

Matlab for Research session 2

Writing your own scripts and functions

Colin Goldblatt, c.goldblatt@uea.ac.uk

October 24, 2007

1 Introduction

This handout and links to the tutorial and associated scripts are at <http://researchpages.net/people/colin-goldblatt/teaching/matlab/>.

2 Programming tutorial

We will use the “Programming in Matlab” tutorial by Edward Neuman (<http://www.math.siu.edu/matlab/tutorials.html>). Work through **sections 2.1 – 2.8** but **skip section 2.2** (inline functions are not that useful).

3 Don’t reinvent the wheel

The advantage of Matlab is that it comes with lots of built in functions, all of which are probably better than anything that any of us could write. For whatever procedure you want to do, the first thing to do is see if there is an inbuilt function which will do the job. Unfortunately, searching the help is notoriously difficult if you do not already know the name of the function you want. Fortunately, there are other ways of finding things...

A good place to start looking for whatever you want to do is, from the *Help Contents*,
MATLAB → *Functions by category*.

Matlab has various toolboxes (bundles of extra functions which MathWorks charge yet more money for), many of which we have a licence for at UEA. Look at all top level options after *MATLAB* in the help contents, and then look in any toolboxes which look useful. A full list of the available toolboxes is at http://www.mathworks.com/products/product_listing/index.html. If you are going to be doing something a lot, it is probably worth paying for it.

If there is not an inbuilt/toolbox function for what you want, have a look on the (MathWorks supported) Matlab user community site (<http://www.mathworks.com/matlabcentral/>) in the file exchange. There is also a newsgroup which you can post questions at. It is highly probable that someone else has already written a function for any “standard” task that you need to do!

4 Programming

The following principles apply to any programming language

- Write a structure plan, i.e. write down the structure of the programme that you need to write, the equations you will use etc.
- Write lots of comments to identify which part of your code do what. Coming back to uncommented code months/years later is a nightmare!
- Try not repeat bits of code (copy + paste); if you need to change them then you have to do it multiple times. Write a function or script with the code in.
- Avoid hard coding constants. i.e. do not write $F = m*9.81$ in lots of places, write $g = 9.81$ at the top of your script, then $F = m*g$ everywhere else. It makes it a lot easier when you need to change g ...

There are whole books written on the subject of writing good code...

5 Programming tasks

1. Write a function to find $y = ab^2$, supplying a and b as arguments.
2. Write a function to find the roots of a quadratic equation

$$ax^2 + bx + c = 0 \quad (1)$$

using the formula

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2ac} \quad (2)$$

3. This exercise practices the idea of using a while loop. Joe the wants to save some money to buy a house. From his small academic salary he can afford to save only 1000 each year. He deposits this in an account which pays him interest at a rate of 6% per annum. Write a simple script to simulate Joe depositing money into an account and receiving interest each year. Use a while loop so the script stops when Joe has saved 15,000. Enough for his deposit!
4. Write a function which accepts the 24-hour clock time as an argument and displays the that time in 12-hour clock format, e.g. 08:00 \rightarrow 8:00am, 15:30 \rightarrow 3:30pm.
5. Write a function to numerically calculate

$$y = \int_a^b \sqrt{x} \, dx \quad (3)$$

Hint: use the Matlab function `quad` (see the help!), in which you will need to refer to the Matlab function `sqrt`, supplying a and b as arguments [skip this exercise if you don't understand the question. If you want a more chalanging exercise, integrate the Planck function to find the irradiance in the visable wavelength region].