



COMPUTER GRAPHICS

ЗАСОБИ ПРОГРАМУВАННЯ КОМП'ЮТЕРНОЇ ГРАФІКИ

ЛЕКЦИЯ 13

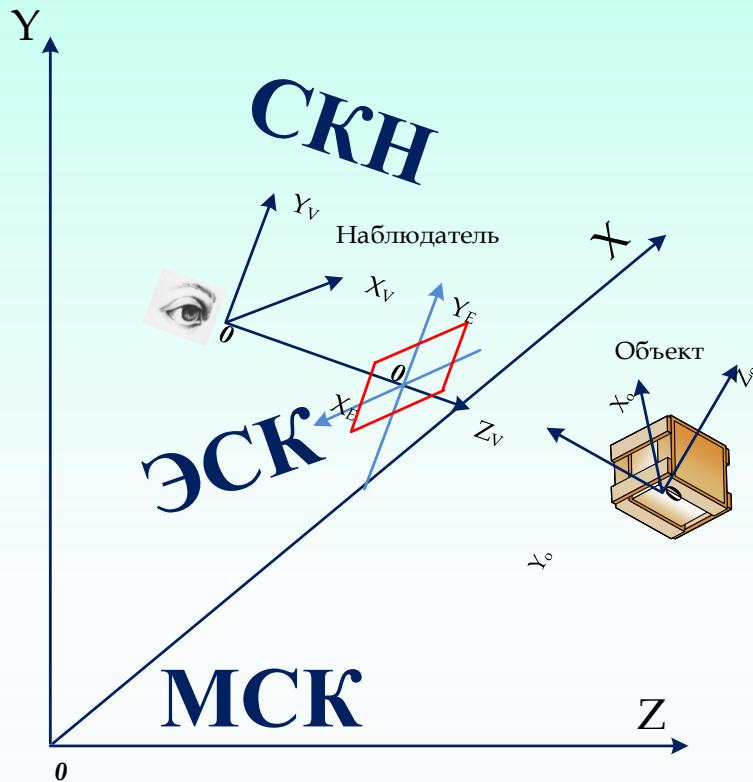
ПРОЕКТИРОВАНИЕ. УДАЛЕНИЕ НЕВИДИМЫХ ПОВЕРХНОСТЕЙ

- Перспективные преобразования
- Центральная перспективная проекция.
- Перспективное преобразование в обобщенных координатах.
- Приведение к координатам экрана дисплея
- Задача удаления невидимых поверхностей.
 - Отбрасывание нелицевых граней
 - Алгоритм художника
 - Z-буфер

ПРОЕКТИВНОЕ ПРЕОБРАЗОВАНИЕ

Мир 3-D \leftrightarrow Дисплей 2D

Как перейти?



Проецирование — это процесс получения отображения объекта на какой-либо поверхности (плоской, сферической, конической, ...) с помощью проецирующих лучей.

Плоские проекции:

- Центральное проецирование
- Плоское проецирование

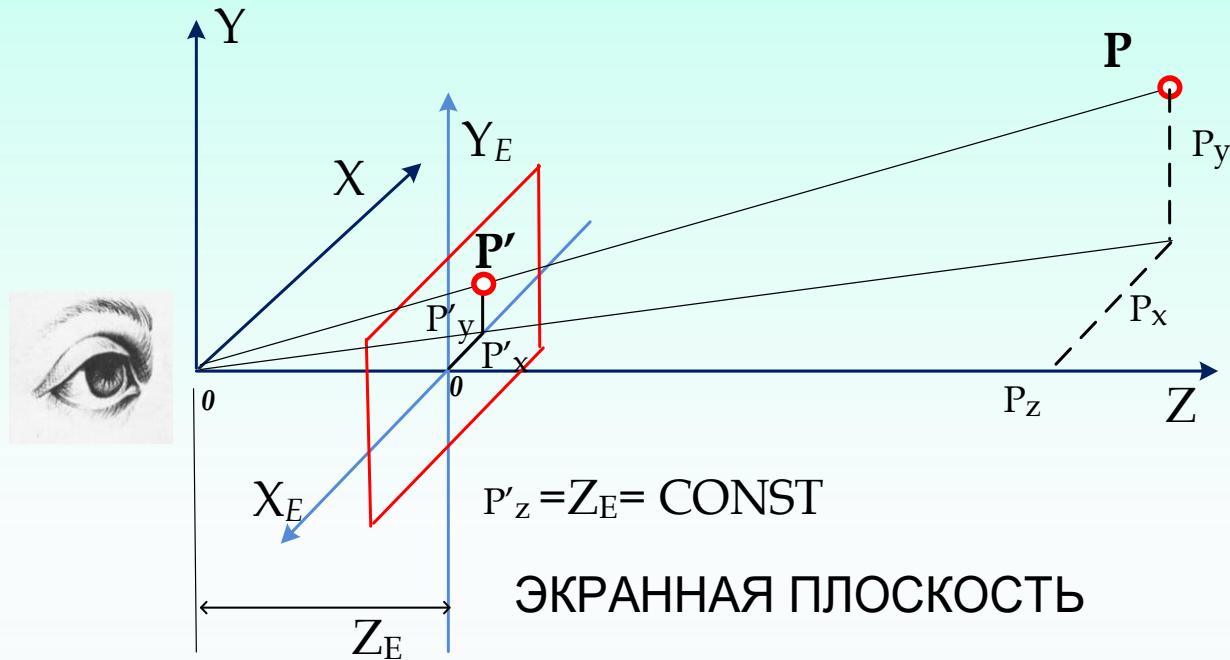
ПРОЕКТИВНОЕ ПРЕОБРАЗОВАНИЕ

- **Проекция:** объекты в СК размерности n , в объект в СК размерностью меньшей, чем n (у нас из 3 в 2).
- В нашем случае: переход от **трехмерных** объектов в СКН к их изображениям на **плоскости** (дисплея) называется **перспективной проекцией**.
- Перспективная проекция является частным случаем **центрального проективного преобразования**.

ЦЕНТРАЛЬНАЯ ПЕРСПЕКТИВНАЯ ПРОЕКЦИЯ

- Центральная перспективная проекция приводит к визуальному эффекту, подобному тому, который дает зрительная система человека.
- При этом наблюдается эффект перспективного укорачивания, когда размер проекции объекта изменяется обратно пропорционально расстоянию от центра проекции до объекта.

ЦЕНТРАЛЬНАЯ ПЕРСПЕКТИВНАЯ ПРОЕКЦИЯ



$$P'_x = -\frac{Z_E}{P_z} P_x$$

$$P'_y = \frac{Z_E}{P_z} P_y$$

Переход из обобщенных координат в декартовы

Если 4 – я координата не равна 0

$$\rightarrow P = \begin{bmatrix} X \\ Y \\ Z \\ w \end{bmatrix}$$

выполняется нормализация $\rightarrow P = \begin{bmatrix} X/w \\ Y/w \\ Z/w \end{bmatrix}$

ЦЕНТРАЛЬНАЯ ПЕРСПЕКТИВНАЯ ПРОЕКЦИЯ. ОБОБЩЕННАЯ МАТРИЦА ПРЕОБРАЗОВАНИЯ

$$\Pi = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1/z_E & 0 \end{bmatrix} \quad P = \begin{bmatrix} P_x \\ P_y \\ P_z \\ 1 \end{bmatrix}$$

$$P' = \Pi * P = \begin{bmatrix} P_x \\ P_y \\ 0 \\ P_z/z_E \end{bmatrix}, \quad P' = \begin{bmatrix} -\frac{z_E}{P_z} P_x \\ \frac{z_E}{P_z} P_y \end{bmatrix}$$

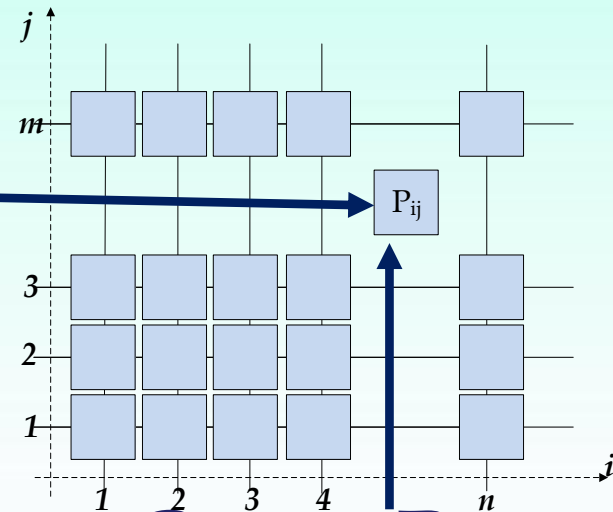
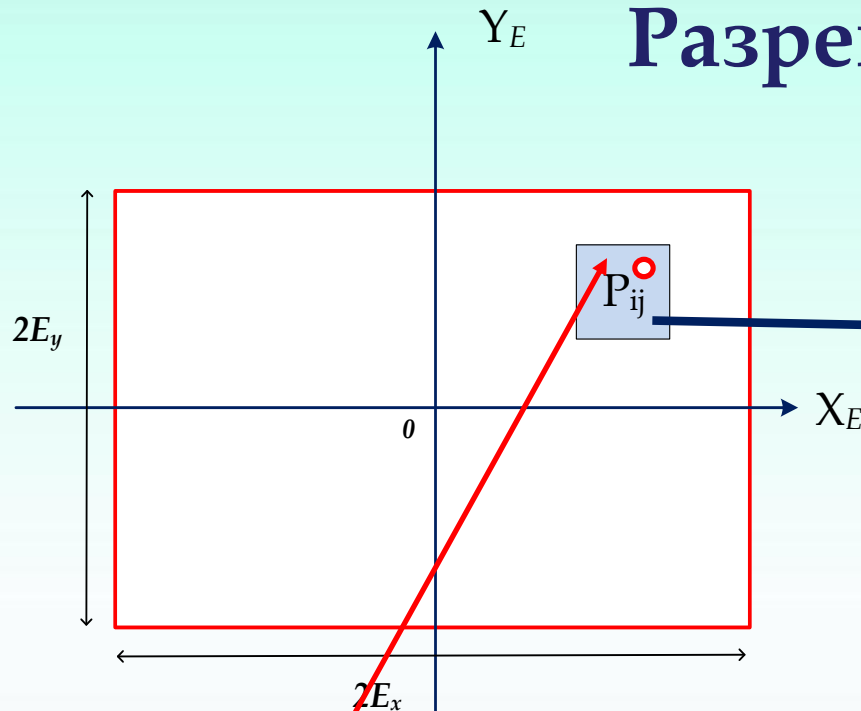
ПРОЕКТИВНОЕ ПРЕОБРАЗОВАНИЕ. ОБОБЩЕННАЯ МАТРИЦА ПРЕОБРАЗОВАНИЯ (Open GL)

$$\Pi = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/z_E & 0 \end{bmatrix} \quad P = \begin{bmatrix} P_x \\ P_y \\ P_z \\ 1 \end{bmatrix}$$

$$P'' = \Pi * P = \begin{bmatrix} -P_x \\ P_y \\ P_z \\ P_z/z_E \end{bmatrix}, \quad P'' = \begin{bmatrix} -\frac{z_E}{P_z} P_y \\ \frac{z_E}{P_z} P_y \\ P_z \end{bmatrix}$$

ПРИВЕДЕНИЕ К КООРДИНАТАМ ЭКРАНА ДИСПЛЕЯ

Разрешение экрана N, M



Точка **Q** на экранной
плоскости

с координатами Q_x, Q_y

Здесь P_{ij} -
 i, j -й пиксель
экрана

Как найти i, j

ПРИВЕДЕНИЕ К КООРДИНАТАМ ЭКРАНА ДИСПЛЕЯ

$$i = FLOOR \left[\left(\frac{Q_x}{E_x} + 1 \right) * \frac{N}{2} \right]$$

$$j = FLOOR \left[\left(\frac{Q_y}{E_y} + 1 \right) * \frac{M}{2} \right]$$

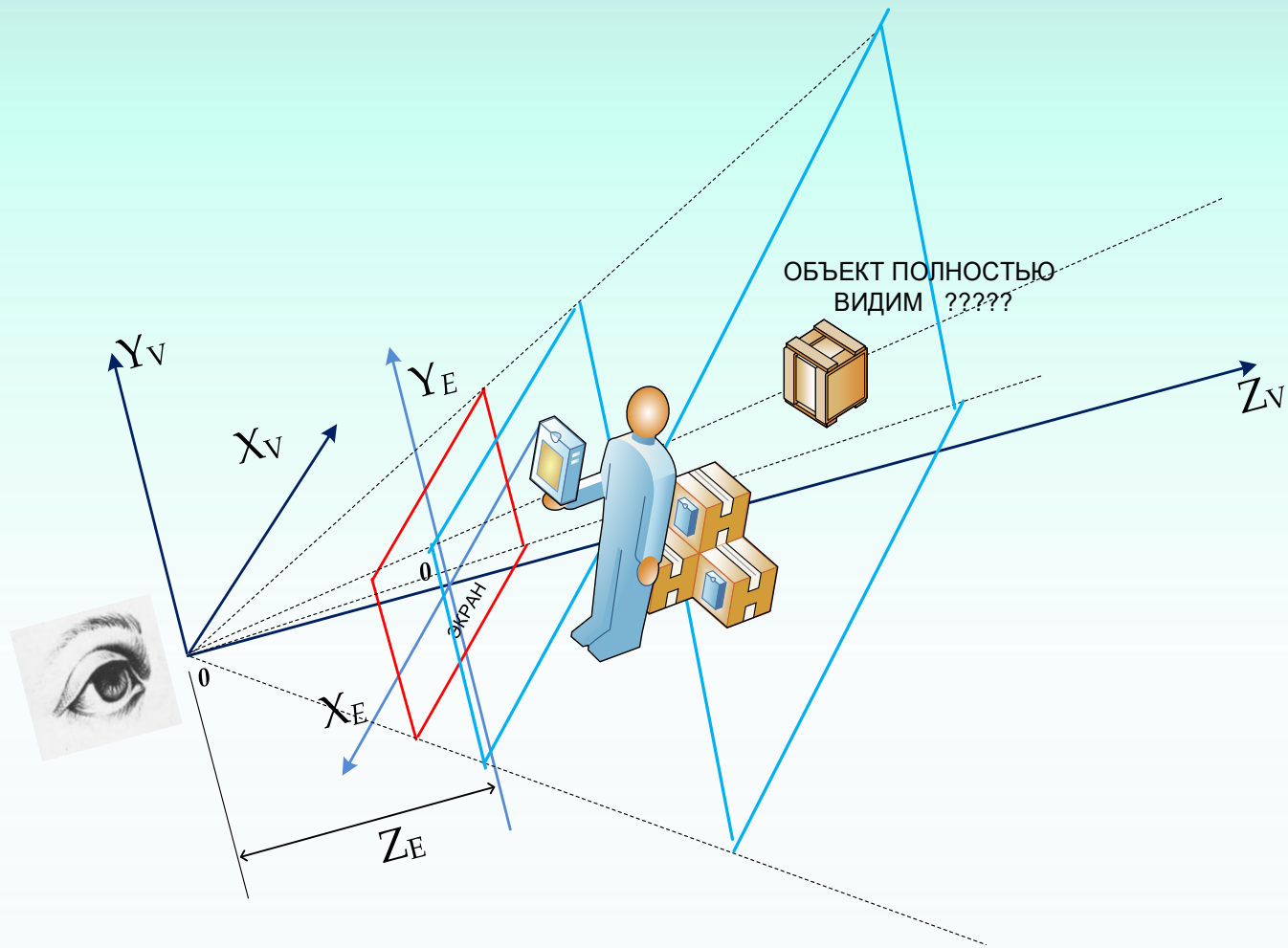
УДАЛЕНИЕ REMOVAL

При проецировании ->

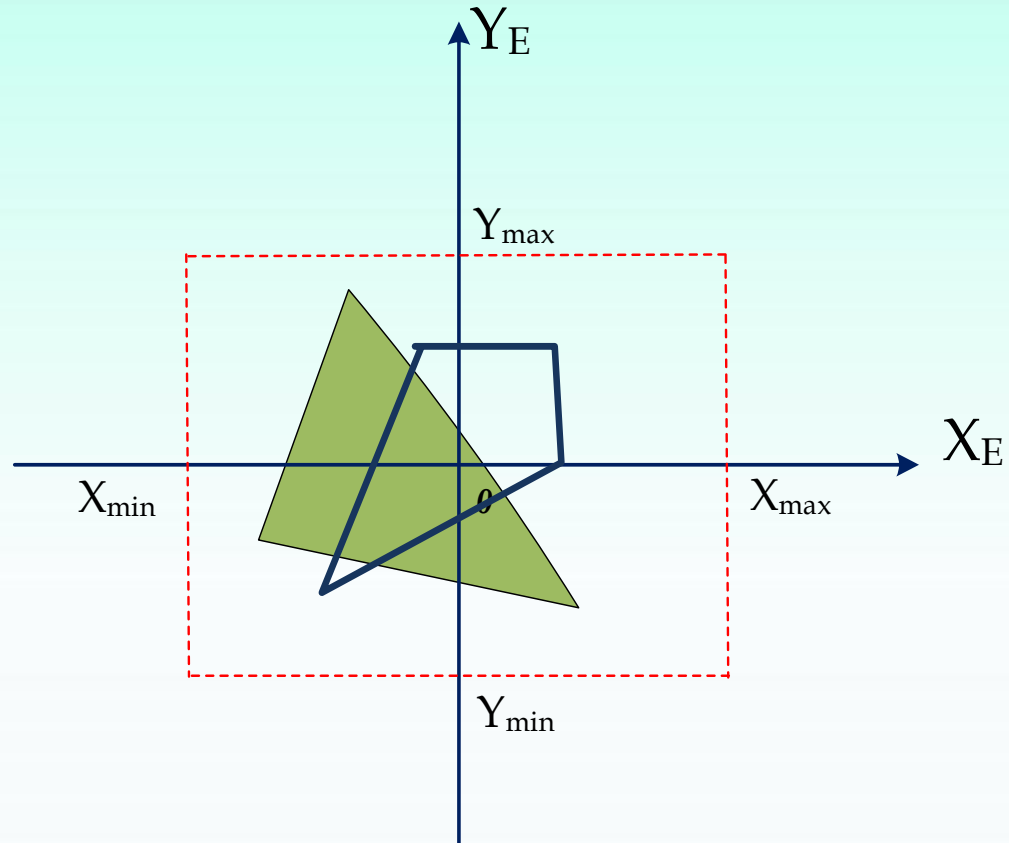
Объект полностью или частично загорожен (перекрывается) другим объектом и, следовательно, невидим наблюдателю.

НУЖНЫ методы (алгоритмы) удаления невидимых поверхностей.

УДАЛЕНИЕ REMOVAL



УДАЛЕНИЕ REMOVAL



УДАЛЕНИЕ REMOVAL КЛАССИФИКАЦИЯ

По способу изображения

- Каркасное
- Сплошное (полигональное)

По пространству

- В мировом пространстве
- В системе координат наблюдателя
- В экранной плоскости

По точности

- Аналитическое
- Приближенное

ОСНОВНЫЕ МЕТОДЫ УДАЛЕНИЯ

Удаление нелицевых граней

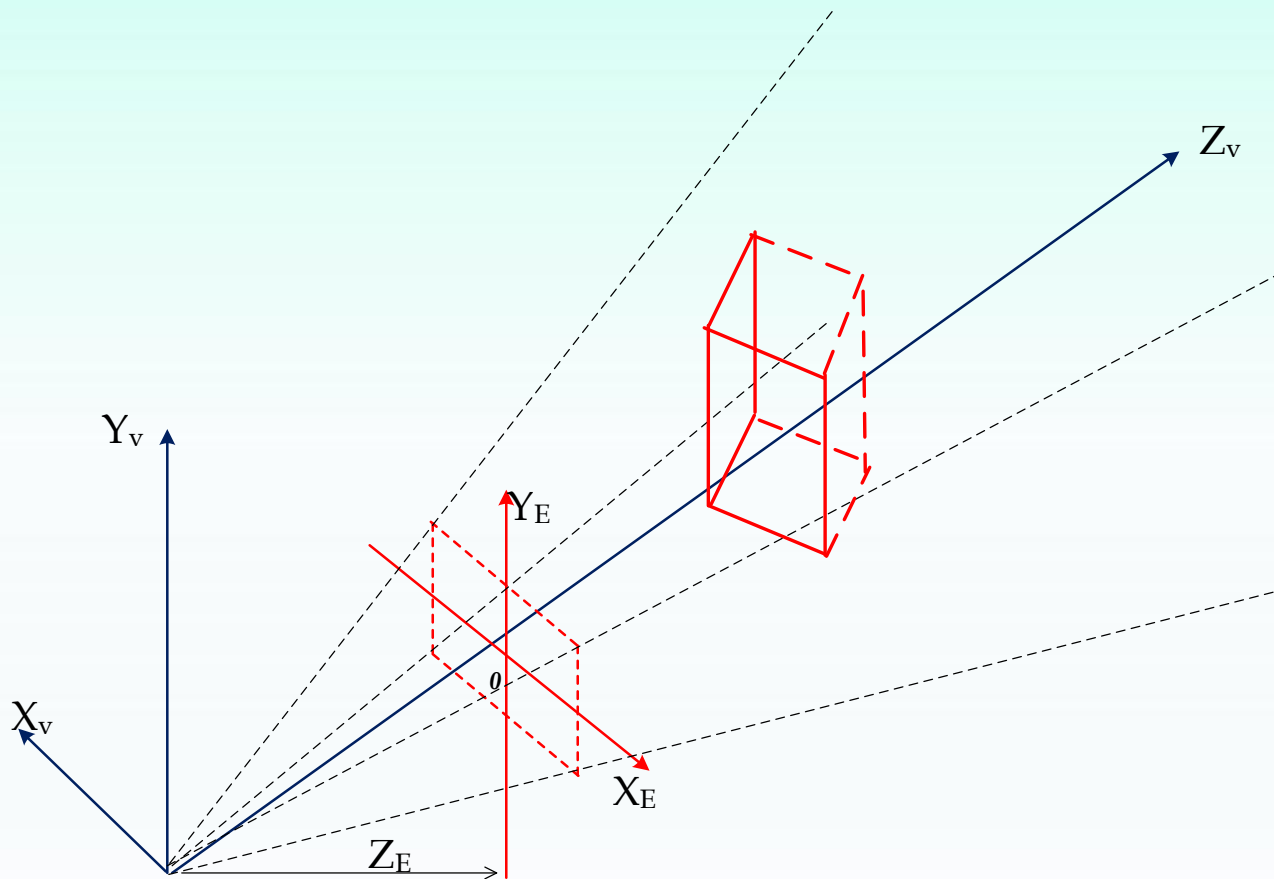
Алгоритм художника

Буфер глубины (z-буфер)

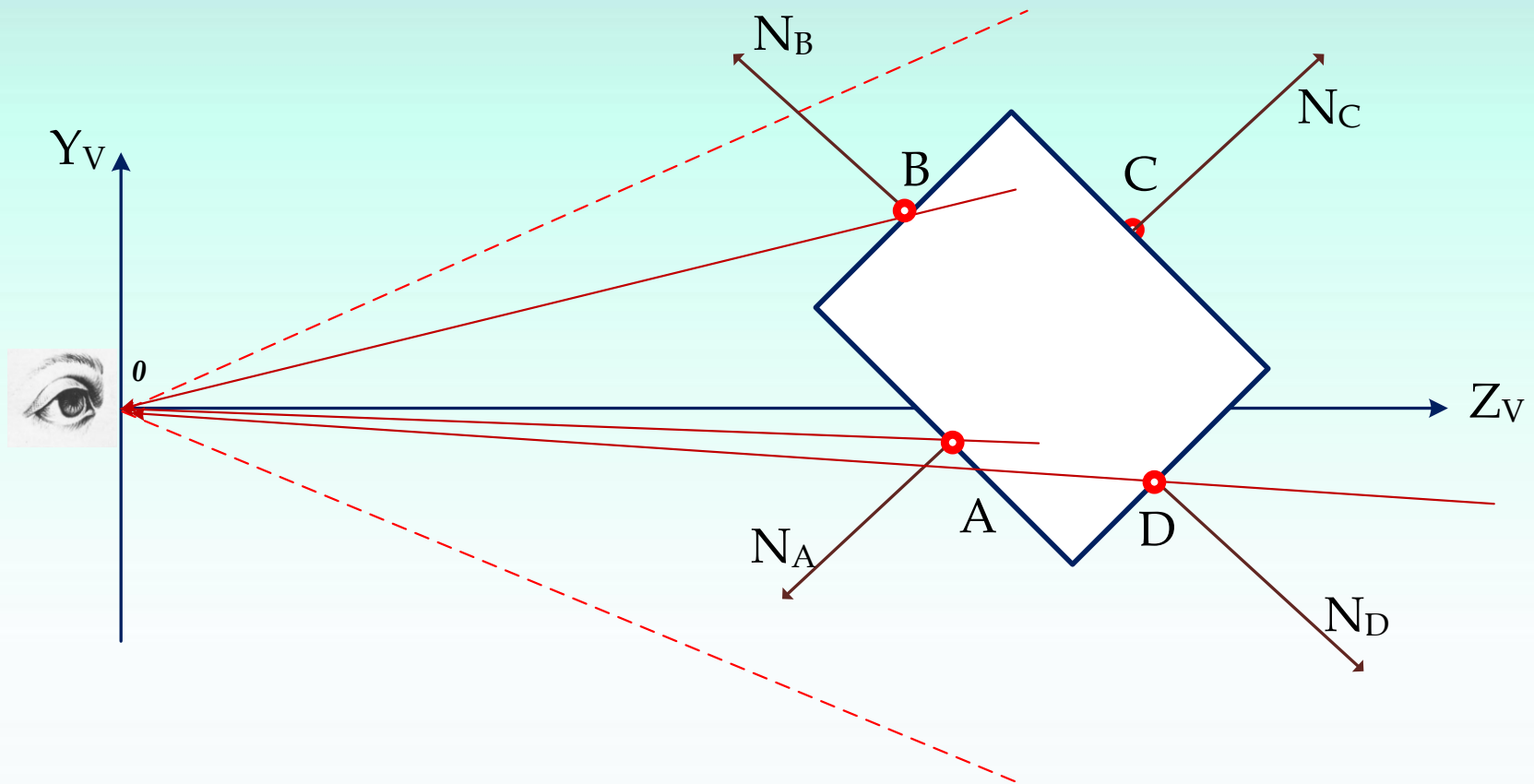
Трассировка лучей

УДАЛЕНИЕ НЕЛИЦЕВЫХ ГРАНЕЙ

Выпуклый объект. Грани заслоняются **самим** объектом (back face).



УДАЛЕНИЕ НЕЛИЦЕВЫХ ГРАНЕЙ



Скалярное произведение нормали к
границе и вектора визирования
ПОЛОЖИТЕЛЬНО

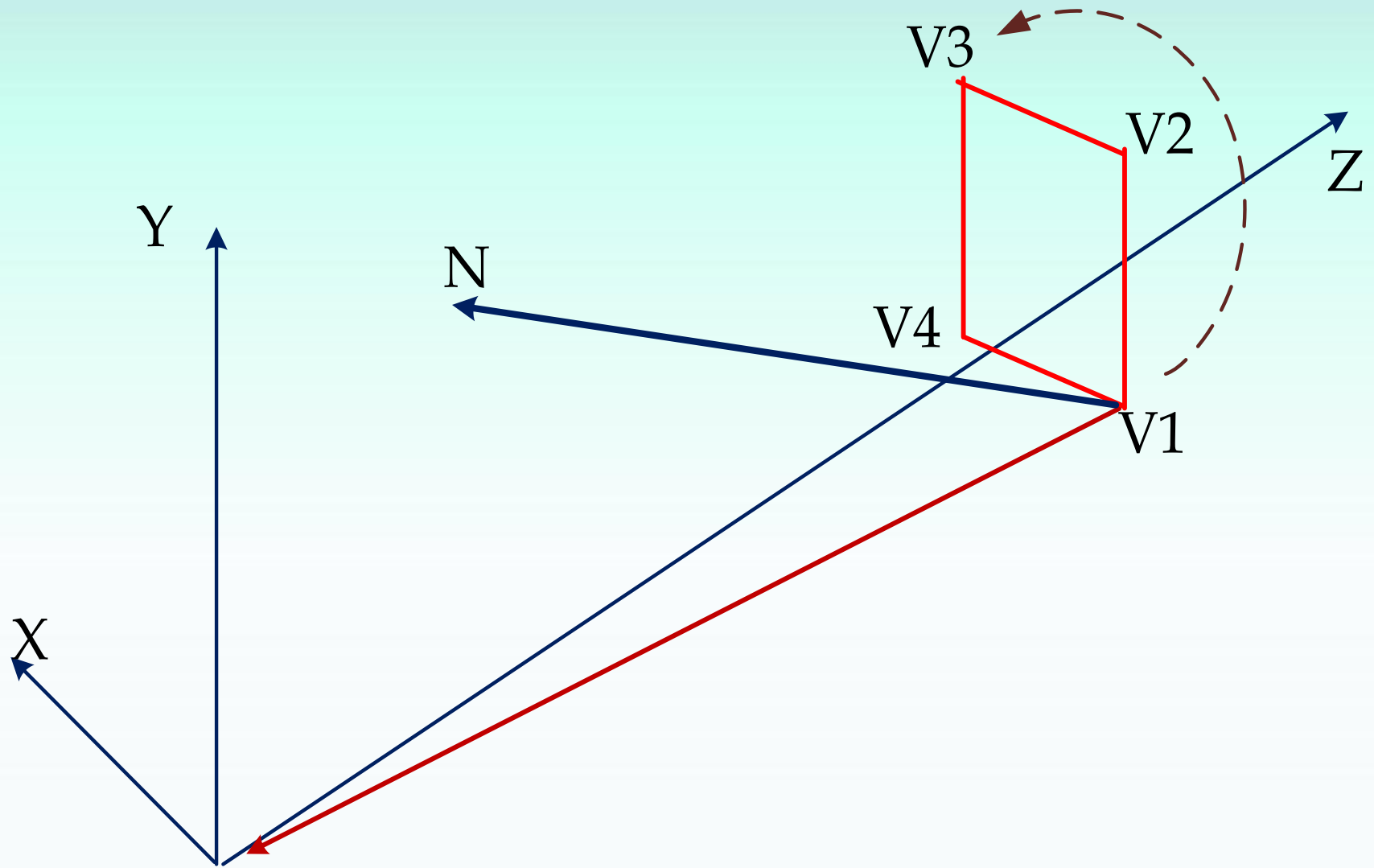
УДАЛЕНИЕ НЕЛИЦЕВЫХ ГРАНЕЙ

Определить, является ли грань **нелицевой** можно по знаку скалярного произведения нормали N и вектора из любой точки на грани A на наблюдателя V . Выражение $N \cdot (V - A) < 0$ означает, что косинус угла между векторами отрицательный, и угол больше 90° , а значит грань **нелицевая**. Ее можно не отрисовывать!

УДАЛЕНИЕ НЕЛИЦЕВЫХ ГРАНЕЙ

Нормаль N и точка на грани A задают плоскость грани, а знак выражения $N \cdot (V - A)$ говорит о том, с какой стороны от этой плоскости лежит точка наблюдателя V , положительный – над плоскостью, и грань лицевая, отрицательный – под плоскостью, грань нелицевая, ноль – на плоскости, грань проецируется в отрезок

УДАЛЕНИЕ НЕЛИЦЕВЫХ ГРАНЕЙ



УДАЛЕНИЕ НЕЛИЦЕВЫХ ГРАНЕЙ

Задание нормали N .

Если посмотреть на поверхность грани и описать его списком вершин обходя многоугольник **против** часовой стрелки, то такая грань называется лицевая. В противном случае – обратная.

$$N = (V2 - V1) \times (V3 - V2)$$

$$N1 = N / |N|$$

Грань видна $N1 \cdot (-V1) > 0 ?$

НОРМАЛЬ N

Три точки $V1, V2, V3$ (не совпадающие, не лежащие на одной прямой)

Находим два вектора

$$\vec{V2V1} = (V2 - V1), \vec{V3V1} = (V3 - V1)$$

Уравнение плоскости, содержащей точки $V1, V2, V3$

$$Ax + By + Cz + D = 0$$

Где A, B, C - Компоненты вектора нормали
 $N = N(A, B, C) !!!$

УРАВНЕНИЕ ПЛОСКОСТИ. НОРМАЛЬ

$$M = \begin{bmatrix} x - x_1 & y - y_1 & z - z_1 \\ x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \\ x_3 - x_1 & y_3 - y_1 & z_3 - z_1 \end{bmatrix}$$

$$A = (y_2 - y_1)(z_3 - z_1) - (y_3 - y_1)(z_2 - z_1)$$

$$B = (x_3 - x_1)(z_2 - z_1) - (x_2 - x_1)(z_3 - z_1)$$

$$C = (x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1)$$

$$D = -(Ax_1 + By_1 + Cz_1)$$

НОРМАЛЬ N

Если уравнение $Ax+By+Cz+D=0$ задает плоскость в которой лежит грань объекта, то неравенства:

$Ax+By+Cz+d>0$ - задает полупространство
ВНЕ ОБЪЕКТА (перед гранью)

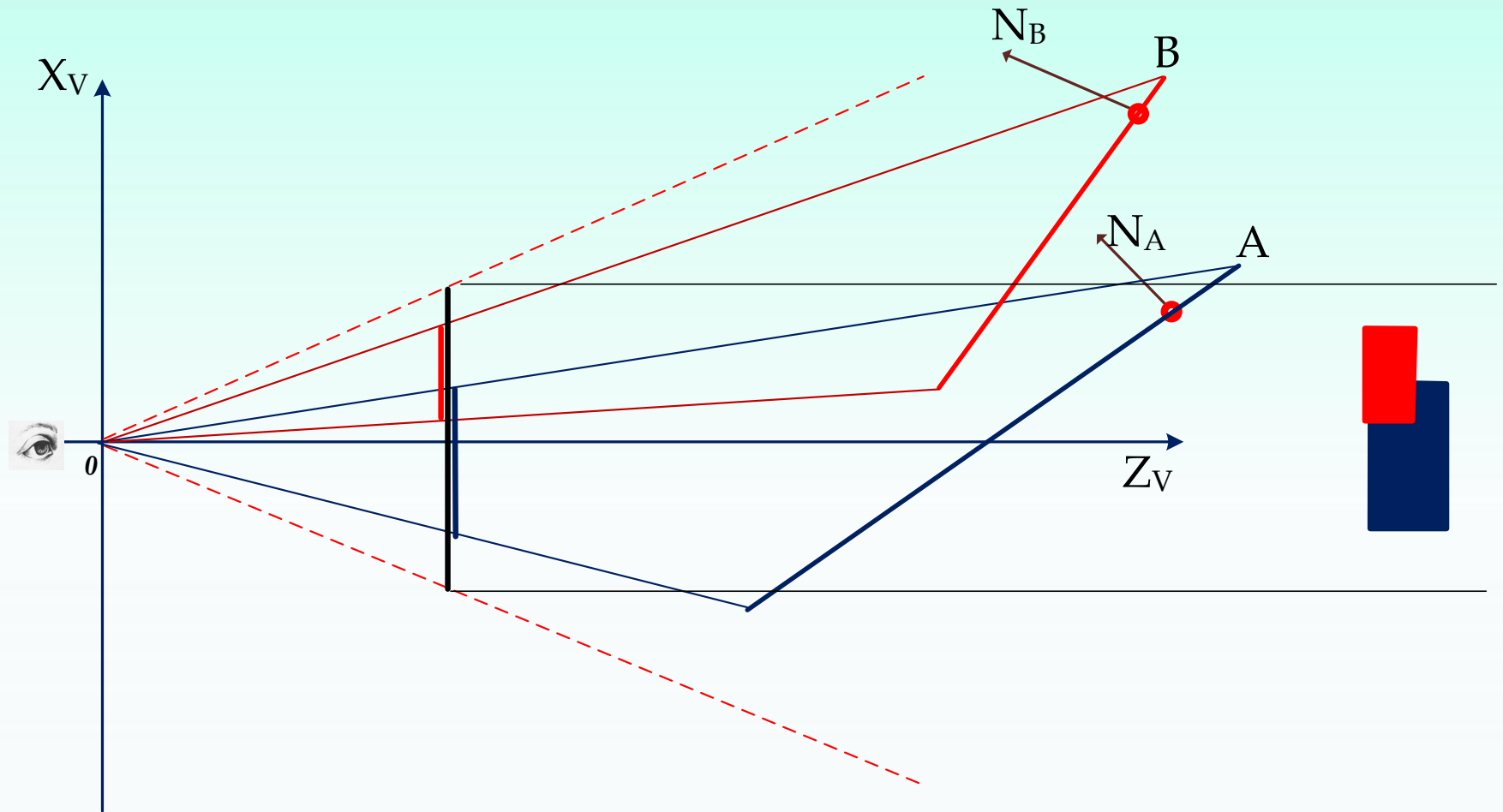
$Ax+By+Cz+D<0$ - задает полупространство
ВНУТРИ ОБЪЕКТА (за гранью)

АЛГОРИТМ ХУДОЖНИКА

Идея: грани сортируются по Z и рисуются от дальних к ближним. В результате - ближние грани автоматически заслоняют дальние.

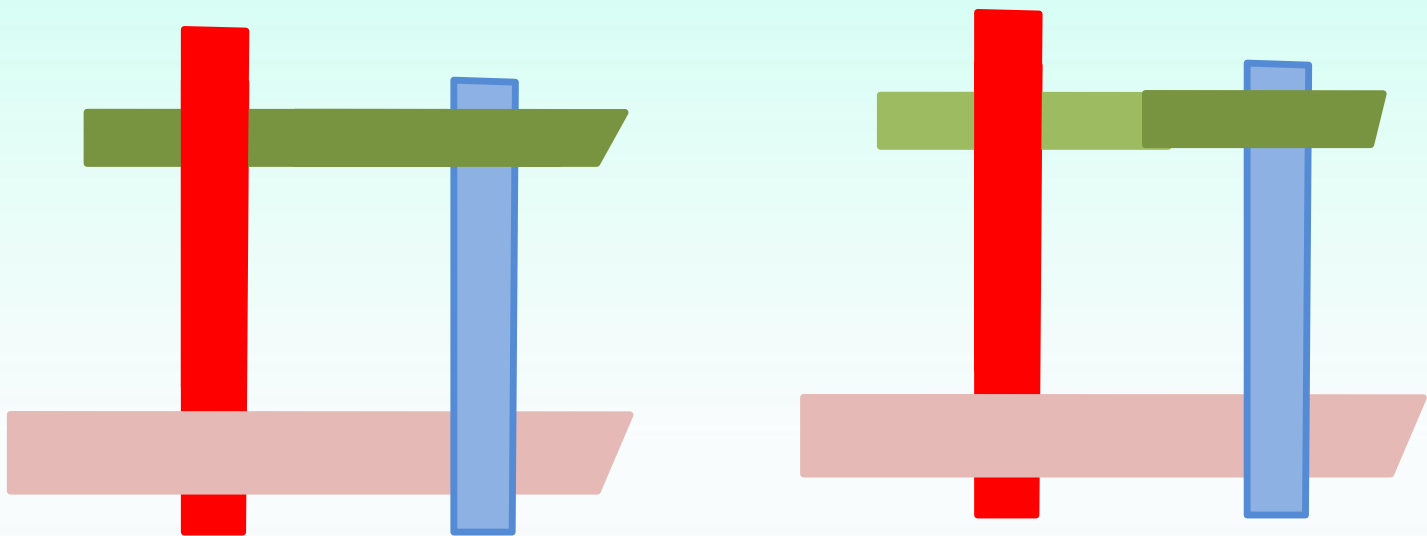
НО! Сортировка – задача далеко не тривиальная.

АЛГОРИТМ ХУДОЖНИКА



АЛГОРИТМ ХУДОЖНИКА

Циклически закрывающие друг друга грани



!!! Одну грань делят на части

АЛГОРИТМ ХУДОЖНИКА

Бинарное разделение пространства

Binary Space Partition tree + List Priority

ИДЕЯ:

Все пространство сцены делится плоскостью на два подпространства.

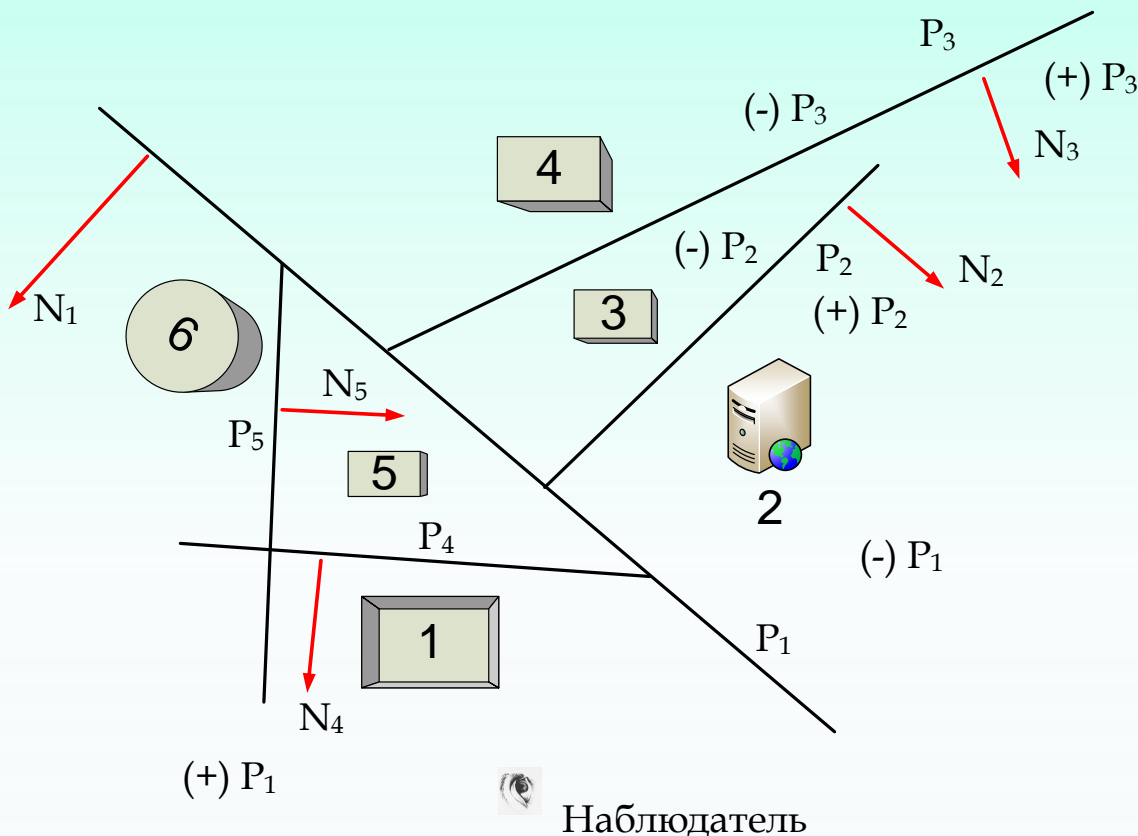
ПОЛОЖИТЕЛЬНОЕ полупространство содержит все объекты, на которые указывает положительная нормаль к плоскости

ОТРИЦАТЕЛЬНОЕ полупространство содержит все остальные объекты (на которые указывает отрицательная нормаль к плоскости).

Оба (+) и (-) полупространства затем делятся следующей дополнительной плоскостью. И т.д., пока каждый объект не будет отделен. Хотя бы одной плоскостью.

АЛГОРИТМ ХУДОЖНИКА

Бинарное разделение пространства

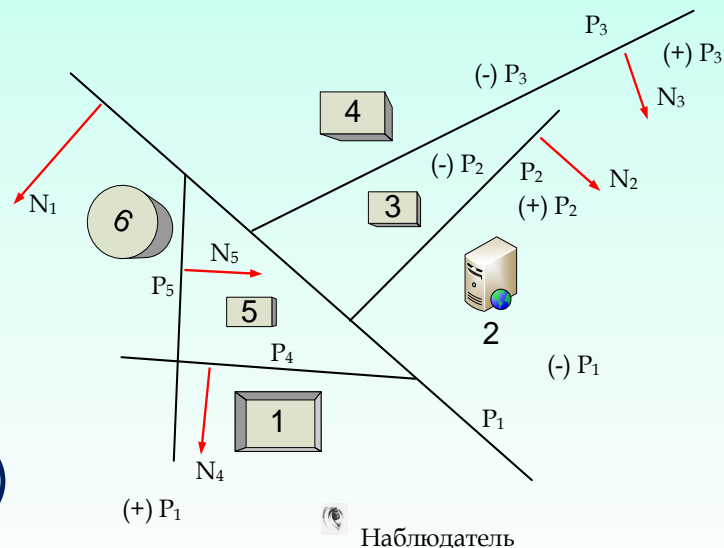
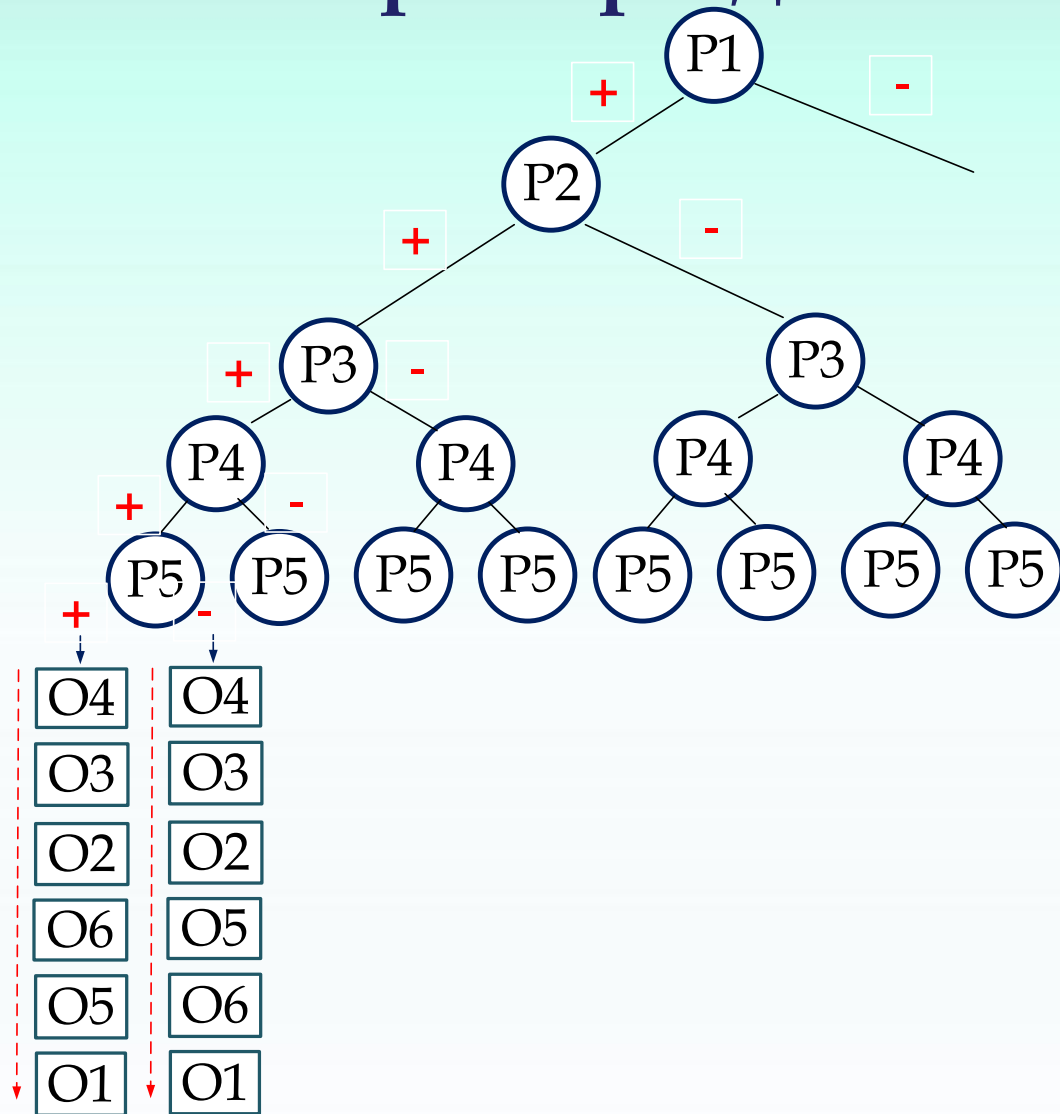


Плоскость P_1 делит пространство на 2 полупространства. Объекты в полупространстве $(+)$, в котором находится наблюдатель, могут закрывать объекты во втором полупространстве $(-)$.

Плоскость P_2 делит пространство на 2 полупространства. И так далее ...

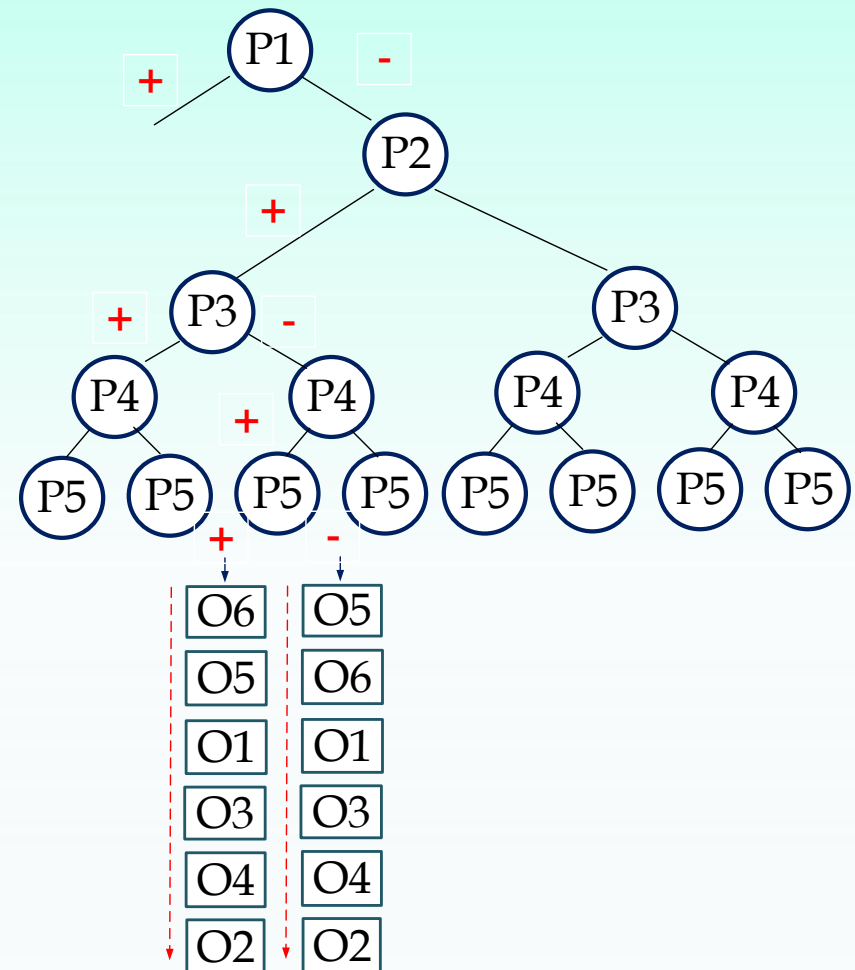
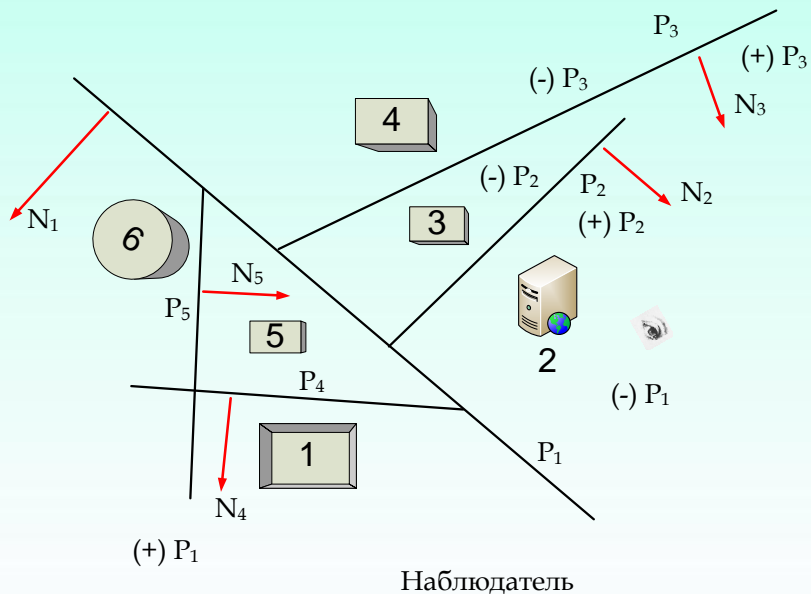
АЛГОРИТМ ХУДОЖНИКА

Бинарное разделение пространства



АЛГОРИТМ ХУДОЖНИКА

Бинарное разделение пространства



АЛГОРИТМ ХУДОЖНИКА

Бинарное разделение пространства

ris(tree) {

IF наблюдатель $\theta + P_i$ THEN

IF rtree не пусто ris(rtree);

Рисуем корневой объект;

IF ltree не пусто ris(ltree);

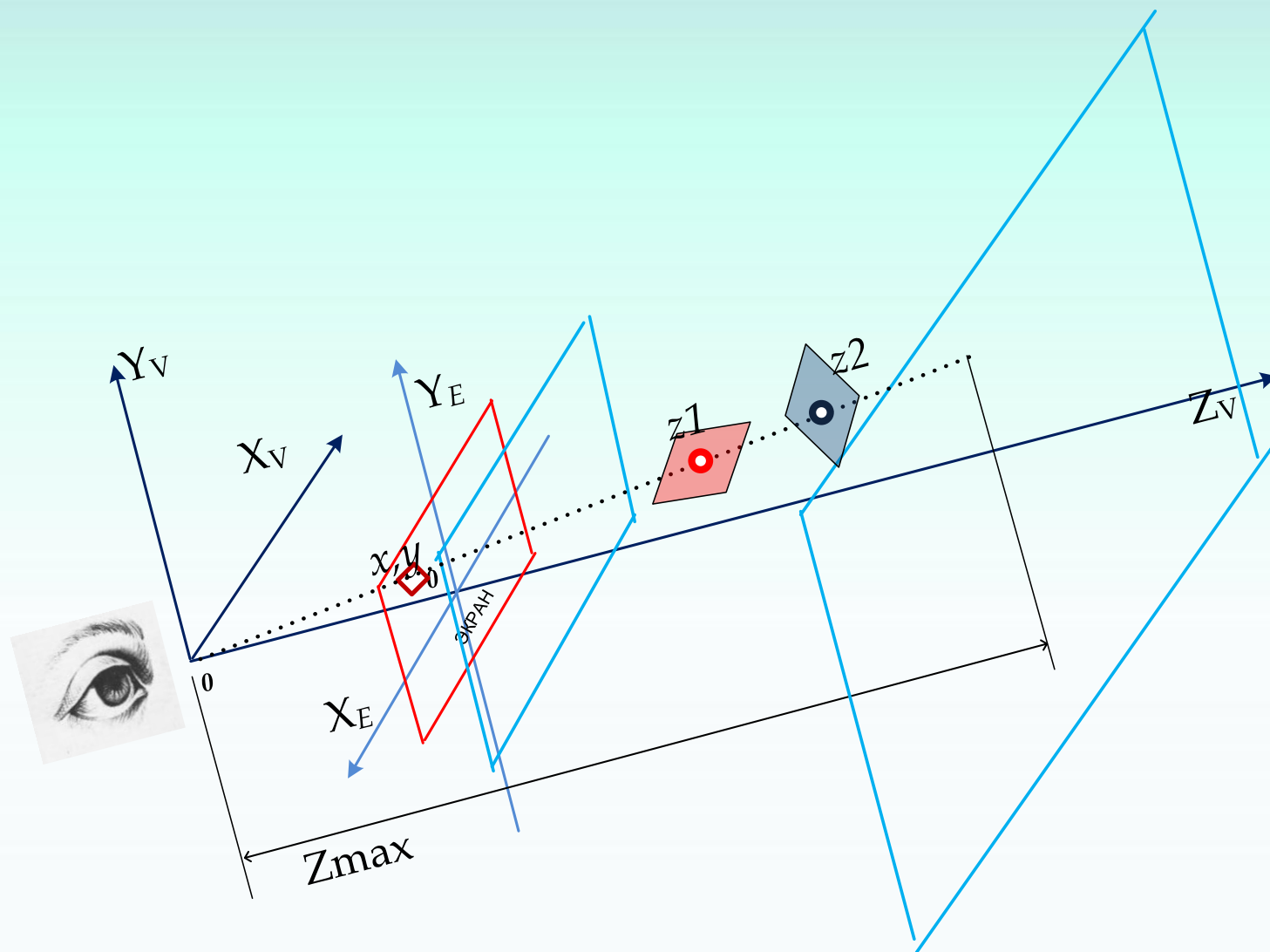
ELSE

IF ltree не пусто ris(ltree);

Рисуем корневой объект;

IF rtree не пусто ris(rtree); }

Z - БУФЕР



Z - БУФЕР

Заполнить буфер кадра фоновым значением цвета

$$\forall(x, y) \rightarrow \mathbf{bufc}[x,y] := \mathbf{Fon_Col}.$$

Заполнить Z-буфер максимальным значением глубины

$$\forall(x, y) \rightarrow \mathbf{bufz}[x,y] := \mathbf{Zmax}.$$

Преобразовать объекты (границы) в растровую форму в произвольном порядке.

Z - БУФЕР

Для каждого объекта (границы):

- для каждого пикселя $[x, y]$ проекции границы вычислить глубину $z(x, y)$;
- сравнить глубину $z(x, y)$ со значением глубины, хранящемся в Z-буфере в позиции **bufz** $[x, y]$;
- если **bufz** $[x, y] > z(x, y)$, то **bufz** $[x, y] := z(x, y)$ и занести в буфер кадра цвет объекта (границы).

Вопросы для экзамена

1. Центральная перспективная проекция. Формулы расчета экранных координат.
2. Приведение к координатам экрана дисплея
3. Постановка задачи удаления невидимых поверхностей.
4. Удаление нелицевых граней.
5. Алгоритм художника. Бинарное разделение пространства.
6. Алгоритм Z –буфера.

Литература:

Компьютерная графика. Курс лекций
Маценко В.Г. Компьютерная графика

END #13