Worksheet #7 - PHY102 (Spr. 2004)

Collisions

Due March 15th, 2004

In this worksheet, we will return to solving equations and solving differential equations.

Often there are multiple ways of accomplishing something in *Mathematica*. Usually one way is easier than another but less elegant. Why might you want to use the elegant method rather than the "easy" one? Because it can often save trouble later on in your mathematica session. Here is an example. Let's say you want to know the distance a mass of 50kg falls in 30s after falling out of an airplane. Obviously we need to use $y = v_0 t + \frac{1}{2}at^2$ where $v_0 = 0$, $t = 30s$ and $a = -g = -9.81m/s^2$. The simplest way is to type into mathematica:
y = -9.81*30^2/2.
An obviously more elegant route is to type:
v0=0;
a=-9.81;
t=30;
y=v0*t + a*t^2/2
or a more space-saving way would be:
{v0,a,t}={0,-9.81,30};
y=v0*t + a*t^2/2
The problem with both of these approaches may come later because you have permanently defined the variables v0, a and t to these values and wherever they appear later in your *Mathematica* notebook these values will be substituted leading to unpredictable results. (It is to clean up messes like this that we regularly use **Remove["Global`*"]**). The most elegant solution to this problem is to define the equation as an equation then seek the solution with particular "substitutions":
y[t_] := v0*t + a*t^2/2
sol1 = y[t] /. {v0 → 0, a → -9.81, t → 30}
This solution is elegant because the function of interest is defined as a function so we can operate on it (for example, find its derivative etc.). We found the particular solution we were looking for (i.e., got the same result as we did when we used the "easy" methods shown above) but didn't permanently reset the values of any internal variables in mathematica. Try it. Enter the

different commands above. Then check what *Mathematica* thinks the variables (e.g., a and v0) are after each case. Use a **Remove["Global'*"]** in between each test. Many of you have already been using this "substitution" technique if you have been using the **DSolve** example given out with worksheet 4. When you use **DSolve** or **Solve**, for example, the solutions to the equation are returned as a list of substitutions. This is discussed in more detail below.

Enter the following code:

```
(*You can structure your equation solvers like this*)
(*This solves two simultaneous linear equations*)
    f1[x_,y_]:=a*x+b*y-c
    f2[x_,y_]:=c*x+d*y-e
    sol=Solve[{f1[x,y]==0,f2[x,y]==0},{x,y}]
    {x,y}={x,y}/. sol[[1]]
    (*This checks to see that the solutions are correct*)
    Simplify[f1[x,y]]
    Simplify[f2[x,y]]
```

Problem 1

Use mathematica to solve the following problem (see the example above) A ball of mass m moving horizontally with a velocity u undergoes a head on *elastic collision* with another ball of mass M travelling at velocity U. Apply conservation of momentum and energy to find expressions for the final velocities of these two particles as a function of m, M, u and U. Verify your solutions by confirming that they preserve energy and momentum conservation.

Now find the final velocities for the following cases:
(i) $m = M$
(ii) $m = 2M$

Problem 2

A particle of mass m strikes a pendulum of length l and of mass M which is initially at rest and becomes embedded in it. The center of mass of the pendulum(plus particle) rises a vertcial distance h.

(i) Find the initial speed of the particle. Find the maximum angle by which the pendulum swings.

(ii) Now suppose $m = 1$, $M = 9$, $g = 10.0m/s^2$, $h = 4m$, $l = 10m$. Solve the non-linear differential equation for the pendulum's motion. Hence plot the behavior of the pendulum angle as a function of time. Now solve the linear pendulum problem using the same parameters. Plot the time dependent oscillations of the linear and non-linear solutions on the same graph. Do you think it is legitimate to approximate the motion in this problem by the linear(simple) pendulum equation?