ASTR 1040 RECITATION 9

10/30/2023

KIRK LONG CU BOULDER

HOUSEKEEPING

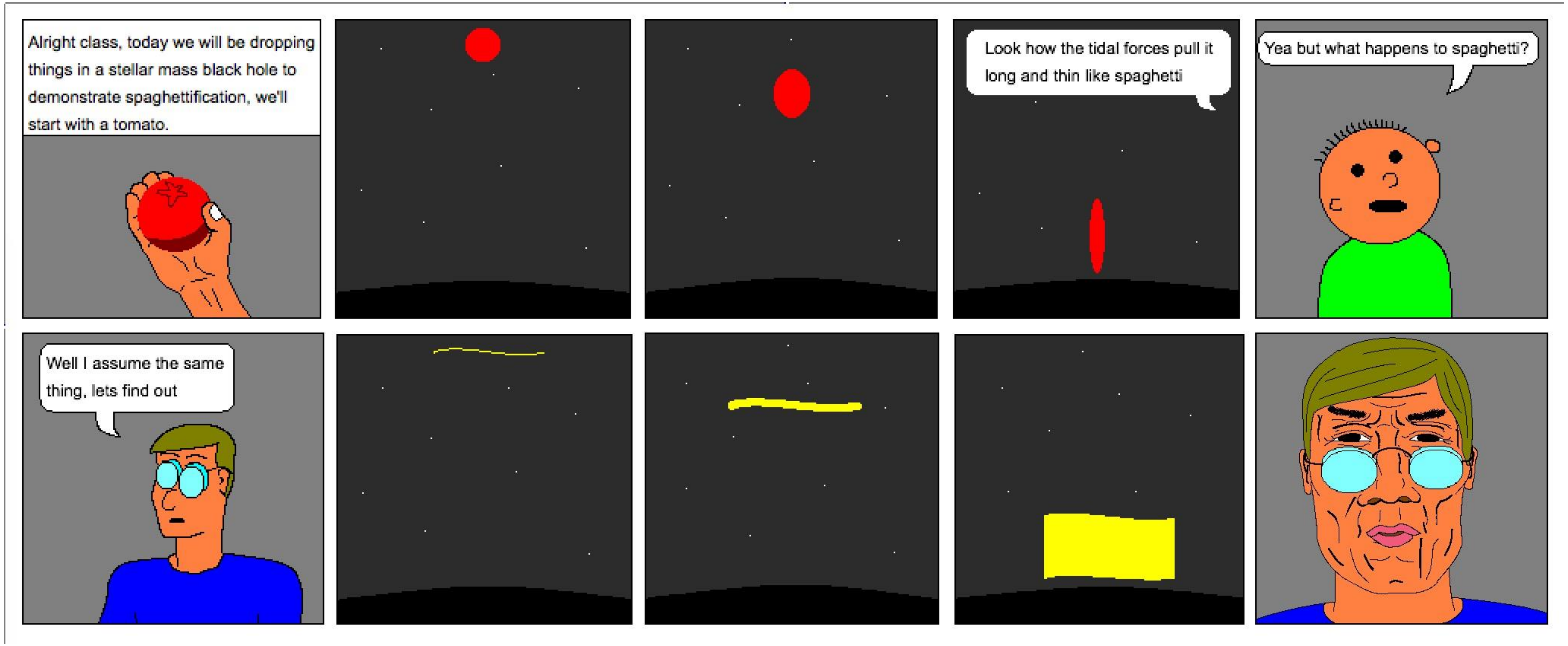
a) Homeworks 3 and 4 graded – solutions are posted, reach out with any questions

b) Midterm 2 will (still) be 11/14, review will be next week in recitation

c) Only two more homeworks for the entire semester:

   1. Homework 8 due next Thursday (11/9)
   2. No homework week of 11/13 (exam), no class 11/16 (time to think about / work on science communication project – if you have an alternative idea propose it to us then!)
   3. No homework week of 11/30 (week to finish science communication projects)
   4. Homework 9 (last homework) due 12/7

SCIENCE COMMUNICATION PROJECT EXAMPLES


Alright class, today we will be dropping things in a stellar mass black hole to demonstrate spaghettification, we'll start with a tomato.
Look how the tidal forces pull it long and thin like spaghetti
Yea but what happens to spaghetti?
Well I assume the same thing, lets find out

SPECIAL VS GENERAL RELATIVITY

Special relativistic time dilation: $\Delta t' = \gamma \Delta t$

General relativistic time dilation (Schwarzschild metric):

a) Stationary observer: $t_o = t_\infty \sqrt{1 - \frac{2GM}{rc^2}}$

b) Orbiting (circular) observer: $t_o = t_\infty \sqrt{1 - \frac{3}{2}\frac{R_s}{r}}\,(r > R_s)$

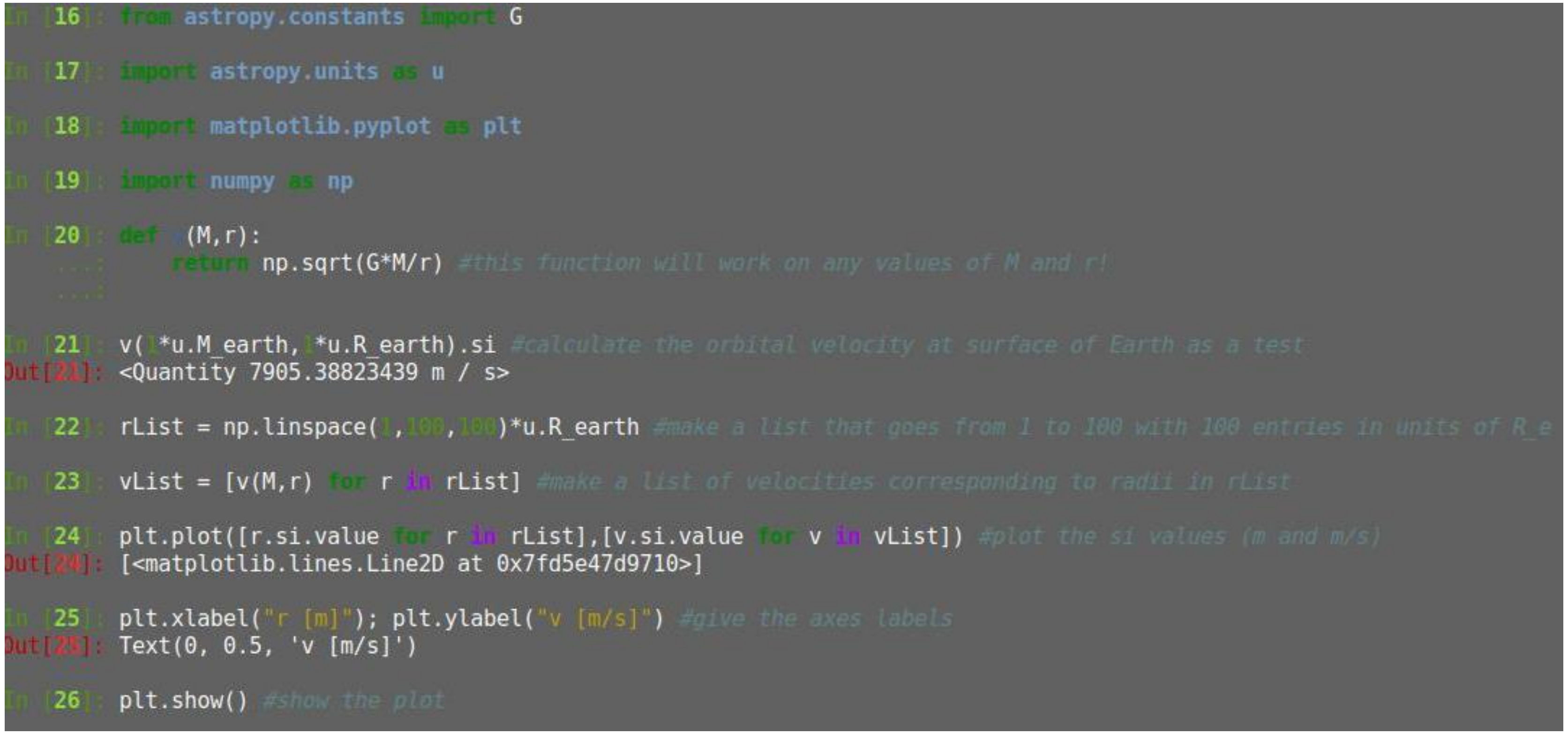
Practice problem: GPS satellites orbit at an altitude of roughly 20,000 km.

a) Calculate the velocity of a circular orbit for such a satellite

b) Compare the magnitudes of the gravitational and special relativistic time dilation.

c) What is the total accumulated difference in time for a GPS satellite per day, compared with the watch of a stationary observer at sea level?

HOW TO MAKE A PLOT / FUNCTION IN PYTHON DEMO



```
In [16]: from astropy.constants import G

In [17]: import astropy.units as u

In [18]: import matplotlib.pyplot as plt

In [19]: import numpy as np

In [20]: def v(M,r):
    ...:     return np.sqrt(G*M/r) #this function will work on any values of M and r!
    ...:

In [21]: v(1*u.M_earth,1*u.R_earth).si #calculate the orbital velocity at surface of Earth as a test
Out[21]: <Quantity 7905.38823439 m / s>

In [22]: rList = np.linspace(1,100,100)*u.R_earth #make a list that goes from 1 to 100 with 100 entries in units of R_e

In [23]: vList = [v(M,r) for r in rList] #make a list of velocities corresponding to radii in rList

In [24]: plt.plot([r.si.value for r in rList],[v.si.value for v in vList]) #plot the si values (m and m/s)
Out[24]: [<matplotlib.lines.Line2D at 0x7fd5e47d9710>]

In [25]: plt.xlabel("r [m]"); plt.ylabel("v [m/s]") #give the axes labels
Out[25]: Text(0, 0.5, 'v [m/s]')

In [26]: plt.show() #show the plot
```

Code link: https://gist.github.com/kirklong/a0ffe958da63bda9fc231386b1fe1766