

Process model formulation and solution, 3E4

Computer software tutorial - Tutorial 1

Kevin Dunn, dunnkg@mcmaster.ca

September 2010

Assignment objectives

- Three questions that will help you get comfortable (again?) with MATLAB or Python
- Creating vectors and matrices
- Mathematical manipulation of variables
- Displaying simple plots of these vectors and matrices

Recap of tutorial rules

- Tutorials can be completed by yourself, or with one other person.
- Tutorials must be handed in by the end of the tutorial session in the lab.
- Please give the TA a paper submission, no electronic submissions - thanks!

Question 1 [1]

Note: This question is a recap of creating vectors and generating plots.

A pellet that is shot vertically upward with initial speed u will have vertical displacement, s , after a time, t , given by the formula:

$$s = s_{\text{up}} + s_{\text{down}} = ut - \frac{gt^2}{2}$$

where $g = 9.81 \text{ m/s}^2$ is the acceleration due to gravity, and we ignore the effect of air resistance. We would like to plot a graph that shows the vertical displacement as a function of time. Also assume that the initial speed is 50 m/s.

1. Create a vector, called τ , that contains the time steps from $t = 0$ to $t = 10.0$ in increments of 0.2 seconds.
2. How many elements in this vector? Write down the line of code used to calculate the vector's size.
3. Create a vector, s_{up} that contains the first term in the above equation.
 - How many elements in this vector?
 - What is the smallest value in the vector?
 - What is the largest value?
4. Also create a vector s_{down} , the downward displacement of the pellet. Write down the line of code used to create this vector.
5. Next create a vector s , as the **sum** of s_{up} and s_{down} . Note that we can add and subtract vectors as long as they have the same dimension.
6. Create a plot of the upward displacement vs time. Your solution should contain the line of code you use.

7. Create a plot of the the downward displacement vs time.
8. Finally, create a plot of the total displacement vs time.
 - label your x- and y-axes with labels that also contain the *units of measurement*,
 - give the plot a suitable title,
 - and make sure there is a grid in the background.

Your solution to this question must include the intermediate questions, but you will be graded on the lines of code and the plot from the last part.

Question 2 [2]

Note: This question is a recap of for-loops and while-loops.

The square root of y , a positive number, can be approximated using Newton's method, a method we will learn about later in the course. The method requires an initial guess, and then iterates (repeats over and over) certain calculations, until the approximation does not improve any more. Here is some pseudo-code for the method:

1. Assign a value to y
2. Set the initial guess of the final answer to $x = y/2.0$
3. Repeat a few times, in this tutorial, repeat 5 times:
 - Replace x with $\frac{x + y/x}{2.0}$. In the future we will write such statements as: $x \leftarrow \frac{x + y/x}{2.0}$
 - Print the value of x to the screen
4. Stop and compare your answer with the built-in `sqrt()` command.

Answer these questions:

- Implement this pseudo code in either MATLAB or Python and show the printed output when calculating $\sqrt{13}$. MATLAB users: please type `format long` so that the output shows 15 digits of precision; Python shows the full precision by default.
- Change your code so that the 3rd step is replaced with a `while`-loop that will stop the code when the change in x from the previous iteration to the current iteration is smaller than 1×10^{-8} .
- How many iterations are required to calculate $\sqrt{65536.0}$ with this criterion?

Bonus question [1.5]

The population count of the United States can be modelled using the formula:

$$P(t) = \frac{197\,252\,000}{1 + e^{-0.03134(t-1913.25)}}$$

where the time, t , is given in years. Write some code that will calculate the population count every **ten** years, from 1800 to 2010. Plot the results and label the plot appropriately.

Does the population ever reach steady-state? If so, roughly when, and what is the steady-state value?

Comment on whether or not this model is realistic. Is it useful?

END