



A diagram of the Main Asteroid Belt, showing a dense cloud of grey asteroids between the orbits of Jupiter and Saturn. The Sun is at the center, with the orbits of Mercury, Venus, Earth, and Mars shown as concentric circles. Jupiter and Saturn are shown as large orange and yellow spheres on their respective orbits. The text "MAIN ASTEROID BELT" is written in white capital letters along the top edge of the asteroid cloud. The title "Recreating the Kirkwood Gaps" is written in large white letters across the center of the diagram.

Recreating the Kirkwood Gaps

By Cassandra Bodin

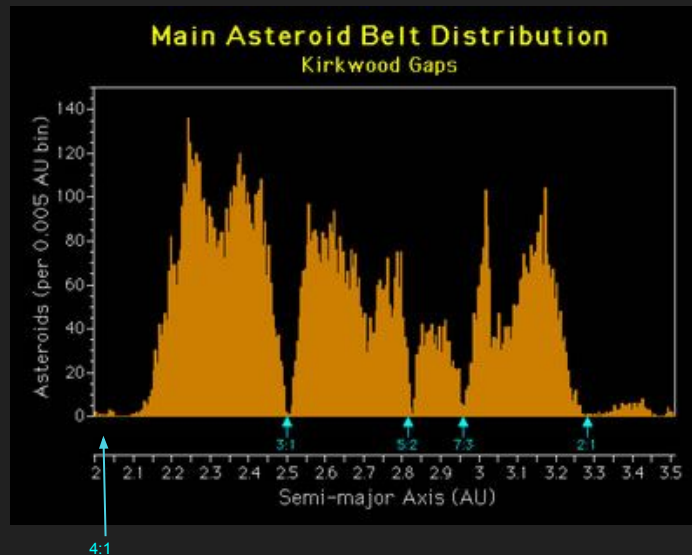
Goal of Project

Simulate the Asteroid Belt, Jupiter and the Sun

- Input n number of asteroids in a range of semimajor axis
- Gather data for every run

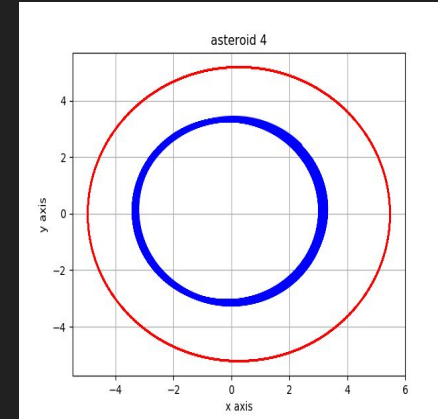
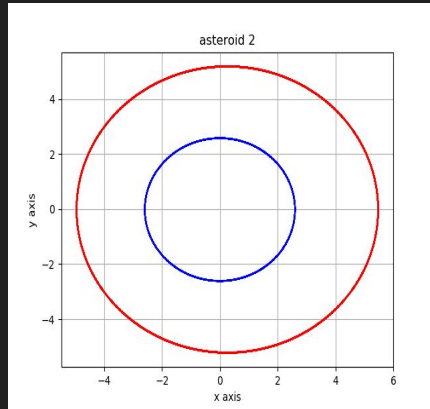
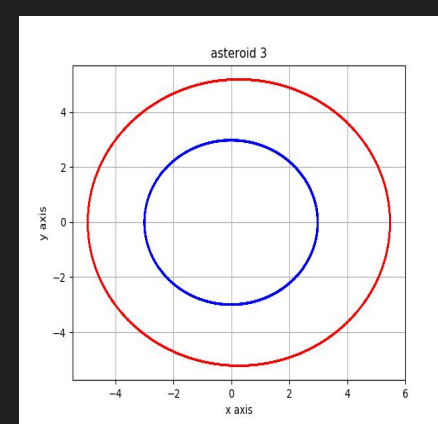
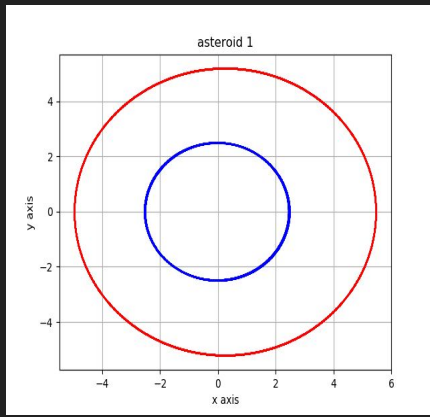
Create a histogram of the data

- Recreate Kirkwood Gaps
 - Little to no asteroids found
 - Resonance \leftarrow Jupiter's gravity clears the orbit



Where did I start?

- 3 body problem: Jupiter, the Sun, and an Asteroid
 - Homework 6: asteroid problem
 - Euler Cromer Method



Modifications and Problems to Face

1. How to input an n number of asteroids and keeping track of the parameters for each asteroid throughout the code

```
for i in range(0,1000):  
    number= 1.5*np.random.random()+2.0
```

```
for i in range(n.size):
```

2. Finding which orbits were stable and finding the semimajor axis of those orbits.
3. Finding a way to get multiple runs of data into one place in order to make a better histogram.

Hardest Solutions- Problem 2 & 3:

3

1

```
r_initial = np.sqrt((xarr[:,0])**2 + (yarr[:,0])**2)
r_final = np.sqrt((xarr[:,1])**2 + (yarr[:,1])**2)
f = open("asteroid_datafile.txt","a")
for i in range(n.size):
    if abs(r_final[i] - r_initial[i]) < 1e-1:
        #print(semimajor_a[i])
        f.write(str(semimajor_a[i]) + "\n")
f.close()
```

4

2

Choosing values to keep:

1. Find initial and final radii
2. Difference is less than a certain percent = stable

Allow multiple runs of program:

3. Create/open txt file and append data to file
4. Write in file the values to keep

Running the program

f' string prints cycle number every 200 cycles

Super cool!!!



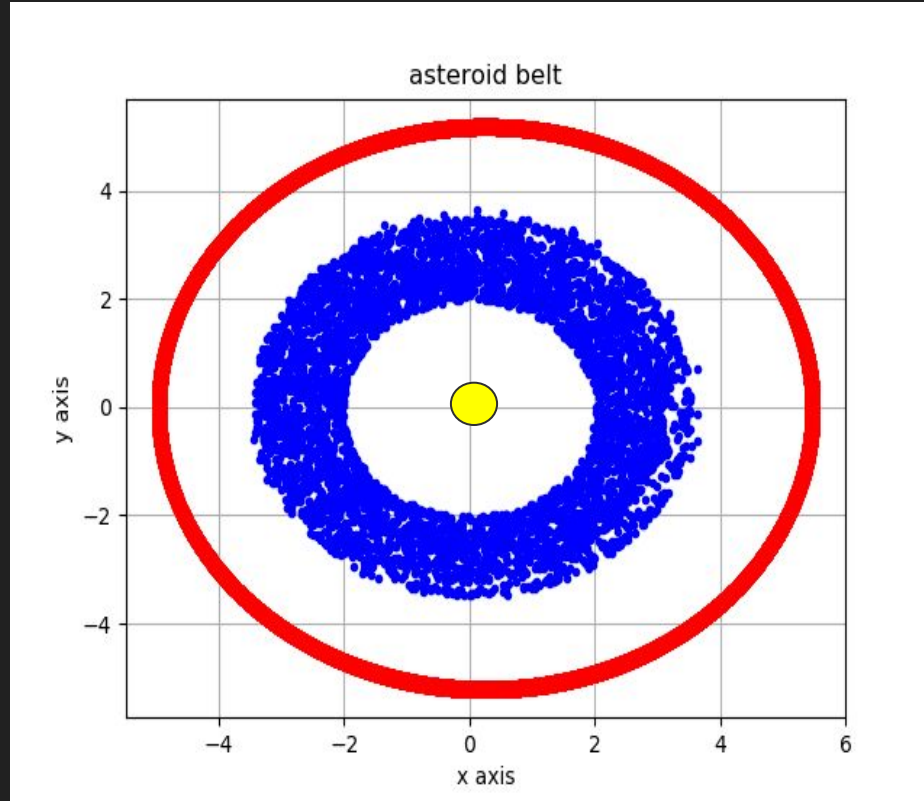
```
if (count %200 ==0):  
    print(f'{count}')
```

Run time:

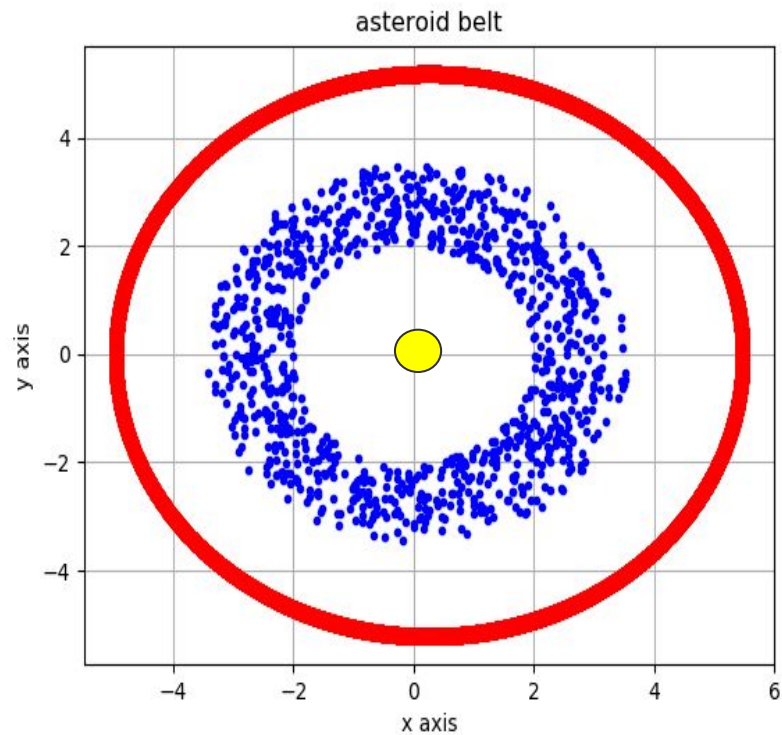
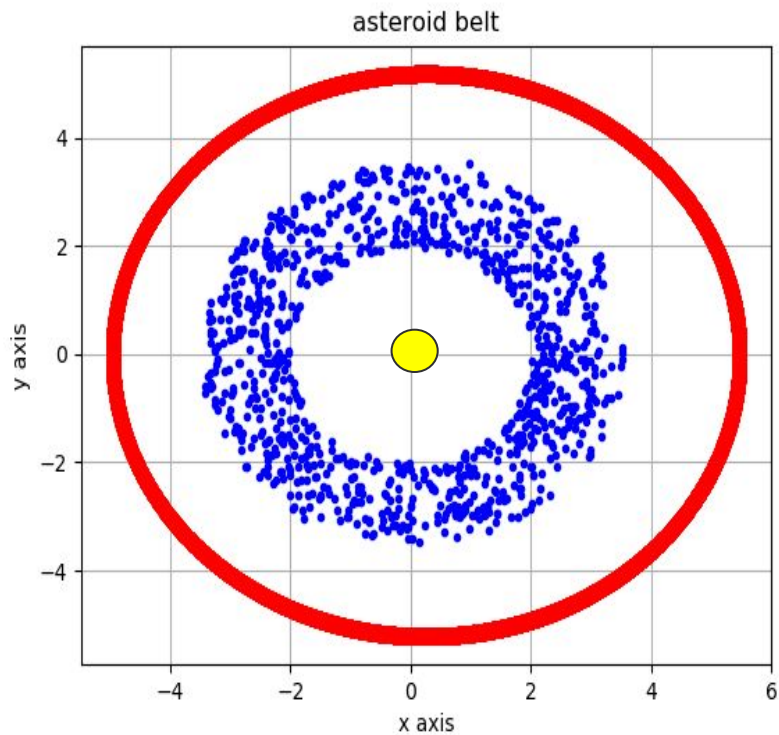
- 5000 asteroids: 7+hrs
- 1000 asteroids: 1hr 17min 32sec
- 500 asteroids: 20min 36sec
- 100 asteroids: 5min 9sec
- 50 asteroids: 5min 3sec

What did the plots of the asteroids look like →

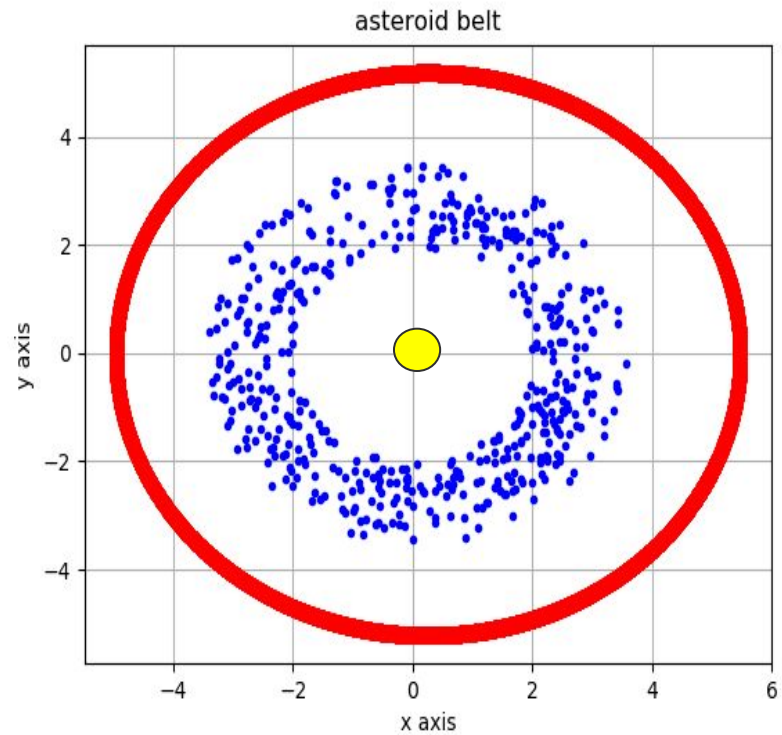
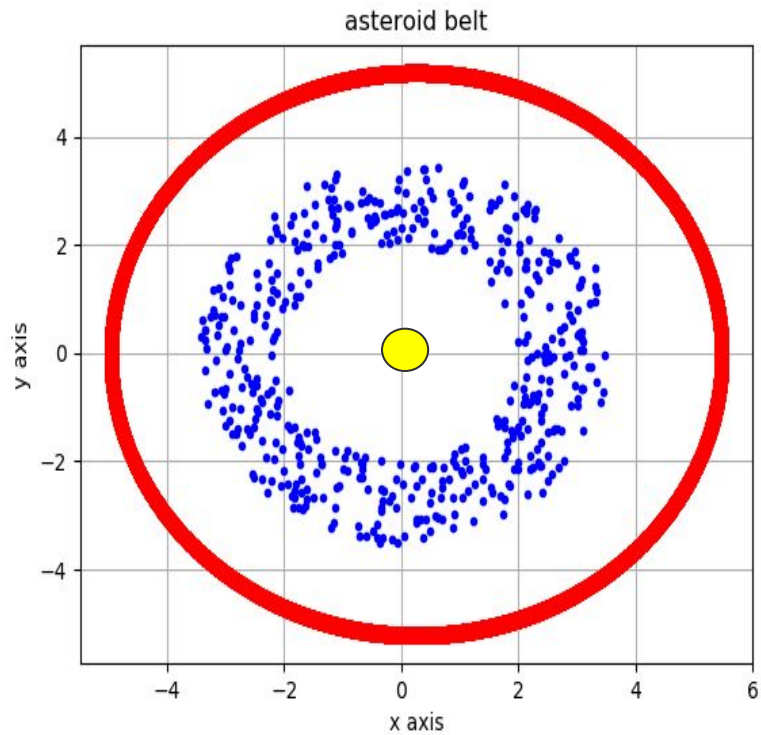
5000 Asteroids



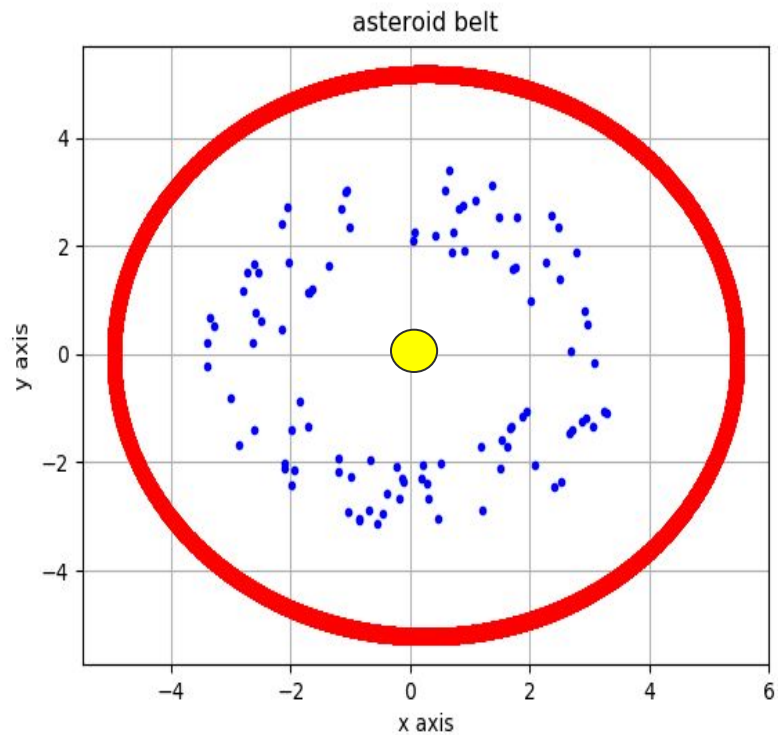
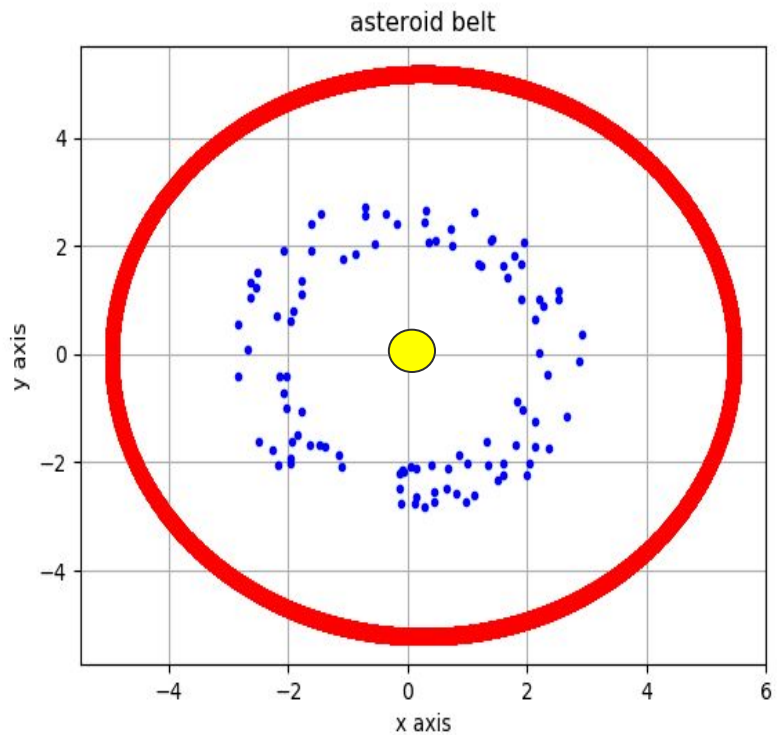
1000 Asteroids



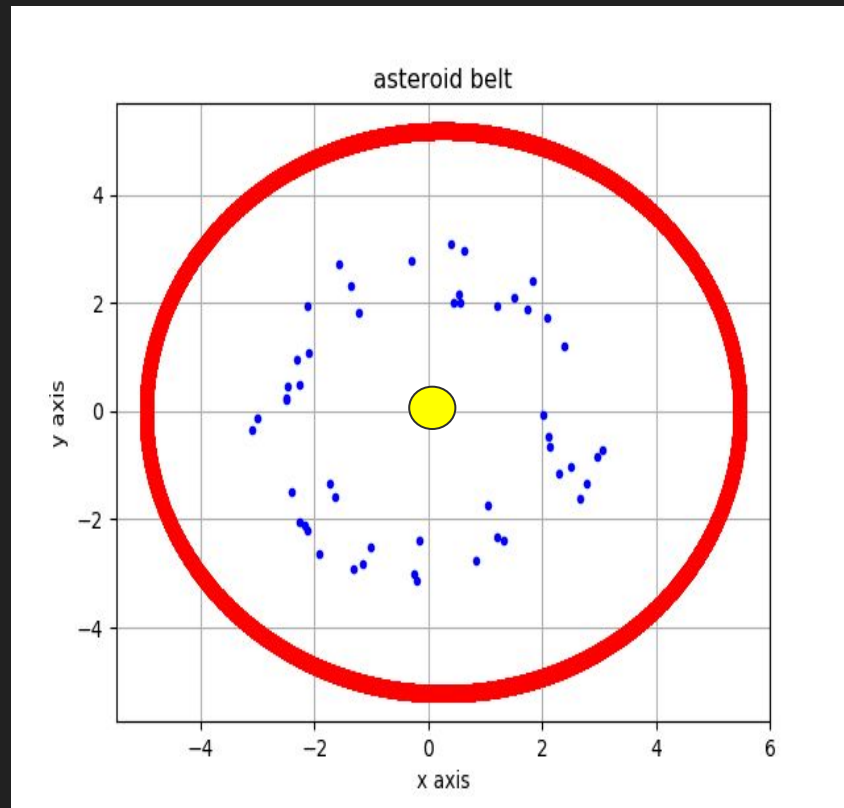
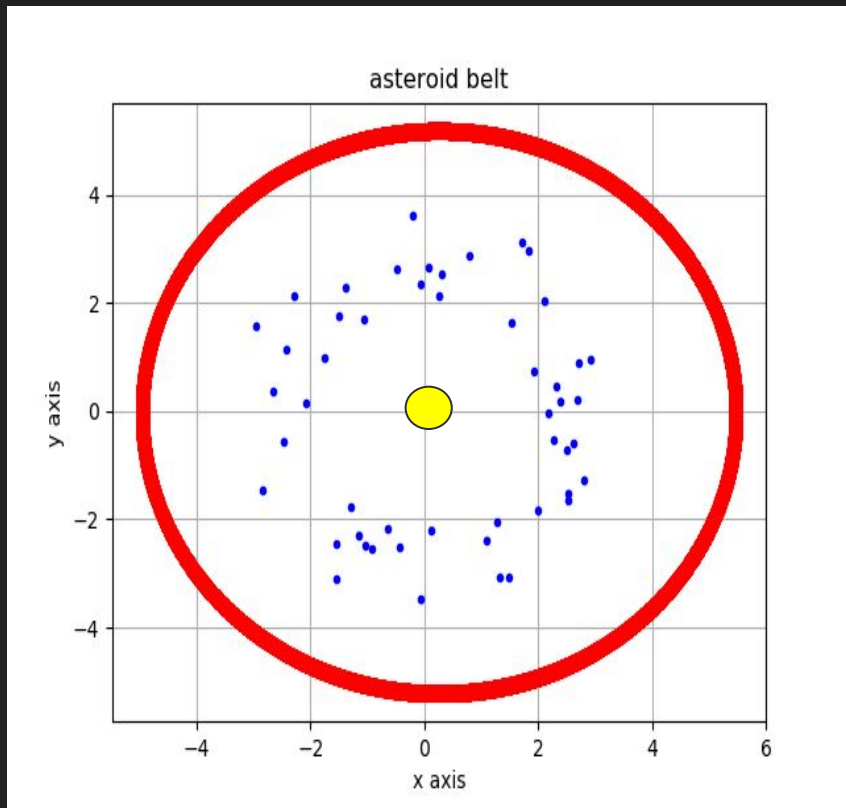
500 Asteroids



100 Asteroids

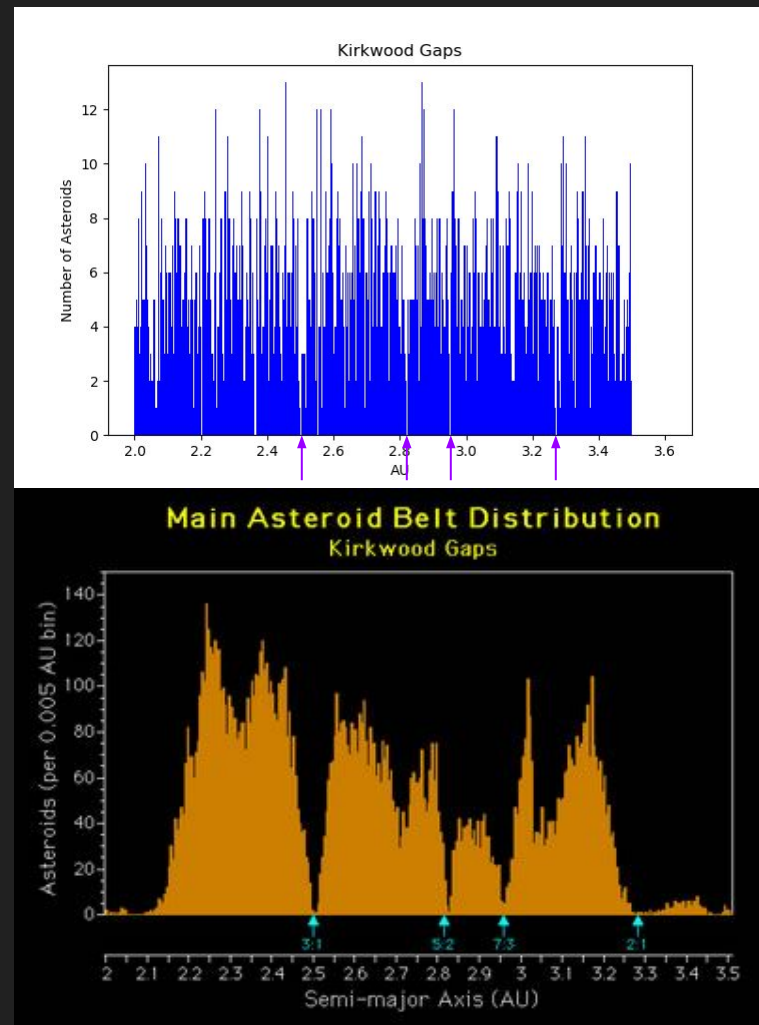


50 Asteroids



Results/Comparison:

- Data File contains 2636 asteroids
- Gaps in the right spots!!!
 - 2.5 AU, 2.82 AU, 2.95AU, 3.27 AU
- My histogram
 - Extra gaps at: 2.2, 2.36, 2.55
 - Does not show resonances



Conclusion:

How accurate was it?

Was there error? What caused it?

- Not as many asteroids
- Used randomly distributed asteroids, not real ones
- Radii deleted from set, deemed unstable

Ideas for future:

- Using real data
- Resetting parameters to sort which are stable or unstable
- Run program more



Questions?



Bibliography

Title page image- <https://space-facts.com/asteroid-belt/> created by Laurine Moreau

Histogram comparison image- <http://astronomy.swin.edu.au/cosmos/K/Kirkwood+Gaps> created by NASA