ECS130 Homework Assignment #1 **Due: 4:00pm, January 23, 2017**

1. (a) Derive, mathematically, the back substitution method for solving the upper triangular linear system

$$Ux = b$$

using componentwise, row-oriented and column-oriented algorithms, respectively.

(b) Write three MATLAB functions:

```
x = mybscomponent(U,b)
x = mybsrow(U,b)
x = mybscolumn(U,b)
```

for the componentwise, row-oriented and column-oriented algorithms, respectively.

(c) Test the correctness of your functions and compare the execution time of these functions for a set of different sizes of upper triangular linear systems.

2. Modify the `lutx` function to a new function called `mylutx` so that it uses explicit `for` loops instead of MATLAB vector notation. For example, one section of your modified program will read

```
% compute the multipliers
for i = k+1:n
    A(i,k) = A(i,k)/A(k,k);
end
```

Test the correctness of your function `mylutx`, and compare the execution time of `mylutx` with `lutx` and with the built-in `lu` function by finding the order of the matrix for which each of the three programs takes about 10 seconds on your computer.

3. The inverse of an $n \times n$ matrix A can be defined as the matrix X whose columns x_j solve the equations

$$Ax_j = e_j \quad \text{for } j = 1, 2, \ldots, n,$$

where e_j is the jth column of of the identity matrix.

(a) Starting with the function `bslashtx`, write a MATLAB function `X = myinv(A)` that computes the inverse of `A`. Your function should call `lutx` only once and should not use the built-in MATLAB backslash operator or `inv` function.

(b) Test your function by comparing the inverses it computes with the inverses obtained from the built-in `inv(A)` on a few test matrices, say from "`gallery`" collection in MATLAB.