

بسم تعالی

وزارت علوم، تحقیقات و فناوری



دانشگاه علم و فرهنگ

مستندات پروژه گرافیک

موضوع پروژه : هاکی دستی سه بعدی

ارائه دهنده : فائزه اسدی

نام استاد : آقای سامعی

چکیده:

بازی هاکی سه بعدی است که تک بازیکنه است و با حریف سیستمی رقابت میکند.

درجه سختی آن از ۰ تا ۱ قابل تغییر است.

سقف میزان امتیاز برای بردن نیز قابل تغییر میباشد.

با دکمه های صفحه کلید حرکت میکند. دکمه ها عبارتند از A چپ، D راست

رویداد کیبورد:

```
1 window.addEventListener('keyup', function(event) { Key.onKeyUp(event); }, false);
2 window.addEventListener('keydown', function(event) { Key.onKeyDown(event); }, false);
3
4 var Key = {
5   _pressed: {},
6
7   A: 65,
8   ⚡: 68,
9   F: 42,
10  SPACE: 32,
11
12  isDown: function(keyCode) {
13    return this._pressed[keyCode];
14  },
15
16  onKeyDown: function(event) {
17    this._pressed[event.keyCode] = true;
18  },
19
20  onKeyUp: function(event) {
21    delete this._pressed[event.keyCode];
22  }
23 };
```

در این قسمت کداسکی در رویداد ها مشخص میشود. کلید A که کداسکی آن ۶۵ است که باعث

میشود دسته به سمت چپ برود. کلید D با کداسکی ۶۸ هم به سمت راست میرود.

## مشخص کردن متریال های مختلف در صفحه

```
var pedall1Material =  
    new THREE.MeshLambertMaterial(  
        {  
            color: 0x5e4dcc  
        });  
// create the pedall2's material  
var pedall2Material =  
    new THREE.MeshLambertMaterial(  
        {  
            color: 0x0000ff  
        });  
// create the plane's material  
var planeMaterial =  
    new THREE.MeshLambertMaterial(  
        {  
            color: 0x2d4200  
        });  
// create the table's material  
var tableMaterial =  
    new THREE.MeshLambertMaterial(  
        {  
            color: 0x111111  
        });  
// create the pillar's material  
var pillarMaterial =  
    new THREE.MeshLambertMaterial(  
        {  
            color: 0x534d0d  
        });  
// create the ground's material  
var groundMaterial =  
    new THREE.MeshLambertMaterial(  
        {  
            color: 0x833888  
        });
```

---

تمامی متریال های موجود در صفحه از جمله بخش های مختلف میزبازی بصورت تک به تک دارای مشخصات خاص خود میباشند.

```
// set a default position for the camera
// not doing this somehow messes up shadow rendering
camera.position.z = 600;
```

محل قرارگیری دوربین در ارتفاع ثابت و در نقطه ای مشخص میباشد.

محاسبات گل ها

```
function ballPhysics()
{
    // if ball goes off the 'left' side (Player's side)
    if (ball.position.x <= -fieldWidth/2)
    {
        // CPU scores
        score2++;
        // update scoreboard HTML
        document.getElementById("scores").innerHTML = score1 + "-" + score2;
        // reset ball to center
        resetBall(2);
        matchScoreCheck();
    }

    // if ball goes off the 'right' side (CPU's side)
    if (ball.position.x >= fieldWidth/2)
    {
        // Player scores
        score1++;
        // update scoreboard HTML
        document.getElementById("scores").innerHTML = score1 + "-" + score2;
        // reset ball to center
        resetBall(1);
        matchScoreCheck();
    }
}
```

```
var ballDirX = 1, ballDirY = 1, ballSpeed = 2;  
: ball.position.x += ballDirX * ballSpeed;  
: ball.position.y += ballDirY * ballSpeed;  
: if (ballDirY > ballSpeed * 2)  
:     ballDirY = ballSpeed * 2;  
: else if (ballDirY < -ballSpeed * 2)  
:     ballDirY = -ballSpeed * 2;  
:     ballSpeed = 0;  
:     ballSpeed = 0;
```

سرعت توپ در صورت وجود زاویه درپرتاب و برخورد با حاشیه زمین افزایش پیدا میکند.

```

var paddle1DirY = 0, paddle2DirY = 0, paddleSpeed = 3;
:   if (Math.abs(paddle2DirY) <= paddleSpeed)
:   // if the lerp value is too high, we have to limit speed to paddleSpeed
:       if (paddle2DirY > paddleSpeed)
:           paddle2.position.y += paddleSpeed;
:       else if (paddle2DirY < -paddleSpeed)
:           paddle2.position.y -= paddleSpeed;
:       paddle1DirY = paddleSpeed * 0.5;
:       paddle1DirY = -paddleSpeed * 0.5;

```

سرعتی مشخص برای دسته های بازی وجود دارد.

دسته های بازی

```

function playerPaddleMovement()
    // if paddle is not touching the side of table
    if (paddle1.position.y < fieldHeight * 0.45)
        paddle1DirY = paddleSpeed * 0.5;
        paddle1DirY = paddleSpeed * 0.5;
    // else we don't move and stretch the paddle
    paddle1DirY = 0;
    paddle1.scale.z += (10 - paddle1.scale.z) * 0.2;
    paddle1.scale.z += (10 - paddle1.scale.z) * 0.2;
    // if paddle is not touching the side of table
    if (paddle1.position.y > -fieldHeight * 0.45)
        paddle1DirY = -paddleSpeed * 0.5;
        paddle1DirY = -paddleSpeed * 0.5;
    // else we don't move and stretch the paddle
    paddle1DirY = 0;
    paddle1.scale.z += (10 - paddle1.scale.z) * 0.2;
    paddle1.scale.z += (10 - paddle1.scale.z) * 0.2;

```

```

var renderer, scene, camera, pointLight, spotLight;
var renderer, scene, camera, pointLight, spotLight;
: // // create a point light
: pointLight =
:   new THREE.PointLight(0xF8D898);
: pointLight.position.x = -1000;
: pointLight.position.y = 0;
: pointLight.position.z = 1000;
: pointLight.intensity = 2.9;
: pointLight.distance = 10000;
: scene.add(pointLight);
: // add a spot light
:   spotLight = new THREE.SpotLight(0xF8D898);
:   spotLight = new THREE.SpotLight(0xF8D898);
:   spotLight.position.set(0, 0, 460);
:   spotLight.intensity = 1.5;
:   spotLight.castShadow = true;
:   scene.add(spotLight);
: // MAGIC SHADOW CREATOR DELUXE EDITION with Lights PackTM DLC
: // Handles camera and lighting logic
: // we can easily notice shadows if we dynamically move lights during the game

```

از نور Spotlight و Pointlight استفاده شده است.