

./oCamera/Create_0.gml

```
/// @description Set up camera
cam = view_camera[0];
follow = oPlayer.id;
view_w_half = camera_get_view_width(cam) * 0.5;
view_h_half = camera_get_view_height(cam) * 0.5;
xTo = xstart;
yTo = ystart;
```

./oCamera/KeyPress_82.gml

```
game_restart();
```

./oCamera/Step_0.gml

```
/// @description Update camera

//Update destination
if (instance_exists(follow)) {
    xTo = follow.x;
    yTo = follow.y;
}

//Update object position
x += (xTo - x) / 25;
y += (yTo - y) / 25;

x = clamp(x,view_w_half,room_width-view_w_half);
y = clamp(y,view_h_half,room_width-view_h_half);

//Update camera view
camera_set_view_pos(cam,x - view_w_half,y - view_h_half);
```

./oEnemy/Create_0.gml

```
vsp = 0; // navpična hitrost
grv = 0.5; // gravitacija
walksp = 4; // hitrost teka
hsp = walksp; // vodoravna hitrost

hp = 10; // število življenj
hitfrom = 0; // smer zadetka

flash = 0;
```

./oEnemy/Draw_0.gml

```
draw_self();

if (flash > 0) {
    flash--;
    shader_set(shWhite);
    draw_self();
    shader_reset();
}
```

./oEnemy/Step_0.gml

```
//Calculate Movement
vsp += grv;

/// VODORAVNI TRK: SE OBRNE
if (place_meeting(x+hsp,y,oWall)) { // if (čez 1 tik bo v zidu)
    while (!place_meeting(x+sign(hsp),y,oWall)) { // dokler je vsaj 1px od zida v smeri gibanja po x osi
        x += sign(hsp); // se premika 1px v smeri gibanja po x osi
    }
    hsp = -hsp; // se obrne (po x osi)
}

/// ROB PLATFORME: SE OBRNE
if (!place_meeting(x+hsp+sign(hsp)*60,y+1,oWall)) { // if (čez 1 tik ne bo več nad zidom)
    hsp = -hsp; // se obrne (po x osi)
}

x += hsp;

//Vertical Collision
if (place_meeting(x,y+vsp,oWall)) {
    while (!place_meeting(x,y+sign(vsp),oWall)) {
        y += sign(vsp);
    }
    vsp = 0;
}
```

```
y += vsp;
```

```
//Animation
if (!place_meeting(x,y+1,owall)) { // (if not on ground)
    sprite_index = sEnemyA;
    image_speed = 0;
    // Up or down image
    if (sign(vsp) > 0) image_index = 1; else image_index = 0;
}
else { // is on ground
    image_speed = 1;
    if (hsp == 0) {
        sprite_index = sEnemy;
    }
    else {
        sprite_index = sEnemyR;
    }
}
if (hsp != 0) image_xscale = sign(hsp);
```

./oEnemy/Step_2.gml

```
/// SMRT
if (hp <= 0) { // if (nasprotnik več nima življenj)
    with (instance_create_layer(x,y,layer,oEnemyCorpse)) { // ustvari truplo na isti poziciji
        direction = other.hitfrom;
        hsp = lengthdir_x(5,direction);
        vsp = lengthdir_x(5,direction) - 3;
        image_xscale = sign(hsp); // obrne truplo v isto smer kot oEnemy
    }
    instance_destroy(); // uniči ta primer oEnemy
}
```

./oEnemyCorpse/Create_0.gml

```
image_speed = 0; // ne animira
global.kills++;

hsp = 3;
vsp = -4;
grv = 0.5;
done = 0;
```

./oEnemyCorpse/Step_0.gml

```
if (done == 0) {
    //Calculate Movement
    vsp += grv;

    //Horizontal Collision
    if (place_meeting(x+hsp,y,owall)) {
        while (!place_meeting(x+sign(hsp),y,owall)) {
            x += sign(hsp);
        }
        hsp = 0;
    }

    x += hsp;

    //Vertical Collision
    if (place_meeting(x,y+vsp,owall)) {
        if (vsp > 0) {
            done = 1;
            image_index = 1;
        }
        while (!place_meeting(x,y+sign(vsp),owall)) {
            y += sign(vsp);
        }
        vsp = 0;
    }

    y += vsp;
}
```

./oEye/Step_0.gml

```
/// @description Insert description here
// You can write your code in this editor
```

./oEyeLeft/Create_0.gml

```
firingdelay = 0; // zakasnitev do strela
```

./oEyeLeft/Draw_0.gml

```
draw_self();

if (oPlayer.pflash > 0) {
    shader_set(shWhite);
    draw_self();
    shader_reset();
}
```

./oEyeLeft/Step_0.gml

```
/// IZSTRELI LASER VSAKIH 13 TIKOV
firingdelay -= 1; // zmanjša zakasnitev za 1
if (mouse_check_button(mb_left)) // if (pritisnjen Levi gumb na miški)
    && (firingdelay < 0) { // && zakasnitev je manj kot 0
        firingdelay = 13; // nastavi zakasnitev na 13
        with (instance_create_layer(x,y,"Lasers",oLaser)) { // ustvari Laser
            speed = 20; // hitrost Laserja
            direction = other.image_angle + random_range(-3,3); // smer Laserja je enaka smeri slike očesa
            image_angle = direction; // smer slike Laserja je enaka smeri slike očesa
        }
}

//Makes sure Eyes don't turn upside-down
if (image_angle > 90) && (image_angle < 270) {
    image_yscale = -1;
}
else {
    image_yscale = 1;
}
```

./oEyeLeft/Step_2.gml

```
// Pozicija glede na oPlayer
x = oPlayer.x - 21; // 21px levo od oPlayer
y = oPlayer.y; // na isti višini

// sledi kazalcu miške
image_angle = point_direction(x,y,mouse_x,mouse_y);
```

./oEyeRight/Create_0.gml

```
firingdelay = 0;
```

./oEyeRight/Draw_0.gml

```
draw_self();

if (oPlayer.pflash > 0) {
    shader_set(shWhite);
    draw_self();
    shader_reset();
}
```

./oEyeRight/Step_0.gml

```
//Shoots a Laser Every 11 Frames
firingdelay -= 1;
if (mouse_check_button(mb_left)) && (firingdelay < 0) {
    firingdelay = 11;
    with (instance_create_layer(x,y,"Lasers",oLaser)) {
        speed = 20;
        direction = other.image_angle + random_range(-3,3);
        image_angle = direction;
    }
}

//Makes sure Eyes don't turn upside-down
if (image_angle > 90) && (image_angle < 270) {
    image_yscale = -1;
}
else {
    image_yscale = 1;
}
```

./oEyeRight/Step_2.gml

```
//Placement Relative to Player
x = oPlayer.x + 21;
y = oPlayer.y;
```

```
//Follows Cursor
image_angle = point_direction(x,y,mouse_x,mouse_y);
```

```
./oFlag/Collision_6aaf4652-920a-4477-9e33-070c8f903434.gml
```

```
winner = 1;
```

```
./oFlag/Create_0.gml
```

```
/// @description Insert description here
// You can write your code in this editor
winner = 0;
```

```
./oGameOver/Draw_64.gml
```

```
draw_set_color(c_white);
draw_set_font(ComicSans);

if ((instance_exists(oPlayerCorpse) || (oPlayer.outofbounds==1)) && oFlag.winner==0) {
    draw_text_transformed(800, 550, "SCORE: " + string(global.kills), 2,2,0);
    draw_text_transformed(800, 850, "Press [R] to restart!", 1,1,0);
    draw_set_color(c_red);
    draw_text_transformed(500, 400, "YOU DIED ':'c", 4,4,0);
}

if (oFlag.winner==1) {
    draw_text_transformed(800, 550, "SCORE: " + string(global.kills), 2,2,0);
    draw_text_transformed(800, 850, "Press [R] to restart!", 1,1,0);
    draw_set_color(c_lime);
    draw_text_transformed(400, 400, "YOU WON \\(^w^)/", 4,4,0);
}
```

```
./oHPCount/Create_0.gml
```

```
/// DOGODEK Create
image_speed = 0; // ne animira
image_index = 3; // začne na 4. sliki

follow = oCamera; // sledi kameri

xTo = xstart; // pozicija ob nastanku
yTo = ystart; // pozicija ob nastanku
```

```
./oHPCount/Step_0.gml
```

```
/// DOGODEK Step
image_index = global.hp; // indeks slike = št. življenj

xTo = follow.x; // nastavi x na x kamere
yTo = follow.y; // nastavi y na y kamere

x += (xTo - x) - 948; // postavi na desni rob kamere
y += (yTo - y) - 528; // postavi na zgornji rob kamere

if (instance_exists(follow)) {}
```

```
./oKillCount/Create_0.gml
```

```
global.kills = 0;
```

```
./oKillCount/Draw_64.gml
```

```
/// IZPIŠE TOČKE
if instance_exists(oPlayer) // if (igralec obstaja)
    && (global.kills > 0) { // IN (je vsaj en nasprotnik mrtev)
        draw_set_color(c_white); // uporabi belo barvo
        draw_set_font(ComicSans); // določi pisavo
        draw_text_transformed // izpiše v zgornjem desnem kotu ...
            (1480, 12, string(global.kills) // ... število mrtvih nasprotnikov ...
            + " Pointless Murders >:3c", 1,1,0); // ... in tekst
    }
}
```

```
./oLaser/Collision_36de0a7c-7208-44d8-b6fe-38128b6231b6.gml
```

```
/// TRK Z NASPROTNIKOM oEnemy
with (other) { // obratuje z oEnemy
    hp--; // zmanjša hp za 1
    hitfrom = other.direction; // hitfrom je enak smeri ...
} // ... od oEnemy do oLaser

instance_destroy();
```

```
flash = 3;
```

./oLaser/Draw_77.gml

```
if (place_meeting(x,y,oWall)) instance_destroy();
```

./oLaser/Other_7.gml

```
image_speed = 0; // konča animacijo  
image_index = 1; // nastavi drugo sliko
```

./oPlayer/Collision_021aa449-8bd2-4b1f-928a-f154907cd7a9.gml

```
/// TRK Z oEnemy  
damagedelay--; // zniža časovnik za 1  
if (damagedelay < 0) { // if (časovnik je manj kot 0)  
    damagedelay = 30; // nastavi časovnik na 30  
    global.hp--; // zmanjša število življenj za 1  
}  
  
pflash = 5;  
///Smrt  
  
// if (ŠTEVILO ŽIVLJENJ <= 0)  
if (global.hp <= 0) {  
    with (oEyeLeft) instance_destroy(); // uniči oEyeLeft  
    with (oEyeRight) instance_destroy(); // uniči oEyeRight  
  
    instance_change(oPlayerCorpse,true); // spremeni oPlayer v oPlayerCorpse,  
                                                                                   // ohrani ID primera  
  
    direction = point_direction(other.x,other.y,x,y); // zazna smer, ki kaže do sredine oEnemy,  
                                                                                   // s katerim je trčil  
  
    hsp = lengthdir_x(6,direction); // premakne za 6px stran od oEnemy po x osi  
    vsp = lengthdir_y(4,direction) -2; // premakne za 4px stran od oEnemy in 2px navzgor po y osi  
  
    image_xscale = sign(hsp); //spremeni SMER SLIKE, da bo enaka SMERI PREMIKANJA PO X OSI  
}
```

./oPlayer/Create_0.gml

```
hsp = 0; // vodoravna hitrost  
vsp = 0; // navpična hitrost  
grv = 0.5; // gravitacija  
walksp = 6; // hitrost teka  
  
damagedelay = 0; // časovnik  
  
global.hp = 3; // število življenj  
  
outofbounds = 0; // zunaj območja igre  
  
pflash = 0; //
```

./oPlayer/Draw_0.gml

```
draw_self();  
  
if (pflash > 0) {  
    pflash--;  
    shader_set(shWhite);  
    draw_self();  
    shader_reset();  
}
```

./oPlayer/Other_0.gml

```
outofbounds = 1;
```

./oPlayer/Step_0.gml

```
/// BERE VHODNE PODATKE  
  
key_left = keyboard_check(vk_left) // leva tipka  
            or keyboard_check(ord("A")); // tipka A  
  
key_right = keyboard_check(vk_right) // desna tipka  
            or keyboard_check(ord("D")); // tipka D  
  
key_jump = keyboard_check(vk_space) // tipka za presledek  
            or keyboard_check(vk_up) // tipka navzgor
```

```

        or keyboard_check(ord("W")); // tipka W

/// RAČUNA HITROST
var move = key_right - key_left; // določi smer gibanja

hsp = move * walksp; // vodoravna hitrost v smeri gibanja
vsp += grv; // navpični hitrosi doda gravitacijo

/// SKOK
if (key_jump) // if (tipka za skok)
    && ( (place_meeting(x,y+1,oWall) // and ( (1px pod mano je zid)
        && !place_meeting(x,y,oWall) ) // and (nisem znotraj zida))
        or place_meeting(x,y+1,oEnemyCorpse) ) { // or (1px pod mano je truplo)
        vsp = -14; // poveča navpično hitrost oz. skoči
    }

/// VODORAVNI TRK
if (place_meeting(x+hsp,y,oWall)) { // if (čez 1 tik bo v zidu)
    while (!place_meeting(x+sign(hsp),y,oWall)) { // dokler je vsaj 1px od zida v smeri gibanja po x osi
        x += sign(hsp); // se premika 1px v smeri gibanja po x osi
    }
    hsp = 0; // se ustavi (po x osi)
}

/// PREMIK PO VODORAVNI OSI
x += hsp; // x koordinati prišteje vodoravno hitrost

/// NAVPIČNI TRK
if (sign(vsp)) { // if (se premika navzdol)
    if ( (place_meeting(x,y+vsp,oWall) // if (čez 1 tik bo v zidu)
        && !place_meeting(x,y,oWall)) // and (trenutno ni v zidu))
        or place_meeting(x,y+vsp,oEnemyCorpse){ // or (čez 1 tik bo v truplu)

        while !(place_meeting(x,y+sign(vsp),oWall) // dokler je vsaj 1px od zida ...
            or place_meeting(x,y+sign(vsp),oEnemyCorpse)) { // ... ali trupla v smeri gibanja po y osi
            y += sign(vsp); // se premika 1px v smeri gibanja po y osi
        }
        vsp = 0; // se ustavi (po y osi)
    }
}

/// PREMIK PO NAVPIČNI OSI
y += vsp; // y koordinati prišteje navpično hitrost

/// Animacija

// if (NI (na tleh) or (na nasprotnikovem truplu))
if (!(place_meeting(x,y+1,oWall) or place_meeting(x,y+1,oEnemyCorpse))) {
    sprite_index = sPlayerA; // določi sprite V ZRAKU
    image_speed = 0; // NI animirano
    if (sign(vsp) > 0) image_index = 1; // if (se premika navzdol) uporabi 2. sliko
    else image_index = 0; // else uporabi 1. sliko
}
else { // if JE na tleh
    image_speed = 1; // JE animirano
    if (hsp == 0) sprite_index = sPlayer; // if (stoji pri miru) določi sprite STOJEČE
    else sprite_index = sPlayerR; // else določi sprite V TEKU
}

if (hsp != 0) image_xscale = sign(hsp); // ko se premika, spremeni SMER SLIKE
// da bo enaka SMERI PREMIKANJA PO X OSI

```

./oPlayerCorpse/Create_0.gml

```

hsp = 0;
vsp = 0;
grv = 0.5;
done = 0;

image_speed = 0;
image_index = 0;

```

./oPlayerCorpse/Step_0.gml

```

if (done == 0) {
    //Calculate Movement
    vsp += grv;

    //Horizontal Collision
    if (place_meeting(x+hsp,y,oWall)) {
        while (!place_meeting(x+sign(hsp),y,oWall)) {
            x += sign(hsp);
        }
        hsp = 0;
    }

    x += hsp;

    //Vertical Collision

```

```
if (place_meeting(x,y+vsp,owall)) {  
    if (vsp > 0) {  
        done = 1;  
        image_index = 1;  
    }  
    while (!place_meeting(x,y+sign(vsp),owall)) {  
        y += sign(vsp);  
    }  
    vsp = 0;  
}  
  
y += vsp;  
}
```