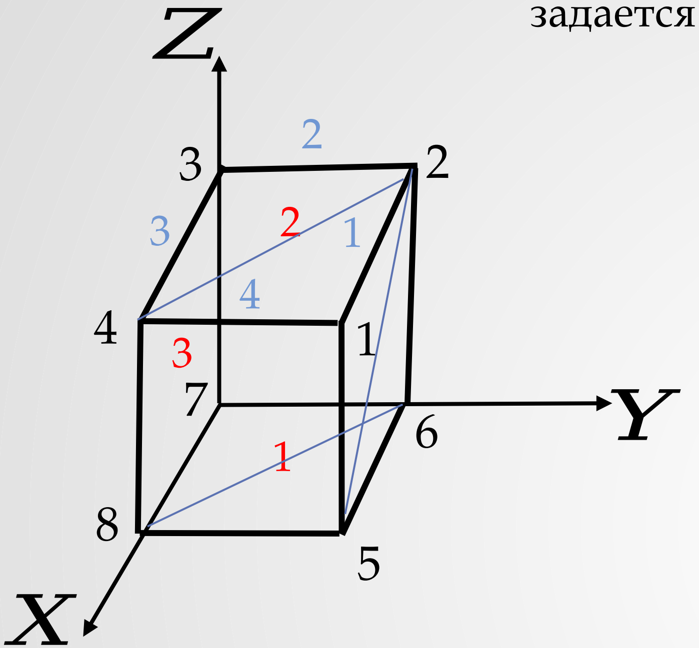
Компьютерная графика

Практическое занятие 3. Каркасная модель трехмерного тела

Хачумов М.В., к.ф.-м.н.,
доцент кафедры информационных технологий

Задание. Полная модель куба

Модель куба задается в виде набора вершин, ребер и граней. Каркас задается в виде набора вершин и ребер



Номер вершины	Координаты вершины		
	X	Y	Z
1	1	1	1
2	0	1	1
3	0	0	1
4	1	0	1
5	1	1	0
6	0	1	0
7	0	0	0
8	1	0	0

Обозначение ребер
1 (1 2) 2 (3 2) 3 4 5 6 7 8 9 10 11 12

Обозначение граней
1 (5 6 7 8) 2 3 4 5 6

Мировые координаты – связаны с центром объекта наблюдения

Если куб не в центре мировых координат, то необходимо осуществить его сдвиг в центр.

$$[x' \quad y' \quad z' \quad 1] = [x \quad y \quad z \quad 1]V$$

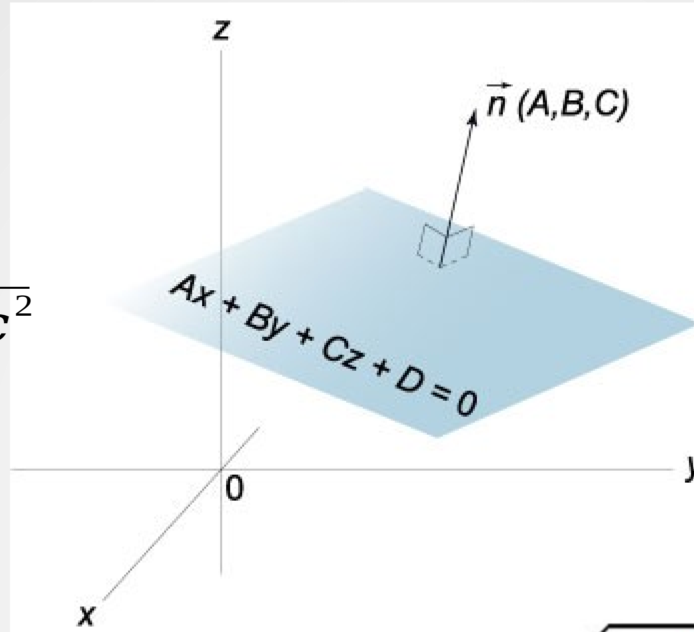
Отрисовка нелицевых граней. Алгоритм 1.

$$ax+by+cz=h$$

$$n=[a,b,c]$$

$$|n|=r=\sqrt{a^2+b^2+c^2}$$

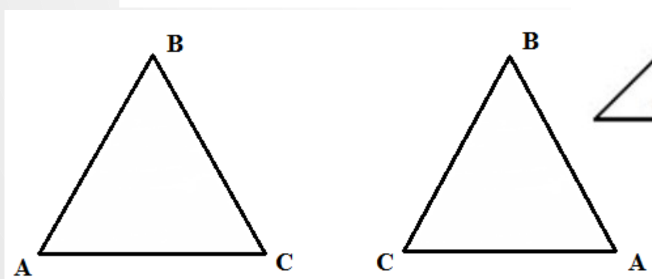
$$n=\left[\frac{a}{r},\frac{b}{r},\frac{c}{r}\right]$$

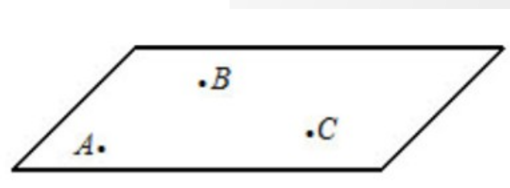


=0

Уравнение плоскости, которая проходит через три точки

C





1 (6 5 8 7)

2 (4 1 2 3)

В исходном файле описываем лицевые грани – против часовой стрелки, нелицевые по часовой стрелке. В таком же порядке подставляем их в уравнение плоскости. При изменении знака h, грань меняет свою видимость

Отрисовка нелицевых граней. Алгоритм 2.

Пусть координаты барицентра объекта (центра тяжести) есть p(*x*,*y*,*z*).

$$x = \frac{\sum_{j=1}^{n} x_j}{n} \qquad y = \frac{\sum_{j=1}^{n} y_j}{n} \qquad z = \frac{\sum_{j=1}^{n} z_j}{n}$$

Поскольку барицентр не лежит в плоскости ни одной из *i*-ой грани тела, то имеет место неравенство

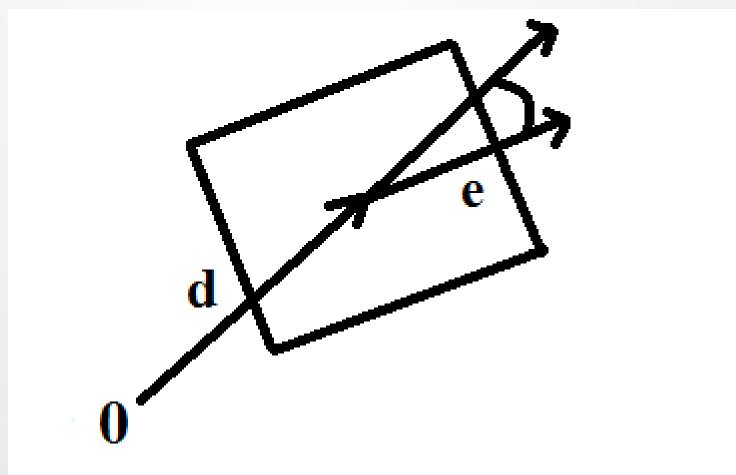
$$L_i(p) = A_i x + B_i y + C_i z + D_i \neq 0$$

где , , − коэффициенты, определяющие плоскость, проходящую через *i*-ую грань. В случае когда наблюдатель смотрит в центр грани *i*, угол между направлением наблюдения и нормалью к *i*-ой грани определяется соотношением:

$$\cos(\gamma) = \frac{d\ e}{|d||e|}$$

$$e = [\mathrm{A}_i, \mathrm{B}_i, \mathrm{C}_i]\,sign(L_i(p))$$

d – смотрит в центр грани
d = [*x*1, *y*1, *z*1], где
*p*1(*x*1, *y*1, *z*1) – центр грани



Грань видима наблюдателю, если

$$0^0 < \gamma < 90^0$$

Закраска многоугольников

Алгоритм закраски основан на построчном сканировании экрана в пределах выделенного окна, ограниченного размерами Xmin, Xmax, Ymin, Ymax.

Количество пересечений строки с ребрами фиксируется специальным счетчиком по принципу "чет-нечет". Производится развертка и при достижении первой точки пересечения с многоугольником (состояние «нечет») происходит включение закраски, а при достижении следующей точки пересечения (состояние «нечет») - выключение закраски.

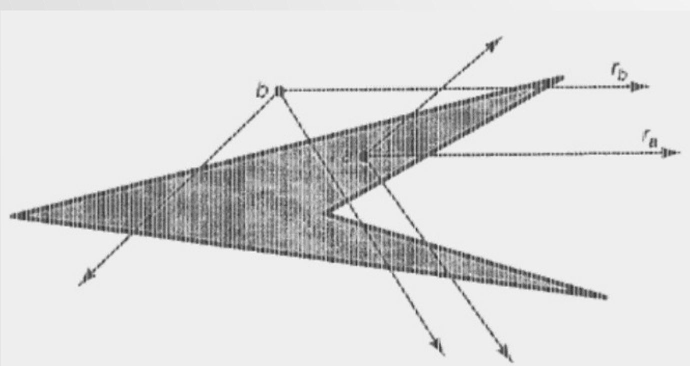
В особых случаях, когда сканирующая строка с номером *j* попадает на концевую точку ребра многоугольника, обработка выполняется по следующим правилам:

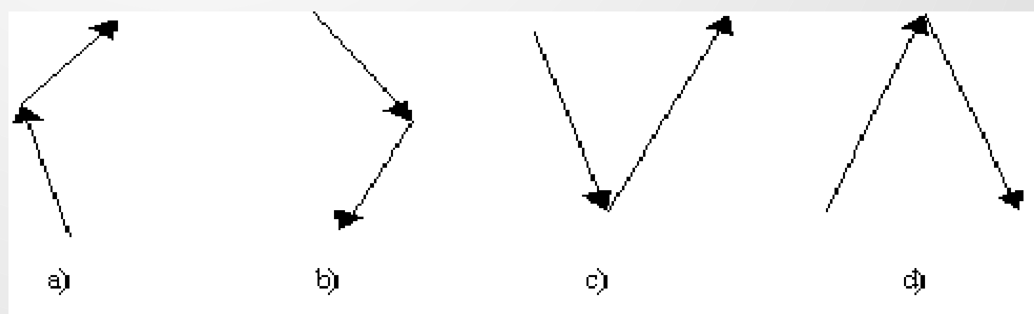
1) если

если *,* , то точка пересечения (,) игнорируется

2) если

если *,* то точка пересечения (,) игнорируется





Закраска пространственных многоугольников. Алгоритм Z-буфера

Алгоритм включает следующие этапы:

1) Вначале в *Z*-буфер заносится максимально возможное значение *Z*, а буфер регенерации заполняется значениями пикселей, описывающих фон.

2) Каждый многоугольник преобразуется в растровую форму и записывается в буфер регенерации, при этом нет необходимости в их предварительном упорядочении.

При разложении в растр дополнительно к рассмотренному выше алгоритму закраски выполняются следующие шаги:

а) вычисление глубины многоугольника $Z(X, Y)$ в точке (X, Y)

б) если $Z(X, Y)$ меньше, чем значение *Z*-буфера в позиции (X, Y), то величина $Z(X, Y)$ заносится в *Z*-буфер. Значение цвета (яркости) пикселя, которое имеет многоугольник в данной точке, помещается в элемент (X, Y) буфера регенерации.

Реализация алгоритма требует большого объема памяти.

Вычисление координаты *Z* для каждой точки сканирующей строки производят в соответствии с уравнением плоскости, проходящей через развертываемый многоугольник: $Ax + By + Cz = 0$,

где коэффициенты *A*, *B*, *C*, *D* определяют по трем точкам (вершинам),