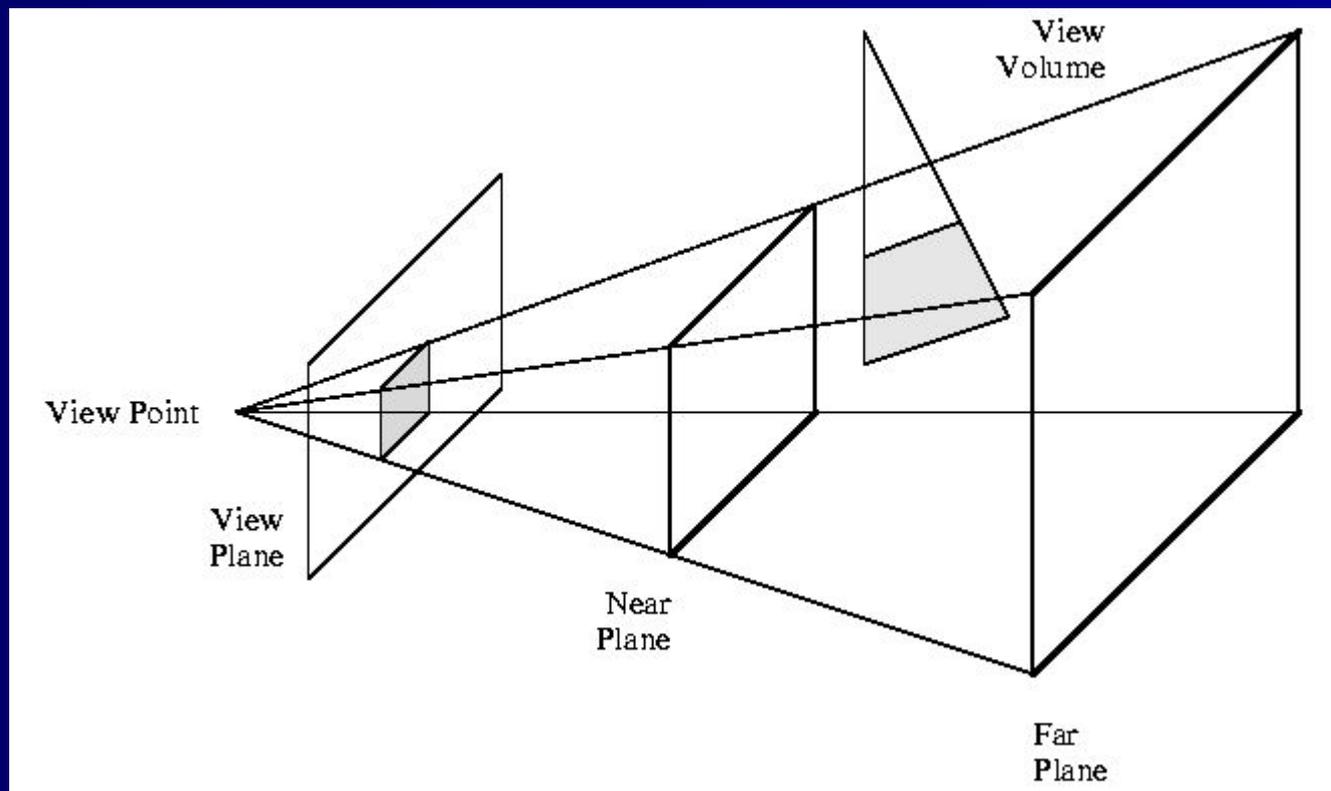


Отсечение невидимых частей объектов (Clipping)

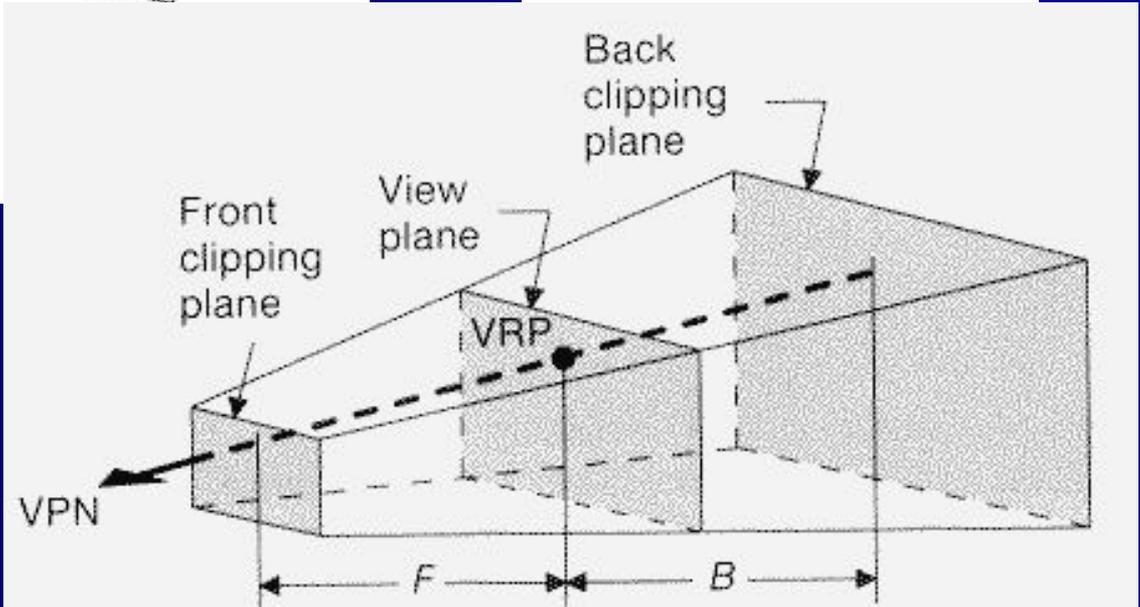
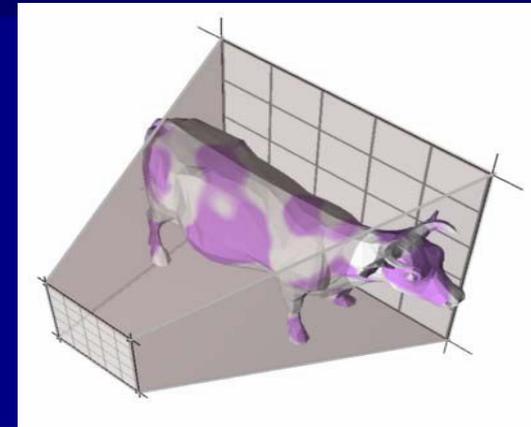
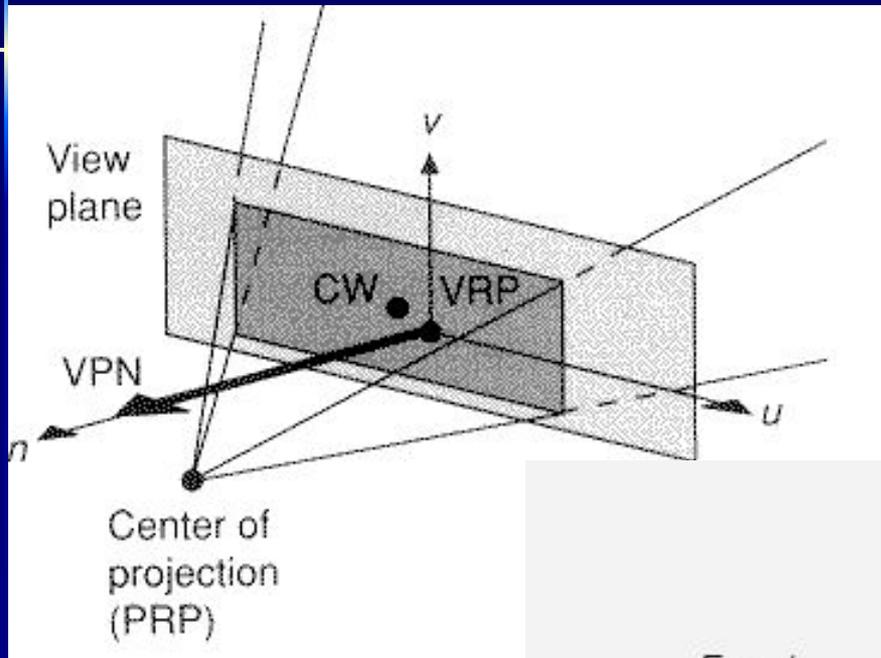
Видимый объем (пирамида \ параллелепипед видимости) – выделенная часть мирового координатного пространства.

Окно вывода (экран монитора, активное окно, вывод на печать и т.п.) – область вывода на физическом устройстве

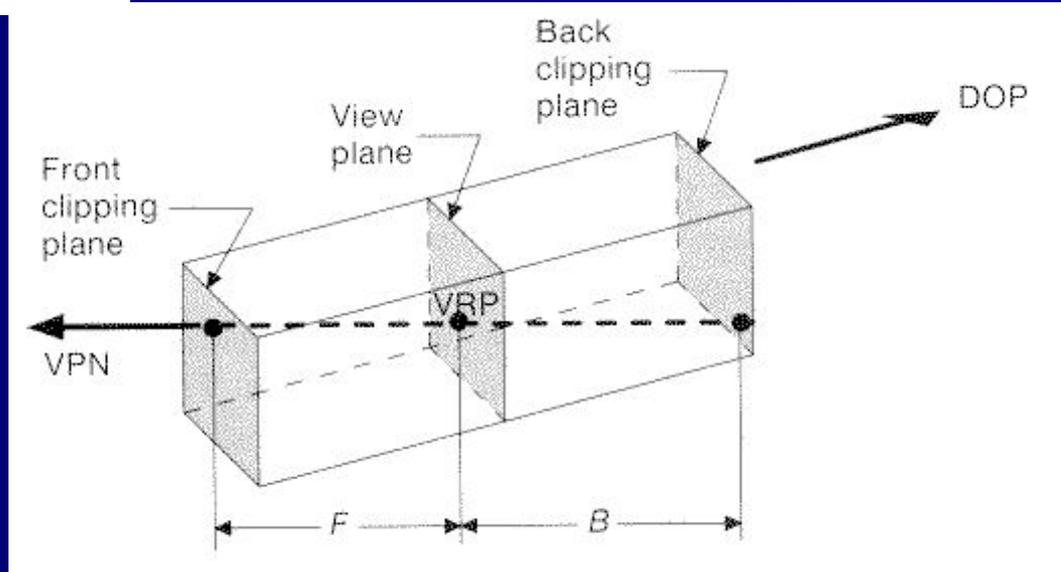
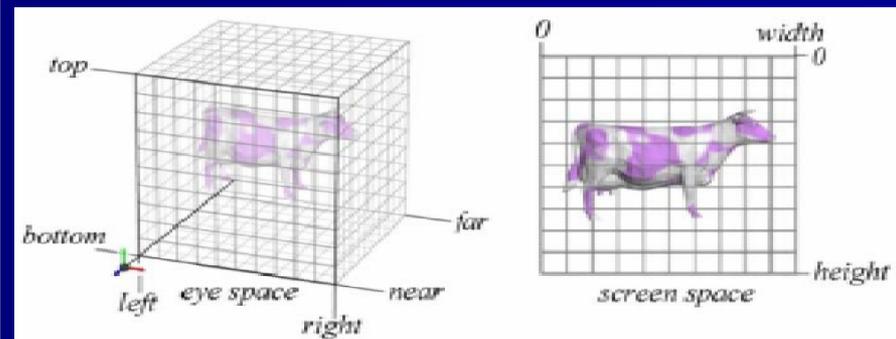
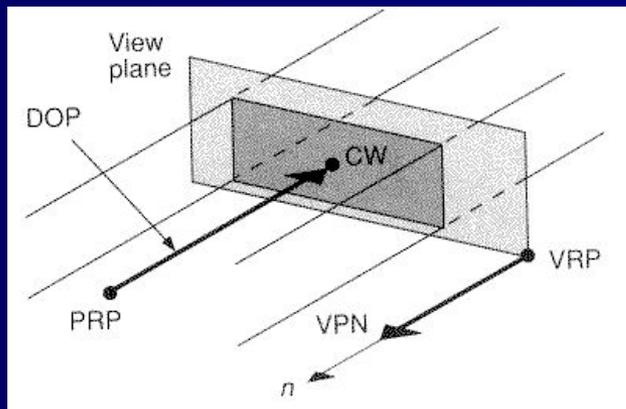
Отсечение (пирамида видимости)



Отсечение (Clipping) видимого объема в окне перспективы



Отсечение (Clipping) видимого объема в окнах параллельных проекций



Алгоритмы отсечения

- Отсечение 2D
- Отсечение 3D

Алгоритмы отсечения могут быть реализованы программно или аппаратно. Алгоритмы отсечения, реализованные программно, зачастую оказываются недостаточно быстродействующими для интерактивных графических приложений.

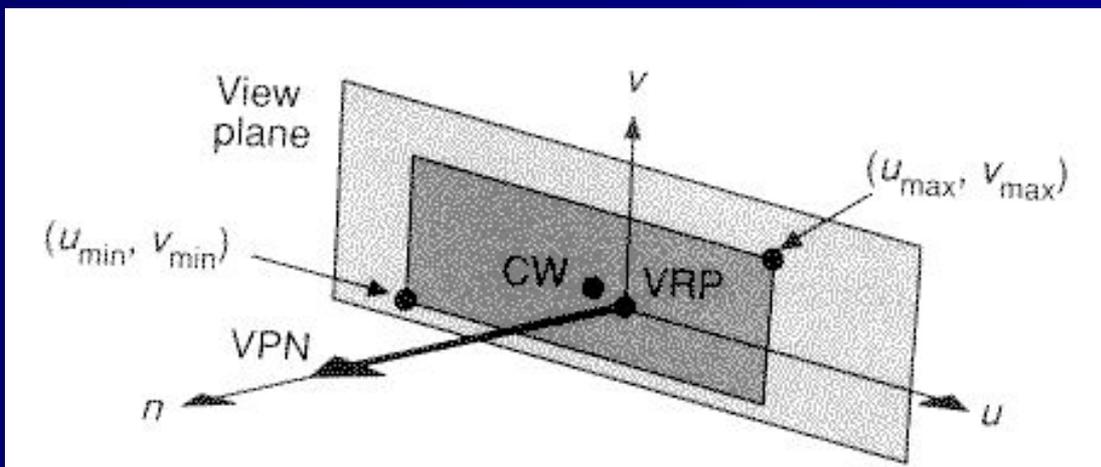
Поэтому как 2D, так и 3D алгоритмы отсечения реализуются аппаратными или микропрограммными средствами.

Алгоритм отсечения 2D

- Плоская сцена
- Окно вывода – выпуклое, регулярной формы
- Цель алгоритма отсечения – определить те точки, отрезки (части отрезков), которые лежат внутри окна вывода.
- Эти точки, отрезки (части отрезков) остаются для визуализации, а все остальное отбрасывается.

Окно вывода

- Регулярным отсекающим окном является прямоугольник, стороны которого параллельны осям физической системы координат. Например, осям координат экрана.

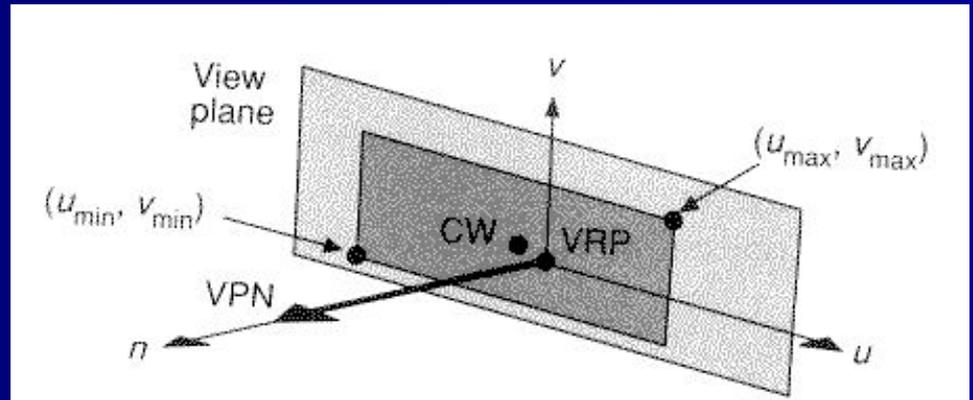


Шаг 1. Отсечение точек

$CW(u,v)$ точка с координатами u и v .

$$U_{\min} \leq U \leq U_{\max}$$

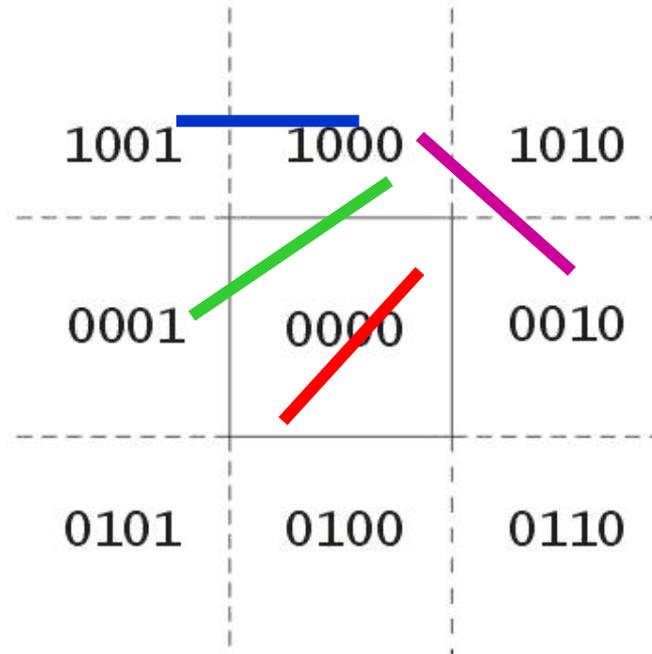
$$V_{\min} \leq V \leq V_{\max}$$



Шаг 2. Отсечение отрезков. Алгоритм Сазерленда-Коэна

- Рассмотрим задачу отсечения отрезков прямых. Некоторые из них полностью лежат внутри области экрана, другие целиком вне ее, а некоторые пересекают границу экрана.
- Алгоритм Сазерленда-Коэна позволяет найти полностью видимые и отсекают полностью невидимые отрезки.

Код окна вывода - 0000



$$x = x_1 + t(x_2 - x_1)$$
$$y = y_1 + t(y_2 - y_1)$$

Категории отрезков

(& - определяется конъюнкция кодов концов отрезка)

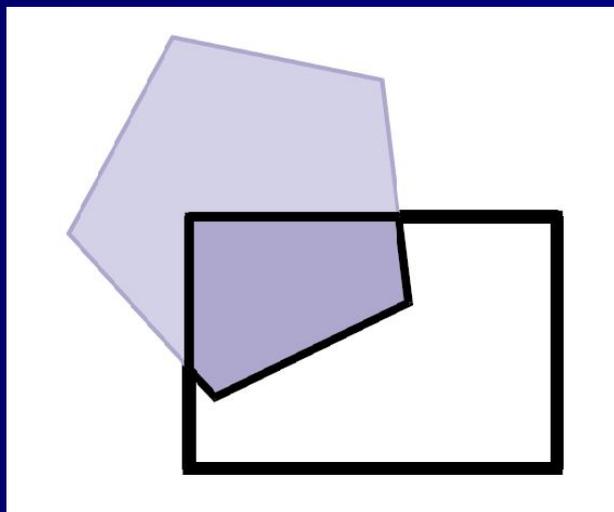
- Полностью видимый – код1 & код2 = 0000.
- Тривиально невидимый - код1 & код2 ≠ 0000.
- Неопределенный - код1 ≠ 0000, код2 ≠ 0000,
код1 & код2 = 0000.

Алгоритм работает в пространстве изображений(растровый), т.к. важно свести задачу к двоичному виду(реализовано аппаратно или микропрограммно), а не к решению геометрической задачи поиска точки пересечения отрезков на основе системы линейных уравнений, $X = x_1 + t(x_2 - x_1)$
 $Y = y_1 + t(y_2 - y_1).$

Шаг 3. Алгоритм разбиения средней точкой (для неопределенных отрезков)

- Для отрезков, пересекающих окно вывода, которые нельзя принять за видимые или полностью отбросить.
- Последовательно определяется средняя точка при делении отрезка пополам до тех пор, пока одна из частей отрезка не будет принята целиком (если есть пересечение), а другая часть – целиком отброшена.
- Максимально необходимое число делений = максимальному числу двоичных разрядов, используемых для представления координат u и v .

**Алгоритм Сазерленда-Козна удобен,
для отсечения частей изображения
значительно выходящих за границы
окна вывода**



Отсечение 3d объектов (отсечение по видимому объему)

Двумя наиболее распространенными формами видимого объема являются:

-  Усеченная пирамида,
-  Прямоугольный параллелепипед.

Как и 2d отсечении, отрезки, которые полностью видимы или тривиально невидимы, можно идентифицировать с помощью алгоритма Сазерленда-Коэна.

В трехмерном случае используется 6-битный код.

В биты кода заносятся 1 и 0 с помощью обобщения двумерной процедуры.

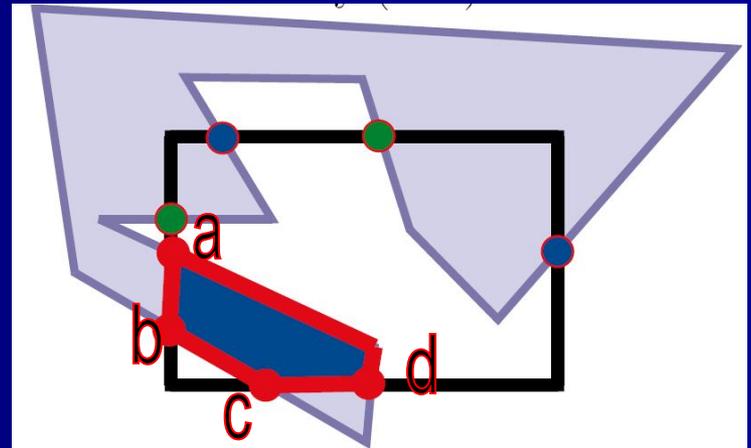
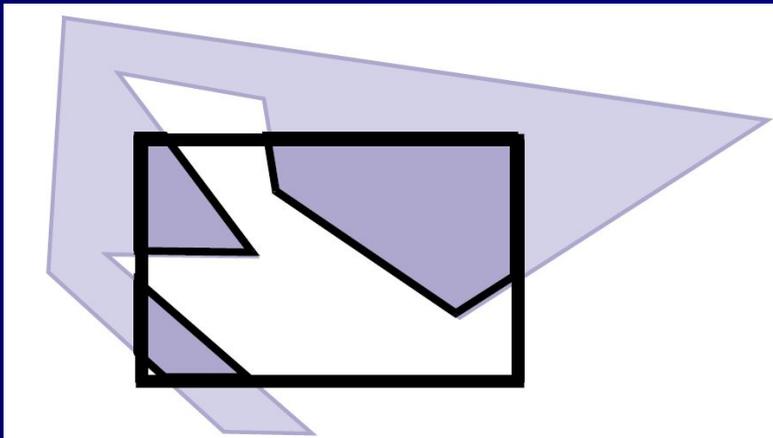
В случае частичной видимости необходимо определить пересечения отрезка с гранями отсекающего объема

Отсечение в однородных координатах

- Когда отсечение необходимо провести в однородной системе координат, то нужно проявить осторожность, если при этом производится еще и проецирование. Отрезок может «заворачиваться» в бесконечности так, что внутри видимого объема (если он не ограничен дальней плоскостью) будут видны сразу обе части отрезка.
- Джим Блинн доказал, что отсечение всех отрезков до завершения процедуры перспективного проецирования (присутствует деление всех координат на значение однородной координаты), удаляет части отрезков, которые «возвращаются» из бесконечности.
- Лианг и Барский получили корректный результат отсечения отрезков в однородной системе координат, используя усеченную пирамиду видимости.
- Доказано, что и к отсекающему объему, и к отрезку перед проецированием можно применить любое аффинное преобразование (т.е. повороты, переносы и т.п.)

Отсечение многоугольников

- Предыдущее обсуждение было связано с отсечением отрезков. Разумеется, многоугольник можно рассматривать как набор отрезков.
- Однако, если замкнутый многоугольник отсекается, как набор отрезков, то части исходного объекта могут превратиться в один или более открытых многоугольников.
- Если многоугольники являются сплошными областями (а не заштрихованными), то необходимо, чтобы замкнутость сохранялась и у результата. Например, отрезки (a,b) и (c,d) должны быть добавлены к описанию видимой части



Алгоритм Сазерленда-Ходжмена. Отсечение многоугольников.

- Основная идея алгоритма состоит в том, что отсечь многоугольник относительно одной прямой легко. Исходный и каждый из промежуточных многоугольников отсекается последовательно относительно одной прямой (одна из сторон окна видимости).
- Исходный многоугольник задается списком вершин - P_1, P_2, \dots, P_n , который порождает список его ребер - $P_1 P_2, \dots, P_{n-1} P_n, P_n P_1$.
- Этот алгоритм способен отсекают любой многоугольник, выпуклый\невыпуклый, плоский\неплоский, относительно окна видимости, являющегося выпуклым многоугольником.
- Порядок отсечения многоугольника разными сторонами окна принципиален.

