LAB 02-3: Modeling motion of a projectile (with no air resistance)

In this activity, you will model the motion of a projectile. We will refer to the projectile as a ball. Before you begin, write down the following data that you will need for your model.

mass of ball	0.6 kg
magnitude of Earth's gravitational field near Earth's surface	9.8 N/kg
net force on the ball	$< 0, -mg, 0 >$
initial position of the ball	$\langle 0, 2, 0 \rangle$ m
initial velocity of the ball	$\langle 5.74, 8.19, 0 \rangle$ m/s

Writing your program

It is helpful to start with a previous program, as opposed to starting one from scratch.

Start with a previous program

1. Open your previous program that modeled the motion of a ball rolling on a track.
2. Save this file with a new name like `projectile.py`.
3. Make the track appear as a floor (or grass or whatever) and change its name to `floor`. Leave its position the same but change its size to 20 m long (x direction) and 10 m wide (z direction). (Hint: use `size=(20,0.05,10)` which sets the size of a box in VPython.
4. Run your program. You should see a 3-D floor that is longer and wider than the original track. The cart, in comparison, will barely show up as a dot in the picture.
5. Delete the cart and instead create a sphere of radius 0.4 m that is at the desired initial position. Name it `ball`. Make its color yellow or green or cyan or magenta or something like that so that it will clearly show up. Note that this radius is unrealistically large (compared to a basketball or soccer ball for example); however, it's intentionally large so that it is clearly visible in the simulation.
6. Change the word "cart" to the word "ball" everywhere in your program.
7. Set the mass of the ball to 0.6 kg (for a basketball).
8. Set the initial velocity vector to the desired initial velocity.
9. Define a constant for the value of g, the magnitude of Earth's gravitational field near Earth, using `g=9.8`.
10. Change the condition of the while statement so that the loop will end when the ball hits the floor. Use something like `while ball.pos.y > 0:`.
11. Define the net force to be equal to the gravitational force by Earth on the ball.

    ```
    Fnet=vector(0,-ball.m*g ,0)
    ```

12. Comment out any print statements because they slow down the animation.
13. Leave all of the other lines of code the same. Run your program.

Drawing graphs

Your previous program already draws the x vs. t and v_x vs. t graphs for the ball. We want to add y vs. t and v_y vs. t graphs and display them in an organized way.

1. In your previous program, set the x and y positions of the x vs. t and v_x vs. t graphs so that they are neatly arranged. For example, use values `x=450, y=0` and `x=450, y=200` for each of the two graph windows.
2. Now, create two new graph windows and plots for graphing y vs. t and v_y vs. t. It is easiest to copy the pertinent lines for the other graphs, paste them, and edit their properties as needed. Set their window positions to `x=450, y=400` and `x=450, y=600` Change all appropriate values. Check the quantities being plotted, the axis labels, the graph titles, etc. to ensure that they are correct.
3. Run the program. Sketch all four graphs below.

Showing a trail for the object's path

You would like to display a trail that shows the path of the object.

1. As the last line before the `while` loop, create a curve named `trail` that has the same color as the ball, as shown below.

```
trail=curve(color=ball.color)
```

2. As the last line inside the `while` loop, place a point on the curve that is the position of the ball, as shown below.

```
        trail.append(pos=ball.pos)
```

3. Run your program. You should see the path of the ball.

Application

1. Simulate the motion of the basketball for the example shown in class. Print the clock reading, position of the ball (ball.pos), and velocity of the ball (ball.v). Find the position and velocity of the ball when it is at y = 3.0 m (and on the way down). This is when it goes through the basket. It's nice to also add the basket to the simulation and see the ball hit the basket. (You can use a sphere or a ring for the basket).

What is the clock reading, position, and velocity of the basketball when it hits the basket?

What is the clock reading, position, and velocity of the basketball when it is at its peak?

2. A soccer player wants to kick a soccer ball a horizontal distance of 50 m at an angle of 45 degrees. What initial velocity is required? (Neglect air resistance in your model.) Compare the results of your computer model and a theoretical calculation.