FIT3139: Assignment 1
(Due by 11:59pm, Friday, 8 May 2020)

[Weight: $15 = 5 + 5 + 5$ marks.]

This assignment has three parts. The objective of the first part is to motivate the effect of computer arithmetic on numerical calculations which are common to problems in computational science; this part is worth 5 of 15 marks. The second part is about discrete dynamical systems and is worth 5 marks. The third part is about dynamical systems in continuous time and it is worth 5 marks.

Follow these procedures to submit this assignment

The assignment must be submitted *online* via Moodle, and should follow the following procedure:

- Accept the Electronic Plagiarism Statement for this Assignment. All your scripts/program will be scanned using MOSS (a plagiarism detection software). Read Monash Student Academic Integrity policy for consequences of plagiarism.

- All your scripts and reports **MUST** contain your name and student ID.

  - **You are free to program the assignment in either MATLAB or Python or Jupyter notebook.**
  - Your submitted archive must extract to a directory named as your student ID.
  - This directory should contain a subdirectory for each of the two questions in the assignment, named as q1/, and q2/.
  - Within each of those subdirectories the contents include MATLAB/Python scripts and corresponding PDF reports or plain text output depending on what is asked of you for that question.
  - When submitting scripts and reports, choose file names that identify the subquestion. (Eg. q1a_script.py, or q1b_report.pdf, or q2_script_driver.m, or q2_output.txt)

- Submit your zipped file electronically via Moodle.

Part 1

(a) A common computation in many applied mathematical problems involves the evaluation of the function $\log(1 + x)$,* which is often represented as log1p(x). Indeed most math libraries

---

*Here $\log()$ is the natural logarithm function to the base e.

(including MATLAB and Python) define an **in-built function** `log1p(x)` for this purpose, over and beyond using the standard **in-built logarithm function** `log(1 + x)`.

One way to compute this function is by using the following Taylor series expansion over infinite terms:

$$\texttt{log1p}(x) = \log(1 + x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \cdots + \frac{(-1)^{n+1}x^n}{n} + \cdots \tag{1}$$

- Write your own function `my_log1p(x)` that accepts a value x as input and evaluates the above series expansion until EITHER 1,000,000 terms in the expansion have been computed OR when the approximation to $\log(1 + x)$ has **changed** by less than 10^{-6}.

- Your function should output the value it evaluated, the number of terms in the expansion it computed before it returned, the relative error compared to the the in-built `log1p(x)`.

- Test and plot the accuracy of your function (in terms of the relative error) for varying values of x, including the values of x **less than** the `machine epsilon` of your floating-point representation.

What should you submit for this question?

Your submission should include your script that drives your function `my_log1p(x)` for various values of x and plots the relative errors.

Your submission must also include a report. Your report should **briefly** comment on the values of x that lead to the **most inaccurate** results in terms of relative error, and identify/describe the advantages/limitations of using the above expansion. It should also specify the values of x for which you would use `my_log1p(x)` over the in-built **logarithm function** `log(1 + x)`, and *vice versa*. Note, this report should be separate from the script, and submitted in a PDF format, should be succinct and comprehensibly written.

(b) Consider **another** way of evaluating `log1p(x)`. For any given x, one can **choose another value** y such that
$$\frac{1 + y}{1 - y} = 1 + x.$$

- Work out the expansion of $\log(1 + x)$ as $\log\left(\frac{1 + y}{1 - y}\right)$.

- Write another function `my_another_log1p(x)` that evaluates `log1p(x)` using this new expansion.[†]

- Test and plot the accuracy of this new function (in terms of relative error wrt the in-built `log1p(x)`).

What should you submit for this question?

Your submission should contain a script that drives `my_another_log1p(x)` for varying values of x and plots the relative errors. Your report should contain a fully worked out expansion

---

[†]The termination of this series expansion is similar to what you have done in part 1(a).

for $\log(1 + x)$ (with intermediate steps that facilitated the expansion). Compare and contrast the results with those analysed in part 1(a). Your report should comment which of the two approaches lead to greater accuracy and why? Suggest any further observations you might have about the comparison between the these approaches.

Part 2

(a) Consider the dynamical system given by the following difference equation:

$$x_{t+1} = mx_t + c$$

Find the steady states equilibrium (or equilibria) of the system and, if any, the conditions necessary for it to exist.

(b) Find an explicit solution to the system above. That is, a function $f(x_0, t)$ that describes the state of the system at time t, starting where the state at $t = 0$ is x_0.

(c) The system above can be shown to show qualitatively different behaviours for different values of m and c as follows:

- Case 1: $|m| < 1$
- Case 2: $|m| > 1$
- Case 3: $m = 1$ and $c = 0$.
- Case 4: $m = 1$ and $c \neq 0$.
- Case 5: $m = -1$

For each case above, sketch a cobweb plot and explain the behaviour of the dynamical system. You can draw the plots by hand or using a computer program, or both.

(d) A non-linear function f defines a dynamical system with the difference equation:

$$x_{t+1} = f(x_t)$$

It is known that $x^* = f(x^*)$. Using a Taylor expansion of f around x^*, linearise the system to determine conditions for local stability of x^*. You can use the findings of Part 2c to find this answer. Local stability means that the system will converge to x^* provided that it is sufficiently close to it. Local stability means that the system will converge to x^* provided that it is sufficiently close to it.

What should you submit for this question?

You will have to submit a report with your answers to all questions in part 2. If you used any program as part of your argument, please submit the code specifying in the name of the program the question it addresses.

Part 3

This part is about continuous time models that result in a system of Ordinary Differential Equations. Consider a basic SEIR model with the following diagram:
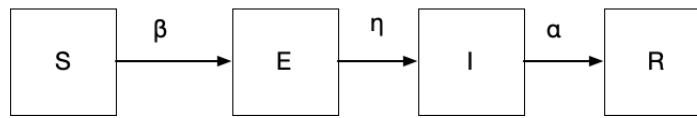

Figure 1:

The compartments stand for Susceptible, Exposed, Infected and Recovered; with a transmission rate β, and recovery rate α. The parameter η is the per-capita rate of becoming infectious, and $\frac{1}{\eta}$ is approximately the length of the so-called latent period, where individuals do not show symptoms and are not contagious. We want to extend this model to have *(i)* some basic demographic features (that is, the population size N is not constant); and *(ii)* to account for asymptomatic transmission.

For all questions, You may use any of the tools studied in the lectures (or combinations of them); including inspecting the dynamics numerically for different parameter values, attempting to find steady state solutions or using phase-plane analysis when appropriate. There is no single way to analyse the models, so we will focus on how you justify what you report on the basis of the techniques of your choice [‡].

(a) Reformulate the SEIR system to include demography, assuming that the birth rate (or rate at which the number of susceptible individuals increase, for example due to migration) is constant, given by Λ, and a constant per-capita death rate μ.

- Draw the flow diagram describing the model.
- Specify the set of equations for the compartments SEIR.
- Using the equations you derived for SEIR, formulate an expression that shows how N changes in time. Use this equation to show that happens to the population size in the long-run.
- How does the long-term behaviour of the system change qualitatively when adding demography, as compared to the closed system that ignores Λ and μ?

(b) Let's now say you can spread the disease without showing symptoms. An asymptomatic compartment is typically added, denoted A. Exposed individuals progress to the symptomatic infectious compartment with probability p, and to the asymptomatic infectious compartment with probability $(1 - p)$. Assume that asymptomatic individuals are contagious, but transmit the disease at a reduced rate q. They also recover at a rate $\gamma > \alpha$. Make sure Λ and μ are also included to reflect the same demographic assumptions of the previous model.

- Draw the flow diagram describing the model.
- Specify the set of equations the compartments SEIR.
- How does the long-term behaviour of the system changes qualitatively when adding asymptomatic transmission? Compare SEIR and SEIAR models for different values of p and q.

[‡]For all numerical explorations, please use $N = 1000$, $E_0 = 1$. You are free to set other parameters to reasonable values. For doing this, it may help to think about the units of the quantities involved. For example, since I and E have units number of people, the units of β must be $1/(\text{number of people} \times \text{unit of time})$. When doing numerical exploration use several parameters to see how things vary and better support your arguments. A good idea is to use similar parameters when making comparisons, for example use the same β values when comparing SEIR and SEIAR.

What should you submit for this question?

You will have to submit a report with your answers to all questions in part 3. Make sure to include graphics and descriptions of the results in the report. Please submit all the code you used for supporting your report, specifying in the name of the program the question it addresses.