**Министерство образования Республики Беларусь**

**Учреждение образования**

**«Белорусский государственный университет информатики   
и радиоэлектроники»**

Кафедра интеллектуальных

информационных технологий

**Отчёт**

по лабораторной работе № 4

по курсу «Графический интерфейс интеллектуальных систем»

Выполнили:

студенты гр. 821701

Бозбей М.С.

Михневич Е.Д.

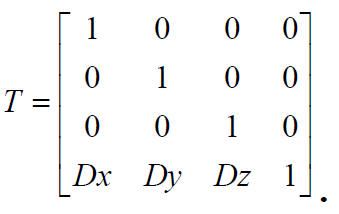
Минск 2011

**Цель**

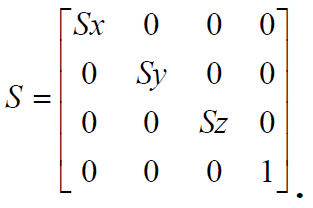
Изучить базовые геометрические операции (поворот, перемещение, скалирование), их математическое и векторное представление. Реализовать функции геометрических преобразований в двумерном и трехмерном пространстве, а также перспективные преобразования.

**Теоритические сведения**

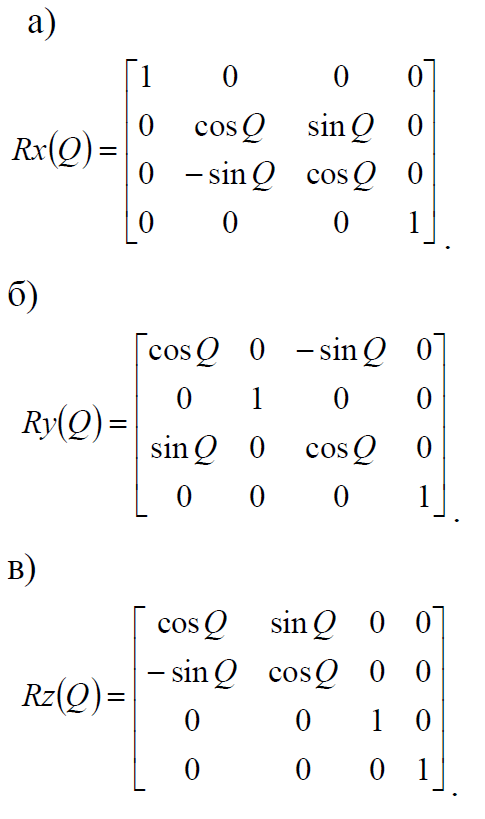
В трехмерном случае есть пять базовых операций:  
1) Перемещение



2) Скалирование



3) Вращение вокруг оси



**Исходный код**

this.zRotate = $M([

[Math.cos(angle), -Math.sin(angle), 0],

[Math.sin(angle), Math.cos(angle), 0],

[0, 0, 1]]);

this.xRotate = $M([

[1, 0, 0],

[0, Math.cos(angle), -Math.sin(angle)],

[0, Math.sin(angle), Math.cos(angle)]]);

this.yRotate = $M([

[Math.cos(angle), 0, Math.sin(angle)],

[0, 1, 0],

[-Math.sin(angle), 0, Math.cos(angle)]]);

this.rotationMatrix = this.xRotate.multiply(this.yRotate.multiply(this.zRotate));

render: function() {

var start;

var end;

for (i = 0; i < this.vertices.length; i++) {

start = this.offset.add(this.scaleMatrix.multiply(this.rotationMatrix.multiply(this.project(this.vertices[i]))));

if (i == 3 || i == 7) {

end = this.offset.add(this.scaleMatrix.multiply(this.rotationMatrix.multiply(this.project(this.vertices[i-3]))));

} else {

end = this.offset.add(this.scaleMatrix.multiply(this.rotationMatrix.multiply(this.project(this.vertices[i+1]))));

}

var wuLine = new BresenhamLine({

pointStart: new Point({x: Math.round(start.e(1)), y: Math.round(start.e(2))}),

pointEnd: new Point({x: Math.round(end.e(1)), y: Math.round(end.e(2))}),

color: 'red'

});

this.canvas.draw(wuLine);

}

for (i = 0; i < 4; i++) {

start = this.offset.add(this.scaleMatrix.multiply(this.rotationMatrix.multiply(this.project(this.vertices[i]))));

end = this.offset.add(this.scaleMatrix.multiply(this.rotationMatrix.multiply(this.project(this.vertices[i+4]))));

var wuLine = new BresenhamLine({

pointStart: new Point({x: Math.round(start.e(1)), y: Math.round(start.e(2))}),

pointEnd: new Point({x: Math.round(end.e(1)), y: Math.round(end.e(2))}),

color: 'red'

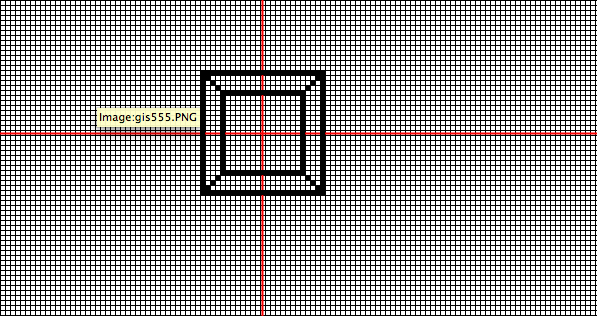
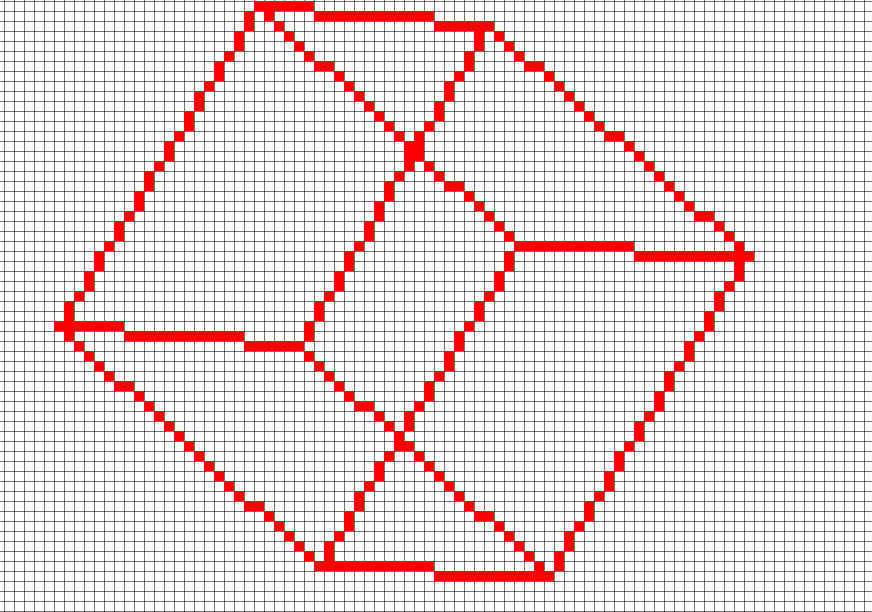
});

this.canvas.draw(wuLine);

}

},

**Пример работы**

**Выводы**

В ходе лабораторной работы были изучены основные механизмы реализации базовых геометрических операций, были реализованы методы для работы с матрицами, константные множители для каждого вида преобразований. Был реализован графический интерфейс, который позволяет производить все выше описанные операции с трехмерным кубом.