ENG1060 ASSIGNMENT – OCT NOV 2020

Due: 8:00PM MYT / 11PM AEDT (Sharp), Friday 22nd January 2021 (Week 11)

Late submissions: A 10% penalty (-1 mark) per day, or part thereof, will be applied. No submissions will be accepted once the penalty has reached 50%.

GUIDELINES

This assignment is to be completed INDIVIDUALLY. Students are advised to review Monash University's policies on academic integrity, plagiarism and collusion. Plagiarism occurs when you fail to acknowledge that the ideas or work of others are being used. Collusion occurs when you work in a manner not authorised by the teaching staff. Again, this is an individual assignment. Do not share your code or code with others. You may discuss ideas with your peers but the approach to coding must be your own. You must have full understanding of it.

All assignments will be checked using the Measure of Software Similarity (MOSS) plagiarism and collusion detection software. Files with high similarity counts will be flagged and reviewed. In the event of suspected misconduct, the case will be reported to the Chief Examiner and the student's unit total will be withheld until the case has been reviewed and a decision has been finalised by the Associate Dean of Education.

INSTRUCTIONS

Download the assignment template files from Moodle and modify the code within the template files (e.g. **`Q1a.m`**, **`Q1b.m`**, etc.). DO NOT rename the template m-files or modify **`run_all.m`**. Check your solutions by running **`run_all.m`** and ensuring all questions are answered as required. The variables will remain in the workspace upon each file call. Do not use close all, clear all, clc in any m-files except run_all.m.

This assignment assesses your ability to apply concepts taught in ENG1060. Therefore, do not use any toolboxes or functions that are not taught in ENG1060, unless otherwise specified.

SUBMISSION

Submit your assignment via Moodle. Read the "**Assignment upload instructions.pdf**" to prepare your **ZIP file** for submission. Your ZIP file must include the following attachments:

a. Solution m-files for assignment tasks (e.g. **run_all, `Q1a.m`**, **`Q1b.m`**, etc.)
b. Any additional function files required by your m-files (e.g. **newraph.m**, etc.)
c. All data files needed to run the code **including the input data provided to you** (e.g. **owid-covid-data-s3.xlsx**, etc.)

It is your responsibility to ensure that everything needed to run your solution is included in your ZIP file. Download your submission from Moodle and test the files to ensure that you have submitted the correct files.

The assignment is based the contents of the submitted files and the output of the **run_all.m** on a Windows-based system (demonstrator's laptop). If you use a MAC, you can test your files on MATLAB Online. The assignment will not be downloaded to your individual laptops for marking. Contact your unit coordinator if you encounter any issues.

Marking

Your assignment will be marked in your usual computer lab session during Week 12**.** You must attend and be present for the assessment to be marked. You will receive a score of 0 if you are absent. Your zip file will be downloaded from Moodle and only these files will be marked on a demonstrator's laptop.

Marking Scheme

This assignment is worth 10% (1 Mark = 1%) of the unit mark. Your assignment will be graded using the following criteria:

1) **`run_all.m`** produces results **automatically** (additional user interaction only if asked explicitly)
2) Your code implements appropriate approaches and produces correct results (printed values, plots, etc…) and adopts good programming practices.
3) Poor programming practice will result in a loss of up to 2 marks out of 10.
4) Your ability to answer the demonstrator's questions that test your understanding of the assignment questions and the submitted code.
5) This assignment assesses your ability to apply concepts taught in ENG1060. Therefore, do not use any toolboxes or functions that are not taught in ENG1060, unless otherwise specified.

Support

1) You may use the function files that you have written in the labs and workshops.
2) You may ask questions in the Discussion Board on Moodle.
3) The m-file templates contain pre-written comments and sections only as a guide. You do not need to follow its structure. You may delete the comments.
4) Hints may be provided during workshops.
5) Bold text has been used to emphasize important aspects of each task. This does not mean that you should ignore all other text.
6) The questions have been split into sub-questions. It is important to understand how each sub-question contributes to the whole, but each sub-question is effectively a stand-alone task that does part of the problem. Each can be tackled individually.
7) It is recommended that you break down each sub-question into smaller parts too and figure out what needs to be done step-by-step. Then you can begin to put things together again to complete the whole.
8) Solve the question, of part thereof, by hand before attempting to code the solution.
9) You may discuss ideas and approaches with peers and demonstrators. However, discussions that lead to similarity in code may result in collusion.

QUESTION 1 **[4 MARKS]**

Background

Coronavirus disease (COVID-19) is an infectious disease caused by a most recently discovered coronavirus which might result in various severity level of respiratory illness. The COVID-19 virus spreads mainly through droplets of saliva or discharge from an infected person. To prevent infection, stay safe by practising physical distancing, wearing a mask, washing your hands or using an alcohol based hand sanitizer. Apart from that, it is better to be well-informed about the disease it causes and how it spreads.
COVID-19 data is collected globally to understand its impact.

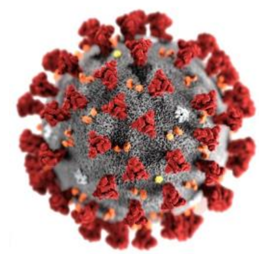


Image: https://www.moh.gov.my/

You are provided with a modified file provided by Our World in Data that contains COVID-19 data. The file is named **owid-covid-data-s3.xlsx**. It contains 60672 rows and 51 columns of data. *You cannot modify any contents of the file.*

Open the file and familiarise yourself with the headers. Some of the key headers include:

- Location: The location/country
- Continent: The continent of the location
- Days_tracked: The days since COVID-19 was first tracked in the location
- Total_cases: The total cases of COVID-19
- Total_deaths: The total deaths due to COVID-19

Source: https://ourworldindata.org/covid-cases

Q1a

In the Q1a.m file, **use importdata() to import the contents of owid-covid-data-s3.xlsx**. It is important to check what fields the imported data structure contains.

Consider the data for Australia, China, India, Indonesia, Malaysia and Vietnam. The majority of the students at Monash University come from the aforementioned countries.

Create *figure(1)*[1] and *figure(2)*[1], plot the following for the 6 countries listed above with a '--.' (dotted-double-dash) line specification. The vertical axis should be logarithmically-scaled and the horizontal axis should be linearly scaled. Turn on the grid and position the legend in the north-west corner.

- [Figure 1] Total cases against days tracked
- [Figure 2] Total deaths against days tracked

Note: The key_locations variable containing the 6 country names has already been provided in the m-file.
Note: Use strcmpi() to compare two strings (case-insensitive).

*You should have two figure windows by the end of this task.

Based on visual inspection, use fprintf() to print a statement that describes which of the 6 countries above has best suppressed the COVID-19 virus.

Q1b

In the Q1b.m file, you are tasked to model the available data for **Malaysia** in order to make several estimates, so that the authorities can make an informed decision on whether the current actions are effective in controlling the spread of the virus.

Create *figure(3)*[1], plot the total cases of Malaysia against days tracked using red circles on the same figure window.

Note the following time periods:

Period	Days Tracked	Action
1	1st – 50th	Initial community transmission stages. Movement Control Order + Local lockdowns introduced
2	51st – 145th	Relaxation of restrictions. Conditional Movement Control Order initiated.
3	146th – 251st	Relaxation of restrictions. Recovery Conditional Movement Control Order initiated.
4	252nd – 275th	New COVID-19 clusters in community. Conditional Movement Control Order and Enhanced Movement Control Order reinstated.
5	276th – 317th	Increment of COVID-19 clusters in community.

Period 1 can be taken as the start of the first wave of COVID-19 cases afflicting Malaysia. Fit Period 1, Period 2 and Period 3 with linear models separately. Subsequently, fit an exponential model to Period 4 and a 2nd order of polynomial to Period 5.

Plot the fitted functions of all periods on the same figure window. Leave the plot formatting up to MATLAB (i.e. don't provide any line specifications). Use ***fprintf()*** to **print the curve fitting equations for each period** to the command window using the **exponential specifier with 2 decimal places.**

Q1c

In the dCdt.m file, **write a function to determine the coefficients of the derivative** of the 2nd order polynomial model obtained in *Q1b*.

Q1d

In the Q1d.m file, **use *Newton-Raphson method* to determine the days tracked when the total cases of Malaysia reach 12×10^4 if the situation worsens.**

Use xi = 300 and precision = 1×10^{-5} for the Newton-Raphson method.

Use ***fprintf()*** to **print a single line answer** to the command window by rounding the answer to an integer.

Note: You will need to use function handle to create two anonymous functions for root finding. You must also use the output of ***dCdt()*** written in *Q1c* in one of the anonymous function.

**You should have three figure windows by the end of this task.*

QUESTION 2 [3 MARKS]

Background

Your mechanical engineering design team is evaluating the average temperature profile of a metal plate with dimensions ($x\epsilon[-2,10]$) and ($y\epsilon[-8,12]$) is depicted in the figure below. The temperature profile is described by:

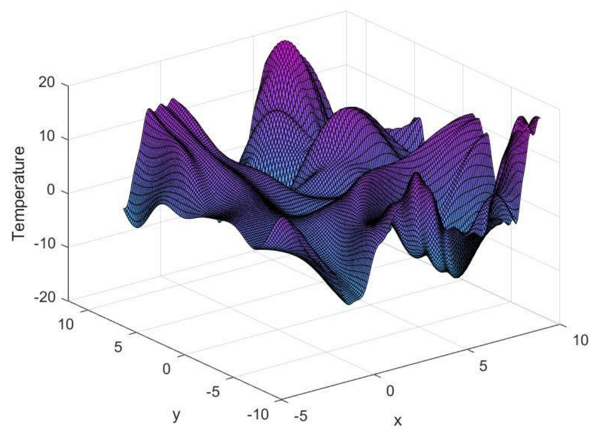
$$T(x,y) = y\cos(x) - x\sin(y) - 3\cos\left(\frac{xy}{3}\right)\sin^2\left(\frac{xy}{5}\right)$$

where *T* represents the temperature.

The average temperature of the metal plate is formulated as:

$$T_{average} = \frac{1}{A}\int_{-8}^{12}\int_{-2}^{10} T(x,y)dxdy$$

where A represents the total surface area of the metal plate.



Q2a

The composite Simpson's 1/3 rule is given by

$$I_{simp\ 1/3} = \frac{h}{3}\left[f(x_1) + 4\left(\sum_{i=2,4,6\ldots,even}^{n-1} f(x_i)\right) + 2\left(\sum_{j=3,5,7\ldots odd}^{n-2} f(x_j)\right) + f(x_n)\right]$$

Also, the composite Simpson's 3/8 rule is given by

$$I_{simp\ 3/8} = \frac{3h}{8}\left[f(x_1) + 3\left(\sum_{i=2,5,8\ldots}^{n-2} f(x_i)\right) + 3\left(\sum_{i=3,6,9\ldots}^{n-1} f(x_i)\right) + 2\left(\sum_{i=4,7,10\ldots}^{n-3} f(x_i)\right) + f(x_n)\right]$$

In the comp_simp13_vector.m and comp_simp38_vector.m file, **create the function file for the Composite Simpson's 1/3 rule Composite Simpson's 3/8 rule, respectively, which integrates the vector inputs.** The function should print an error statement if an incorrect number of points is supplied for integration.

Q2b

In the Q2b.m file, create a 3-by-1 subplot plot in ***figure(4)***[1]. **In each subplot, plot the temperature of the plate along y=2 from x between -2 to 10 (inclusive)** as a red line of thickness 2, using 127 equally spaced points. Additionally, fit the (x,T) data using polynomials (see below) using a blue line. Turn the grid on and include the polynomial degree in the title of each subplot.

- Top subplot: polynomial degree 3
- Middle subplot: polynomial degree 5
- Bottom subplot: polynomial degree 7

**You should have four figure windows by the end of this task.*

Q2c

In the Q2c.m file, **calculate the average temperature of the plate using the comp_simp13_vector()** written in Q2a with the minimum number of spaced points required in each direction (x and y) to achieve within an absolute difference (error) of 4×10^{-6}. The absolute difference (error), *err_diff* is calculated by $err_{diff} = |T_{average,comp\_simp} - T_{average,MATLAB}|$. $T_{average,MATLAB}$ is the average temperature value obtained using MATLAB's integral2() function.

Repeat step above to **calculate the average temperature of the plate using comp_simp38_vector() function file** written in Q2a with the minimum number of spaced points required in each direction (x and y) to achieve within an absolute difference (error) of 4×10^{-6}.

Use **fprintf to print the method which requires less number of spaced points** as shown by the example below.

The ?? requires only ?? of spaced points to obtain the average temperature of ??.

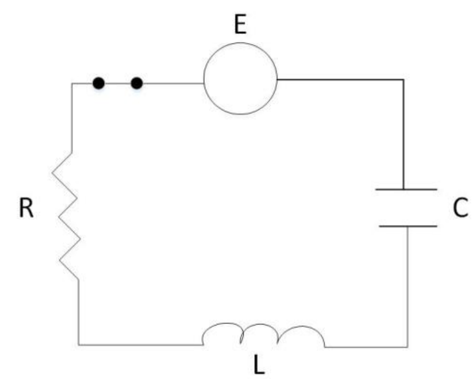
Hint: Start by evaluating the inner integral along the x dimension for each value of y. The resulting values can then be integrated along the y dimension.

QUESTION 3 **[3 MARKS]**

Background

The RLC circuit can be used as oscillator circuits, radio tuning system, filter, and etc. It is an electrical circuit made up of resistor (R), an inductor (L), and a capacitor (C), connected in series or parallel.

Kirchoff's voltage rule states that the total amount of voltage drops within any closed loop is equal to zero. By considering the total voltage drops E(t) across resistor $E_R(t)$, inductor $E_L(t)$ and, capacitor $E_C(t)$, the RLC series circuits can be modelled by the following 2nd-order ordinary differential equation (ODE):



$$L\frac{d^2q}{dt^2} + R\frac{dq}{dt} + \frac{1}{C}q = E(t)$$

where q represents the charge of capacitor (Coulomb), t is time (s), L is the inductance (H), R is the resistance (ohm), and C is the capacitance (F).

The 2^{nd}-order ODE can be solved as a set of 1^{st}-order ODEs, as formulated below. Let's assume z is a 'dummy' variable.

$$\frac{dq}{dt} = z$$

$$L\frac{dz}{dt} = -Rz - \frac{1}{C}q + E(t)$$

You are an electrical engineer and have been tasked to investigate the RLC series circuit. The two 1^{st}-order ODEs above are coupled and must be solved simultaneously. i.e. the solutions are dependent on each other and therefore cannot be solved individually. The first iteration will solve for q_1 and z_1 using initial conditions q_0 and z_0. The second iteration will solve for q_2 and z_2 using q_1 and z_1, and so forth. **You are strongly encouraged to complete a few iterations by hand to fully understand the process.**

Q3a

<u>In the midpoint2.m file</u>, complete the function file to perform the midpoint method to solve two 1st-order ODE equations simultaneously.

Q3b

<u>In the Q3b.m file</u>, consider initial conditions $q(0) = 0$ and $z(0) = 8$.
The parameters are: $R = 0.4\ ohm$, $C = 0.5\ F$, $L = 0.2\ H$, and $E(t) = 50\ V$

Solve for the charge of capacitor q and the rate of flow of charge $\frac{dq}{dt}$ using the midpoint2() function written in Q3a with a time step of 0.2, 0.1, and 0.05 s from $t = 0$ to 10 s.

Create a 2-by-1 subplot arrangement in *figure(5)*[1], and then plot the following panels using the 'colours' variable provided in the m-file to represent time steps of 0.2, 0.1, and 0.05 s, respectively.

- [Top panel] q against t
- [Bottom panel] $\frac{dq}{dt}$ against t

*You should have five figure windows by the end of this task.

Q3c

<u>*In the Q3c.m file*</u>, **use the midpoint2() function to solve the charge of capacitor with $\boldsymbol{R = 0.2, 0.5, 1.0, 3.0\ \Omega}$** of underdamped, critically damped and overdamped cases using a time step of 0.05s. Use the same initial conditions and parameters as in *Q3b*, unless otherwise specified in this question.

Create *figure(6)*[1] and produce a plot of $\frac{dq}{dt}$ (Coulomb/s) against t (s) for each value of R as solid lines. Use the RGB values defined in the 'colourmap' variable as provided in the m-file to colour solutions for all four cases of R (top row to the bottom row).

Note: the legend syntax provided in the m-file will produce the appropriate legend for the plot if R is a row vector.

*You should have six figure windows by the end of this task.

Poor Programming Practices [-2 Marks]
(Includes, but is not limited to, poor coding style, hardcoding, inefficient coding, insufficient documentation, unlabelled figures, output/printing, etc.)

(END OF ASSIGNMENT)