Exercise : Use the worksheet to wite down the PLU factorization.

$$P_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, L_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, U_1 = \begin{bmatrix} 1/3 & 0 & 0 \\ 1 & 1 & 1 \\ 1/2 & 0 & 1 \end{bmatrix}$$

$$P_2 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, L_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, U_2 = \begin{bmatrix} 1 & 1 & 1 \\ 1/3 & 0 & 0 \\ 1/2 & 0 & 1 \end{bmatrix}$$

$$P_3 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, L_3 = \begin{bmatrix} 1 & 0 & 0 \\ 1/3 & 1 & 0 \\ 1/2 & 0 & 1 \end{bmatrix}, U_3 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & -1/3 & -1/3 \\ 0 & -1/2 & 1/2 \end{bmatrix}$$

$$P_4 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, L_4 = \begin{bmatrix} 1 & 0 & 0 \\ 1/2 & 1 & 0 \\ 1/3 & 0 & 1 \end{bmatrix}, U_4 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & -1/2 & 1/2 \\ 0 & -1/3 & -1/3 \end{bmatrix}$$

$$P_5 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, L_5 = \begin{bmatrix} 1 & 0 & 0 \\ 1/2 & 1 & 0 \\ 1/3 & 2/3 & 1 \end{bmatrix}, U_5 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & -1/2 & 1/2 \\ 0 & 0 & -2/3 \end{bmatrix}$$

Exercise : Let

$$A = \begin{bmatrix} 10^{-16} & 1 \\ 1 & 1 \end{bmatrix}, b = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

(a) Solve $Ax = b$ exactly.

$$\begin{bmatrix} 10^{-16} & 1 \\ 1 & 1 \end{bmatrix} x = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

$$\begin{bmatrix} 10^{-16} & 1 \\ 0 & 1 - 10^{16} \end{bmatrix} x = \begin{bmatrix} 2 \\ 3 - 2 * 10^{16} \end{bmatrix}$$

$$\begin{bmatrix} 10^{-16} & 0 \\ 0 & 1 - 10^{16} \end{bmatrix} x = \begin{bmatrix} 2 - \frac{3-2*10^{16}}{1-10^{16}} \\ 3 - 2 * 10^{16} \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x = \begin{bmatrix} \frac{2}{10^{-16}} - \frac{3-2*10^{16}}{10^{-16}(1-10^{16})} \\ \frac{3-2*10^{16}}{1-10^{16}} \end{bmatrix}$$

$$x = \begin{bmatrix} \frac{2}{10^{-16}} - \frac{3-2*10^{16}}{10^{-16}(1-10^{16})} \\ \frac{3-2*10^{16}}{1-10^{16}} \end{bmatrix} = \begin{bmatrix} \frac{2-2*10^{16}-(3-2*10^{16})}{10^{-16}(1-10^{16})} \\ \frac{3-2*10^{16}}{1-10^{16}} \end{bmatrix} = \begin{bmatrix} \frac{-1}{10^{-16}-1} \\ \frac{3-2*10^{16}}{1-10^{16}} \end{bmatrix} \approx \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

As a check:

$$Ax = \begin{bmatrix} 10^{-16} & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 10^{-16} + 2 \\ 3 \end{bmatrix} \approx \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

(b) What is the 2-norm condition number for A? Is A well behaved in the 2-norm?
The condition number is 2.6180. This is well conditioned since errors is output will be on the same order as errors in imput.

(c) Here is my solution to solving a matrix without pivoting

```
function x=matrixSolve(A,b)
#make A upper triangular
for i=1:(length(A)-1)
    for j=i+1:length(A)
        k=A(j,i)/A(i,i);
        #A(j,i)=0; These values are never used again,
        #don't overide just remember they are zero
        A(j,[i+1:length(A)])-=k*A(i,[i+1:length(A)]);
        b(j)-=k*b(i);
    end
end
#our matrix A is "upper diagonal", not really but we have implied zeros.
#now make A diagonal
for i=length(A):-1:2
    for j=1:i-1
        b(j)-=b(i)*A(j,i)/A(i,i);
        #implied zeroing out of A
    end
end
#our matrix A is "diagonal", not really but we have implied zeros.
#now make A the identity matrix
for i=1:length(A)
    b(i)/=A(i,i);
    #A(i,i)/=A(i,i); implied
end
#our matrix A is "ientity", not really but implied.
x=b;

endfunction
```

And the solution I get:

```
>> x=matrixSolve(A,b)
x =

   4.4409
   2.0000

>> A\b
ans =

   1
   2

>> diary off
```

The problem with this method is that adding and subtracting large numbers causes errors:

```
>> 3+10^16
ans =    1.0000e+16
>> ans-10^16
ans =  4
>> diary off
```

Exercise : From the worksheet on implementing partial pivoting, show your code for mylu.m. Then show your answer to problem 10. Also, use your function usolve from the last homework and lsolve from the course web page to solve $Ax = b$ where A is the matrix from problem 10 of the worksheet and $b = \begin{bmatrix} -1 \\ 6 \\ -8 \end{bmatrix}$.

```
function [L,U]=mylu(A)
    n=length(A);
    L=eye(n);
    U=A;
    for i=1:(n-1)
        for j=i+1:n
            k=U(j,i)/U(i,i);
            L(j,i)=k;
            U(j,i)=0;
            U(j,i+1:n)-=k*U(i,i+1:n);
        end
    end
endfunction
```

This is my demonstration of L U factorization:

```
>> A=[-4,3,3;20,-13,-14;-16,8,8]
A =

   -4    3    3
   20  -13  -14
  -16    8    8

>> [L,U]=mylu(A)
L =

   1    0    0
  -5    1    0
   4   -2    1

U =

```

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

>> L*U
ans =

   -4    3    3
   20  -13  -14
  -16    8    8

>> diary off
```

Here are my usolve, isolve, and general solve without partial pivoting:

```
function x=usolve(U,y)
    n=length(y);
    x=zeros(n,1);
    for(i=n:-1:1)
        #calc x
        x(i)=y(i)/U(i,i);
        #eliminate entries
        #note that we will not acctualy change the values of U only
        #remember that all terms furthur out than i are zero's
        y(1:(i-1))=y(1:(i-1))-x(i)*U(1:(i-1),i);
    end
endfunction
```

```
function x=lsolve(L,y)
U=L(length(L):-1:1,length(L):-1:1);
y=y(length(y):-1:1);

x=usolve(U,y);

x=x(length(x):-1:1);
endfunction
```

```
function x=noPPsolve(A,y)
    [L,U]=mylu(A);
    x=usolve(U,lsolve(L,y));
endfunction
```

So solving the desired problem:

```
>> A=[-4,3,3;20,-13,-14;-16,8,8]
A =

   -4    3    3
   20  -13  -14
```

```
  -16    8    8

>> b=[-1;6;-8]
b =

  -1
   6
  -8

>> x=noPPsolve(A,b)
x =

   1
   0
   1

>> A*x
ans =

  -1
   6
  -8

>> diary off
```

Exercise : To compute the inverse of an $n \times n$ matrix A you need to find n vectors v_i such that $Av_i = e_i$, where e_i is the vector of all zeros, except that e_i has a one in its i th entry. E.g., if $n = 4$ then $e_3 = [0, 0, 1, 0]^T$. Once the vectors v_i are known, then $A^{-1} = [v_1|v_2|\cdots|v_n]$. How many floating point operations are required to perform an LU decomposition of A and then solve for the n vectors v_i? If one wants to compute the solution of $Ax = b$ by computing A^{-1} and then mutiplying to obtain $x = A^{-1}b$, how many floating point operations does this take? Compare this number with the number of floating point operations to solve $Ax = b$ by LU decomposition without computing A^{-1} .

Doing a LU decomposition, lets calculate the cost by examining the solution, converting for loops into summations, we see imeadiately that there are

$$\sum_{1}^{n-1}\sum_{i+1}^{n} 1 = \sum_{1}^{n-1} n - i = (n-1)n - \frac{(n-1)n}{2} = 1/2n^2 - 1/2n$$

devisions. There will be the same number of subtractions as multiplications so we get

$$\sum_{1}^{n-1}\sum_{i+1}^{n}\sum_{i+1}^{n} 1 = \sum_{1}^{n-1}\sum_{i+1}^{n} n-i = \sum_{1}^{n-1}(n-i)(n-i) = \sum_{1}^{n-1} n^2-2ni+i^2 = n^2(n-1)-2n\frac{(n-1)n}{2}+\frac{(n-1)(n)(2n-1)}{6}$$

multiplications or devisions. Well these numbers are nasty so lets work with order of magnitude. LU factorization costs $\frac{2n^3}{3} + O(n^2)$ FLOPs.

Solving a unit lower triangular matrix would require a multiply and a add for each entry

below the diagonal, this would be $n^2/2 - n$ adds and multiplys. The solve on the upper triangular matrix works out the same but there are n additional devisions, since the diagonal is not zeroed. so a compleate LU solve requires $4 * (n^2/2 - n) + n = 2n^2 + O(n)$ FLOPs.
If we want to find the inverse of a matrix by doing LU factoring and n LU solves it will take $\frac{2n^3}{3} + O(n^2) + n(2n^2 + O(n)) = 2\frac{2n^3}{3} + O(n^2)$, If we then multiply to solve we take a additional $2n^2 + O(n)$ operations for a total of $2\frac{2n^3}{3} + O(n^2)$ FLOPs. This is much worse than a simple LU factor and solve method since a LU factor and solve method only takes $\frac{2n^3}{3} + O(n^2) + (2n^2 + O(n)) = \frac{2n^3}{3} + O(n^2)$ FLOPs.

Exercise w: orksheet problem 16

```
>> L=[1 0 0 0;2 1 0 0 ;3 4 1 0;5 6 0 1]
L =

   1   0   0   0
   2   1   0   0
   3   4   1   0
   5   6   0   1

>> L([3 4],[1 2])=L([4 3],[1 2])
L =

   1   0   0   0
   2   1   0   0
   5   6   1   0
   3   4   0   1

>> diary off
```

Exercise w: orksheet problem 17

This is a working solution

```
function [P,L,U]=myplu(A)
    n=length(A);
    p=[1:n];
    L=eye(n);
    U=A;
    for i=1:(n-1)
        U
        [x,xi]=max(abs(U(i:n,i)));
        xi=xi+i-1;

        temp=p(i);
        p(i)=p(xi);
        p(xi)=temp;

```

```
        temp=U(i,i:n);
        U(i,i:n)=U(xi,i:n);
        U(xi,i:n)=temp;
        if i~=1
            temp=L(i,1:(i-1));
            L(i,1:(i-1))=L(xi,1:(i-1));
            L(xi,1:(i-1))=temp;
        end
        U
        disp("next");
        #we have now pivoted
        for j=(i+1):n
            k=U(j,i)/U(i,i);
            L(j,i)=k;
            U(j,i)=0;
            U(j,i+1:n)-=k*U(i,i+1:n);
        end
    end
    I=eye(n);
    P=I(p,:);
endfunction
```

Here is a demonstration that it works:

```
>> a=rand(4,4)*2-1
a =

  -0.1010625   0.9362113   0.0139713   0.4418893
  -0.5953604   0.7011362  -0.4407313  -0.7969206
  -0.9817731  -0.2038775   0.2376541   0.2044053
  -0.0058744   0.1644356  -0.8053894   0.8938794

>> b=rand(4,1)
b =

   0.74662
   0.17145
   0.29130
   0.36871

>> [P,L,U]=myplu(a);
U =

  -0.1010625   0.9362113   0.0139713   0.4418893
  -0.5953604   0.7011362  -0.4407313  -0.7969206
  -0.9817731  -0.2038775   0.2376541   0.2044053
  -0.0058744   0.1644356  -0.8053894   0.8938794

U =

  -0.9817731  -0.2038775   0.2376541   0.2044053
  -0.5953604   0.7011362  -0.4407313  -0.7969206
  -0.1010625   0.9362113   0.0139713   0.4418893
```

```
  -0.0058744   0.1644356  -0.8053894   0.8938794

next
U =

  -0.98177  -0.20388   0.23765   0.20441
   0.00000   0.82477  -0.58485  -0.92087
   0.00000   0.95720  -0.01049   0.42085
   0.00000   0.16566  -0.80681   0.89266

U =

  -0.98177  -0.20388   0.23765   0.20441
   0.00000   0.95720  -0.01049   0.42085
   0.00000   0.82477  -0.58485  -0.92087
   0.00000   0.16566  -0.80681   0.89266

next
U =

  -0.98177  -0.20388   0.23765   0.20441
   0.00000   0.95720  -0.01049   0.42085
   0.00000   0.00000  -0.57581  -1.28350
   0.00000   0.00000  -0.80500   0.81982

U =

  -0.98177  -0.20388   0.23765   0.20441
   0.00000   0.95720  -0.01049   0.42085
   0.00000   0.00000  -0.80500   0.81982
   0.00000   0.00000  -0.57581  -1.28350

next
>> P
P =

Permutation Matrix

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

>> P*a*b-L*U*b
ans =

   0
   0
   0
   0

>> diary off
```