```
from visual import *
ball = sphere(pos=(-5,10,0), radius=0.5, color=color.cyan)
wallB = box(pos=(0,0,0), size=(12,0.2,12), color=color.green)
ball.velocity = vector(1,0,0)
ball.trail = curve(color=ball.color)
deltat = 0.005
t=0
while t < 12:
    rate(100)
    if ball.pos.y < wallB.pos.y:

        #note to have the ball bounce realistically change the code below:

        ball.velocity.y = ball.velocity.y

        # to this:ball.velocity.y = C*ball.velocity.y
        # where 'C' is a tuneable parameter less than 1

    ball.velocity.y = ball.velocity.y - 9.8 * deltat
    ball.pos = ball.pos + ball.velocity*deltat
    ball.trail.append(pos=ball.pos)
    print (t,ball.pos.y)
    t = t + deltat
```

← this is Student code

for the ball we used 1m drop → 0.8 m rebound. the parameter that works should be ~ 0.89

Kinematics

$V_f^2 = V_0^2 + 2g(y - y_0)$ ⇒ $V^2 = V_0^2 + 2g(0.8y - 0)$

$V_f^2 = 2gy$ $\quad 0^2 = V_0^2 + 2g(0.8)y$

if $y_{new} = 0.8y$ $\quad V_0 = \sqrt{0.8}\, V_f^{(1)}$

$V_f^{(1)} = 2gy$ $\quad \therefore C = \sqrt{\text{Height Ratio}}$