***Задание №1***

**Целью работы является ознакомление с основами векторной графики и получение навыков работы с базовыми функциями графического API и трехмерными графическими примитивами.**

**Требуется при помощи стандартных функций бибилиотеки (OpenGL или Direct3D) изобразить указанные объекты и произвести необходимые преобразования.**

**【ЛИ】Задание 30.**

Изобразить каркасный тор и поместить его в каркасный куб. Размеры и местоположение примитивов на экране задать самостоятельно.

1. Промасштабировать тор с коэффициентом 0,3
2. Изобразить конус, сферу и куб, где вершина конуса является центром сферы, куб помещен внутрь сферы (центры куба и сферы совпадают).
3. Переместить сферу таким образом, чтобы ее центр совпал с центром основания конуса

**Задание 32.**

1. Изобразить каркасный тор и сферу одного радиуса. Размеры и местоположение примитивов на экране задать самостоятельно.
2. Выполнить сдвиг отора на dy=30, масштабирование сферы с коэффициентом 0,5..
3. Изобразить конус и куб, где основание конуса является вписанным в верхнюю грань куба.
4. Переместить куб на вершину конуса.

***Задание №2***

**Целью работы является ознакомление с основными функциями API, описывающими свойства материалов объектов и позволяющими задавать параметры источника освещения.**

Требуется разработать программу,изображающую заданный набор из трех предметов с указанными свойствами материалов и параметры источника освещения. При этом в качестве базового набора объектов выступают 3D примитивы, указанные в вашем варианте задания №1. Следует наделить один из объектов свойствами прозрачности (значение параметра должно быть выше 0,5). Другой выбранный объект должен имитировать отполированную поверхность (shininess, значение указывается максимальным). В качестве такого объекта следует выбирать примитивы с выпуклыми поверхностями, например - цилиндр, тор, конус, сферу, чайник. Третий объект должен быть диффузно-рассеивающим, матовым.

В сцене обязательно должен быть как минимум один источник освещения, с возможностью менять его параметры: ***местоположение****,* ***интенсивность, цвет освещения.***

Окончательный этап – текстурирование одного из объектов (матового). Возможно при этом также использовать микроискажение нормалей при помощи bump-mapping

**Пример задания:**

*Например, было в* ***вашем*** *варианте первого задания:*

1. Изобразить каркасный куб и каркасную сферу, расположенные на некотором расстоянии друг от друга.
2. Совместить одну вершину куба и центр сферы.
3. Изобразить тор и цилиндр. Размеры и местоположение примитивов задать самостоятельно.
4. Выполнить последовательно сначала поворот цилиндра вокруг оси Х, а затем растяжение тора в 2 раза.

*Соответственно, вариант второго задания может выглядеть так:*

Базовый набор был – куб, сфера, цилиндр, тор.

Изображаем – полупрозрачную сферу, отполированный тор, текстурированный куб.

Включаем источник света – например, точечный, белый. Показываем его перемещение и, например, изменение цвета освещения.

# *Задание №3.*

## Цель задания - отобразить изменение формы объекта, т.е. осуществить преобразование одного трехмерного объекта в другой. Изменение должно быть плавным, пошаговым, предусмотреть не менее 8 шагов морфирования. Задание выполняется при помощи библиотек OpenGL или DirectX

## Объекты должны изображать *узнаваемые* предметы, однако могут при этом быть комбинацией примитивных форм. В сцену должен быть включен источник освещения.

## Материалы объектов, определяющие отражение света поверхностью объекта, требуется задать самостоятельно.

## *Далее приводятся (как пример) несколько способов морфирования. Реализация приведенных примеров не может быть зачтена как сделанная работа!!! Требуется задать и морфировать собственные объекты!.*

## *1. Моделирование изменений формы при помощи поверхностей Безье.*

Параметрическая поверхность Безье описывается следующим образом:

m n

P(s,t) = ∑ ∑ Cim si (1 – s) m – i  Cjn tj (1 – t) n – j Pi j  ; 0≤t≤1, 0≤s≤1.

i=0 j=0

При помощи опорных точек нужно задать начальную (например, плоскость) и конечную (например, труба) форму объекта и затем при помощи интерполяции создать ряд промежуточных положений для каждой точки. В итоге происходит трансформация исходной поверхности в заранее заданную с показом промежуточных этапов.

Порция поверхности Безье должна задаваться как минимум 16-ю опорными точками. допустимо использовать функции из библиотеки GL (glMap2[fd], glMapGrid2[fd], glEvalMesh2).

Можно также применить несколько участков поверхности Безье, как разрывных, так и сопряженных.

***2. Морфирование какой-либо готовой формы.***

Взяв за основу математическое описание какого-либо стандартного объекта, изменить его форму.

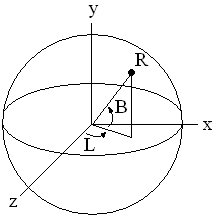
*Например*, параметрическое описание сферы :

x = Fx(B,L) = R cosB sinL

y = Fy(B,L) = R cosB cosL

z = Fz(B) = R sinB

- 90° < B < 90°, 0° < L < 360°



Со сферой можно, *например*, проделать следующие преобразования:

- эллипсоиды:

варьируя значения z (z’ = 0.5z; z’ = 2z), получаем различные формы эллипсоидов.

- «груша»:

z’ = z + 2.5R (z/R – 0.5)2, если z > R/2

z’ = z – иначе.

- «капля»:

z’ = z + R (B/90°)4, если B > 0°

z’ = z – иначе.

- «яйцо»:

z’ = 2z, если z > 0

z’ = z – иначе.

***3. Морфирование при помощи опорных точек.***

Начальный и конечный объекты задаются списками вершин. Затем производится расчет промежуточных положений (траектории движения) для каждой вершины исходного объекта с переходом в вершину конечного объекта. Если количество вершин не совпадает, то на определенном шаге часть из них «исчезает» или, наоборот, «появляется» - при помощи слияния или дублирования вершин.

*Например*, превращаем куб в усеченную пирамиду.

***Курсовая работа***

***МОДЕЛИРОВАНИЕ ДИНАМИЧЕСКОЙ СЦЕНЫ***

**Целью работы является визуализация динамических процессов, имитирующих реальные.**

**Требуется при помощи стандартных функций библиотеки (OpenGL или DirectX) изобразить указанные объекты, затем рассчитать и визуализировать передвижение объекта, имитирующее реальное. Основная задача работы – так рассчитать перемещения объектов, чтобы наблюдателю казалось, что объекты обладают физическими свойствами – а именно, имеют большую массу и являются твердыми. Объекты при передвижении по плоскости должны как-бы «переваливаться» через ребро, затем «падать» соседней гранью на плоскость, не «протыкая» ее при этом, а опираясь на нее. Если объект больших размеров, то при таком движении также будет присутствовать и видимое ускорение. Всё это нужно заложить в алгоритм передвижения.**

**Достаточно реализовать несколько таких «перекатываний» и последующее замедление движения и остановку.**

**Объекты должны быть текстурированы и освещены одним или несколькими источниками света.**

**【ЛИ】Задание 30.**

1. Изобразить текстурированные икосаэдр и сферу, расположенные на плоскости.
2. Реализовать освещение (один источник).
3. Рассчитать и изобразить перекатывание икосаэдра по плоскости вокруг сферы.

**Задание 32.**

1. Изобразить текстурированные куб и расположенный на нем октаэдр.
2. Реализовать освещение (один источник).
3. Рассчитать и изобразить перекатывание октаэдра по верхней грани куба с последующим падением с нее на плоскость.