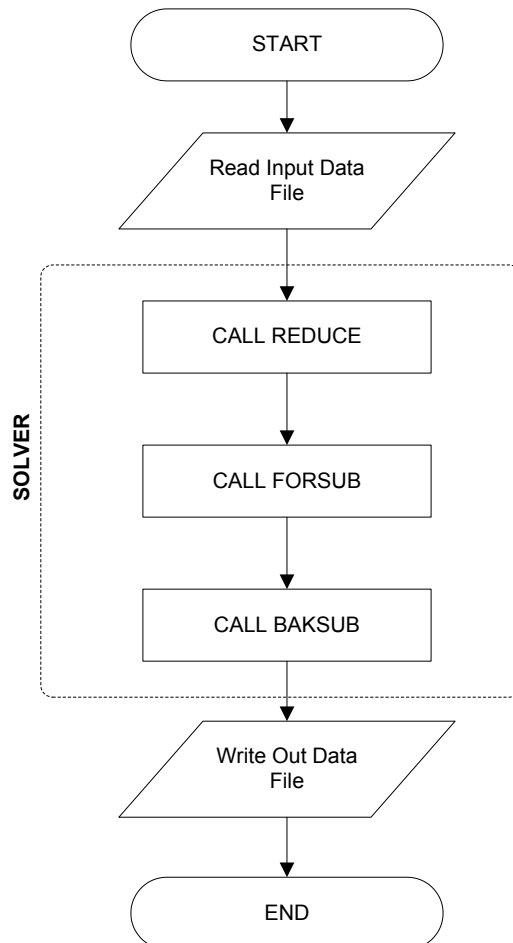


FLOWCHART

This program for solving a system of linear equations ($AX = B$), employing LU decomposition.



ส่วนของโปรแกรมหลักที่ใช้ในการแก้สมการ ($AX = B$) โดยเรียกโปรแกรมย่อย REDUCE , FORSUB , BAKSUB
เมื่อให้

เมตริก $[A]$ มีขนาดเท่ากับ $[N \times N]$,
เวกเตอร์ด้านขวา $\{B\}$ มีขนาดเท่ากับ $[N \times 1]$,
เวกเตอร์คำตอบ $\{X\}$ มีขนาดเท่ากับ $[N \times 1]$

```
C..... Perform solve equations
      CALL REDUCE (A,N)
      CALL FORSUB (A,B,Y,N)
      CALL BAKSUB (A,Y,X,N)
```

ส่วนของโปรแกรมย่อย REDUCR เพื่อ REDUCE เมตริก [A] ให้เป็น เมตริกสามเหลี่ยมบน [U]

```
SUBROUTINE REDUCE (A,N)
IMPLICIT DOUBLE PRECISION (A-H,O-Z)
INTEGER          :: I,J,K
DIMENSION        A(50,50)
DO J=1,N-1
  I1 = J+1
  DO I=I1,N
    !      Check divide error
    IF (A(J,J) == 0) THEN
      WRITE (*,*) 'Divide error!!! (Ajj = 0)',J
    END IF
    C = -1.0*(A(J,I)/A(J,J))
    DO K = J,N
      A(I,K) = A(I,K)+A(J,K)*C
    END DO
  END DO
END DO
RETURN
END SUBROUTINE REDUCE
```

ส่วนของโปรแกรมย่อย FORSUB เพื่อทำ FORWARD-SUBSTITUTION เพื่อหาค่า {Y}

```
SUBROUTINE FORSUB (U,B,Y,N)
!=====!
!
!   This subprogram to solve (LY = B) for Y using forward substitution !
!
!=====!
IMPLICIT DOUBLE PRECISION (A-H,O-Z)
INTEGER          :: I,J
DIMENSION        U(50,50),B(50),Y(50)
Y(1) = B(1)
DO I=2,N
  J1 = 1
  J2 = I-1
  SUM = 0
  DO J =J1,J2
    SUM = SUM + U(J,I)/U(J,J)*Y(J)
  END DO
  Y(I) = B(I) - SUM
END DO
RETURN
END SUBROUTINE FORSUB
```

ส่วนของโปรแกรมย่อย BAKSUB เพื่อทำ BACK-SUBSTITUTION เพื่อหาค่า $\{X\}$

```
SUBROUTINE BAKSUB (U,Y,X,N)
!=====!
!
!   This subprogram to solve (UX = Y) for X using back-substitution   !
!
!=====!
IMPLICIT DOUBLE PRECISION (A-H,O-Z)
INTEGER      :: J,K
DIMENSION    U(50,50),Y(50),X(50)
X(N) = Y(N)/U(N,N)
DO K = (N-1),1,-1
    J1 = K+1
    J2 = N
    SUM = 0
    DO J = J1,J2
        SUM = SUM + U(K,J)*X(J)
    END DO
    X(K) = (Y(K)-SUM)/U(K,K)
END DO
RETURN
END SUBROUTINE BAKSUB
```