Crout Factorization

1) Ask the user for a matrix A. Needs to be a squared matrix.
2) Ask the use for vector b and make sure that is the same length as matrix A, and ask for a whole number n.
3) Next, we create 2 more matrices, we call them L and U. Matrices L and U need to have the following characteristics
   a) Have the same size as matrix A
   b) L will have the same elements that are below the diagonal in A but with the opposite sign. The diagonal in L will be the same as A, and all other elements will be 0.
   c) U will have the same elements in A that are above the diagonal with opposite signs, and will have the same elements in the diagonal as A and the rest of the elements will be 0.

4) Time for a cycle
   a) For k = 1 < n, with 1 step (k++)
      (i) sum1 = 0
      (ii) For p = 1 < k-1, with 1 step (p++)
      (iii) sum1 = sum1 + $L_{kp} * U_{kp}$
      (iv) $L_{kk} = \sqrt{a_{kk} + sum1}$
      (v) For i = k+1 < n , with 1 step (i++)
      (vi) sum2 = 0
         (1) For r = 1 < k-1 , with 1 step (r++)
         (2) sum2=sum2+$L_{ir} * U_{rk}$
      (vii) $L_{ik} = (a_{kk} - \text{sum2})$
   b) For j = k+1 < n , with 1 step (j++)
      (i) sum3 = 0
         (1) For s = 1 < s-1, with 1 step (s++)
            (A) sum3 = sum3 + $L_{ks} * U_{sj}$
            (B) $U_{kj} = (a_{kj} - \text{sum3}) / L_{kk}$
5) z = progressive substitution (1,b) x = regressive substitution (u,z)
6) Regressive Replacement
   a) Para i = n < 1 , with 1 step (i++)

(i) sum = 0

(1) for j = i +1 < n , with 1 step (j++)

(A) sum = sum+a_{ij}* x_{ij}

(2) xi =(a_{in+1}− sum)/a_{ii}

7) Progressive replacement

a) Para i = 1 < n , with 1 step (i++)

(i) sum = 0

(1) for j = i +1 < n , with 1 step (j++)

(A) sum = sum+a_{ij}* x_{ij}

(2) xi =(a_{in+1}− sum)/a_{ii}