Государственное учреждение образования

«Белорусский государственный технологический университет»

Кафедра информационных систем и технологий

**Лабораторная работа № 10**

**Изучение аффинных преобразований в пространстве. Изучение принципов построения 3D – изображений. Изучение методов удаления невидимых граней для выпуклого многогранника.**

**Выполнила:**

 студентка 2 курса 2 группы

Кохнович Алина Сергеевна

Минск

2016

**Ход работы**

В ходе данной работы были реализованы функции аффинных преобразований, которые были добавлены в библиотеку LibPyramid:

**CMatrixCreateViewCoord(double r,doublefi,double q)**

CMatrix CreateViewCoord(double r,double fi,double q)

// Создает матрицу пересчета точки из мировой системы координат в видовую

// (r,fi,q)- координата ТОЧКИ НАБЛЮДЕНИЯ(начало видовой системы координат)

// в мировой сферической системе координат( углы fi и q в градусах)

{

double fg=fmod(fi,360.0);

double ff=(fg/180.0)\*pi; // Перевод в радианы

fg=fmod(q,360.0);

double qq=(fg/180.0)\*pi; // Перевод в радианы

CMatrix VM(4,4);

VM(0,0)=-sin(ff); VM(0,1)=cos(ff);

VM(1,0)=-cos(qq)\*cos(ff); VM(1,1)=-cos(qq)\*sin(ff); VM(1,2)=sin(qq);

VM(2,0)=-sin(qq)\*cos(ff); VM(2,1)=-sin(qq)\*sin(ff); VM(2,2)=-cos(qq); VM(2,3)=r;

VM(3,3)=1;

return VM;

}

**CMatrixVectorMult(CMatrix& V1,CMatrix& V2)**

CMatrix VectorMult(CMatrix& V1,CMatrix& V2)

// Вычисляет векторное произведение векторов V1 и V2

//Векторное произведение — это псевдовектор, перпендикулярный плоскости, построенной по двум сомножителям,

//являющийся результатом бинарной операции «векторное умножение» над векторами в трёхмерном Евклидовом пространстве.

//Векторное произведение полезно для «измерения» перпендикулярности векторов — длина векторного произведения двух векторов

//равна произведению их длин, если они перпендикулярны, и уменьшается до нуля, если векторы параллельны либо антипараллельны.

{

int b1=(V1.cols()==1)&&(V1.rows()==3);

int b2=(V2.cols()==1)&&(V2.rows()==3);

int b=b1&&b2;

if(!b)

{

char\* error="VectorMult: неправильные размерности векторов! ";

MessageBox(NULL,error,"Ошибка",MB\_ICONSTOP);

exit(1);

}

CMatrix W(3);

W(0)=V1(1)\*V2(2)-V1(2)\*V2(1);

//double x=W(0);

W(1)=-(V1(0)\*V2(2)-V1(2)\*V2(0));

//double y=W(1);

W(2)=V1(0)\*V2(1)-V1(1)\*V2(0);

//double z=W(2);

return W;

}

**double ScalarMult(CMatrix& V1,CMatrix& V2)**

double ScalarMult(CMatrix& V1,CMatrix& V2)

// Вычисляет скалярное произведение векторов V1 и V2

//Скаля́рное произведе́ние — операция над двумя векторами, результатом которой является число (скаляр),

//не зависящее от системы координат и характеризующее длины векторов-сомножителей и угол между ними.

//Данной операции соответствует умножение длины вектора x на проекцию вектора y на вектор x. Эта операция

//обычно рассматривается как коммутативная и линейная по каждому сомножителю.

{

int b1=(V1.cols()==1)&&(V1.rows()==3);

int b2=(V2.cols()==1)&&(V2.rows()==3);

int b=b1&&b2;

if(!b)

{

char\* error="ScalarMult: неправильные размерности векторов! ";

MessageBox(NULL,error,"Ошибка",MB\_ICONSTOP);

exit(1);

}

double p=V1(0)\*V2(0)+V1(1)\*V2(1)+V1(2)\*V2(2);

return p;

}



