**Лабораторная работа №10**

**Темы:**

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

**Задание.**

1. Задать координаты вершин усеченной пирамиды (рис.1) в мировой декартовой системе координат и положение камеры (наблюдателя) в мировой сферической системе координат .
2. В пакете Mathcad построить изображение пирамиды (рис.1) без удаления невидимых граней, рис. 1а. Построенный рисунок должен моделировать изображение пирамиды в прямоугольной области окна Windowsс координатами:

* – оконные координаты левого верхнего угла области ;
* – оконные координаты правого нижнего угла области ;

1. Создать приложение Windows для изображения:

* усеченной пирамиды без удаления невидимых граней, рис. 1а;
* усеченной пирамиды с удалением невидимых граней, рис.1б.

90°

A1

B1

C1

A1

B1

X

Y

C

A

B

90°

X

C

A

B

Y

Z

Z

C1

|  |  |
| --- | --- |
| Рис. 1а | Рис. 1б |

В режиме изображения пирамиды с удалением невидимых граней противоположные грани (верхнюю и нижнюю) закрасить разными цветами.

Использовать аксонометрическая проекцию фигуры на картинную плоскость.

Изображения строятся в режиме отображенияMM\_TEXT.

Координаты вершин задать в конструкторе по умолчанию.Положение камеры (наблюдателя) задаётся в мировой сферической системе координат. Начальные значения  определяются в конструкторе по умолчанию.

Каждое из изображений пирамиды появляется на экране при выборе соответствующего пункта меню.

Обеспечить изображение фигуры при перемещении камеры по углу  (клавиши «→» и «←») и углу (клавиши «↑» и «↓»).

Обеспечить масштабирование фигуры при изменении размеров окна.

Класс **CPyramid** создается в отдельном файлеLibPyramid

**class**

**{**

**CMatrixVertices;** // Координаты вершин

**void GetRect(CMatrix& Vert,CRectD&RectView);**

// Вычисляет координаты прямоугольника,охватывающего проекцию

// пирамиды на плоскость XY в ВИДОВОЙ системе координат

// Vert - координаты вершин (в столбцах)

// RectView - проекция - охватывающий прямоугольник

**public:**

**CPyramid(); //** Конструкторпоумолчанию

**voidDraw(CDC&dc,CMatrix&P,CRect&RW);**

// Рисует пирамиду С УДАЛЕНИЕМ невидимых ребер

// Самостоятельный пересчет координат из мировых в оконные (MM\_TEXT)

// dc - ссылка на класс CDC MFC

// P - координаты точки наблюдения в мировой сферической системе //координат

// (r,fi(град.), q(град.))

// RW - область в окне для отображения

**voidDraw1(CDC&dc,CMatrix&P,CRect&RW);**

// Рисует пирамиду БЕЗ удаления невидимых ребер

// Самостоятельный пересчет координат из мировых в оконные (MM\_TEXT)

// dc - ссылка на класс CDCMFC

//P - координаты точки наблюдения в мировой сферической системе

//координат

// (r,fi(град.), q(град.))

// RW - область в окне для отображения

**};**

Требуемые функции:

**CMatrixCreateViewCoord(double r,doublefi,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( углыfi и q в градусах)

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

// Вычисляет векторное произведение векторов V1 (3x1) и V2(3x1)

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

// Вычисляет скалярное произведение векторов V1(3x1) и V2(3x1)