Добавление теней

Способы упрощения трассировки лучей

* Каждый полигон будем затенять целиком
* Источник света исключительно точечный
* Источник света только один

Допишем класс полигон

class Polygon{

constructor(//блабла){

//блабла

this.center = new Point;

this.R = 1;

}

}

Math3D.js

getVector(a, b){

return {

x: b.x – a.x,

y: b.y – a.y,

z: b.z – a.z,

}

}

multVector(a,b){

return {

x: a.y \* b.z – a.z \*b.y,

y: -a.x \* b.z + a.z \* b.x,

z: a.x \* b.y – a.y \* b.x,

}

}

moduleVector(a){

return Math.sqrt(a.x\*\*2 + a.y\*\*2 + a.z\*\*2);

}

// Метод считает центр каждой фигуры и записывает в свойство центра полигона

calcCenter(surface){

//calcIllumination поможет

}

calcRadius(surface){

const points = surface.points;

surface.polygons.forEach(polygon =>{

const center = polygon.center;

const p1 = points[polygon].points[0]];

const p2 = points[polygon].points[1]];

const p3 = points[polygon].points[2]];

const p4 = points[polygon].points[3]];

polygon.R = this.moduleVector(this.getVector(center, p1)) + this.moduleVector(this.getVector(center, p2) + this.moduleVector(this.getVector(center, p3) + this.moduleVector(this.getVector(center, p4);

});

}

calcShadow(polygon, scene, LIGHT){

const result = {isShadow:false};

const M1 = polygon.center;

const r = polygon.R;

const S = this.getVector(M1, LIGHT);

scene.forEach((surface, index) => {

if (polygon.index === index) return;

surface.polygons.forEach(polygon2 => {

const M0 = polygon2.center;

if (M1.x === M0.x && M1.y === M0.y && M1.z === M0.z) return;

if (polygon2.lumen < polygon.lumen) return;

const dark = this.moduleVector(this.multVector(this.getVector(M0,M1), S))/this.moduleVector(S);

if (dark < r) {

result.isShadow = true;

result.dark = dark / 1.3;

}

});

});

return result;

}

Graph3D.js

renderScene(){

const polygons = [];

this.scene.forEach((surface, index) =>

this.math3D.calcCenter(surface);

this.math3D.calcRadius(surface);

this.math3D.calcDistance(surface, this.WIN.CAMERA, ‘distance’);

this.math3D.calcDistance(surface, this.LIGHT, ‘lumen’);

surface.polygons.forEach(polygon => {

polygon.index = index;

polygons.push(polygon);

});

});

this.Math3D.sortByArtistAlgorithm(polygons);

polygons.forEach(polygon => {

//const points = […]

let {r,g,b} = polygon.color;

const (isShadow, dark) = this.math3d.calcShadow(poltgon, this.scene, this.LIGHT);

const lumen = this.math3d.calcIllumination(polygon.lumen, this.LIGHT.lumen \* (isShadow?dark:1));

//блабла

Домашка: Реализовать код

\*Не рисовать невидимые полигоны

\*\* Фигура не должна отбрасывать на себя тень