How to incrementally move a car propelled by a rocket engine (constant mass & no drag). (Programming drag and a decreasing mass due to fuel consumption is not difficult, but has been omitted to keep the example simple.)

1. "Read in" the data from the Excel spreadsheet describing the mass of the object, the initial conditions Xo and Vox, information related to Fnet (in this case the Thrust), the total time (Total_time) the simulation is allowed to run, and the suitably short time interval Δt (deltaT) that the object will be successively moved. *Actual "Read" & "Write" commands are not shown.*

2. Create a loop that iterates (or repeats itself) over and over while moving the object by means of the kinematics equations for a little bit of time deltaT each loop.

3. Use the final position & velocity at the end of each loop as the initial conditions for the next loop.

4. "Write" Xf, Vfx, and t back onto the spreadsheet to be used as data for a graph that tracks the motion

5. Repeat the process until the Total_time has been reached.

Read in mass, Thrust, Xo, Vox, Total_time, and deltaT from the spreadsheet.

The loop begins with the command "Do" which tells the computer to do whatever follows.

Do

F_{net} and a_x are calculated.

Fnetx = thrust
ax = Fnetx / mass

The kinematics equations are applied for the short interval deltaT to determine the final location and velocity at the end of each loop.

Xf = Xo + Vox*deltaT + (1/2)*ax*deltaT^2
Vfx = Vox + ax*deltaT

Xf and Vfx are made to be the initial conditions Xo and Vox for the next iteration/loop.

Xo = Xf
Vox = Vfx

t = t + deltaT

The current time "t" is updated by the increment of time, deltaT that is spent in each incremental motion. "t" is only required if we wish to plot any of the data as a function of time.

Write Xf, Vfx, and t at the end of each loop back onto the Excel spreadsheet.

The looping continues the simulation until the Total_time has been reached or exceeded.

Loop Until t >= Total_time