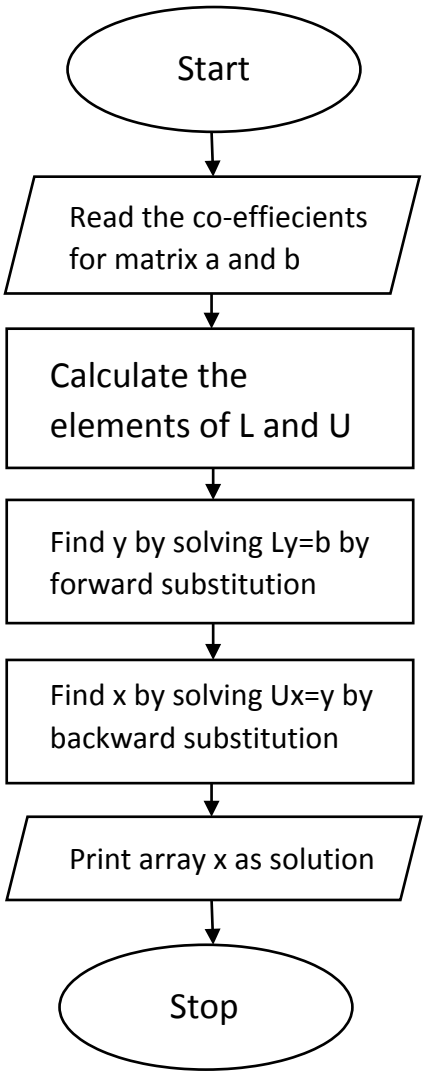
Algorithm

1. Define the arrays b[n], x[n], y[n], a[n][n], L[n][n], U[n][n].
2. Read the co-efficients in a[i][j], for i = 1 to n and for j = 1 to n.
3. Read the co-efficients in b[i], for i = 1 to n.
4. Initialize x[i] = 0, y[i] = 0, L[i][j] = 0, U[i][j] = 0 for i = 1 to n, and j = 1 to n+1.
5. Now for i = 1 to n, and j = 1 to n, set sum = 0, then inside these for loops add one more for loop for k = 1 to n. For k = 1 to n, if j==k, U[k][j] == 1 and for the same k = 1 to n, sum = sum + L[i][k]*U[k][j].
6. Now if i >= j, L[i][j] = a[i][j] – sum.
7. Else U[i][j] = (a[i][j] – sum)/L[i][i].
8. Now for i = 1 to n, sum = 0, for j = 1 to n, sum = sum + L[i][j]*y[j] y[i] = (b[i] – sum)/L[i][i].
9. Then again for i = 1 to n, sum = 0, for j = 1 to n, sum = sum + U[i][j]*x[j], x[i] = (y[i] – sum)/U[i][i].

   Finally print the solution by printing out x[i] for i = 1 to n.

Flowchart


Start
Read the co-effiecients for matrix a and b
Calculate the elements of L and U
Find y by solving Ly=b by forward substitution
Find x by solving Ux=y by backward substitution
Print array x as solution
Stop