In implementing Part 1 I first did the specific integration estimates defined in step \_\_.

When it came time to try multiple intervals I initially ran my program with .1, .05, and .001. The difference in the errors was orders of magnitude. The magnitude of the errors was very, very small. The .1 interval reached magnitudes 0.00026236 while the .05 interval reached 0.0000019487 and the .001 reached 0.00000000000027898. It was interesting to me that all of these ran on my laptop without a noticeable difference in run time from the original program which just had a single loop and a maximum interval of .1. I realize when the integration is being performed for a simulation with many objects moving in the scenario it requires exponentially more processing power both to move the objects and to account for collisions. So I wondered what happened to the error if I reduced the integration interval. Therefore, I ran the same code after modifying the intervals to be .1, .2 and .5. I suppose I could have tried to get sophisticated and create a GUI to prompt for parameters such as the output file name and the intervals which would make this more versatile. I then created graphs for the exact solution plus the RK4 integration for these three intervals. Even at an interval of .5 the error only reached 3.4194% in the range of 0 to 10 using RK4 estimation. I can see why, depending on the specific simulation needs, a less precise integration estimation which is also computationally less expense can be very appropriate.