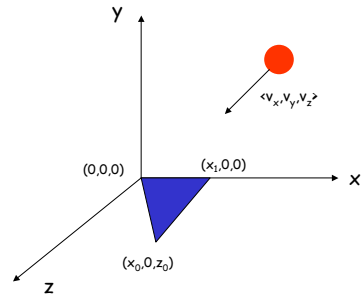
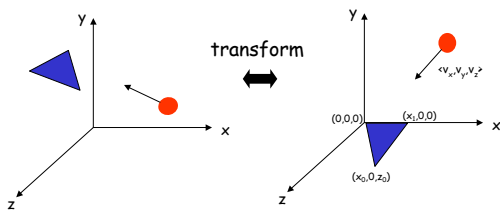


collision detection prototype

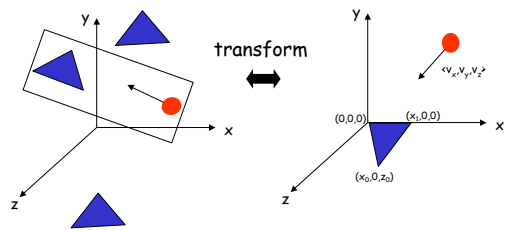
the world



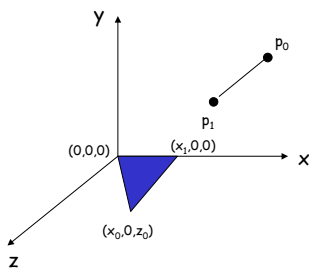
later



later later



"easy" case



1. ball center moves from p_0 to p_1
2. ball does not intersect triangle when center at p_0
3. ball has radius r

for now ...

- ignore edge & vertex collision
- ignore collisions on the "back" of the triangle

why

- i'll give you code for edge and vertex collisions
- we'll make sure we can never collide with the back of a triangle

for now ...

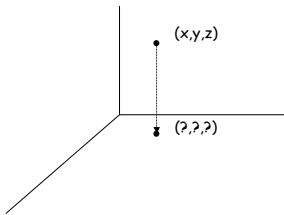
- ignore edge & vertex collision
- ignore collisions on the "back" of the triangle



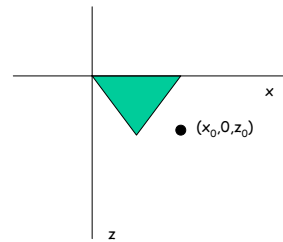
p_0 is at least r units above the x - y plane and the ball is moving toward the x - y plane

some tricks: projection

what is the projection of (x,y,z) into the x - z plane?

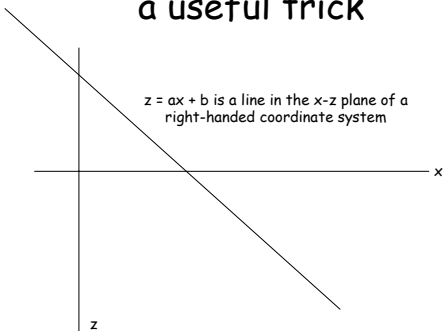


some tricks: in/out



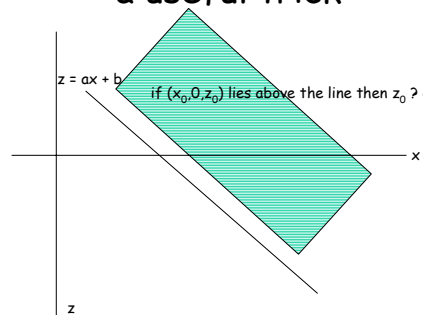
a useful trick

$z = ax + b$ is a line in the x - z plane of a right-handed coordinate system

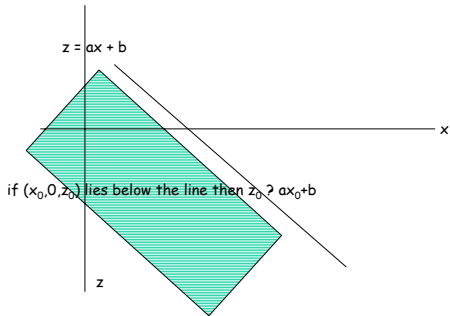


a useful trick

$z = ax + b$
if $(x_0, 0, z_0)$ lies above the line then $z_0 > ax_0 + b$



a useful trick



collision prototype

- I can set the ball's initial position
- I can set the ball's initial velocity
- I can set the ball in motion. It stops when there is a front face collision or it leaves the screen.

collision prototype cont.

- due thursday 10/23
- you'll submit your code (use `cs121submit`) and demo in class
- start from the test harness