Problem 1: Find the expression for the gradient and Hessian for the following functions, and consequently find a local minima, if there exists one (use MATLAB if necessary):
(a) \( f = x_1 - \frac{10}{x_1x_2} + 5x_2 \); (b) \( f = 100(x_2 - x_1)^2 + (1 - x_1)^2 \); (c) \( f = 5x_1 + x_2 \)

Problem 2: Given a generic quadratic function in two variables:
\( f = 0.5Ax^2 + Bxy + 0.5Cy^2 + Dx + Ey + F \)
classify the various types of contour plots one can expect depending on the various coefficients.

Problem 3: Write a MATLAB function to find the gradient of a spring system:
```matlab
function [GradPE] = GradPEofSpringSystem_LastName(springSystem,freeNodes,uvOfFreeNodes)
```
where:
- springSystem is a structure with four entries (see code from website): (a) initialNodeLocations, (b) springConnectivity, (c) Forces and (d) stiffness
- freeNodes is a (1 x Nf) where Nf is the number of free nodes
- uvOfFreeNodes is a (1 x 2*Nf) vector with entries \((u1, v1, u2, v2, \ldots)\) where \((ui, vi)\) is the displacement of the ith node.
- GradPE is a (2*Nf x 1) vector with entries \((d(PE)/du1, d(PE)/dv1, \ldots)\)

Compare the output with finite difference gradient (as a sanity check). As a second verification, use the above code to find the stationary points of various spring systems, via MATLAB's fsolve. No need to submit the latter code, but compare with fminunc method, and provide comments if any.

Problem 4: Consider two circles centered at \((x_1, y_1)\) and \((x_2, y_2)\) with radii \(r_1\) and \(r_2\).
Write a Matlab code to find a pair of points \((x_1^*, y_1^*)\) and \((x_2^*, y_2^*)\) on the two circles such that the distance between the two points is a local minima. Assume that the two circles are not concentric. Pose the problem as an unconstrained minimization problem, and use appropriate MATLAB optimization routines. Your code format should be:
```matlab
function [x1S,y1S,x2S,y2S] = closestPoints_LastName(x1,y1,r1,x2,y2,r2)
```
Test your code before submitting!