Midterm take-home exam (total 10 points), due date: Friday, 10 March 2023 at 11:59 pm

This exam is to be completed individually. By completing and submitting this exam, you commit with the university's section K. 'Integrity and Conduct' of the University Calendar.

Instructions on how to prepare your exam:

You are a computational scientist who gives consultation to other scientists experiencing problems with their computer simulations and algorithms. Read the issues brought by the three scientists and assist them.

The questions of scientists #1 and #2 need to be answered with your own words and no coding. Provide one answer per page, i.e., answer for item 1.(a) is written on page 1, answer for 1.(b) is written on page 2, and so on. Write your answers in the order of appearance as given on the midterm. Indicate on the first line of each page the item that the answer refers to. This can be done by adding '1.(a)', '1.(b)', '1.(c)', etc. on the very first line of each page. Write your answers for scientists #1 and #2 in any text editor of your preference but **it is important that you follow the same format guidelines as in our regular assignments**. Generate then a pdf file to be submitted to the Gradescope dropbox folder.

The problem raised by scientist #3 needs coding. Write a Jupyter Notebook code that can solve the rocket trajectory problem and upload your code to the Gradescope dropbox folder. Your code needs to be fully commented and documented. Your code needs to generate at least 2 plots and output at least three quantities as requested by scientist #3. The quantities cannot be simply mentioned or typed, we need to see your code generating the numbers and/or plots without any interference or syntax error. You are allowed to make visual estimations of values as long as your plots evidence and confirm your estimations. Together with your code, upload a pdf file write-up of 1-page maximum that explains how your code works **(just as in the code workflow section of our regular assignments and with the same format settings)**. The figures should be presented directly in the Jupyter Notebook code (not in the code workflow section). To solve the problem raised by scientist #3, you will need the Jupyter Notebook code provided in the D2L midterm module named `random_generator_masses_ex3.ipynb`.

In total, you will submit to the Gradescope dropbox folder three 3 files:

- 1 pdf write-up containing the answers for scientists #1 and #2;
- 1 Jupyter Notebook code (fully commented) that solves scientist #3 problem;
- 1 pdf write-up containing the code workflow section of the submitted code.

Include your name and UCID in all three files.

ONLY write-ups in pdf format will be accepted. DO NOT use file compressors (such as zip). Submit all 3 files to the Gradescope dropbox folder.

1 Scientist #1: finding the minima of a potential

I am trying to describe the intermolecular interaction between two Krypton (Kr) atoms with the Lennard-Jones potential energy given below

$$U(r) = 4\epsilon \left\{ \left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6} \right\} \tag{1}$$

where r is the interatomic separation, ϵ is the well-depth and a measure of how strongly the two particles attract each other, and σ is the distance at which the potential energy between the two particles is zero. σ gives a measurement of how close two nonbonding particles can get and it is also referred to as the van der Waals radius. The values of ϵ and σ were provided by my experimental colleagues as $\epsilon = 18.2$ meV and $\sigma = 3.7$ Å. I am a bit confused about how to characterize the equilibrium properties of this dimer and how to use computers to determine these properties.

(a) How can I quantify analytically the most stable separation between the atoms and the minimum value of the interatomic potential energy described as in equation (1)? And what are those values? **(1 point)**

(b) I am planning in writing a computer code to obtain the values mentioned in item (a) above. What are the advantages of using computers and coding to obtain the most stable separation and the minimum value of the interatomic potential energy of molecular systems? **(1 point)**

(c) I came across two numerical methods of finding roots of functions: (i) the bisection and (ii) the Newton-Raphson method. I do not know which one to choose to analyze stable equilibrium of the Kr diatomic molecule. Which method would you advise me to use and why? Explain how the algorithm of the method you recommend works. **(1.5 points)**

******

2 Scientist #2: Driven damped oscillator problem

I need to solve the system of differential equations describing the motion of a driven damped oscillator (spring-mass system) given as

$$m\frac{d^2x}{dt^2} + b\frac{dx}{dt} + kx = F_0\cos(\omega t) \quad (2)$$

$$\frac{dx}{dt} = v \quad (3)$$

where t is the time, x describes the position of the object of mass m, v is the velocity, k is the spring constant, b is a constant setting the strength of the damping, ω is the driving frequency, and F_0 is the amplitude of the driving force. I need to solve this system of differential equations and I do not know how to start.

(a) I heard you are experienced with three numerical ordinary differential equation (ODE) solvers: (i) the Euler, (ii) the Trapezoid, and (iii) the Runge-Kutta method. Which one would you recommend me to use in my code and why? Provide evidence and founded arguments for why you are recommending such a method. **(1 point)**

(b) I learned that Runge-Kutta methods can admit higher-order numerical schemes such as RK2 (second order Runge-Kutta) and RK4 (fourth order Runge-Kutta). What is the difference between the RK2 and the RK4? Based on your experience, which RK order do you recommend I implement and why? Provide evidence and founded arguments for why you are recommending such a method **(1 point)**

(c) I believe I still need more help on how to solve the driven damped oscillator problem given above. Please, write a code workflow section that can help me design my own code that solves the system of equations (2-3). **(1.5 points)**

******

3 Scientist #3: rocket trajectory (3 points)

I am planning the launching of a rocket with meteorological equipment and I would like to simulate its trajectory in the most realistic scenario as possible. The rocket will be launched, having its fuel tank full and I would like to determine its maximum altitude until its fuel tank gets empty. After that, the rocket will have no thrust and it will start to fall. I also would like to know how long it takes for the rocket to reach terminal velocity during its fall. I have some information for you about the rocket and the physics we will be using in this simulation. I would like to look at the one-dimensional trajectory of the rocket moving vertically only. Let's omit for now any deviation from the vertical trajectory.

When the rocket is fully loaded with fuel, it has a total mass of M (in kg). Its mass without fuel is M_r (in kg). The rocket consumes fuel at a rate of 130 kg/s. In ascending stage, the rocket is subjected to three forces denoted as F_{thrust}, F_g, and F_{drag}. Their definitions are presented below.

F_g is the force due to gravity where $F_g = Mg$ with M being the total mass of the rocket (plus fuel) and $g = 9.81\, m/s^2$. Note that this mass will gradually reduce as the rocket's fuel is burned with time t. F_{drag} is the drag force opposing the rocket upward motion and it is given by

$$F_{drag} = \frac{1}{2}\rho\, v^2 C A \tag{4}$$

where ρ is the density of air given by 1.22 kg/m^3 and v is the rocket velocity. C is the drag coefficient, a unitless factor related to the shape of the rocket. In this case, $C = 0.125$. A is the rocket cross-sectional area; the rocket has 1.70 m in diameter. The last force, F_{thrust}, is the one that propels the rocket upward. This force is given by

$$F_{thrust} = v_e \frac{dm}{dt} \tag{5}$$

with v_e being the ejection velocity given by 2000 m/s and dm/dt is the change in mass per unit of time as the rocket uses fuel as given above. This force is zero once the rocket consumed all its fuel.

Use Euler numerical integration to solve the equations of motion for the velocity and height of the rocket. Consider four stages of the rocket motion: (i) launching from the ground, (ii) ascending with its fuel and total mass M reducing with time, (iii) motion after all its fuel is burned and its mass does not change anymore, and finally (iv) descending until it hits the ground. **I would like to see two figures, please, one of the rocket velocity versus time and another of the height versus time for all stages of the rocket motion. From the two plots, determine the maximum velocity of the rocket, its maximum height, and how long it takes for the rocket to reach terminal velocity once it is falling down.**

And one last analysis, please: if I launch the same rocket but in Mars, how will the maximum velocity and the maximum height of the rocket be affected? Consider only the value for the highest atmospheric density on Mars which is 0.020 kg/m^3 and the average gravitational acceleration on Mars being $g_m = 3.72\, m/s^2$.

The initial masses M and M_r need to be generated with the Jupyter Notebook provided in the D2L midterm module. Run that code as instructed and get the masses for your simulation.

PS: Simulate all stages of the rocket trajectory from its launching, consuming fuel until its last drop, and eventually falling down until it reaches the ground. If the correct physics model is implemented, all stages will appear naturally in the simulation.

*****