53 + 50 = 103

$\boxed{103/117}$

Phys 304: Assignment 1

Mary Smith*
Haverford College Department of Physics
(Dated: February 8, 2024)

[Survey question: This homework took me 2.5 hours to complete. I learned how to plan my work ahead of time with a flowchart, build equations, then use those equations in a physical context. The problem set had a just right length, considering the high content of information.]

1. PROBLEM 1: A SATELLITE IN ORBIT

A satellite makes a complete rotation around the Earth with period T.

1.1. Part a: Deriving the altitude h

We know that the gravitational force F_G is given by:

$$F_G = \frac{GmM}{r^2}, \tag{1}$$

where the gravitational constant $G = 6.67 * 10^{-11} m^3 kg^{-1} s^{-2}$, m is the mass of the satellite, M is the mass of the Earth, and r is the radius of the satellite's orbit from the center of the Earth. We also know that the centripetal force is given by

great description!

$$F_C = \frac{mv^2}{r}, \tag{2}$$

where m is once again the mass of the satellite, v is the velocity of the satellite, and r is again the radius of the satellite's orbit from the center of the Earth.

When the satellite is in orbit, the gravitational force must equal the centripetal force. Otherwise, the satellite will either crash into Earth or fly off into space. Setting $F_G = F_C$, we can plug in the value of v to reduce it further. Plugging in $v = \frac{2\pi r}{T}$, where r is the same as above and T is the period of the orbit, we get:

$$\begin{aligned} F_G &= F_C \\ \frac{GmM}{r^2} &= \frac{mv^2}{r} \\ \frac{GM}{r} &= (\frac{2\pi r}{T})^2 \\ r^3 &= \frac{GMT^2}{4\pi^2} \\ r &= (\frac{GMT^2}{4\pi^2})^{1/3}. \end{aligned} \tag{3}$$

Remembering that r represents the orbit of the satellite from the center of the Earth, it can also be expressed as $r = R + h$, where R is the radius of the Earth and h is the altitude of the satellite. This means we can express Eq. 3 as:

$$h = (\frac{GMT^2}{4\pi^2})^{1/3} - R. \tag{4}$$

1.2. Part b: Code for desired T to h

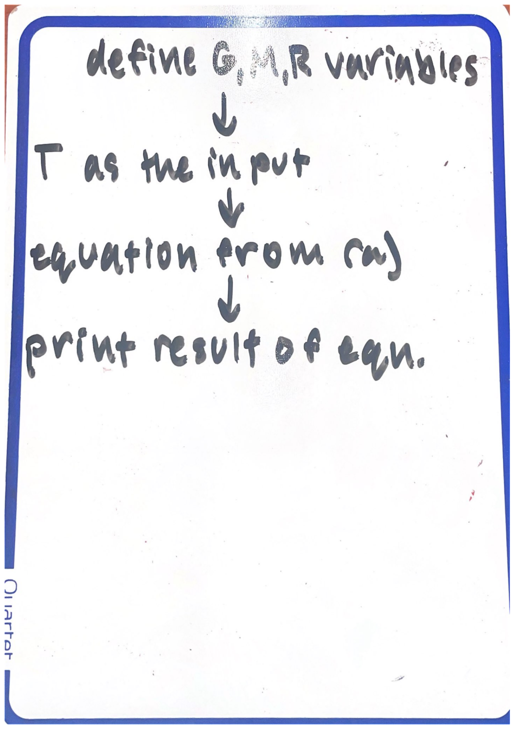

FIG. 1: [Flowchart of the code for Problem 1b. The code prints the altitude of the satellite in meters for a given period in seconds.]

1.3. Part c: Calculating orbits

what values for G, M, and R did you use?

Using the code created above, a satellite that orbits the Earth once a day does so at an altitude of $h = 35855910.18m$. A satellite that orbits the Earth every 90 minutes does so at an altitude of $h = 279321.63m$.

---

*Electronic address: masmith@haverford.edu

survey questions?

Lastly, according to the code I developed, a satellite that orbits the Earth every 45 minutes does so at an altitude of $h = -2181559.9m$. This negative number is likely due to the fact that it is impossible for a satellite to make a complete rotation of the Earth in that small of a period.

1.4. Part d: The sidereal day

A true geosynchronous satellite orbits the Earth once every sidereal day, or 23.93 hours. This is because a sidereal day is the amount of time it takes to make a full rotation about its axis with respect to far away stars, versus with respect to the Sun. Therefore, a satellite rotating with a period of a sidereal day will have a more precisely geosynchronous orbit than if the satellite with a period of a solar day. A satellite with a period of a sidereal day will orbit at $h = 35773762.33m$. This means there is a difference of $h = 82147.85m$ between a sidereal orbit and a solar orbit, which presents quite a large difference.

2. PROBLEM 2: CATALAN NUMBERS

The Catalan numbers C_n can be found using recursion in the following code.

include the equation for C_n along with a description

All final answers should be in your write-up, such as C_{100}

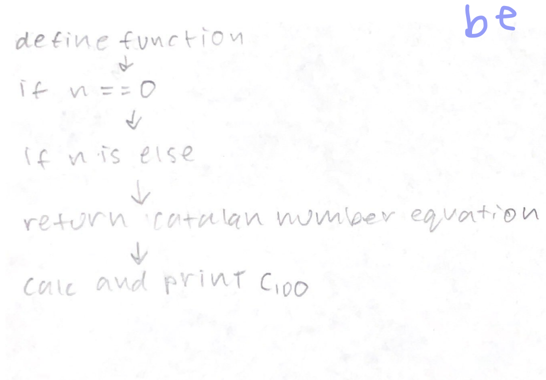

FIG. 2: [Flowchart of the code for Problem 2. The code finds the function for Catalan numbers C_n and prints C_100.]

Ex. 2.2 53/56

Computational Physics/Astrophysics, Winter 2024:

Grading Rubrics [1]

Haverford College, Prof. Daniel Grin

For coding assignments, roughly 56 points will be available per problem. Partial credit available on all non-1 items.

4 1. Does the program complete without crashing in a reasonable time frame? (+4 points)

1 2. Does the program use the exact program files given (if given), and produce an answer in the specified format? (+2 points) answers to (c) and (d) should print out without needing an input -1

3 3. Does the code follow the problem specifications (i.e numerical method; output requested etc.) (+3 points)

5 4. Is the algorithm appropriate for the problem? If a specific algorithm was requested in the prompt, was it used? (+5 points)

4 5. If relevant, were proper parameters/choices made for a numerically converged answer? (+4 points)

4 6. Is the output answer correct? (+4 points).

3 7. Is the code readable? (+3 points)

. 5.1. Are variables named reasonably?

. 5.2. Are the user-functions and imports used?

[1] Inspired by rubric of D. Narayanan, U. Florida, and C. Cooksey, U. Hawaii

. 5.3. Are units explained (if necessary)?

. 5.4. Are algorithms found on the internet/book/etc. properly attributed?

2 8. Is the code well documented? (+3 points)

. 6.1. Is the code author named?

Please comment your name at the top of your code -1

. 6.2. Are the functions described and ambiguous variables defined?

. 6.3. Is the code functionality (i.e. can I run it easily enough?) documented?

9. Write-up (up to 28 points)

- 5 . Is the problem-solving approach clearly indicated through a flow-chart, pseudo-code, or other appropriate schematic? (+5 points)
- ✓ . Is a clear, legible LaTeX type-set write up handed in?
- 2 . Are key figures and numbers from the problem given? (+ 3 points)

  make sure to define all variables in the problem -1
- 4 . Do figures and or tables have captions/legends/units clearly indicated. (+ 4 points)
- 3 . Do figures have a sufficient number of points to infer the claimed/desired trends? (+ 3 points)
- 2 . Is a brief explanation of physical context given? (+2 points)
- 1 . If relevant, are helpful analytic scalings or known solutions given? (+1 point)
- 3 . Is the algorithm used explicitly stated and justified? (+3 points)
- 2 . When relevant, are numerical errors/convergence justified/shown/explained? (+2 points)

2 . Are 3-4 key equations listed (preferably the ones solved in the programming assignment) and algorithms named? (+2 points)
1 . Are collaborators clearly acknowledged? (+1 point)
2 . Are any outside references appropriately cited? (+2 point)

Ex. 2.13 50/56

Computational Physics/Astrophysics, Winter 2024:

Grading Rubrics [1]

Haverford College, Prof. Daniel Grin

For coding assignments, roughly 56 points will be available per problem. Partial credit available on all non-1 items.

4 1. Does the program complete without crashing in a reasonable time frame? (+4 points)

1 2. Does the program use the exact program files given (if given), and produce an answer in the specified format? (+2 points) All outputs should include a description, such as "C_{100} is ..." −1

3 3. Does the code follow the problem specifications (i.e numerical method; output requested etc.) (+3 points)

5 4. Is the algorithm appropriate for the problem? If a specific algorithm was requested in the prompt, was it used? (+5 points)

4 5. If relevant, were proper parameters/choices made for a numerically converged answer? (+4 points)

4 6. Is the output answer correct? (+4 points).

3 7. Is the code readable? (+3 points)

. 5.1. Are variables named reasonably?

. 5.2. Are the user-functions and imports used?

[1] Inspired by rubric of D. Narayanan, U. Florida, and C. Cooksey, U. Hawaii

. 5.3. Are units explained (if necessary)?

. 5.4. Are algorithms found on the internet/book/etc. properly attributed?

2 8. Is the code well documented? (+3 points)

. 6.1. Is the code author named? please comment your name at the top of your code -1

. 6.2. Are the functions described and ambiguous variables defined?

. 6.3. Is the code functionality (i.e. can I run it easily enough?) documented?

9. Write-up (up to 28 points)

5 . Is the problem-solving approach clearly indicated through a flow-chart, pseudo-code, or other appropriate schematic? (+5 points)

✓ . Is a clear, legible LaTeX type-set write up handed in?

2 . Are key figures and numbers from the problem given? (+ 3 points) Final answer should be written -1

4 . Do figures and or tables have captions/legends/units clearly indicated. (+ 4 points)

3 . Do figures have a sufficient number of points to infer the claimed/desired trends? (+ 3 points)

. Is a brief explanation of physical context given? (+2 points)

1 . If relevant, are helpful analytic scalings or known solutions given? (+1 point)

2 . Is the algorithm used explicitly stated and justified? (+3 points) algorithm (recursion) is stated but not justified -1

2 . When relevant, are numerical errors/convergence justified/shown/explained? (+2 points)

0 . Are 3-4 key equations listed (preferably the ones solved in the programming assignment) and algorithms named? (+2 points)

All equations in the problem shoud be given -2

1 . Are collaborators clearly acknowledged? (+1 point)

2 . Are any outside references appropriately cited? (+2 point)