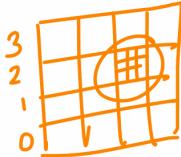


- Simulating Multiple Intensities of Color with An Ink-Pen Output Device: Red, Green, Blue ink...
- 4" atomic output": 
 - Use a 3×3 mega-pixel:  $4^9 = 2^{27} = 2^8 \approx 250,000$ possibilities

Application:
given: (r, g, b) triple
real number



$$r, g, b \in [0, 1]$$

Alg: determine how many R, G, B pixels to "turn on"

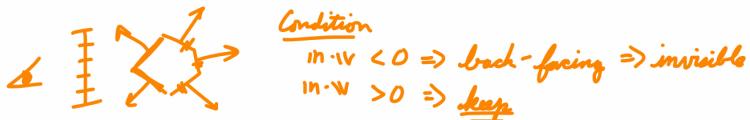
- 1) # on "intensity" = $\text{round}(9 \times \max\{r, g, b\})$
- 2) # RED = $\frac{r}{r+g+b} \times \# \text{on}$
- 3) # GREEN = $\frac{g}{r+g+b} \times \# \text{on}$
- 4) # BLUE = $\frac{b}{r+g+b} \times \# \text{on}$

round to INTEGERS
take largest intensity as % on

Hidden Surface Removal

'Objects': Closed objects polygonal/polyhedral models consisting of a set of planar faces/facets (triangles)
(Objects can be convex or non-convex!)

First: "Back-face Culling" - Eliminate all back-facing faces from scene



When does this work or not work / fail?

① 1 convex object



\Leftarrow I \Downarrow  works

② 1 non-convex object



\Leftarrow I \Downarrow  not back-facing

\Rightarrow keep \Downarrow

③ Multiple objects - ①



\Leftarrow I \Downarrow  works

④ Multiple objects - ②



object actually occluded but depth \Downarrow

\rightarrow Needed: List-priority algorithms to render set of triangles - in back to front ORDER ②

② must SET all triangles w.r.t. depth (z_e -values)

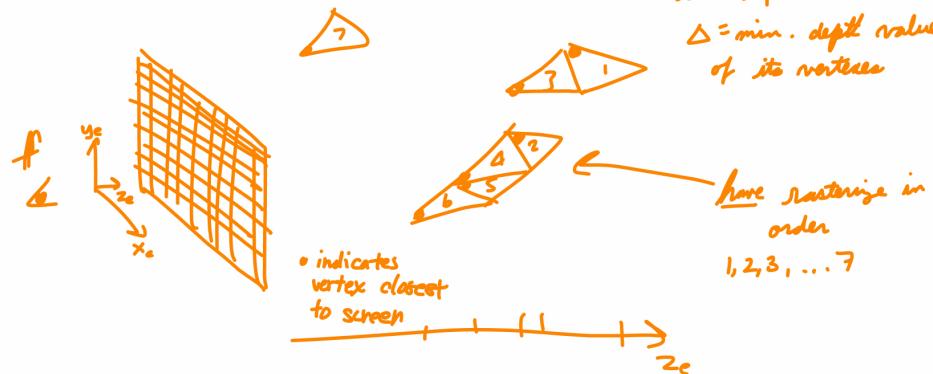
Ex: Painter Algorithm

① Order all (front-facing) triangles w.r.t. their DEPTH

② Renderize order triangle set

③ Quick sort necessary:

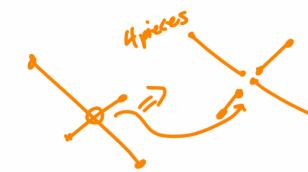
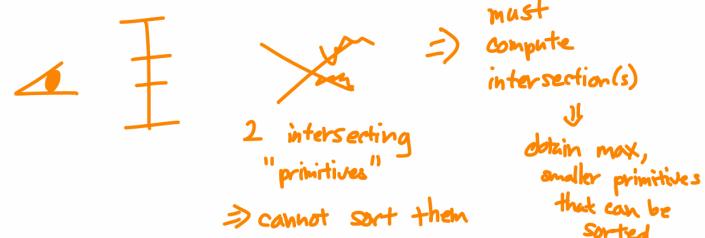
Illustrations



DEF: Depth of a triangle
 $\Delta = \min.$ depth value of its vertices

have rasterize in order
 1, 2, 3, ..., 7

Ugly case: Intersecting objects/triangles



Store z value in z buffer

if pixel in same ij has lower z value, override update z buffer

Reality: Z-buffer

"when rasterizing a pixel (i, j) also store its z_e -value;
 when a pixel is to be "turned on", determine whether this new pixel is closer to the eye than the pixel possibly already rendered there"