



CUDA Assignment #3

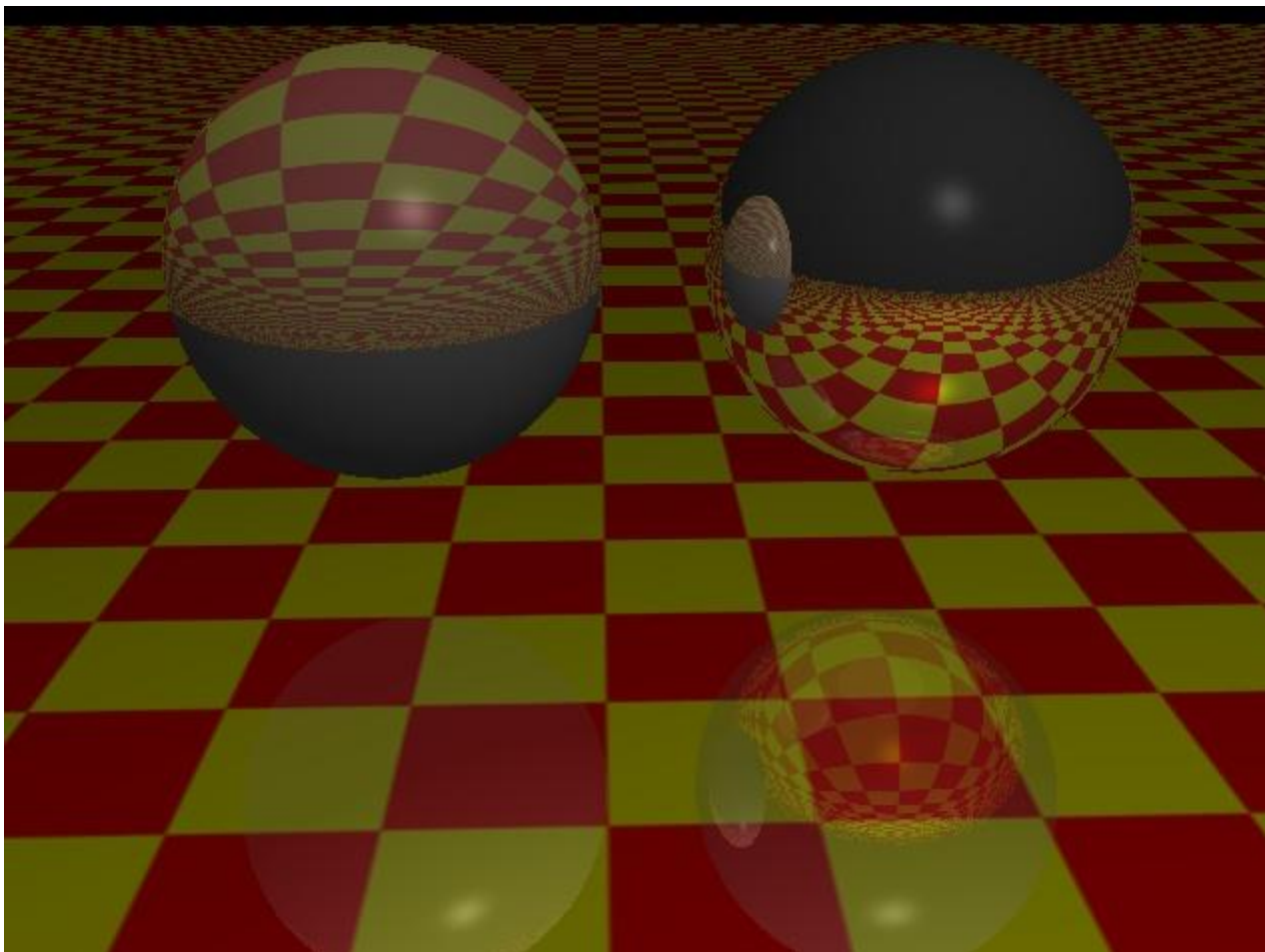
Работа с текстурами, взаимодействие с
OpenGL

Задание на выбор



- ☐ Трассировка лучей: Отражения
- ☐ Трассировка лучей: Преломления

Задание на выбор



Задание на выбор

- Необходимо реализовать один из двух эффектов на выбор:
 - Отражение
 - Преломление
- Сцена:
 - Сферы (минимум 2)
 - С отражением
 - С преломлением
 - Сферы должны быть расположены на расстоянии не превышающим своего диаметра от плоскости
 - Плоскость $z = 0$
 - Можно представить как 2 треугольника
 - **Если выбран вариант с отражением – то плоскость обязательно отражающая**
 - Плоскость обязательно с текстурированием
- Сферы должны двигаться –
 - можно задать простое движение по окружности вокруг
- Замер FPS
 - FPS можно выводить название окна
- Минимальное разрешение экрана: 640x480

Генерация лучей

□ Виртуальный глаз

- Находится в положении `float3 eyePos`
- Смотрит в направлении `float3 eyeDir`
- Имеет вектора ориентации `float3 eyeUp` и `float3 eyeRight`
 - `eyeRight` можно получить как векторное произведение `eyeDir` и `eyeUp`
 - В зависимости от того как взять векторное произведение получится левая или правая тройка векторов
 - Очевидно, что если `eyeUp` и `eyeDir` коллинеарны – то векторное произведение нулевое – это значит что вектор «вверх» выбран не верно.

Генерация лучей

□ Экран

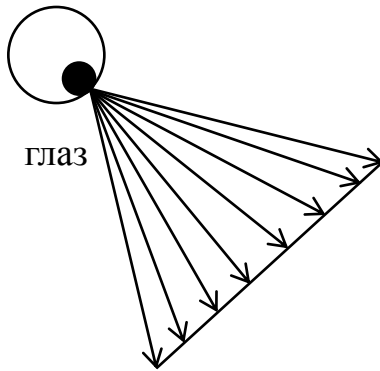
- Шириной `int ScreenW` и высотой `int ScreenH`
- Расстояние до экрана – `float zNear`



Генерация лучей

- Экран

- Через каждый пиксел экрана пускается луч



Пересечение луча и треугольника

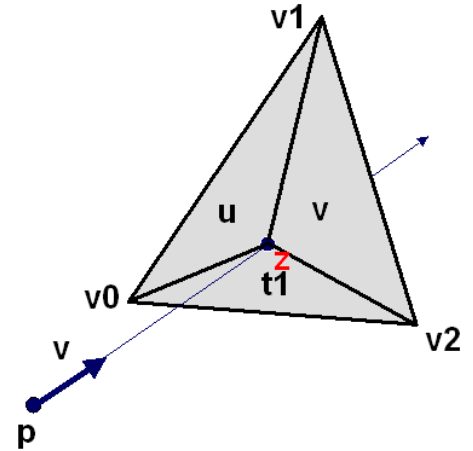
- Оптимизированный вариант
 - Барицентрические координаты
 - $u := u/S, v := v/S, t1 := t1/S$
 - $t1 = 1 - u - v$

$$z(u, v) = (1 - u - v) * v1 + u * v2 + v * v0$$

$$z(t) = p + t * d$$

$$p + t * d = (1 - u - v) * v1 + u * v2 + v * v0$$

- 3 уравнения, 3 неизвестных



Пересечение луча и треугольника

□ Оптимизированный вариант

$$\begin{bmatrix} t \\ u \\ v \end{bmatrix} = \frac{1}{\text{dot}(P, E1)} * \begin{bmatrix} \text{dot}(Q, E2) \\ \text{dot}(P, T) \\ \text{dot}(Q, D) \end{bmatrix}$$

$$E1 = v1 - v0$$

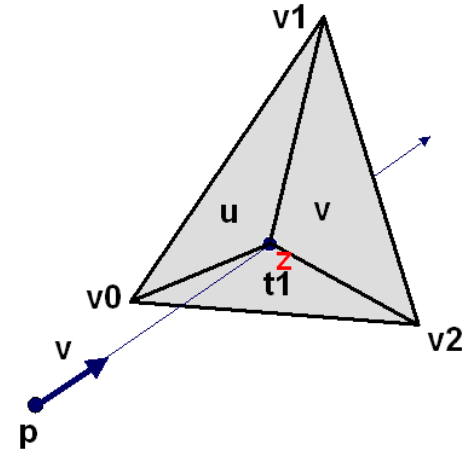
$$E2 = v2 - v0$$

$$T = p - v0$$

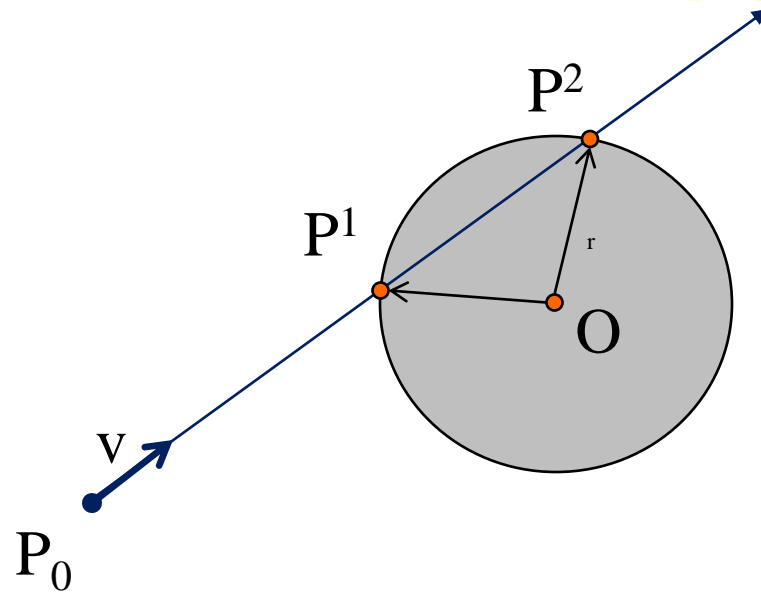
$$P = \text{cross}(D, E2)$$

$$Q = \text{cross}(T, E1)$$

$$D = v$$



Пересечение луча и сферы



Луч: $P = P_0 + tV$

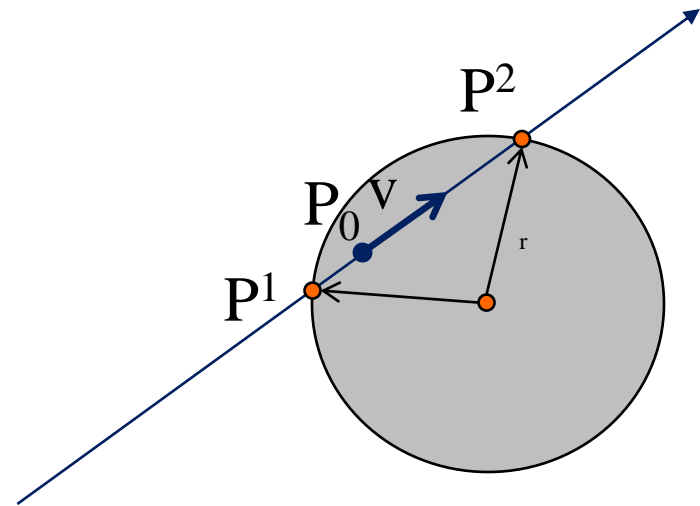
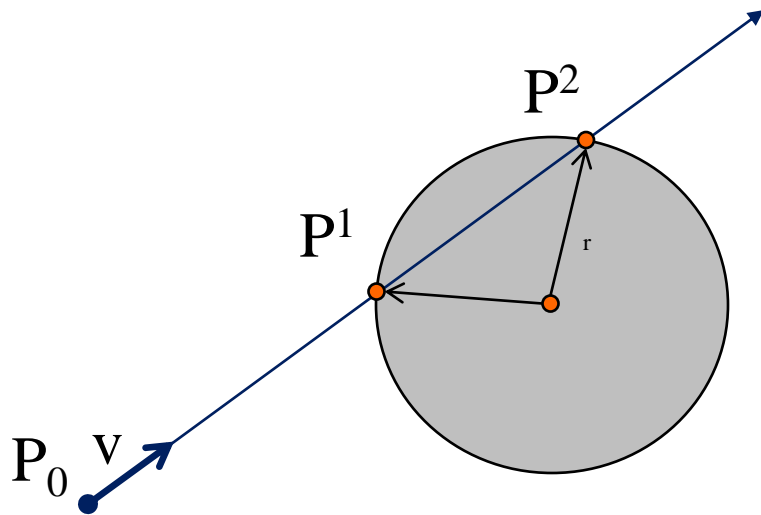
Пересечени: $|P - O| - r^2 = 0$

$$|P_0 + tV - O| - r^2 = 0$$

$$\Rightarrow \begin{cases} at^2 + bt + c = 0 \\ a = 1 \\ b = 2V(P_0 - O) \\ c = |P_0 - C|^2 - r^2 = 0 \end{cases}$$

Пересечение луча и сферы

- При положительном дискриминанте выбирается минимальное положительное t

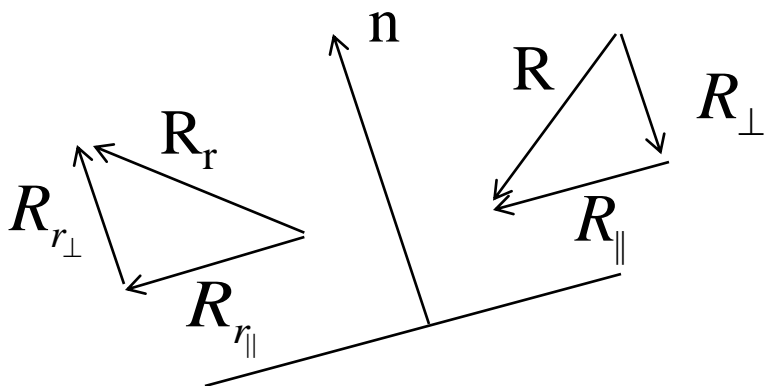


Отражение

□ Посчитать нормаль

□ Для плоскости ее задать просто

□ На сфере [нормаль] = [точка пересечения] – [цент сферы]



$$R = R_{\perp} + R_{\parallel}$$

$$R_r = R_{r\perp} + R_{r\parallel}$$

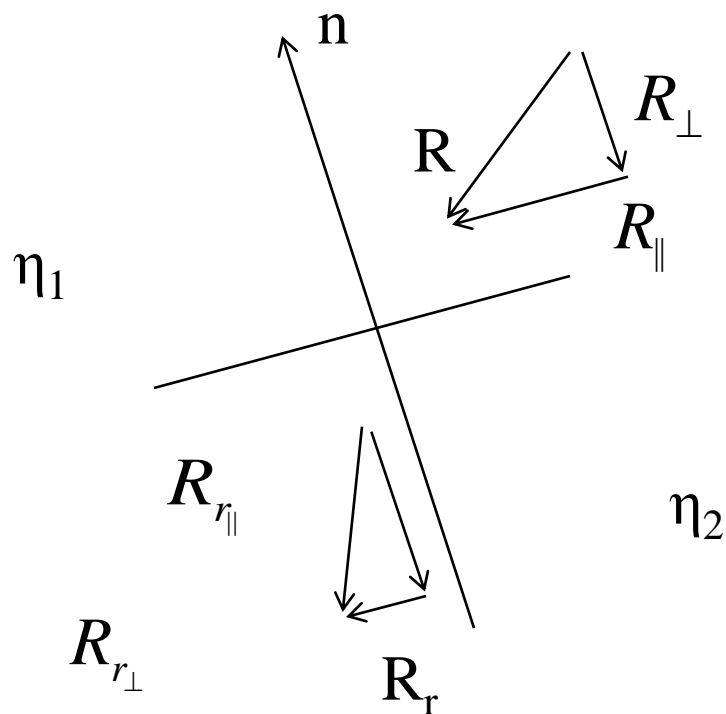
$$R_{\perp} = \text{dot}(R, n)$$

$$R_{\parallel} = R - R_{\perp}$$

$$R_{r\perp} = -R_{\perp}$$

$$R_{r\parallel} = R_{\parallel}$$

Преломление



$$\eta_1 \sin \theta_1 = \eta_2 \sin \theta_2$$

$$R = R_{\perp} + R_{\parallel}$$

$$R_r = R_{r\perp} + R_{r\parallel}$$

$$R_{\perp} = \text{dot}(R, n)$$

$$R_{\parallel} = R - R_{\perp}$$

$$R_{r\perp} = -\sqrt{1 - |R_{r\parallel}|^2} n$$

$$R_{r\parallel} = \frac{\eta_1}{\eta_2} R_{\parallel}$$

Текстуры



- Храните геометрию в текстурах или в разделяемой памяти. Так как геометрии мало, она туда поместится.
- Обязательно текстурировать треугольник
 - $\text{texCoord} = u * \text{texCoordA} + v * \text{texCoordB} + t1 * \text{texCoordC};$
 - Используйте режим нормализованной адресации и wrap для заворачивания текстурных координат
 - Изображение шахматной доски не нужно грузить – его можно создать в отдельном кернеле

Отрисовка



- ☐ Реализовать взаимодействие с OpenGL
- ☐ Копировать цвет в PBO
- ☐ Нарисовать quad на весь экран с текстурой, содержащей результирующий цвет
 - ☐ Копировать цвет в оперативную память и рисовать с помощью `glDrawPixels` – защитано не будет

Вопросы

