1. **Когда возникает необходимость в афинных преобразованиях?**

* При составлении сцены из нескольких объектов, где каждый объект имеет свою ориентацию;
* Когда некоторые объекты имеют определенную симметрию;
* Когда может понадобиться рассмотреть объект с различных точек наблюдения и получить изображение с каждой из них;
* Когда в компьютерной анимации несколько объектов должны от кадра к кадру двигаться один относительно другого.

1. **В чем отличия преобразования объекта от преобразования координат?**

**При преобразовании объектов** координаты каждой точки объекта изменяются в соответствии с некоторыми законами, а соответствующая система координат остается неизменной.

**При преобразовании координат** старая система координат преобразуется в новую и все точки объекта получают представление в этой новой системе координат.

1. **Что представляют собой афинные преобразования?**

**Афинное преобразование** – это отображение плоскости или пространства в себя, при котором параллельные прямые переходят в параллельные прямые, пересекающиеся в пересекающиеся, скрещивающиеся в скрещивающиеся.

1. **Какое преобразование называется афинным?**

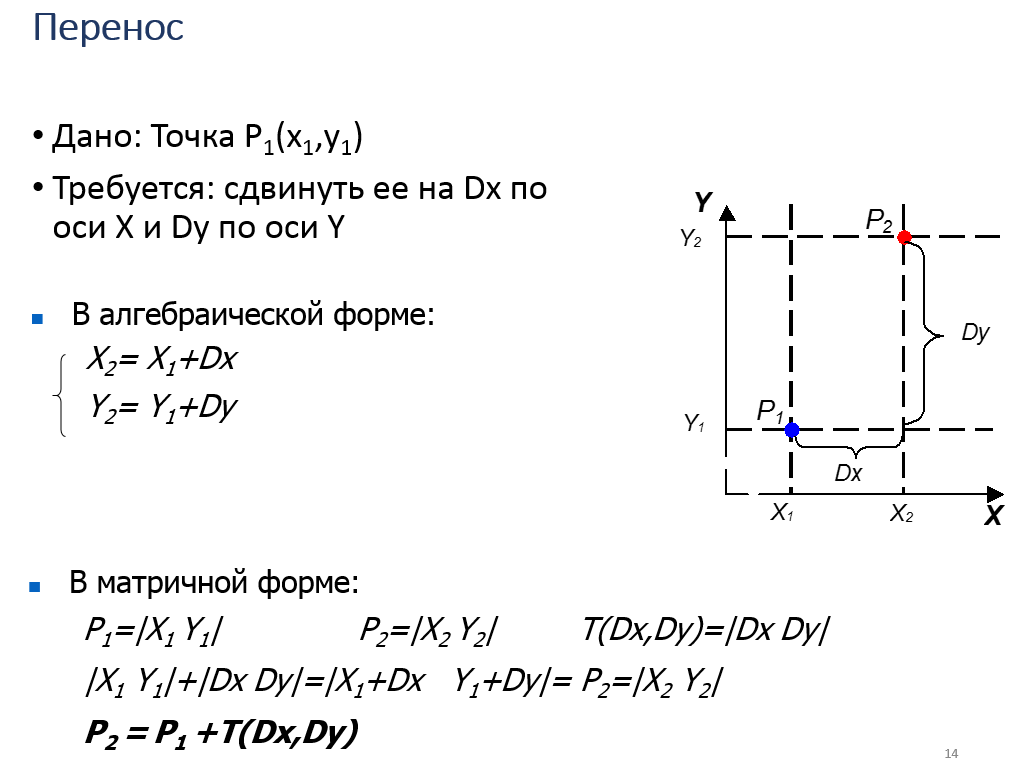
Преобразование называется афинным, если его можно получить следующим образом:

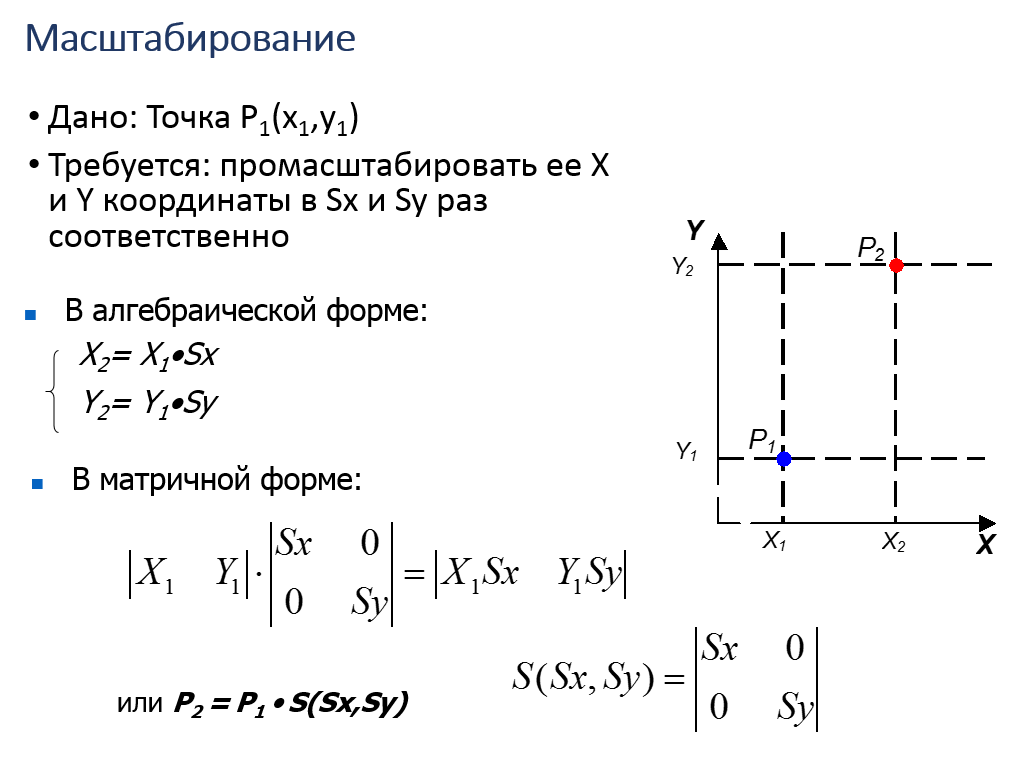
* Выбрать новый базис пространства с новым началом координат ; ;
* Каждой точке x пространства поставить в соответствие точку f(x), имеющую те же координаты относительно “новой” системы координат, что и в x в “старой”.

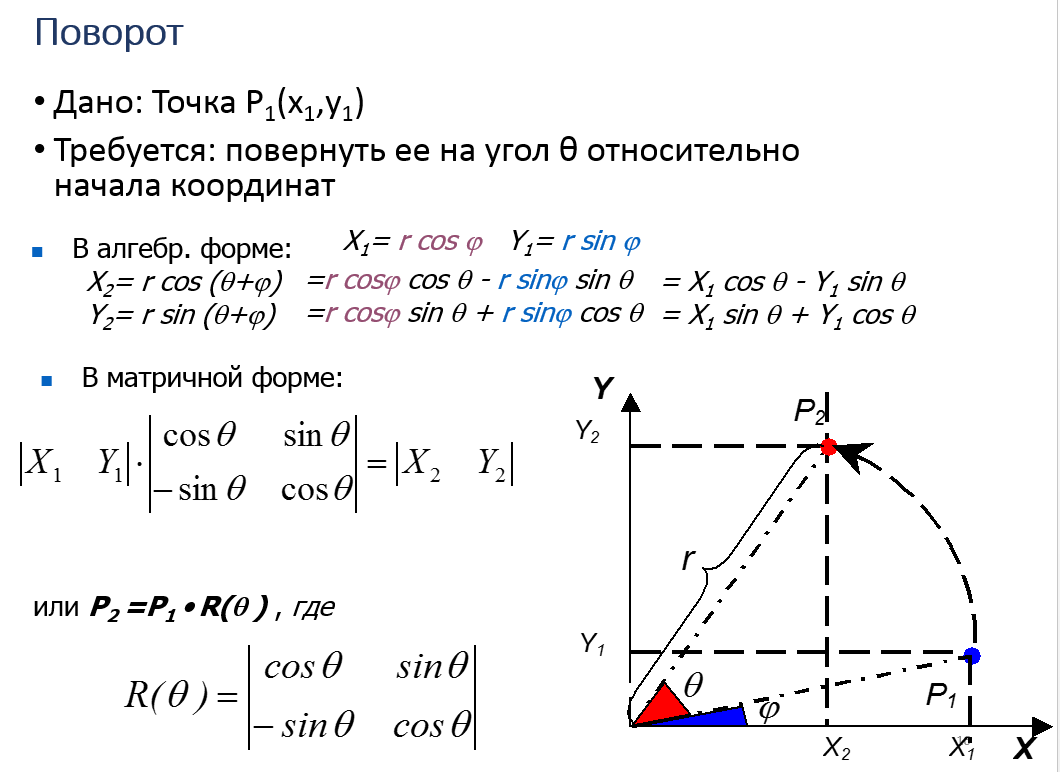
1. **Назовите базовые двухмерные преобразования**

Перенос, масштабирование (относительно начала системы координат), поворот (относительно начала системы координат).

1. **Напишите в матричной и алгебраической форме все двухмерные базовые преобразования**



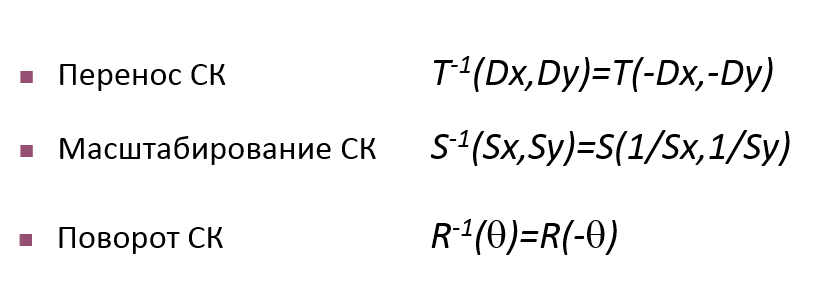




1. **В чем заключается суть обратных преобразований? Напишите формулы обратных преобразований**

Три базовых преобразования (перенос, масштабирование и поворот) имеют и соответствующие им обратные преобразования (операции).

**Обратное преобразование** – это преобразование системы координат, сам объект же остается «неподвижным». Матрица T такого преобразования будет иметь верхний индекс -1.



1. **Для каких целей используется однородная система координат? В чем ее преимущества и недостатки?**

**В неоднородной системе координат** все базовые операции, кроме переноса в матричной форме выражаются через умножение, поэтому последовательность операций невозможно объединить (сгруппировать) в одну операцию.

**В однородной системе координат** все базовые операции в матричной форме выражаются через умножение, что позволяет их объединить (сгруппировать) в одну операцию.

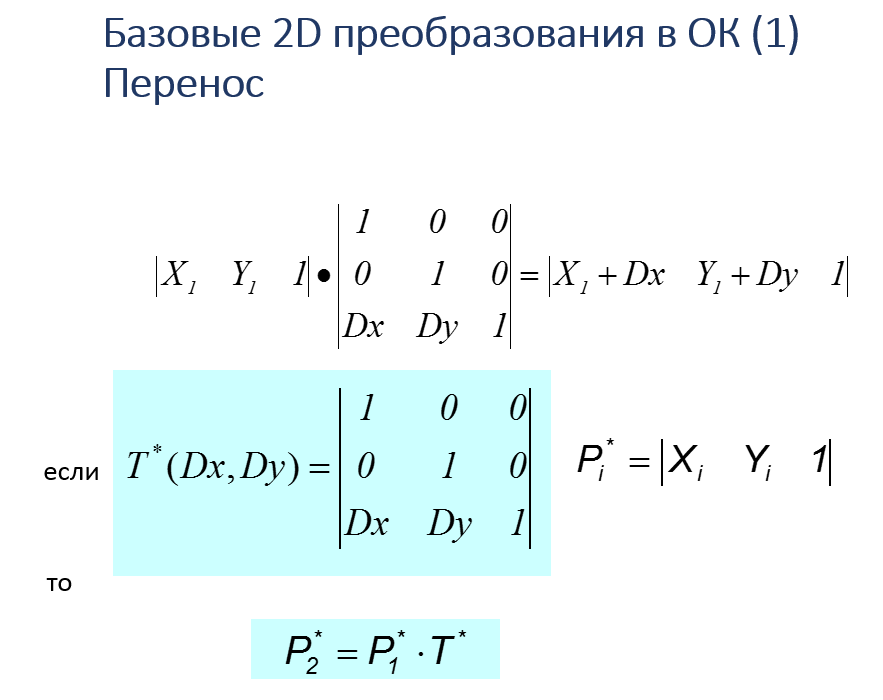
Преимущества:

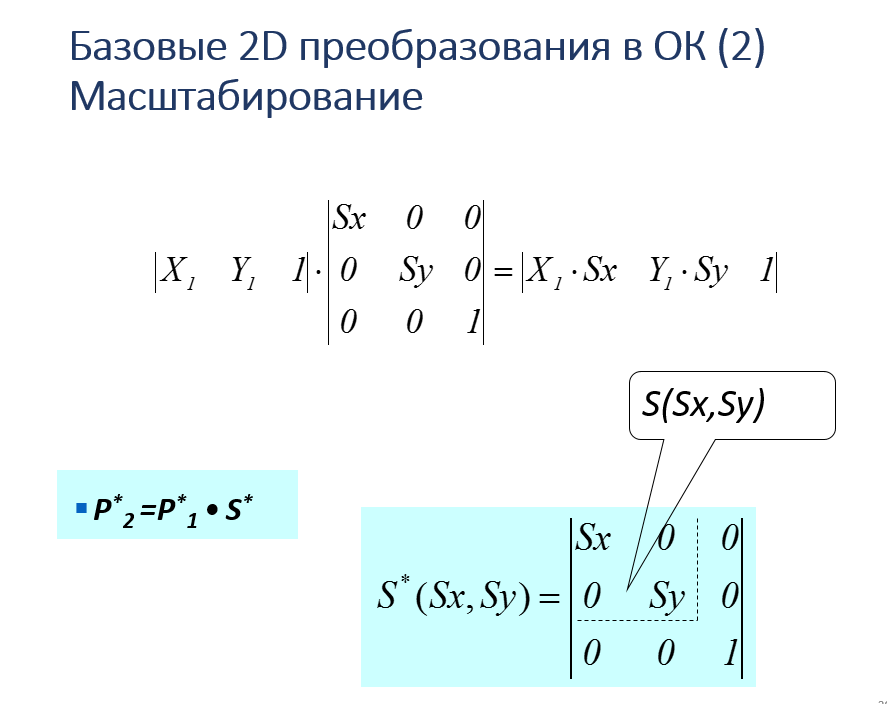
* Инкапсуляция;
* Быстродействие (за счет сокращений при перемножении).

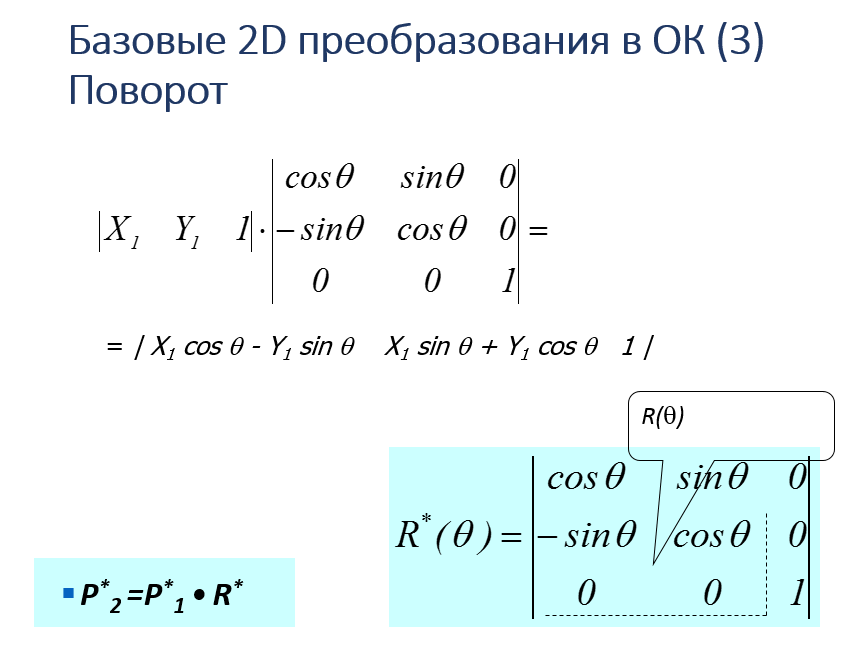
Недостаток:

* Неуниверсальность (для каждого сложного действия необходимо рассчитать свою единую матрицу и запрограммировать свою функцию/подпрограмму).

1. **Напишите в матричной форме всех двухмерные базовые преобразования в однородной системе координат**







1. **Что представляет собой композиция двухмерных преобразований?**

**Композиция** – это объединение последовательности нескольких преобразований в одно, т.е. преобразование произведения матриц в одну результирующую матрицу.

1. **Назовите шаги, которые следует выполнить при организации поворота одной точки относительно другой точки**

* Перенос системы координат в точку P0(x0, y0), после чего центр системы координат будет находиться в точке, относительно которой осуществляется поворот;
* Поворот точки P1 вокруг нового центра системы координат (базовый поворот);
* Перенос системы координат на прежнее место (обратный перенос).

1. **Назовите шаги, которые следует выполнить при организации масштабирования двухмерного объекта относительно заданной точки**

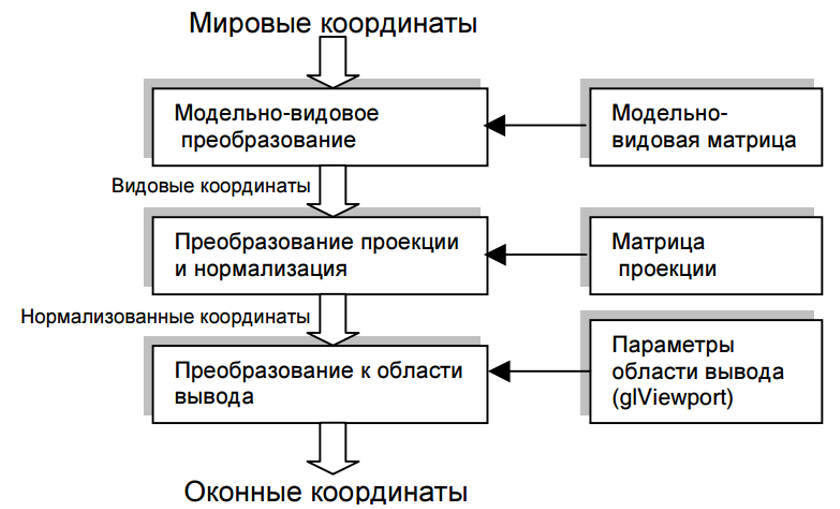
* Перенос системы координат в точку P0(x0, y0), после чего центр системы координат будет находиться в точке, относительно которой осуществляется масштабирование;
* Масштабирование точки P1 относительно нового центра системы координат (базовое масштабирование);
* Перенос системы координат на прежнее место (обратный перенос).

1. **Какие матрицы используются в OpenGL для реализации различных преобразований?**

* **Видовая матрица** определяет преобразования объекта в мировых координатах, такие как параллельный перенос, изменение масштаба и поворот;
* **Матрица проекций** задает, как будут проецироваться трехмерные объекты на плоскость экрана (в оконные координаты);
* **Матрица текстур** определяет наложение текстуры на объект.

Все они имеют размер 4х4.

1. **Назовите последовательность действий, которая используется для отображения объектов сцены в окно приложения OpenGL**



1. **Назовите матрицу, с помощью которых производится преобразования объектов в OpenGL**

Слайд 41

1. **Назовите функции, которые используются при работе с матрицами в OpenGL**

Функция void **glMatrixMode**(GLenum mode), где параметр mode может принимать значения:

GL\_MODELVIEW – модельно-видовая матрица;

GL\_PROJECTION – матрица проекции;

GL\_TEXTURE – матрица текстуры.

Эта функция позволяет выбрать, какую матрицу надо изменить и включает режим работы с соответствующей матрицей.

1. **Назовите функции, которые используются для реализации видовых преобразований**

* **Перенос**: void **glTranslate**[f d](GLtype x, GLtype y, GLtype z);
* **Поворот** против часовой стрелки на угол angle (измеряется в градусах) вокруг вектора (x, y, z): void **glRotate**[f d](GLtype angle, GLtype x, GLtype y, GLtype z);
* **Масштабирование**: void **glScale**[f d](GLtype x, GLtype y, GLtype z).

Все эти функции производят изменения текущей матрицы. В самих функциях строятся матрицы преобразования и затем основная (например, модельно-видовая матрица) умножается на созданную матрицу преобразования, таким образом получается измененная основная (например, модельно-видовая матрица), на которую будут умножаться координаты объектов при вызове команды glVertex\* и некоторых других.

Т.е. функции не изменяют объекты, а изменяют модельно-видовую матрицу. А модельно-видовая матрица в свою очередь хранит коэффициенты, на которые должны умножаться координаты объектов.

1. **Какая функция OpenGL преобразует текущую матрицу в единичную?**

Изначально все матрицы являются единичными (единицы по главной диагонали, остальные нули). При умножении на единичную матрицу координаты объекта не изменяются.

Функция void **glLoadIdentity**(void) заменяет текущую матрицу на единичную.

1. **Какие шаги следует выполнить, чтобы обрабатывать сообщения с клавиатуры в OpenGL?**

Необходимо зарегистрировать функцию, которая будет выполнять обработку клавиш, для этого в GLUT предоставляет две функции:

* void **glutKeyboardFunc** – для обработки “нормальных” клавиш (буквы, цифры, все, что имеет ASCII код), а в качестве параметра указывается функция, которая будет обрабатывать эти нажатия (входными параметрами такой функции являются сам ключ, координаты x и y);
* void **glutSpecialFunc** – для обработки специальных клавиш, таких как PAGE\_UP, PAGE\_DOWN, KEY\_UP, KEY\_DOWN, HOME, END и т.д.