**Game Programming 12**

Moving Objects Review 01

*Consider moving objects without using the speed and direction variables.*

**How do we move a player 4 pixels to the right (no checking for collisions) ?**

x = x + 4

**You don't want a player to go through walls! How would you code checking for wall objects before moving the player to the right?**

if place\_meeting(x+4, y, o\_wall) = false { x=x+4 } OR

if instance\_place(x+4,y,o\_wall) < 0 { x = x + 4 }

**How would you code: if the right key is pressed and there are no walls in the way, move to the right.**

if keyboard\_check(vk\_right) and place\_meeting(x+4, y, o\_wall)=false { x=x+4 }

**Consider a player object inside of a grid type game. The choice of player speed, sprite sizes, and collision mask sizes is important. Explain using a grid game where the objects are all 20X20 square sprites.**

You must decide on a grid size, say 32 X 32. Once you know the grid size, you should select your sprite sizes and speeds so that they ‘work’ well with the grid. You would want to make sprites 32X32 or 16X16. You would want speeds that divide nicely into 32, like 32, 16,8,4,2, and 1. If you don’t you will get sprites ending up at intersections at positions that are not exactly in the center of the intersection and then have problems moving.

You also want to make sure that you set the collision box for the sprite to be the full 32X32 space. If you don’t, the shoulder or arm or leg of the sprite can ‘catch’ onto corners. You do not want to use precise collision detection!

**In projectile motion and space motion we used the variables x, y, vx, vy. How were these variables used to give objects motion without using the speed and direction variables?**

Instead of using speed and direction, we manually jump the objects with x and y positions. We keep changing them in the step method to make the object move. In the step, you would want to change the x and y position with x = x + vx and y=y+vy

**A tower is going to always fire balls to the right at a speed of 8. Code the movement of the ball without using the speed and direction variables.**

in create add vx=8
in step add x=x+vx

**How would you make the ball leave the tower at an angle of 30 degrees without using the speed and direction variables (build off of the previous question).**

in create add vx= 8 \* cos(degtorad(30))

 vy = -8\*sin(degtorad(30))
in step add x = x + vx
 y = y + vy

**A rock is going to start stationary and fall downward with gravity, speeding up. You only need to consider 'y' direction movement. How would you code this movement without the speed and direction variables?**

in create add vy = 0
in step add vy=vy+0.05
 y=y+vy

**How is wind coded differently from gravity?**gravity is a constant pull downward with one value.
the affect of the wind depends on the *difference* between the speed of the object and the speed of the wind, affect of wind = (global.wind-objectspeed ) / 20