Name & ID

University of Alabama
Department of Physics and Astronomy

PH 125 / LeClair Fall 2017

Python exercise - mass and spring

Imagine we have a ball of mass m initially at speed v_i striking a spring with spring constant k. How far will the mass compress the spring? We know we can solve this problem easily with conservation of energy. As the mass begins to compress the spring, its kinetic energy will be transferred to elastic potential energy of the spring. When all of the kinetic energy has been transferred, the block stops for an instant and then reverses direction. Thus, the maximum compression is when the mass' initial kinetic energy is equal to the spring's elastic potential energy at maximum compression $x_{\max}$:

$$K_i + U_i = K_f + U_f \tag{1}$$

$$\frac{1}{2}mv_i^2 + 0 = 0 + \frac{1}{2}kx_{\max}^2 \tag{2}$$

$$x_{\max} = \sqrt{\frac{m}{k}}v_i \tag{3}$$

What if we knew about conservation of energy, but didn't know how to solve the equations? In this case it is quite easy, but we can easily imagine situations where we can't solve the equations analytically (e.g., if air resistance were added). How we simulate the result?

What we could do is calculate our starting energy, let the mass move a tiny bit and compress the spring, and then re-calculate our kinetic and potential energy. We keep repeating that process until the kinetic energy is zero. In pseudocode,

1. define constants k, m, v_i
2. write function $K(m, v_i)$ to calculate kinetic energy for a given m, v_i
3. write function $U(k, x)$ to calculate the potential energy for a given compression x
4. start at $x = 0$, calculate starting kinetic energy K_i.
5. step forward by Δx (say, 0.01 m), calculate new potential energy U and kinetic energy ($K = K_i - U$)
6. repeat as long as $K >= 0$

Task 1: write a program based on the pseudocode above to return the particle's stopping position (which is also the maximum compression of the spring) for a given initial velocity. Print your code to turn in.

As a test case: for $k = 1.3\,\mathrm{N/m}$, $m = 1.0\,\mathrm{kg}$, and $v_i = 5.0\,\mathrm{m/s}$, you should find a compression of $x_{\mathrm{stop}} = 4.39\,\mathrm{m}$. More generally, we could test this by doing the calculation for a variety of initial

velocities. Our equations above indicate that a plot of x^2_{stop} vs v_i^2 should yield a straight line of slope m/k.

> Task 2: for at least 10 values of v_i, verify that x^2_{stop} vs v_i^2 is linear with a slope equal to the m/k ratio you chose. Print your graph with a trend-line and trend-line equation.

Probably you just kept changing the value of v_i and re-ran the code 10 times. We can do better! Using `for` and `range` statements, modify your program to automatically produce 10 values of x_{stop} and v_i. Pro-tip: if you print the output like `print "%f%f" %(x,v_i)` you can copy and paste the values directly into an excel sheet. Make the computer do the tedious stuff!

> Task 3: modify your code to print out at least 10 values of x_{stop} and v_i.

We can modify our program to do a few more things. We might like to know what the velocity as a function of position is, for example.

> Task 4: for a given (single) v_i, modify your first code to print velocity as a function of position. Make a graph of v versus x and turn in your plot.

Finally, what we might really be interested in is the acceleration of the mass. Clearly it won't be constant for a spring, it should depend on how far the spring has been compressed. So far, we have no idea how to handle non-constant accelerations, but we can certainly simulate it. If at a given position we know the velocity at that point and the *previous* position, we can use

$$v_f^2 - v_i^2 = 2a\Delta x \tag{4}$$

to find the acceleration. We calculate v each time around our loop, we need only store the previous value now before updating it. The Δx value in this case is just how far you move the particle between steps in your loop.

> Task 5: solve the equation above for a, and modify your code to produce values of a for each x. Make a plot of x vs a. You should find a straight line graph. What does the slope represent?

Essentially, what we have just done is arrived at Hooke's force law for springs using energy principles. More on that next week!