EXTRACTING SOLUTIONS FROM THE SOLVE FUNCTION

In lab we studied how to use the Solve function in Mathematica to, well, solve both simple and more complex equations. Let' s apply this to the example in class : determine how long an object will be in flight if it is thrown vertically down with an initial speed of 3 m/s from a height of 100 m. We know that the equation of motion for this object (neglecting friction) is

$$y(t) = 100 - 3t - 4.9t^2$$

and solving the resulting quadratic when y (t) = 0 will yield the time of flight. In Mathematica, this is done simply via :

In[54]:=
```
Clear[y, t]
y[t_] := 100 - 3 t - 4.9 t^2
Solve[y[t] == 0, t]
```

Out[56]=
```
{{t → -4.83402}, {t → 4.22178}}
```

And we get two solutions, knowing that only the second one is meaninfgul. How can we extract this one solution from the set of solutions :

In[57]:=
```
Solve[y[t] == 0, t][[2]]
```

Out[57]=
```
{t → 4.22178}
```

The notation [[Q]] will extract the Qth element of a list. However, suppose we wanted to use this value of time to do a further calculation. Suppose our object wasn' t thrown down vertically, but thrown horizontally with an initial horizontal velocity of 5 m/s. How could we determine the horizontal distance traveled by multiplying the horizontal velocity by this time of flight? We use the "slash-dot" command :

In[58]:=
```
timeofflight = t /. Solve[y[t] == 0, t][[2]]
range = 5 timeofflight
```

Out[58]=
```
4.22178
```

Out[59]=
```
21.1089
```

You see in the first output line that the time of flight appears as a number (not an element in a list), and the second output line is the correct value of range. The slash dot command substituted into t the result of Solve[y[t] == 0, t][[2]]. In the way I solved the semester project, I made use of this formalism to get very important intermediate results. There are many ways of solving the semester project, but this is a useful technique to know.