Practical 4: Hand in your solution to MyUni before Thur March 29th at 1pm.

You should read the practical, and prepare before the actual session.

The goals of this practical are

1. To help you with debugging your Matlab code.
2. To get you to modularise your Matlab code into functions, not just scripts.
3. To learn about diffusion using finite difference methods.
4. To improve your LaTeX skills, particularly using the `figure` environment, and internal references in a document.

Tasks:

1. Download the file `debugging.pdf` from http://www.maths.adelaide.edu.au/matthew.roughan/notes/AMP1/04handout.html, and read. You may also want to read the file `matlab_top10.pdf`.
2. Download the file `broken_diffusion.m` from http://www.maths.adelaide.edu.au/matthew.roughan/notes/AMP1/11other.html. This file contains my attempt to write a Matlab function to perform a 1D finite difference method analysis of diffusion. I haven't had time to get it working. Please fix it!

   - The function should take 4 inputs:

     ```
     dt    = the size of the time interval
     dx    = the size of the x interval
     u0    = the initial state of the diffusion system
     n_t   = the number of time steps to run
     alpha = the diffusivity
     ```

   - The function should output the final state of the diffusion system after T steps.
   - In order to test it, you will have to create simple test cases (scripts that call your function and test it), plot results, and provide some convincing evidence that it works correctly.
   - There are 4 bugs that I know about.

   Note that, in general, using your own functions is a great way to write Matlab code. You can then reuse the functions, and test them more carefully. I will (from now on) ask for many exercises to be done by constructing a function.

Hand in work:

- As usual, put your work into an Overleaf document, and hand up a PDF version of the document. However, this week the handed in version of your work should be a bit more sophisticated.
- Use figures and logic to argue that your new function works correctly.
- Please embed your plots from MATLAB in a *figure* environment. See

  `https://en.wikibooks.org/wiki/LaTeX/Floats,_Figures_and_Captions`

  This is a *floating* environment, so it can take time to get used to it. It can put a caption on the figure – please always include a detailed and meaningful caption.

  It also allows you to build *references*. Create a `\label` inside the figure environment, and then you can create a `\ref` somewhere else in the document. In fact, you should always do so whenever you include a figure.
- Please provide a description of your figures in the document, using references to refer to the exact figure numbers.
- Please embed your MATLAB function from the 1st question in the document using either the `verbatim` environment (as in the example), or the `listing` environment if you would like it to be "pretty printed."
- Have a look at the rubric for report writing to get an idea of what I am looking for here.

  `http://www.maths.adelaide.edu.au/matthew.roughan/notes/AMP1/08project.html`

Make sure your document reads well as a stand-alone document. Do not assume that the assessor has access to the questions!

Optional Extension exercise

- Now that you have a function that can perform finite difference analysis on a diffusion system, you can apply this to its own output.
- Use the function iteratively to calculate and plot the results of the diffusion every 100 time steps.
- Now construct a recursive version that does the same thing.
- Create a 2D version of the diffusion function.