AST101: Our Place in the Universe  
Lab 5: Simulating Orbits (II)  
Version for Online Or In-Person

*(If you are using Google Docs, you should choose to “Make a Copy” of this document. That way, you can edit it. Your whole group should edit it together. If one member of your group is not present, list their name and write “absent” beside it.)*

**Submission Instructions:** When you are done with your lab, make sure your document is still set to the sharing option “Anyone with the link can view”. Then email a link to your shared document to suast101labs@gmail.com. The subject line in your email should be “Lab 5 – Group #### – <your names>”.

|  |  |  |  |
| --- | --- | --- | --- |
| **Group Number and Name:** | |  | |
| **Member Name #1:** |  | **Email #1:** |  |
| **Member Name #2:** |  | **Email #2:** |  |
| **Member Name #3:** |  | **Email #3:** |  |
| **Member Name #4:** |  | **Email #4:** |  |
| **Lab Time and Date:** |  | | |
| **Collaboration Method (in-person, online on Collaborate, etc.)** |  | | |

*If someone in your group does not show up and did not tell you why, write “unexpected absence” by their name. If someone does not show up and has a good reason, write “expected absence” by their name, and describe what happened below.*

|  |
| --- |
|  |

**Overview:**

Last week, during our “recess” from normal labs, you got familiar with an orbit simulator written in Python, and explored its basic properties.

Now you’re going to use it to explore Kepler’s laws in depth. You do not need to know anything about computer programming beforehand. However, one step in this lab requires you to look at some computer code and think about what it does. We’re not going to require you to write code, but we want to give you a little exposure to what it looks like. Python is a language used by many scientists and web developers (and other people) for all kinds of things; it is one of the most popular programming languages today.

**Materials:** You will need a computer or tablet. Any computer or tablet will work, since this is a web-based simulation. You could use a cellphone, too, but the small screen and lack of keyboard will likely be annoying.  
  
  
  
**Kepler’s laws of orbits**

Kepler realized that if planets moved in *elliptical* orbits rather than circular ones, their movements fit the observations very well. He formulated three laws of orbits that describe the motion of the planets. Paraphrased, they are:

1. Planets move in elliptical orbits with the Sun at one focus
2. The line connecting a planet to the sun sweeps out equal area in equal time, so planets move faster when they are closer to the Sun, and slower when they are further away
3. The time it takes a planet to orbit the Sun depends only on the length of its semimajor axis. In particular, if the semimajor axis is multiplied by a factor A, the period P is multiplied by

**Newton’s laws of gravity and motion**

Newton realized several things about moving objects:

1. Forces cause objects to accelerate by an amount a = F/m in the direction of those forces.
2. Gravity is one such force. All objects attract each other with a force equal to

where r is the distance between the two objects, and and are their masses.

**The Orbit Simulator:**This program works in the following way:

* Choose the mass of your planet, as well as the mass of the star it orbits.
* Choose a starting position for the planet, as well as its velocity.
* Repeat the following steps many times:
  + Use Newton's law of gravity to find the size and direction of the gravitational force
  + Use Newton's law of motion to determine how this force changes the objects' motion over a tiny amount of time, and then change the motion
  + Along the way, draw some stuff so you can see how the planet moves

You won’t need to write your own code here – only modify my code to see what happens, and then look at some parts of it to see how it works. You can’t break anything; if something goes wrong, you can just click the three-lines menu in the top left and choose “Reset”.

If you press “Run” and your code gives you an error (a big red bar at the bottom), either hit Ctrl-Z to undo your last few changes and then try again, or refresh your browser. Your GTA might be able to assist you as well (most of the graduate students will know some Python).

Computers, at heart, do math on variables. Look at line 5: this assigns the value 1.017 to the variable start\_distance. A computer program is just a list of math instructions in order, along with other instructions that do things like draw on the screen and repeat some segments of math. It is important that you don’t change the names of any of the variables here: if you rename start\_distance to begin\_distance, even though those mean the same thing in English, the computer won’t know what is going on later.

It is also important that you don’t add extra spaces before the beginning of any lines, or remove any that are there. Spaces before a line mean something special in Python, and if you remove them, the program will mean something different than it did before (and probably won’t work). Also, notice the text that’s displayed in green and follows a # sign. These are comments – the computer ignores them. They’re just in the code to give you “signposts” to describe what parts of the code do.

**Describing Orbits:**

A reminder from the previous lab about what some technical words mean:

* Aphelion: The distance from a planet to its star at the furthest point in its orbit
* Perihelion: The distance from a planet to its star at the closest point in its orbit
* Period: How long it takes the planet to go around its star
* Eccentricity: A measure of how ``stretched-out'' an ellipse is. The minimum value is 0 (this is just a circle); the maximum value is 1.  
    
  Eccentricity can be calculated from perihelion and aphelion distances as   
    
    
    
  Note that the computer will do this for you; you won’t need to do any mathematics yourself.

**Exploring Kepler’s laws**

Go to the orbit simulator and set it to fullscreen. Then run the simulator. You’ve played with it before for Lab 4, but now we’re going to make sure that our computer program’s orbits follow Kepler’s laws.

**Kepler’s first law** says that planets orbit the Sun in elliptical orbits with the Sun at a focus. Does

the orbit that you see follow Kepler’s first law? How do you know?

|  |
| --- |
|  |

Now, try different starting velocities for the planet, and sketch the planet’s orbit around its star on a piece of paper. Notice that the starting point (1.02 AU) will always either be perihelion or aphelion. For each of the following, describe:

* How close, or how far, the planet gets to the Sun on the other side (the computer will tell you this; it’s the other perihelion/aphelion point)
* The eccentricity of the planet’s orbit
* Whether anything weird happened that wasn’t at all what you expected

If your planet moves too slowly or too quickly, change the value of seconds\_per\_year that controls how many seconds in the simulation corresponds to one year of real time.

Try starting velocities of

1. 2.5 AU/year
2. 5 AU/year
3. The value you found during your prelab that matches Earth’s actual orbit
4. 8 AU/year
5. 10 AU/year

Record what you see below.

|  |  |  |  |
| --- | --- | --- | --- |
| **Starting Velocity** | **Perihelion/Aphelion on Other Side** | **Eccentricity** | **Other Observations?** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

What patterns do you notice here? Do they make physical sense? (Imagine “throwing” the Earth faster and faster in its orbit.)

|  |
| --- |
|  |

What happened when you gave the Earth a large starting velocity of 10 AU/year? Is this what you expected? Do you think that the Earth would really move in this way if its velocity were suddenly increased to 10 AU/year?

|  |
| --- |
|  |

Remember that the focus of an ellipse is close to the center of the long axis if the ellipse is not very eccentric, but close to the edge if it is strongly eccentric. Look back at your drawings for the previous question. Is this true in the orbits you observed?

|  |
| --- |
|  |

Now, let’s look at **Kepler’s second law**, which compares the speed of a planet at different places in its orbit. By changing the starting distance from the Sun and the starting velocity, put your planet into an orbit with eccentricity at least 0.6.  
  
Does your planet move more quickly near perihelion or aphelion? In evaluating this, look at both the animation and the graph to make sure that they agree. How does your graph show this?

|  |
| --- |
|  |

Does your planet spend more time near perihelion or aphelion?

|  |
| --- |
|  |

Does your planet appear to follow Kepler’s second law? How do you know?

|  |
| --- |
|  |

Now, let’s look at **Kepler’s third law**, which compares the time that different planets take to go around

the Sun.

Kepler’s original formulation of the third law says that . This means that if you calculate this fraction for any planet orbiting the Sun, it should be the same.

How can you determine the length of the long axis if you know the perihelion and aphelion distances? (A diagram on scratch paper may help you figure this out!)

|  |
| --- |
|  |

Simulate two different orbits with different starting distances and velocities, and fill out the table below. (Remember, if your planet flies off, your starting velocity is too high!) You can use Google Calculator to square and cube the numbers you get.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Aphelion | Perihelion | Long axis | Orbital period |  |
|  |  |  |  |  |
|  |  |  |  |  |

Do your two orbits appear to follow Kepler’s third law?

**Does the Mass Matter?**

The mass of the planet is set to the mass of Earth by default: 0.000003 times the mass of the Sun. Change the mass of the planet to the mass of Jupiter: 0.001 times the mass of the Sun.

Did the planet’s orbit change? Describe how it changed, or explain why it didn’t.

|  |
| --- |
|  |

Now, change the *star’s* mass. Does this affect the orbit? Why or why not?

|  |
| --- |
|  |

Kepler’s description of planet orbits – where the Sun doesn’t move and the planet orbits around it in an ellipse – are based only on the motions of the planets in our solar system. They are true only if:

* A much less massive object orbits a much more massive one (like Earth around the Sun)
* The gravity of the larger object being orbited is the only force that causes much of an effect on the smaller object.

Earlier, you saw that changing the mass of the planet orbiting the Sun doesn’t affect its orbit. But what happens if you make it very massive – almost as massive as the Sun itself?

Increase the mass of the planet to at least ten percent of the star’s mass. What happens now? Is this what you expect?

|  |
| --- |
|  |

Is this behavior something that matches Kepler’s description of planetary orbits? If it’s not, which of the assumptions above are no longer true?

|  |
| --- |
|  |

Suppose that you looked in space and saw a star wiggling back and forth. We can see stars with telescopes, but can’t see planets since they don’t make their own light. If you saw a star wiggling back and forth, what could you conclude about it?[[1]](#footnote-0)

|  |
| --- |
|  |

We know that many stars are binary stars – there are two stars that orbit each other closely. Since we know that gravity affects stars in the same way as it affects planets, we can use this program to simulate binary stars too.

Set the mass of the “planet” to nearly the same as the mass of the “star”: both of them will now represent stars. See if you can create an orbit where the two stars orbit each other. Show your orbit to your TA when you’ve got it.

**How’s This All Work?**

This program may look complicated. But most of the code has to do with making the animation and table you see! The only code that actually performs the simulation – where the actual physics is – is between line 50 and 65. Even if you’ve never taken physics before or written a computer program, it’s actually not that hard to look at the code that controls this simulator and see what is happening!

All of the laws of nature that control the motion of orbits are the following:

* Forces cause things to accelerate, where the acceleration is equal to the force divided by the mass of that thing *(Newton’s second law of motion)*
* Any two objects attract each other with a force equal to where the force points toward the other object *(The law of gravity)*
* Forces come in pairs: if the star’s gravity exerts a force on the planet, the planet’s gravity exerts a force back on the star that goes in the opposite direction and has the same size *(Newton’s third law of motion)*
* In a small interval of time, an object’s position changes by an amount equal to its velocity multiplied by the amount of time. (You know this already: if you are traveling at 50 miles per hour, and you do this for two hours, you will go 100 miles...) *(Definition of velocity)*
* In a small interval of time, an object’s velocity changes by an amount equal to its acceleration multiplied by the amount of time. *(Definition of acceleration)*

The computer simulation in this program works as follows. First, we divide time up into very tiny intervals (the length is given on line 40, if you want to change it).

Then, repeat the following:

1. Determine the distance from the planet to the star
2. Determine the direction from the planet to the star
3. Using Newton’s law of gravity and the distance and direction you just found, determine the force the two objects feel from each other
4. Using Newton’s second law of motion, determine the acceleration that this causes on each object
5. Using the definition of velocity, change the positions of the star and the planet by the amount that they will move during the small interval
6. Using the definition of acceleration, change the velocities of the star and the planet by the amount that they will change during the small interval

This is all it does!

Look at the code from lines 50-65. Look at the code and try to identify what each group of lines does. Which lines do which tasks?

|  |  |
| --- | --- |
| **Task** | **Line number(s)** |
| Determine the distance from the planet to the star |  |
| Determine the direction from the planet to the star |  |
| Using Newton’s law of gravity and the distance and direction you just found, determine the force the two objects feel from each other |  |
| Using Newton’s second law of motion, determine the acceleration that this causes on each object |  |
| Using the definition of velocity, change the positions of the star and the planet by the amount that they will move during the small interval |  |
| Using the definition of acceleration, change the velocities of the star and the planet by the amount that they will change during the small interval |  |

As you saw earlier, this simulation has all of the features that Kepler observed in the orbits of the actual planets. However, Kepler’s laws don’t appear anywhere in this code; there are no mentions of ellipses, long and short axes, moving faster near perihelion, or anything like that! (You can check this in the code!)

You’ve already seen how this program works. Why do you think it correctly simulates the features of the planets’ orbits, even though it doesn’t have any of the substance of Kepler’s laws inside it?

|  |
| --- |
|  |

**Submission Instructions:** When you are done with your lab, make sure your document is still set to the sharing option “Anyone with the link can view”. Then email a link to your shared document to suast101labs@gmail.com. The subject line in your email should be “Lab 5 – Group #### – <your names>”.

1. We actually have an extraordinarily sensitive way to measure very small motions of stars -- we can detect a star’s velocity changing by ten meters per second from light-years away. This is done using the Doppler effect combined with detailed measurements of its color -- stay tuned for our third part of the class! [↑](#footnote-ref-0)