Transient-Response Analysis of 1st Order Systems

Due Time: 23:59, 8 April 2020

Earnings: 6% of your final grade

NOTE: Plan to finish a few days early to avoid last minute hardware/software holdups for which no allowance is given.

NOTE: The code in this assignment must be your own work. It must not be code taken from another student or written for you by someone else, even if you give a reference to the person you got it from (attribution); if it is not entirely your own work it will be treated as plagiarism and given a fail mark, or less.

Purpose: Find the solution of 1st order Ordinary Differential Equations (ODE) using well known methods; namely, Euler's and Runge-Kutta 4nd Order Methods.

Discussion: For the thin, glass-walled mercury thermometer system shown in Figure 1, assume that the temperature of the bath changes based on certain chemical process occurring between two substances reacting with each other inside the bath. It is found that the equation that describes this process is given as follows:

$$\frac{d\theta(t)}{dt} + 2\theta(t) = \cos 4t$$

It can be found that the actual solution of the response of the thermometer, $\theta(t)$, is given by the following equation:

$$\theta(t) = 0.1\cos 4t + 0.2\sin 4t + 2.9\, e^{-2t}$$

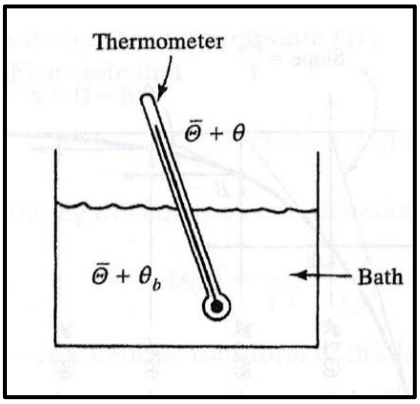

Figure 1 Thin, glass-walled mercury thermometer system

The ODE given above can be solved using many numerical methods, such as Euler's and Runge-Kutta 4nd Order Methods.

1. Write a C/C++ program that computes the solution $\theta(t)$ using Euler's Method. For this step, use the following information: $h = 0.8, 0.2, 0.05$, $\theta_0 = 3$ ℃, $0 \le t \le 2$ second. Find the discrete values of $\theta(t)$ at each h step value.

2. Modify the previous code and implement Runge-Kutta 4th method to solve the same ODE using the following information: $h = 0.8, 0.2, 0.05$, $\theta_0 = 3$ ℃, $0 \le t \le 2$ second. Find the discrete values of $\theta(t)$ at each h step value.

3. Calculate the relative error of the resultant solution at each time for each h step. Your output of your code should show a table that shows the exact temperature, the estimated temperature, and the relative error. The user will choose one method and one step size.

What to Submit: Use Brightspace to submit this assignment as a zip file (**not** RAR, not 9zip, not 7 zip) containing the source code file (ass3.cpp) and a pdf file that shows the plots (bonus). The name of the zipped folder **<u>must</u>** contain your name as a prefix so that I can identify it, for example using my name the file would be kadriaAss3CST8233.zip. It is also vital that you include the file header (as specified in the Submission Standard) so the file can be identified as yours. Use comment lines in the file to include the header.

There is a late penalty of 25% per day - even one minute is counted late.

You may lose 60% or more if:

- The output is wrong
- Your application won't build in Visual Studio 2019
- Your application crashes in normal operation
- I can't build it because you submitted the wrong files or the files are missing, even if it's an honest mistake – this gets 100% deduction.

Don't send me the file as an email attachment – it will get 0.

Example Output

The output of the code should look like below. The results of test case when using Euler's and Runge-Kutta for h = 0.2 are shown in the table below.

```
>> Choose the method for solving the ODE:

1. Euler's Method

2. Runge-Kutta 4th Order Method

>> 1

>> Choose step size "h" (0.8, 0.2, 0.05)

>> 0.2

Time(second)   Exact Temp(C)  Estimated Temp(C)     Percentage Error(%)

0.2            2.157          2.000                 7.28
0.4            1.500          1.339                 10.71
0.6            0.935          0.798                 14.66
0.8            0.474          0.331                 30.13
1.0            0.176          -0.001                100.54
1.2            0.073          -0.131                280.86
1.4            0.128          -0.061                148.01
1.6            0.241          0.118                 50.86
1.8            0.299          0.270                 9.76
2.0            0.236          0.283                 19.89


>> Choose the method for solving the ODE:

1. Euler's Method

2. Runge-Kutta 4th Order Method

>> 2

>> Choose step size "h" (0.8, 0.2, 0.05)
```

```
>> 0.2
```

Time(second)	Exact Temp(C)	Estimated Temp(C)	Percentage Error(%)
0.2	2.157	2.157	0.01
0.4	1.500	1.500	0.02
0.6	0.935	0.935	0.03
0.8	0.474	0.474	0.06
1.0	0.176	0.176	0.15
1.2	0.073	0.073	0.34
1.4	0.128	0.128	0.15
1.6	0.241	0.241	0.05
1.8	0.299	0.299	0.02
2.0	0.236	0.236	0.00

```
>> Choose the method for solving the ODE:
1. Euler's Method

2. Runge-Kutta 4th Order Method
```