ASE 211 Homework 4 Solution

1. By hand, compute the LU decomposition of the following matrix:

$$A = \begin{bmatrix} 3 & 2 & -1 \\ 6 & 1 & 0 \\ -3 & 6 & 4 \end{bmatrix}.$$

$$A = \begin{bmatrix} 3 & 2 & -1 \\ 6 & 1 & 0 \\ -3 & 6 & 4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & -\frac{8}{3} & 1 \end{bmatrix} \begin{bmatrix} 3 & 2 & -1 \\ 0 & -3 & 2 \\ 0 & 0 & \frac{25}{3} \end{bmatrix}.$$

2. For the matrix in problem 1, use forward and backward substitution to solve $A\mathbf{x} = \mathbf{b}$, where

$$\mathbf{b} = \begin{bmatrix} 8 \\ 16 \\ 42 \end{bmatrix}.$$

First solve $L\mathbf{y} = \mathbf{b}$ to get $\mathbf{y} = (8, 0, 50)$. Then solve $U\mathbf{x} = \mathbf{y}$ to get $\mathbf{x} = (2, 4, 6)$. Check that $A\mathbf{x} = \mathbf{b}$.

3. Using the Matlab built-in **lu** command, repeat question 1.

```
>> A=[3 2 -1; 6 1 0; -3 6 4];
>> [L,U,P]=lu(A)

L =

    1.0000         0         0
   -0.5000    1.0000         0
    0.5000    0.2308    1.0000


U =

    6.0000    1.0000         0
         0    6.5000    4.0000
         0         0   -1.9231
```

```
P =

     0     1     0
     0     0     1
     1     0     0

>> L*U

ans =

     6     1     0
    -3     6     4
     3     2    -1

>> P*A

ans =

     6     1     0
    -3     6     4
     3     2    -1

>> diary
```

4. Write Matlab m-files forsolve.m and backsolve.m which perform forward and backward substitution, given the LU decomposition of the matrix. Test your m-files on the system given in problems 1 and 2.

```
function x=forsolve(A,b,n)
for i=1:n
  x(i)=b(i);
  for k=1:i-1
     x(i)=x(i)-A(i,k)*x(k);
  end
  x(i)=x(i)/A(i,i);
end
```

```
function x=backsolve(A,b,n)
for i=n:-1:1
  x(i)=b(i);
  for k=n:-1:i+1
     x(i)=x(i)-A(i,k)*x(k);
  end
  x(i)=x(i)/A(i,i);
end
```

```
>> b=[8; 16; 42];
>> n=3;
>> b=P*b;
>> y=forsolve(L,b,n)

y =

   16.0000   50.0000  -11.5385

>> x=backsolve(U,y,n)

x =

     2     4     6
```

5. Use the Matlab code you have written in problems 3 and 4 to solve the system in problem 3.36 in the book, with the four different right hand sides given. Remember you only have to do the LU decomposition once.

```
>> A=[-5 0 -4 1 4 5; 3 5 -2 -4 3 -2; -1 -3 3 4 3 1; 0 1 1 1 -1 -4; -4 -1 -4 -3 2 0; -

A =

    -5     0    -4     1     4     5
     3     5    -2    -4     3    -2
    -1    -3     3     4     3     1
     0     1     1     1    -1    -4
    -4    -1    -4    -3     2     0
    -3    -3    -4     5     3     1
```

```
>> [L,U,P]=lu(A)

L =

    1.0000         0         0         0         0         0
   -0.6000    1.0000         0         0         0         0
    0.6000   -0.6000    1.0000         0         0         0
    0.8000   -0.2000    0.3962    1.0000         0         0
    0.2000   -0.6000   -0.2736   -0.4443    1.0000         0
         0    0.2000   -0.4434   -0.5035   -0.2090    1.0000


U =

   -5.0000         0   -4.0000    1.0000    4.0000    5.0000
         0    5.0000   -4.4000   -3.4000    5.4000    1.0000
         0         0   -4.2400    2.3600    3.8400   -1.4000
         0         0         0   -5.4151   -1.6415   -3.2453
         0         0         0         0    5.7613   -1.2247
         0         0         0         0         0   -6.7106


P =

     1     0     0     0     0     0
     0     1     0     0     0     0
     0     0     0     0     0     1
     0     0     0     0     1     0
     0     0     1     0     0     0
     0     0     0     1     0     0

>> b1=[14; -26; 0; -16; 0; -17];
>> b1=P*b1;
>> n=6;
>> y1=forsolve(L,b1,n);
>> x1=backsolve(U,y1,n)

x1 =
```

```
   -3.0000    0.0000    3.0000   -2.0000   -3.0000    5.0000

>> b2=[19; -3; -15; -4; 15; 10];
>> b2=P*b2;
>> y2=forsolve(L,b2,n);
>> x2=backsolve(U,y2,n)

x2 =

   -2.0000    1.0000   -3.0000    0.0000   -2.0000    1.0000

>> b3=[-16;29;-4;2;-13;-31];
>> b3=P*b3;
>> y3=forsolve(L,b3,n);
>> x3=backsolve(U,y3,n)

x3 =

    2.0000    3.0000    3.0000   -2.0000    2.0000    0.0000

>> b4=[-44;3;-23;3;-10;-40];
>> b4=P*b4;
>> y4=forsolve(L,b4,n);
>> x4=backsolve(U,y4,n)

x4 =

    3.0000   -2.0000    1.0000   -4.0000   -4.0000   -1.0000
```