1: Understanding Looped Calculations

Consider the following piece of Mathematica code and its output.

```
For[ i = 1; x = 0,
        i < 10,
        i++,
        x = i^2;
        Print[x]
]
```

Output:

1 4 9 16 25 36 49 64 81

(a) Explain what the code is doing. In particular, point out different statements in the code and their purpose in this algorithm. If any lines are unclear, try the Mathematica help.

(b) Now, write a piece of code using Mathematica that divides the number 1500 by the first 100 integers. Run that code. (*Hint*: Where should you move the **Print[x]** command so that you only get one answer? How do you get a numerical result instead of an analytical result?)

✓ **CHECK** your results with your instructor!

2: Numerically solving a simple differential equation

(a) Consider an object that moves with a constant velocity of +5 m/s. Write down the differential equation which describes this motion.

We can use a **for loop** to numerically integrate this differential equation. Consider this piece of code:

```
For[ v = 5; x = 0; t = 0; dt = 0.01,
        t < 20,
        t = t + dt,
        x = x + v*dt;
        Print[x]
]
```

(b) Without actually running the code yet, explain what this code is doing. In particular, point out the different statements in the code and their purpose in this algorithm.
Make sure to think carefully (connect it to part a) about what this line does: **x = x + v*dt**.

(c) Again, without actually running the code yet, what would the output of this code look like? You can just write down the first few results. How many times would this loop be executed?

✓ **CHECK** your results with your instructor!

3: Storing computations and plotting

Printing the output from the loop is not particularly useful (that's why I didn't have you run the code yet). What you really would like to do is visualize it (*i.e.*, plot the results). We can start to do that by storing the results in a Mathematica table (essentially a list of numbers).

This piece of code will perform the numerical integration (constant 5 m/s velocity) we defined in Part 2 and plot the results:

```
xx = tt = vv = Table[0, {i, 1, 201}];
dt = 0.1;
ifinal = 201;                    (A semicolon at the end of a command suppresses the output for the command.)

For[ i = 1,
        i < ifinal,
        i++,
        vv[[i]] = 5;
        xx[[i + 1]] = xx[[i]] + vv[[i]]*dt;
        tt[[i + 1]] = tt[[i]] + dt;
]
ListLinePlot[Table[{tt[[i]], xx[[i]]}, {i, 1, 200}],
        AxesLabel → {"t (s)", "x (m)"}           (typing –> gives a →)
]
```

(a) Try it out! Explain what this code is doing. In particular, point out the different statements in the code and their purpose in this algorithm. Which differential equation(s) is this code modeling?

(b) How would the code change if instead we had an object traveling with constant acceleration?

(c) Modify the above code for an object starting with $x = 0$ m and $v = 0$ m/s, but accelerating at a constant acceleration of +1 m/s^2. Run it. Do the plots of x and v look right?

✓ **CHECK** your results with your instructor!