == DOWN) {
        it->yPos -= v;
        if (it->yPos <= bottom) {
            it->visible = false;
            bullet.erase(it);
            it--;
            continue;
        }

        if (it->xPos >= shipX - 0.05 && it->xPos <= (shipX + 0.05) &&
            it->yPos >= -0.9 && it->yPos <= -0.9 + 0.14) {
            playerWasHit();
            bullet.erase(it);
            it--;
            continue;
        }
    } else if (it->direction == UP) {
        it->yPos += v;
        if (it->yPos >= top) {
            it->visible = false;
            bullet.erase(it);
            it--;
            continue;
        }

        if (checkUpBulletCollision(it)) {
            bullet.erase(it);
            it--;
            continue;
        }
    }
}
}

```

Código 1: código de update das balas.

```

/** Rendering Funcion */
void draw()
{
    //Changes to the
    glMatrixMode(GL_MODELVIEW);
    //Initialize the transformations matrix
    glLoadIdentity();

    // Cleans the screen
    glClear(GL_COLOR_BUFFER_BIT);

    if (GAME_STATE == GAMESTATE_WON) {
        drawWonScreen();
    } else if (GAME_STATE == GAMESTATE_LOST) {
        drawLostScreen();
    } else if (GAME_STATE == GAMESTATE_GAME) {
        drawBullets();
        drawShip();
        drawAlien();

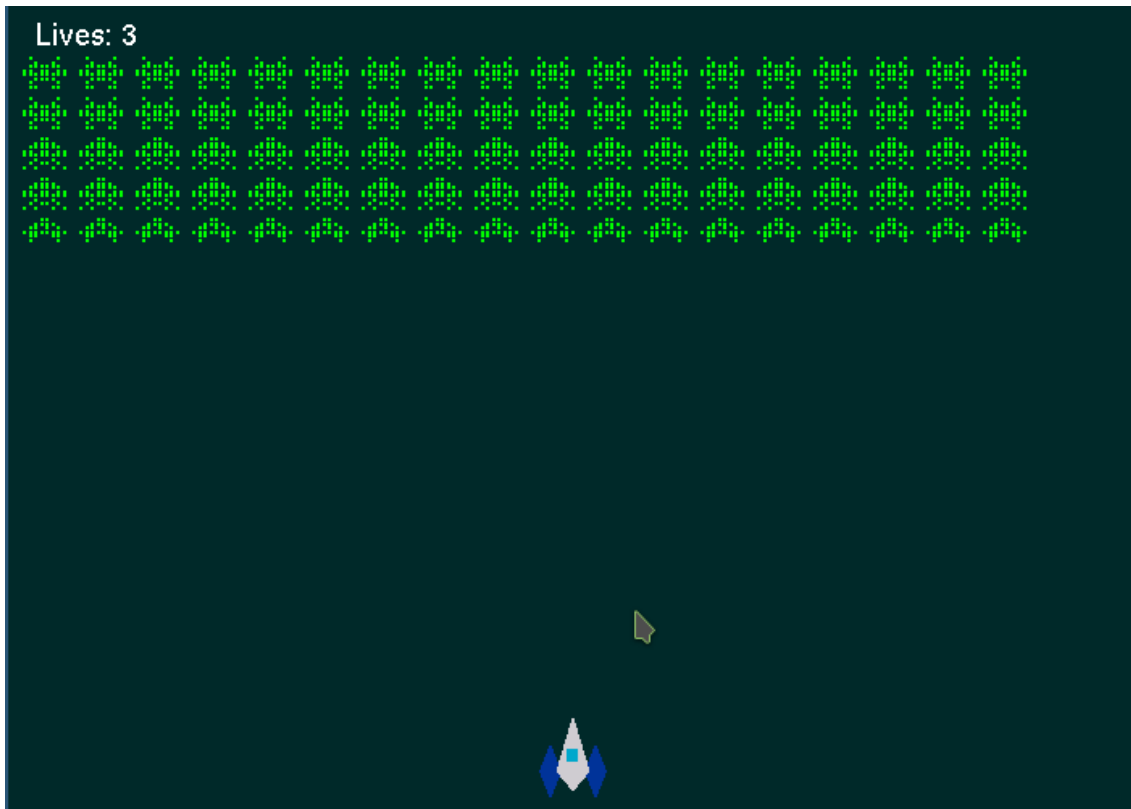
        drawLives();
    }

    // Update the screen
    glutSwapBuffers();
    glFlush();
}

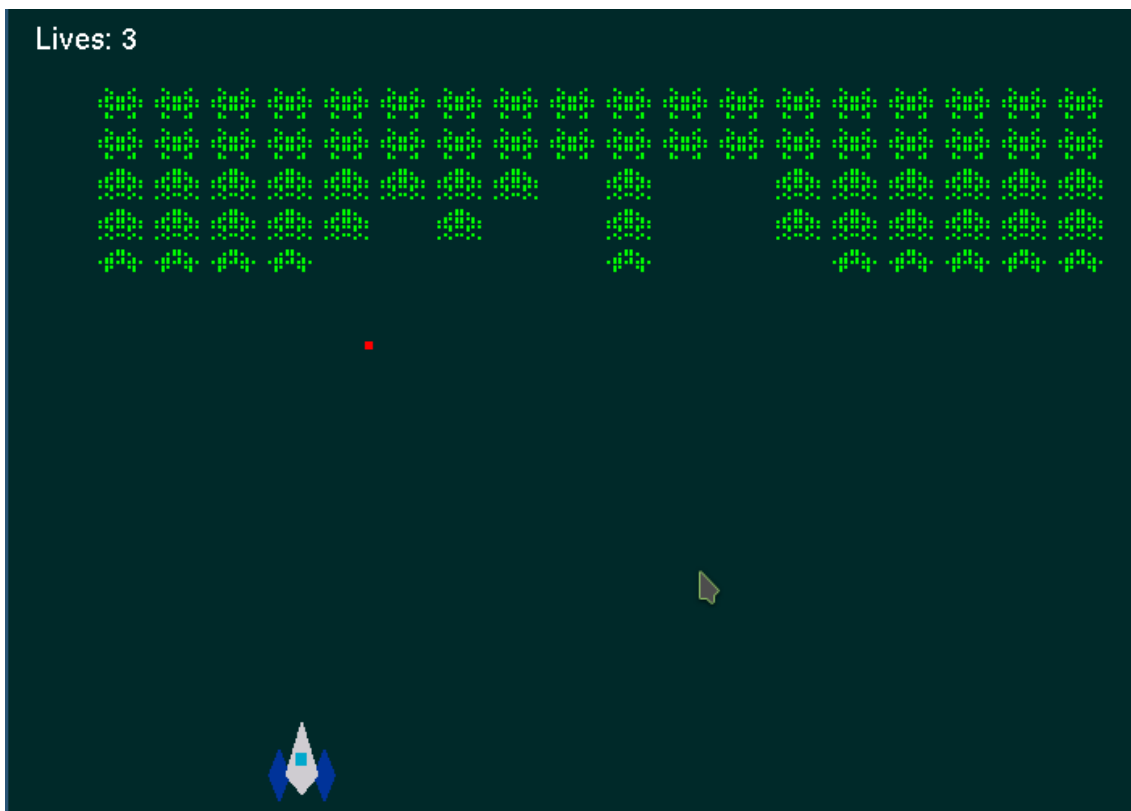
```

Código 2: código de desenho na tela.

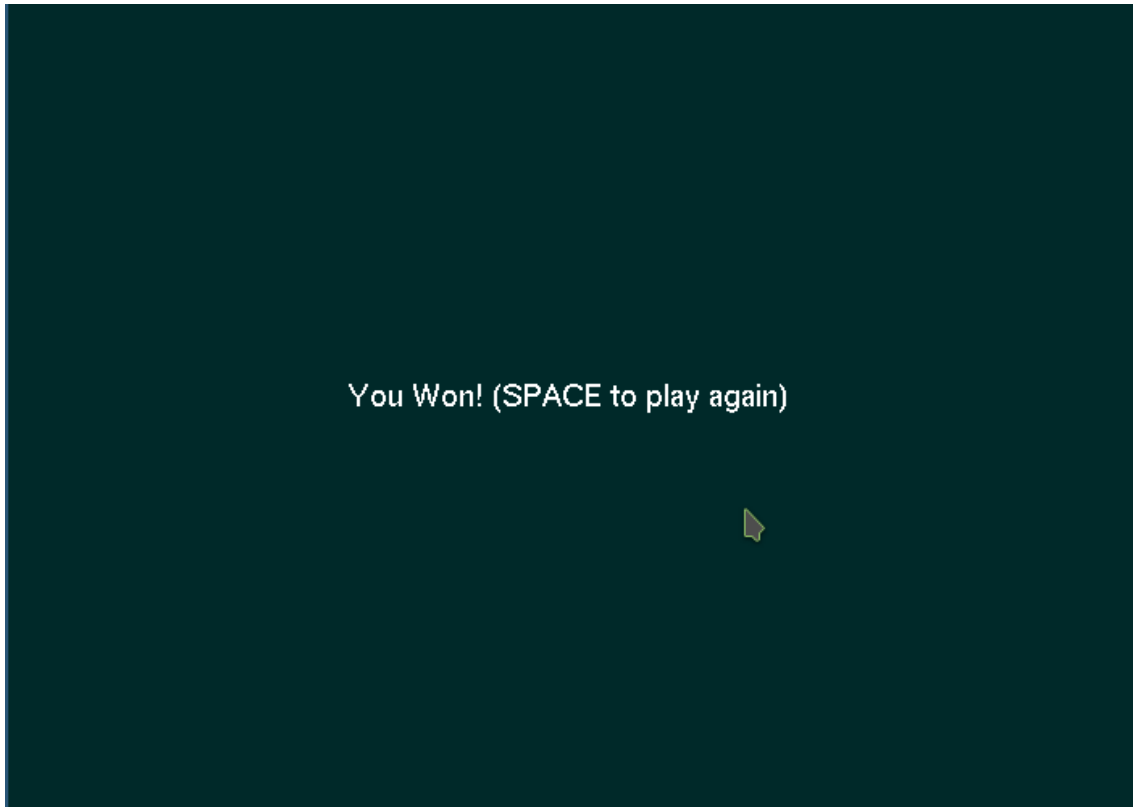
Demonstração de funcionalidades: Ao iniciar o jogo, como vemos na imagem 1, temos a nave, que pode ser movimentada com as teclas direcionais do teclado para a direita e para a esquerda, e os aliens, que automaticamente se moverão e atacam. Pressionar a barra de espaço atirará para cima, caso este tiro acerte um alien, o alien morrerá. Ao perder todas as três vidas (número de vidas mostrado no canto esquerdo superior da tela) ou ao deixar que os aliens atinjam a parte de baixo da tela (o mesmo nível em que o jogador se encontra), o jogador perde. Caso o jogador destrua todos os aliens, o jogador ganha. Ao ganhar ou perder, é possível jogar novamente pressionando a barra de espaço. É possível, também, terminar o jogo a qualquer momento, pressionando a tecla “esc”.



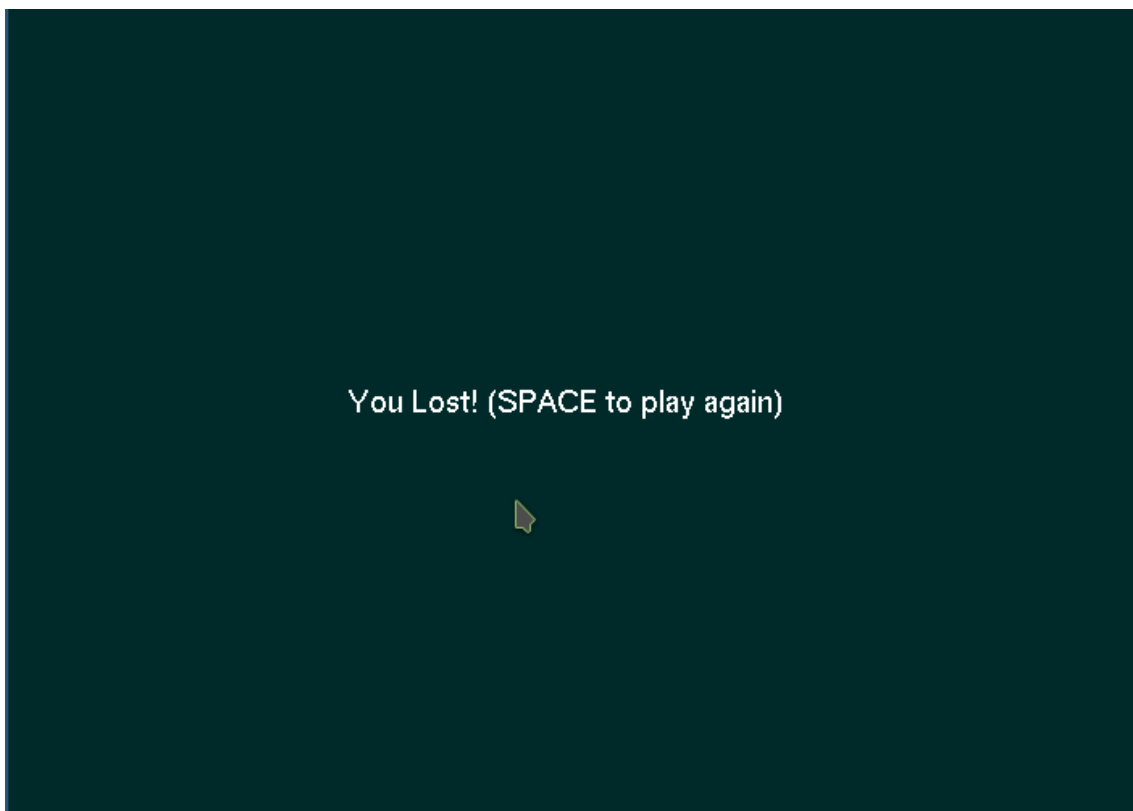
Tela 1: jogo em seu estado inicial.



Tela 2: jogo com alguns aliens destruídos.



Tela 3: tela de vitória.



Tela 4: tela de derrota.