Name: ______________________________

Date: ______________________________

Collaborators: ____

(Collaborators submit their individually written assignments together)

Question:	1	2	3	4	Total
Points:	20	45	15	10	90
Score:					

Instructor/grader comments:

Boundary value problem for ODEs

1. (20 points) Let u represent the electrostatic potential between two concentric conducting spheres of radii r_1 and r_2, ($r_1 < r_2$). The potential of the inner sphere is kept constant at $u(r_1) = V_1$, and the outer sphere is grounded, $u(r_2) = 0$. The potential in the region between the two spheres is governed by the following equation:

$$\frac{\mathrm{d}^2 u}{\mathrm{d}r^2} + \frac{2}{r}\frac{\mathrm{d}u}{\mathrm{d}r} = 0.$$

   Write a matlab script, **hw06p1.m**, to find the numerical solution of the boundary value problem above using the linear shooting algorithm. Use the following numerical values of the parameters:

$$r_1 = 2 \text{ a.u.}, \quad r_2 = 4 \text{ a.u.}, \quad V_1 = 100 \text{ a.u..}$$

   Use one of the matlab's own ode IVP solvers as a part of your method. Explain your choice in your README file.

   On the same figure plot your numerical solution as well as the following analytic solution of the BVP:

$$u(r) = \frac{r_1}{r}\left(\frac{r_2 - r}{r_2 - r_1}\right)V_1.$$

2. Apply multivariate Newton's method to find the single point of intersection of the spheres with center $(1,0,1)$ and radius $\sqrt{8}$, center $(0,2,2)$ and radius $\sqrt{2}$, and center $(0,3,3)$ and radius $\sqrt{2}$.

   Hint: The equation of the sphere with the center at (a,b,c) and radius R is

$$(x_1 - a)^2 + (x_2 - b)^2 + (x_3 - c)^2 - R^2 = 0.$$

   (a) (5 points) Write down the system of equations that you need to solve

   (b) (10 points) Write matlab function `hw06p2eqs` that accepts the coordinates of a point and returns a vector of residuals of your equations at that point.

(c) (10 points) Write down the Jacobian matrix for your equations

(d) (10 points) Write matlab function `hw06p2jac` that accepts the coordinates of a point and returns jacobian matrix of calculated at that point.

(e) (10 points) Write matlab script `hw06p2` that solves the problem using your code and the code developed in class. Chose your own initial approximation.

Hint: Do not chose the initial approximation in the vicinity of the origin of you coordinate system.

3. (15 points) Use Broyden's method to find the solution of Problem 2. Write matlab script `hw06p3` that solves the problem using your code and the code developed in class. Choose your own initial approximation with the relative error of 5–10% off the solution you found in Problem 2. Compare the results of Broyden's and Newton's methods in your README file.

Gitlab

4. (10 points) Create a gitlab project called **hw06** (name it exactly as shown). Upload **all** required matlab code and create your README.md file. Share the project with the instructor.