Problem 1

We want to write an algorithm to detect, isolate and transform letter e in the image of figure 1-left to a different letter or character of your choice. The main difficulty in the problem arises from the fact that letter e (as an image) is not exactly the same.

1) Develop an algorithm and write code to detect and isolate letter e in the image (everything else must go).
2) Transform letter e to single points and count their number in the image.
3) Transform letter e to another letter or character of your choice.
4) Show your work and discuss your steps, method, and results.

Problem 2

Use the Hough transform to detect the lines in a given image of you choice. Pick an image with visible lines. Use command `houghpeaks` to identify the peaks and then mark them with a small rectangle or circle.
Next in the file we find the functions for updating the states, compute_dx, and the output, compute_y. Both these functions hold argument lists, with the output to be computed (dx or y) at position 1, after which follows all variables and parameters required to compute the right-hand side(s) of the state and the output equations, respectively.

The first step in these functions is to unpack the model parameters that will be used in the subsequent equations. Any valid variable name (except for those used in the input argument list) can be used to provide physically meaningful names of the individual parameters.

As is the case in C, the first element of an array is stored at position 0. Hence, dx[0] in C corresponds to dx(1) in MATLAB® (or just dx in case it is a scalar), the input u[0] corresponds to u (or u(1)), the parameter A1[0] corresponds to A1, and so on.

The two tank model file involves square root computations. This is enabled through the inclusion of the mathematical C library math.h. The math library realizes the most common trigonometric functions (sin, cos, tan, asin, acos, atan, etc.), exponential (exp) and logarithms (log, log10), square root (sqrt) and power of functions (pow), and absolute value computations (fabs). The math.h library must be included whenever any math.h function is used; otherwise it can be omitted. See "Tutorials on Nonlinear Grey Box Model Identification:"