บทที่ 3
การเขียนโปรแกรม

3.1 บทนำ

จากที่ได้กล่าวมาแล้วว่า ในส่วนของโปรแกรมเราได้แบ่งออกเป็น 2 ส่วนหลักๆ ดังนั้นในด้านการเขียนโปรแกรมจึงจะขอแยกการอธิบายเป็น 2 หัวข้อ คือ

1. การเขียนโปรแกรมแก้ปัญหาโครงข้อหมุนโดยใช้วิธีการรวมสติฟเนสโดยตรง
2. การเขียนโปรแกรมแก้สมการเชิงเส้น (K * u = P) โดยใช้ วิธีชอเลสกี้แบบประยุกต์

3.2 การเขียนโปรแกรมวิเคราะห์โครงข้อหมุน

ในการเขียนโปรแกรมโดยใช้ วิชวลเบสิก6.0 จะทำการเขียนโปรแกรมเชิงวัตถุ (Object Oriented Programming) นั่นคือ จะเขียนโปรแกรมตามปุ่มคำสั่งที่จะตอบสนองต่อเหตุการณ์ต่าง ๆ ดังนั้น Flow Chart ในโปรแกรมนี้จะแบ่งตามปุ่มคำสั่งที่จะตอบสนองกับผู้ใช้

3.2.1 สัญลักษณ์สำคัญที่ใช้ใน Program

สัญลักษณ์	คำนิยาม
AJ( )	แรงภายนอกกระทำที่ Joint (ทิศทางตาม global coordinate)
AM( )	แรงภายในที่ปลายของชิ้นส่วนแต่ละชิ้น
AMD( )	แรงกระทำที่ปลายของชิ้นส่วนที่มีการยึดรั้งเนื่องจากแรงกระทำภายนอก (ทิศทางตาม local coordinate
AR( )	แรงปฏิกิริยาในแนวต่าง ๆ
DF( )	การเคลื่อนที่ของจุดต่ออิสระ (ทิศทางตาม global coordinate)
DJ( )	การเคลื่อนที่ของจุดต่อ (ทิศทางตาม global coordinate)
E	ค่าโมดูลัสความยืดหยุ่น (modulus of elasticity)
EL( )	ความยาวของชิ้นส่วน
I	หมายเลขประจำตัวชิ้นส่วน
ID	Displacement indexes สำหรับจุดต่อ
IM	Displacement indexes สำหรับชิ้นส่วน

สัญลักษณ์	คำนิยาม
IR , IC	หมายเลขประจำแถวและคอลัมน์ตามลำดับ
J , K	หมายเลขประจำจุดต่อ
JRL( )	รายการจุดยึดรั้ง
M	จำนวนชิ้นส่วนทั้งหมด
MD	จำนวนคู่ลำดับการเคลื่อนที่ของชิ้นส่วน (MD = 4)
N	จำนวนดีกรีอิสระ
NB	ความกว้างครึ่งแบนของเมตริกซ์สติฟเนส
ND	จำนวนคู่ลำดับการเคลื่อนที่ของจุดต่อ
NDJ	จำนวนการเคลื่อนที่ของจุดต่อทั้งหมด
NJ	จำนวนจุดต่อทั้งหมด
NLJ	จำนวนแรงกระทำที่จุดต่อทั้งหมด
NR	จำนวนแรงยึดรั้งที่ support
NRJ	จำนวนจุดยึดรั้งทั้งหมด
SFF( )	เมตริกซ์สติฟเนสของการเคลื่อนที่ (สำหรับจุดต่ออิสระ)ในแนวแกนโกลบัล
SMS( )	เมตริกซ์สติฟเนสของชิ้นส่วนในแนวแกนโกลบัล

3.2.2 ภาพรวมของโปรแกรม

โปรแกรมวิเคราะห์โครงสร้างนี้ประกอบไปด้วยโปรแกรมย่อยทั้งหมด 11 โปรแกรมแบ่งออกได้เป็น 3 กลุ่มใหญ่คือ

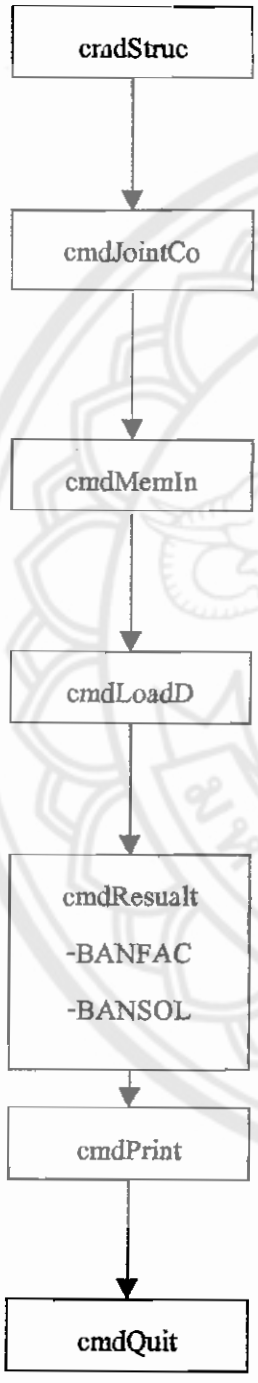
กลุ่มที่ 1 มีหน้าที่รับข้อมูล ได้แก่ โปรแกรมย่อย cmdStrc, cmdJointCo, cmdMemIn, cmdJointRe, cmdLoadD

กลุ่มที่ 2 มีหน้าที่วิเคราะห์ผล ได้แก่ โปรแกรมย่อย cmdResualt, BANFAC, BANSOL

กลุ่มที่ 3 มีหน้าที่ปลีกย่อยอื่น ๆ ได้แก่ โปรแกรมย่อย cmdPrint, cmdQuit

ลักษณะและลำดับการทำงานของแต่ละโปรแกรมย่อยได้แสดงไว้ใน Flow chart 1

Flow Chart 1 : Main Program



เป็นโปรแกรมย่อยสำหรับรับค่าข้อมูลของโครงสร้าง

เป็นโปรแกรมย่อยรับค่าโคออร์ดิเนตที่จุดต่อ

เป็นโปรแกรมย่อยรับค่าข้อมูลของชิ้นส่วนย่อย

เป็นโปรแกรมย่อยที่จะรับค่าจำนวนแรงและข้อมูลของแรงกระทำที่จุดต่อ

เป็นโปรแกรมย่อยที่ใช้ในการวิเคราะห์โครงข้อหมุนและแสดงผลลัพธ์ทางจอภาพ โดยภายในโปรแกรมจะมีการเรียกใช้โปรแกรมโปรแกรมย่อยอีก 2 โปรแกรมคือ โปรแกรมย่อย BANFAC และโปรแกรมย่อย BANSOL

เป็นโปรแกรมย่อยที่ใช้ในการสั่งให้เครื่องพิมพ์ทำการพิมพ์คำตอบที่ได้ออกมา

เป็นโปรแกรมย่อยเพื่อที่เราจะออกจากโปรแกรม

3.2.3 Subprogram cmdStruc

ทำหน้าที่รับค่าข้อมูลของโครงสร้าง นั้นคือจำนวนชิ้นส่วน (M) จำนวนจุดต่อทั้งหมด (NJ) จำนวนแรงปฏิกิริยา (NR) จำนวนจุดรองรับ (NRJ) และค่าโมดูลัสความยืดหยุ่น (E) จากนั้นนำค่าจำนวนชิ้นส่วน (M) มาคำนวณหา จำนวนดีกรีความอิสระ (degree of freedom,N) แล้วพิมพ์ทั้งหมดออกทางหน้าจอดังแสดงใน Flow Chart 2

Flow Chart 2 : Subprogram cmdStruc

ป
QA
76.76
.B3
ก141ก
2544

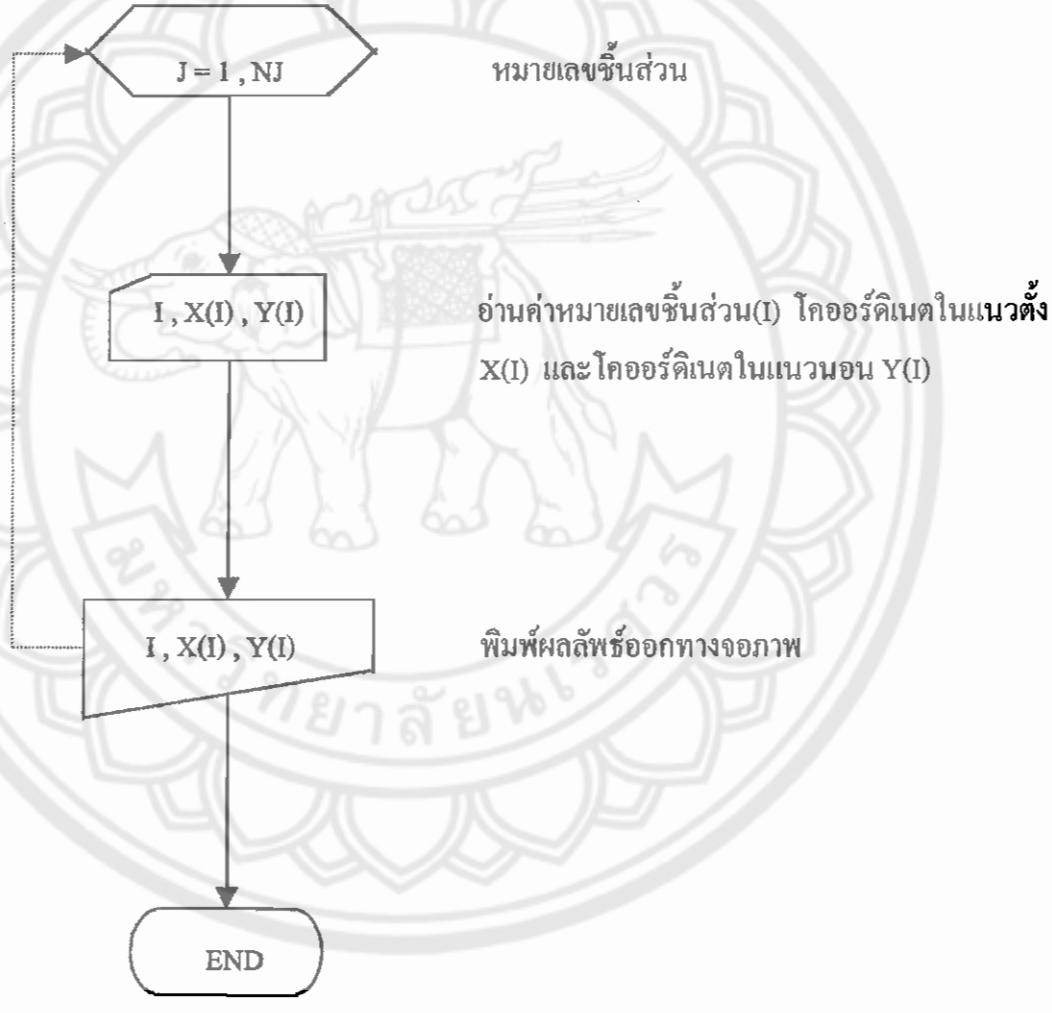
- 3 ก.ค. 2544
4440203

สำนักหอสมุด

3.2.4 Subprogram cmdJointCo

ทำหน้าที่รับค่าโคออร์ดิเนตของจุดต่อ นั้นคือ หมายเลขชิ้นส่วน โคออร์ดิเนตในแนวนอน(X) โคออร์ดิเนตในแนวตั้ง(Y) จากนั้นจะพิมพ์ผลลัพธ์ออกทางจอภาพ ดังแสดงใน Flow Chart 3

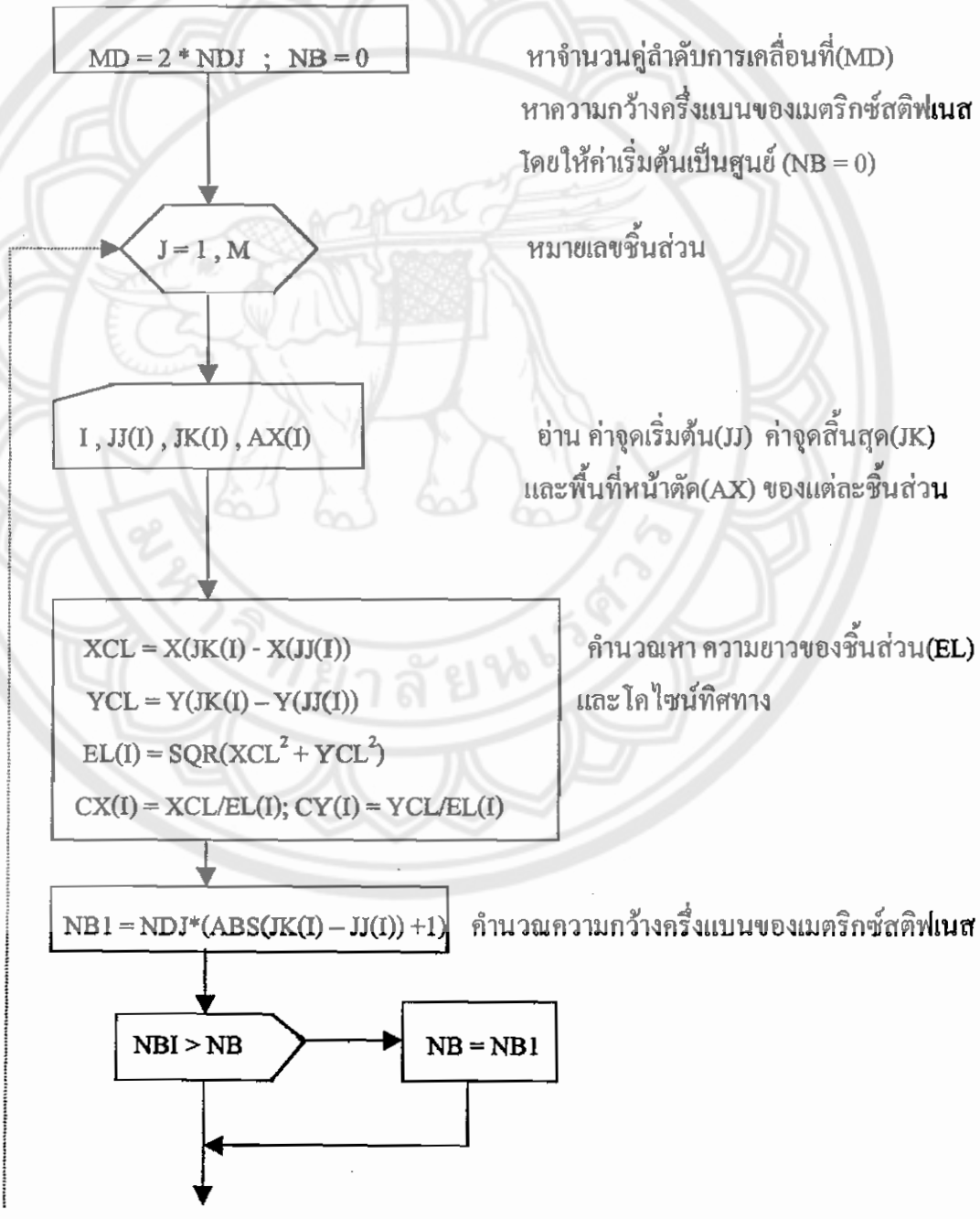
Flow Chart 3 : Subprogram cmdJointCo

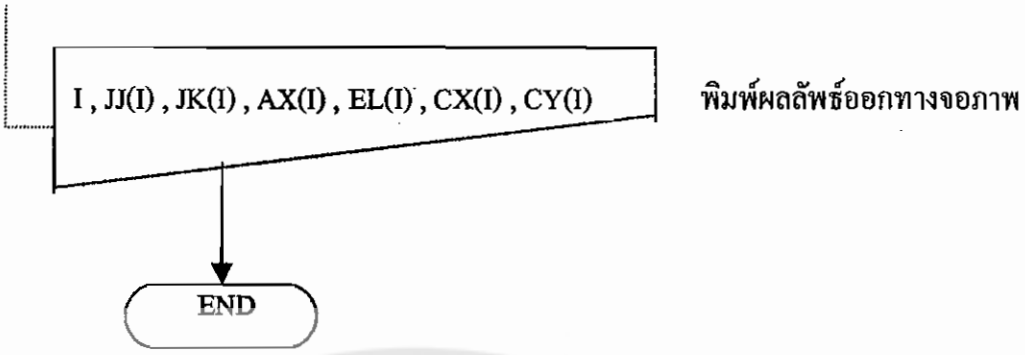


3.2.5 Subprogram cmdMemIn

ทำหน้าที่รับค่าข้อมูลเกี่ยวกับ หมายเลขชิ้นส่วน จุดต่อเริ่มต้น(JJ) จุดต่อสิ้นสุด(JK) พื้นที่หน้าตัด(AX) โคไซน์ทิศทางในแนวตั้ง(CX) และโคไซน์ทิศทางในแนวนอน(CY) จากนั้นจะพิมพ์ผลลัพธ์ออกทางหน้าจอ ดังแสดงใน Flow Chart 4

Flow Chart 4 : Subprogram cmdMemIn

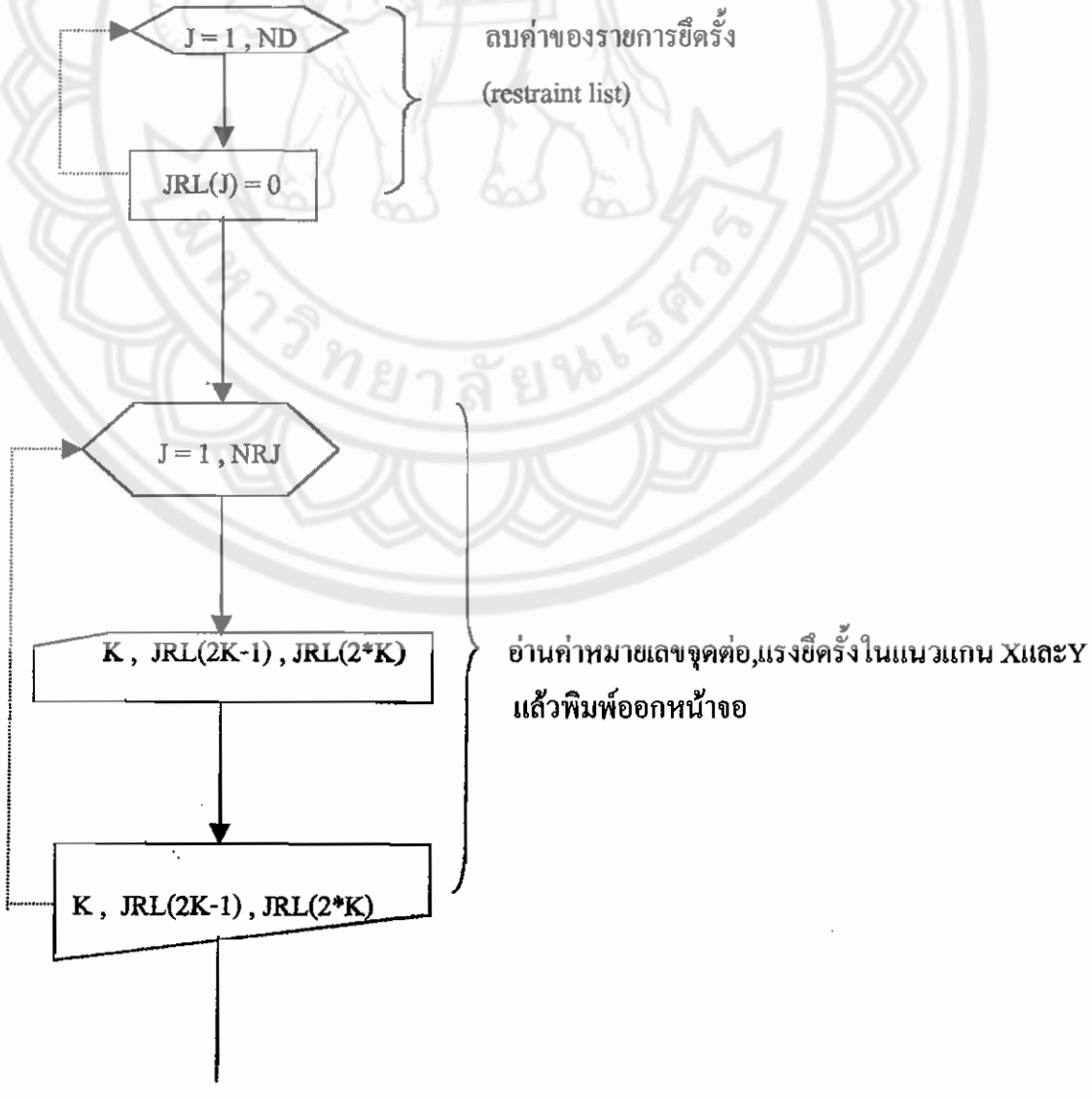


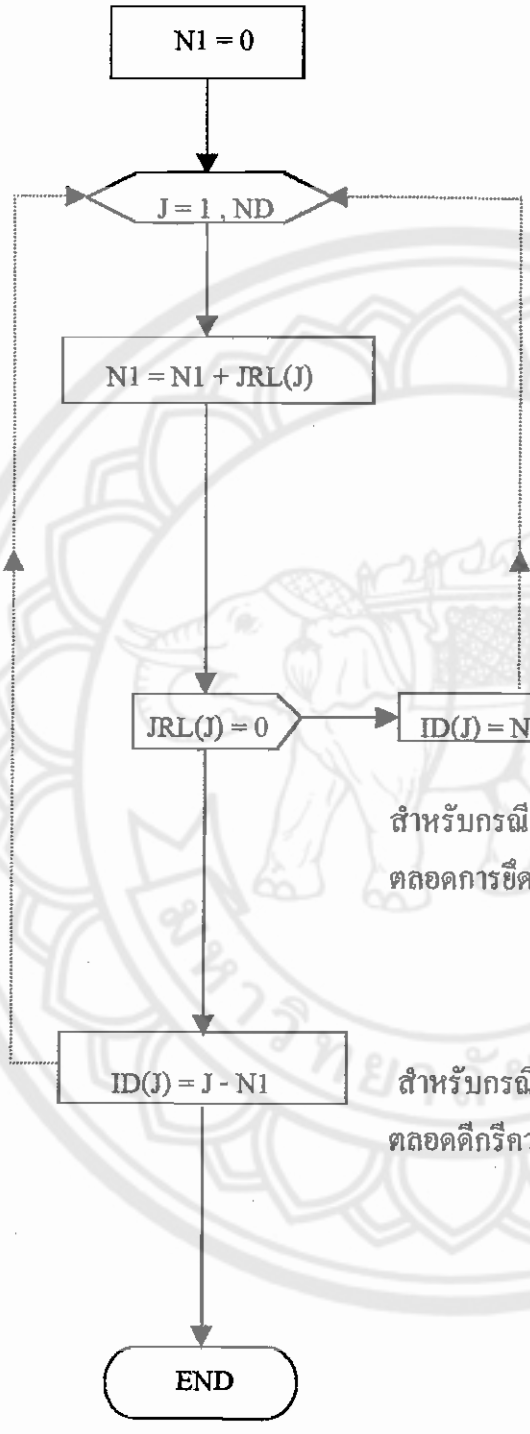


3.2.6 Subprogram JointRe

ทำหน้าที่รับค่าสภาพยึดรั้งของจุดต่อ และพิมพ์ค่าการยึดรั้งออกทางหน้าจอ จากนั้นจะทำการตั้งค่า Joint displacement index เพื่อแสดงหมายเลขที่มีการยึดรั้ง และหมายเลขที่ไม่มีการยึดรั้ง ดังแสดงใน Flow Chart 5

FIow Chart 5 : Subprogram cmdJointRe

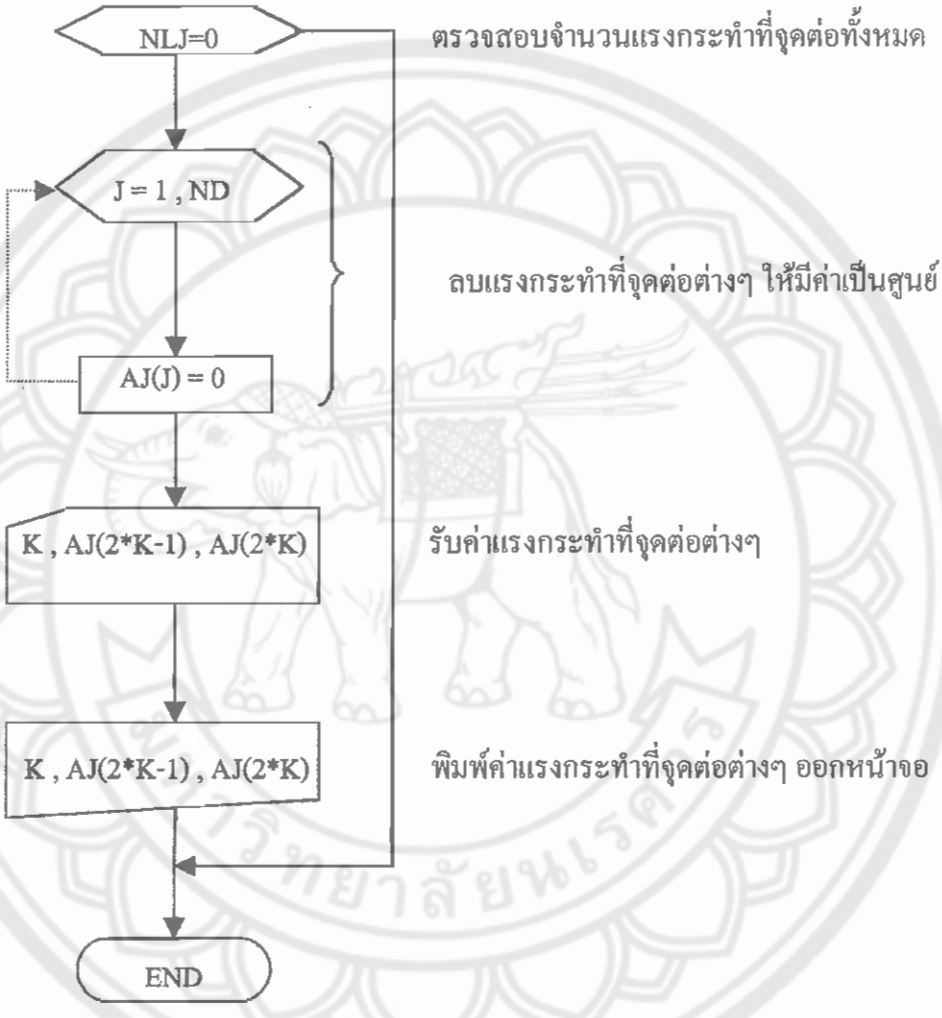



N1 = 0
ตั้งค่าเริ่มแรกให้ N1 = 0
J = 1 , ND
N1 = N1 + JRL(J)
หาจำนวนยึดรั้งสะสม
JRL(J) = 0
ID(J) = N + N1
สำหรับกรณีมีการยึดรั้ง ตั้งค่า joint displacement index
ตลอดการยึดรั้งที่ฐานรองรับ
ID(J) = J - N1
สำหรับกรณีไม่มีการยึดรั้ง ตั้งค่า joint displacement index
ตลอดดีกรีความอิสระ
END

3.2.7 Subprogram cmdLoadD

ทำหน้าที่รับแรงที่กระทำที่จุดต่อดังแสดงใน Flow Chart 6

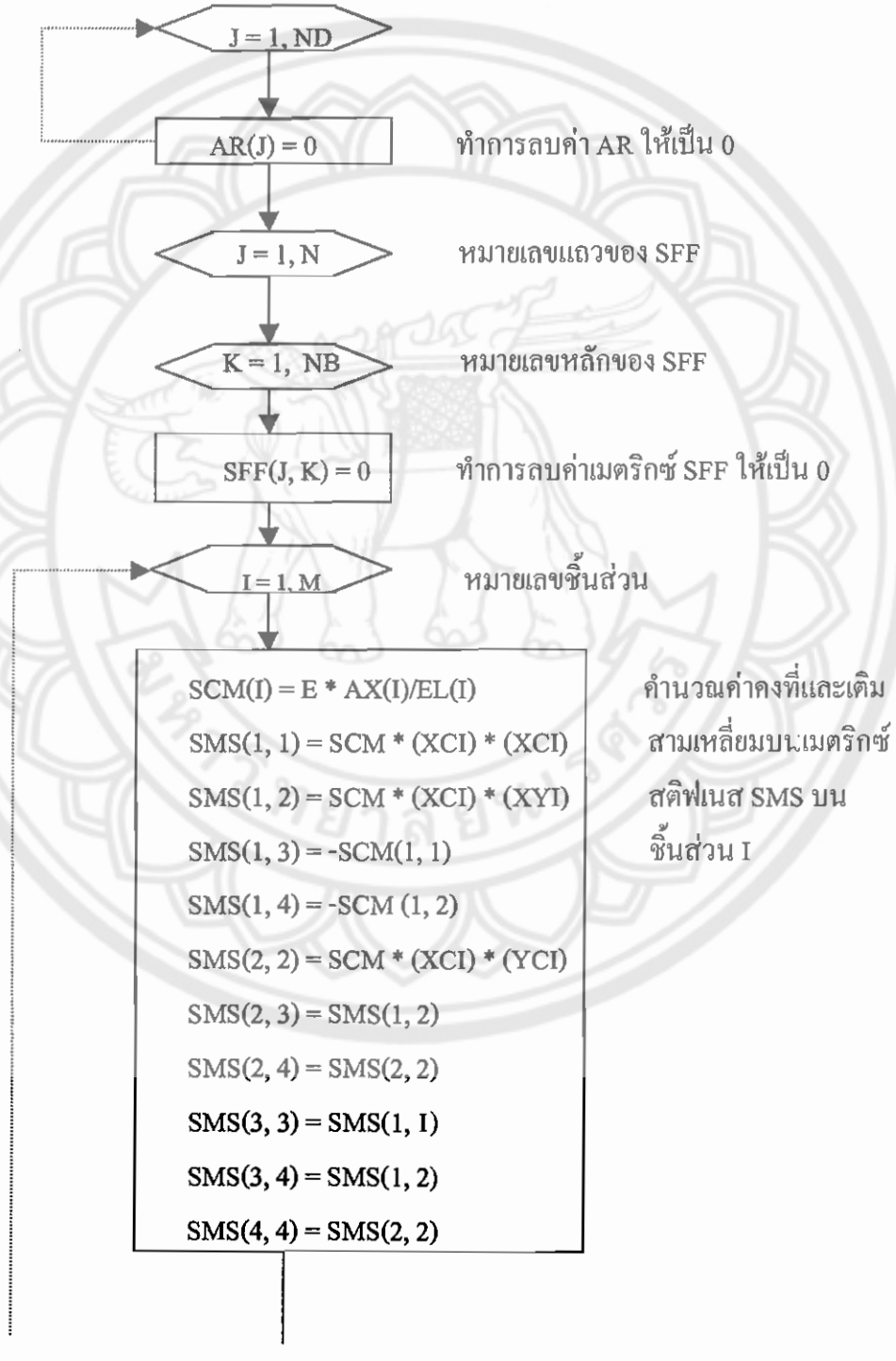
Flow Chart 6 : Subprogram cmdLoadD

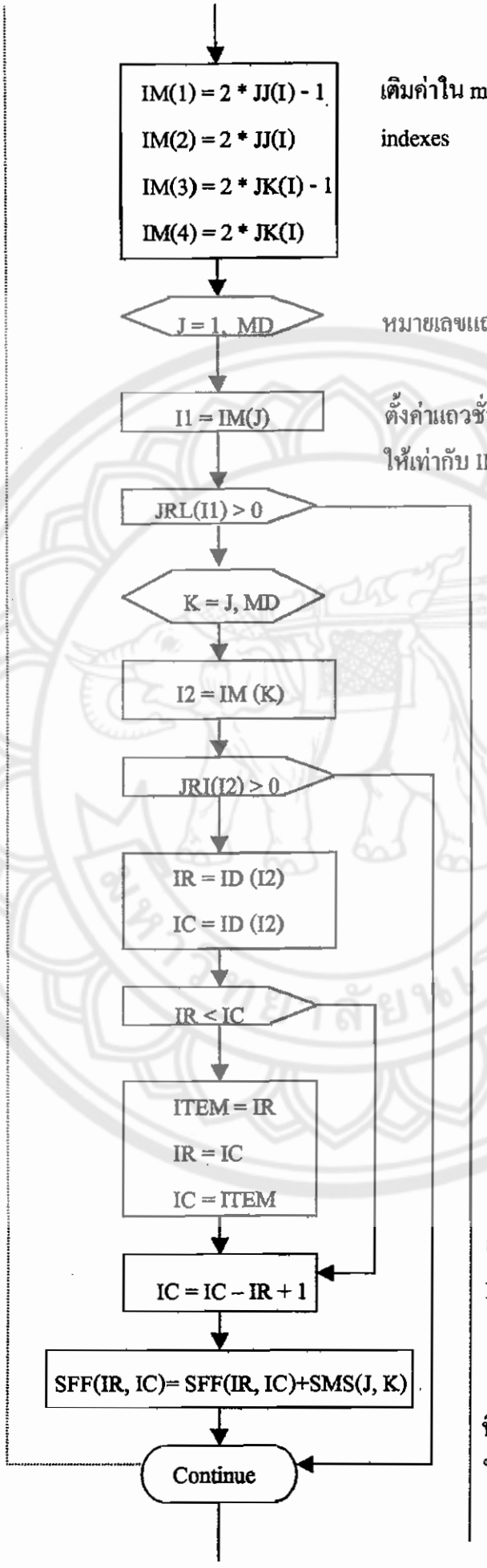


3.2.8 Subprogram cmdResult

ทำหน้าที่วิเคราะห์โครงสร้างโดยภายในโปรแกรมจะมีการเรียกใช้Subprogram BANFAC และSubprogram BANSOL จากนั้นจะทำการคำนวณหาค่า Joint- Displacement Member–end Action และSupport Reaction โดยมีขั้นตอนดังแสดงใน Flow Chart 10

Flow Chart 10 : Subprogram and Result





เติมค่าใน member displacement indexes

หมายเลขแถวของ SMS

ตั้งค่าแถวชั่วคราว T1 สำหรับเมตริกซ์ SFF ให้เท่ากับ IM(I)

ตรวจสอบสำหรับการยึดรั้ง ถ้าเท่ากับ 1 ให้ข้ามไปทำขั้นตอนอื่น

หมายเลขของคอลัมน์ SMS

ตั้งคอลัมน์ชั่วคราว I2 สำหรับเมตริกซ์ SFF เท่ากับ IM (K)

ตรวจสอบค่ายึดรั้งที่สอดคล้องกับ I2

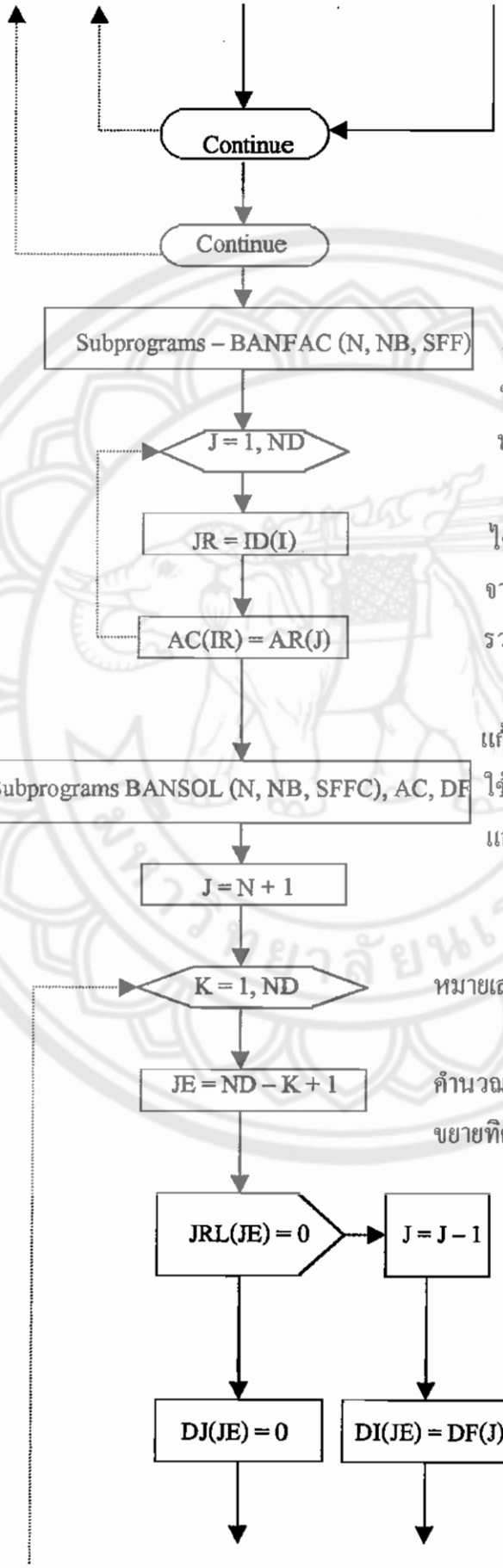
ได้รับค่าแถวจริงและคอลัมน์จริง (IR และ IC) จาก ID

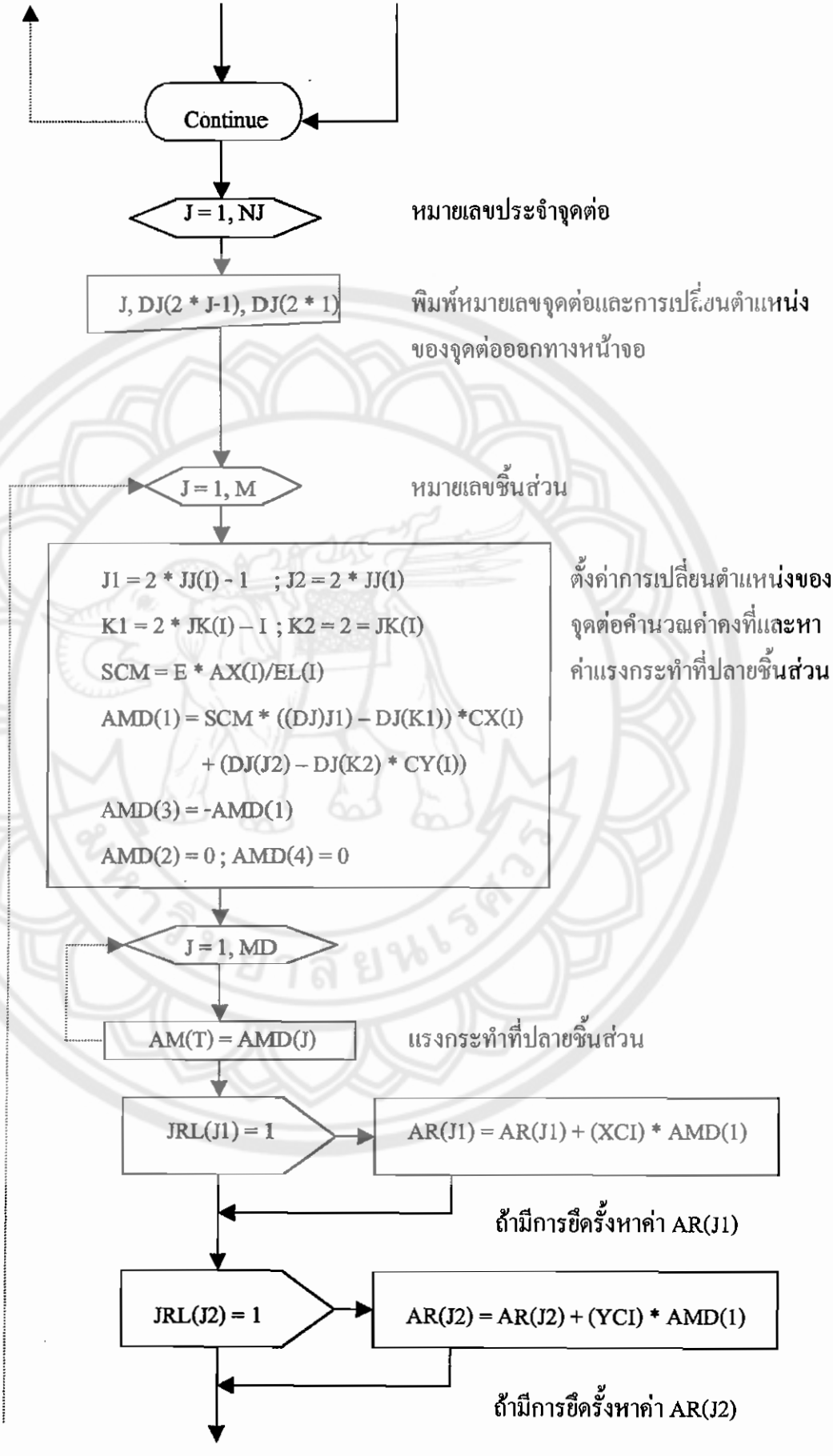
ตรวจสอบตำแหน่งสามเหลี่ยมบน ถ้าใช่ให้ข้ามไป

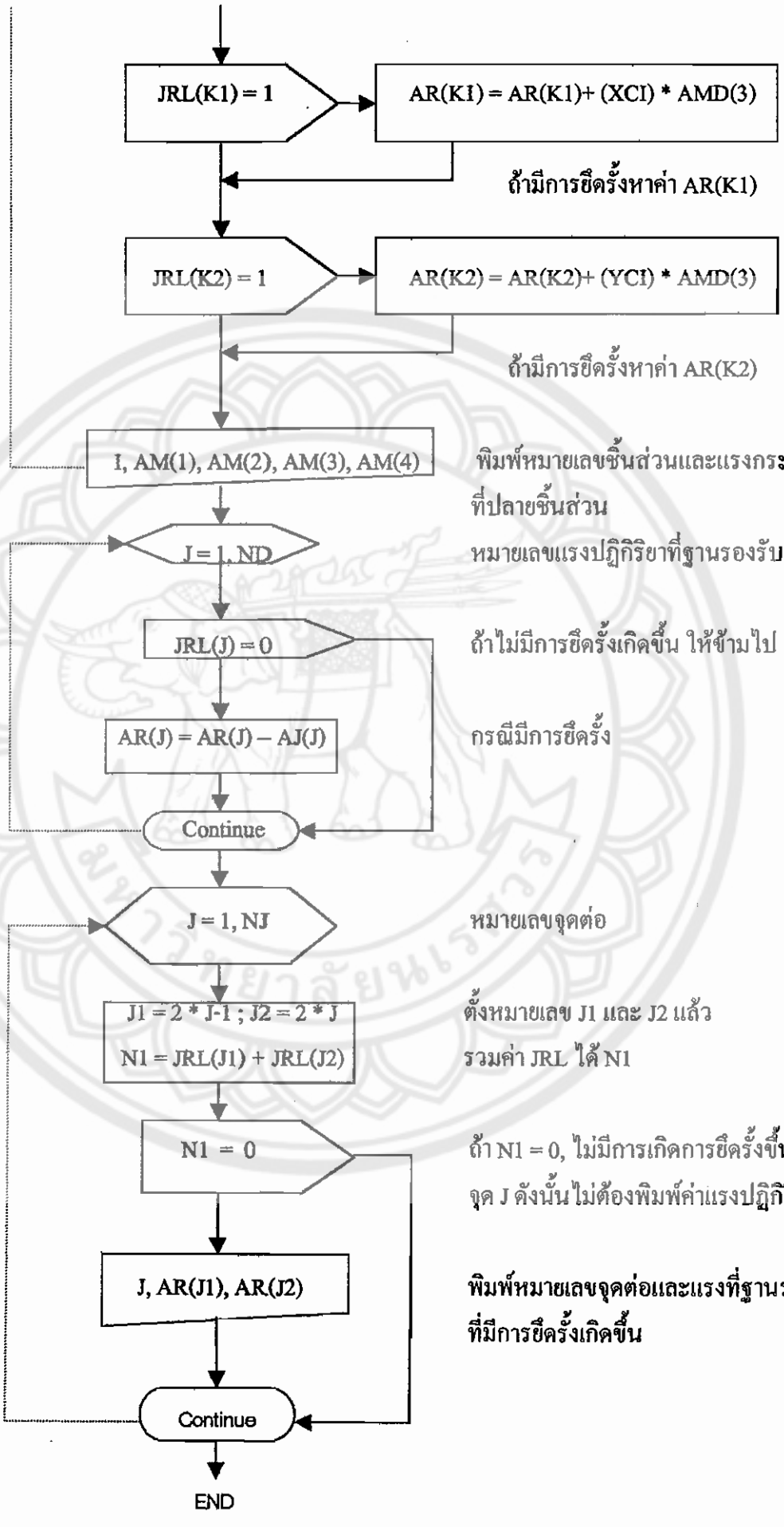
สำหรับตำแหน่งสามเหลี่ยมล่างให้เปลี่ยนเป็นตำแหน่งสามเหลี่ยมบน

ยกเลิกหมายเลขคอลัมน์สำหรับการเก็บ Band ส่วนบนของเมตริกซ์ SFF เป็นอาเรย์สี่เหลี่ยมผืนผ้า

ย้ายสมาชิกของ SMS ไปตำแหน่งที่เหมาะสม ในเมตริกซ์ SFF โดยให้เป็น IR และ IC


Continue
Continue
Subprograms – BANFAC (N, NB, SFF)
แยกตัวประกอบของ band ส่วนบน
ของเมตริกซ์ SFF
J = 1, ND
หมายเลขแรงกระทำที่จุดต่อ
JR = ID(I)
ได้หมายเลข JR สำหรับการจัดเรียงใหม่
จาก ID
AC(IR) = AR(J)
รวมแรงที่กระทำที่จุดต่อจริง
แก้สมการสำหรับ free joint displacement(DF)
Subprograms BANSOL (N, NB, SFFC), AC, DF
ใช้การแยกตัวประกอบเมตริกซ์ SFF
และสมาชิก N ตัวแรกในทิศทาง AC
J = N + 1
K = 1, ND
หมายเลขการเปลี่ยนตำแหน่งของจุดต่อ
JE = ND – K + 1
คำนวณค่าได้มาของหมายเลข JE สำหรับการ
ขยายทิศทางการเปลี่ยนตำแหน่ง
JRL(JE) = 0
J = J – 1
สำหรับกรณีไม่มีจุดยึดรั้ง ลดค่า J
ลง 1 เพื่อบอกตำแหน่งสมาชิก
ของ DF
DJ(JE) = 0
DI(JE) = DF(J)
ย้ายสมาชิกของ DF ไปยังตำแหน่ง
ที่เหมาะสมใน DJ


Continue
J = 1, NJ
หมายเลขประจำจุดต่อ
J, DJ(2 * J-1), DJ(2 * 1)
พิมพ์หมายเลขจุดต่อและการเปลี่ยนตำแหน่ง
ของจุดต่อออกทางหน้าจอ
J = 1, M
หมายเลขชิ้นส่วน
J1 = 2 * JJ(I) - 1 ; J2 = 2 * JJ(1)
K1 = 2 * JK(I) – I ; K2 = 2 = JK(I)
SCM = E * AX(I)/EL(I)
AMD(1) = SCM * ((DJ)J1) – DJ(K1)) *CX(I)
+ (DJ(J2) – DJ(K2) * CY(I))
AMD(3) = -AMD(1)
AMD(2) = 0 ; AMD(4) = 0
ตั้งค่าการเปลี่ยนตำแหน่งของ
จุดต่อคำนวณค่าคงที่และหา
ค่าแรงกระทำที่ปลายชิ้นส่วน
J = 1, MD
AM(T) = AMD(J)
แรงกระทำที่ปลายชิ้นส่วน
JRL(J1) = 1
AR(J1) = AR(J1) + (XCI) * AMD(1)
ถ้ามีการยึดรั้งหาค่า AR(J1)
JRL(J2) = 1
AR(J2) = AR(J2) + (YCI) * AMD(1)
ถ้ามีการยึดรั้งหาค่า AR(J2)

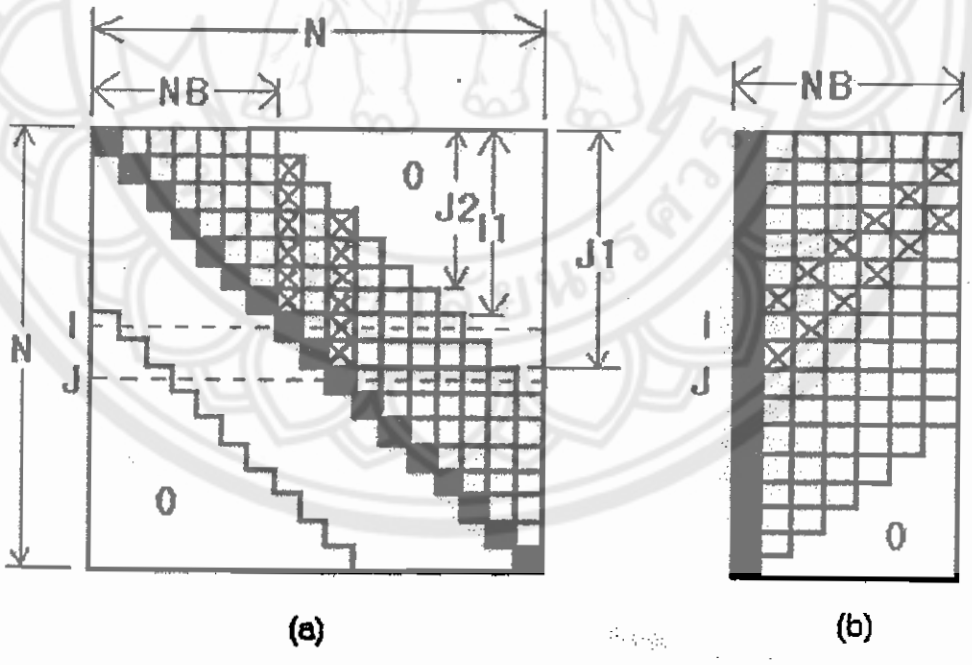

JRL(K1) = 1
AR(K1) = AR(K1)+ (XCI) * AMD(3)
ถ้ามีการยึดรั้งหาค่า AR(K1)
JRL(K2) = 1
AR(K2) = AR(K2)+ (YCI) * AMD(3)
ถ้ามีการยึดรั้งหาค่า AR(K2)
I, AM(1), AM(2), AM(3), AM(4)
พิมพ์หมายเลขชิ้นส่วนและแรงกระทำภายใน
ที่ปลายชิ้นส่วน
J = 1, ND
หมายเลขแรงปฏิกิริยาที่ฐานรองรับ
JRL(J) = 0
ถ้าไม่มีการยึดรั้งเกิดขึ้น ให้ข้ามไป
AR(J) = AR(J) – AJ(J)
กรณีมีการยึดรั้ง
Continue
J = 1, NJ
หมายเลขจุดต่อ
J1 = 2 * J-1 ; J2 = 2 * J
N1 = JRL(J1) + JRL(J2)
ตั้งหมายเลข J1 และ J2 แล้ว
รวมค่า JRL ได้ N1
N1 = 0
ถ้า N1 = 0, ไม่มีการเกิดการยึดรั้งขึ้นที่
จุด J ดังนั้นไม่ต้องพิมพ์ค่าแรงปฏิกิริยา
J, AR(J1), AR(J2)
พิมพ์หมายเลขจุดต่อและแรงที่ฐานรองรับ
ที่มีการยึดรั้งเกิดขึ้น
Continue
END

3.3 การเขียนโปรแกรมแก้ระบบสมการเชิงเส้น (K * u = P)

3.3.1 Subprogram BANFAC

เนื่องจากเมตริกซ์สติฟเนสที่ได้มีลักษณะสมมาตรและเป็น banded matrix ดังนั้นเราจึงทำการเขียนโปรแกรม เพื่อที่จะนำ banded matrix มาแยกเป็นสมการ 2.20 แล้วทำการหาค่าของเมตริกซ์ $[\overline{U}]$ โดยโปรแกรมนี้จะถูกเรียกใช้ในโปรแกรมย่อย cmdResult ซึ่งวิธีการแยกตัวประกอบแบบ banded matrix จะมีประสิทธิภาพดีกว่าแบบเติมเป็นอาเรย์ เพราะว่าจะไม่มีการคำนวณใน ส่วนที่อยู่นอก band รูป 3.1a แสดงรูปร่างโดยทั่วไปของ banded symmetric matrix สัญลักษณ์ NB ที่แสดงในรูปแทนความกว้างครึ่ง band (semi – band width) และ N คือ ขนาดของเมตริกซ์ มี เพียงส่วนบนของ band (รวมทั้งส่วนที่อยู่ในแนวทแยงมุม) ที่เราจะทำการเก็บข้อมูลไว้แสดงด้วย สี่เหลี่ยมจัตุรัสเล็ก ๆ ดังรูป 3.1a

รูปแบบการเก็บ band ครึ่งบนตามแนวทแยงมุมของ matrix แสดงในรูป 3.1b ในการจัดเรียงแบบที่จะจัดเรียงเป็นอาเรย์สี่เหลี่ยมผืนผ้า โดยมีค่าตามแนวทแยงมุมอยู่ในคอลัมน์แรก



รูป 3.1 Banded matrix : (a) usual form of storage and
(b) upper band stroed as a rectangular array

ดังแสดงใน Flow Chart 1 แสดงขั้นตอนของโปรแกรมย่อย ที่องค์ประกอบ band ส่วนบนของเมตริกซ์สมมาตรถูกเก็บไว้เป็นอาเรย์สี่เหลี่ยมผืนผ้า ชื่อของโปรแกรมย่อยคือ

BANFAC (N, NB, A)

N เป็นเลขจำนวนเต็ม หมายถึง ขนาดของเมตริกซ์

NB เป็นเลขจำนวนเต็ม หมายถึง ความกว้างครึ่ง band

A หมายถึง เมตริกซ์สมมาตรซึ่งมีสมาชิกเป็นจำนวนจริง

นอกจากนี้ยังมีสัญลักษณ์ J2 ซึ่งถูกนำเข้ามาด้วยจุดประสงค์ที่จะคำนวณขอบเขตขององค์ประกอบที่ไม่เป็นศูนย์ (non – zero element) เมื่อคอลัมน์ J มีค่าเกินความกว้างครึ่งแบบ (NB) ในรูป 3.1a เทอมแรกที่ไม่ใช้ศูนย์ในคอลัมน์นั้นจะมีหลายเลขแถว

$$J2 = J - NB + 1 (NB \leq J \leq N) \qquad (3.1)$$

นอกจากนี้สำหรับ NB คอลัมน์แรก (ยกเว้นคอลัมน์ 1)

$$J2 = 1 \qquad (1 < J \leq NB) \qquad (3.2)$$

ลำดับของการปฏิบัติการใน Subprogram BANFAC เป็นไปตาม Flow Chart 1

เมื่อ Subprogram ทำการแยกตัวประกอบเมตริกซ์ $[\overline{U}]$ จะถูกเก็บไว้ในเมตริกซ์ A และนอกจากนี้องค์ประกอบในแนวทแยงมุม Dii จะถูกเก็บไว้ในคอลัมน์ A เพื่อประโยชน์ในการคำนวณ

3.3.2 Subprogram BANSOL

เป็นโปรแกรมย่อยที่ประยุกต์ใช้กับ banded matrix โปรแกรมนี้จะได้รับค่า Upper band ของเมตริกซ์ $[\overline{U}]$ จากโปรแกรมย่อย BANFAC และทำการแก้สมการหาตัวไม่ทราบค่าในระบบเดิมของสมการ ชื่อของโปรแกรมย่อยนี้คือ

BANSOL (N , NB, U, B, X)

โดยสัญลักษณ์ที่เพิ่มขึ้นมามีคำนิยามดังนี้

U หมายถึง เมตริกซ์ที่ได้มาจาก Subprogram BANFAC

B และ X หมายถึง เวกเตอร์จริง (real vectors) ของเทอมที่มีค่าคงที่และตัวไม่ทราบค่าตามลำดับ

ใน Flow Chart 2 แสดงขั้นตอนของ Subprogram BANSOL ในการแทนค่า ไปข้างหน้าและแทนค่ากลับ (ดูสมการ 2.23 และ 2.25) สัญลักษณ์ J จะถูกใช้วิเคราะห์เทอมที่

ไม่ เป็นศูนย์ที่สรุปไว้ในการคำนวณ ในการแทนค่าไปข้างหน้าเลขประจำแถวสำหรับค่าที่ไม่เป็นศูนย์ ค่าแรกในคอลัมน์ I คือ

$$J = 1 - NB + 1 \qquad (NB < I \leq N) \qquad (3.3)$$

นอกจากนี้สำหรับคอลัมน์ NB แรก (ยกเว้นคอลัมน์ 1)

$$J = 1 \qquad (1 < I \leq B) \qquad (3.4)$$

คล้ายคลึงกับการแทนค่าย้อนกลับเลขประจำคอลัมน์สำหรับส่วนที่ไม่เป็นศูนย์สุดท้ายในแถว I ของเมตริกซ์ $[\overline{U}]$ คือ

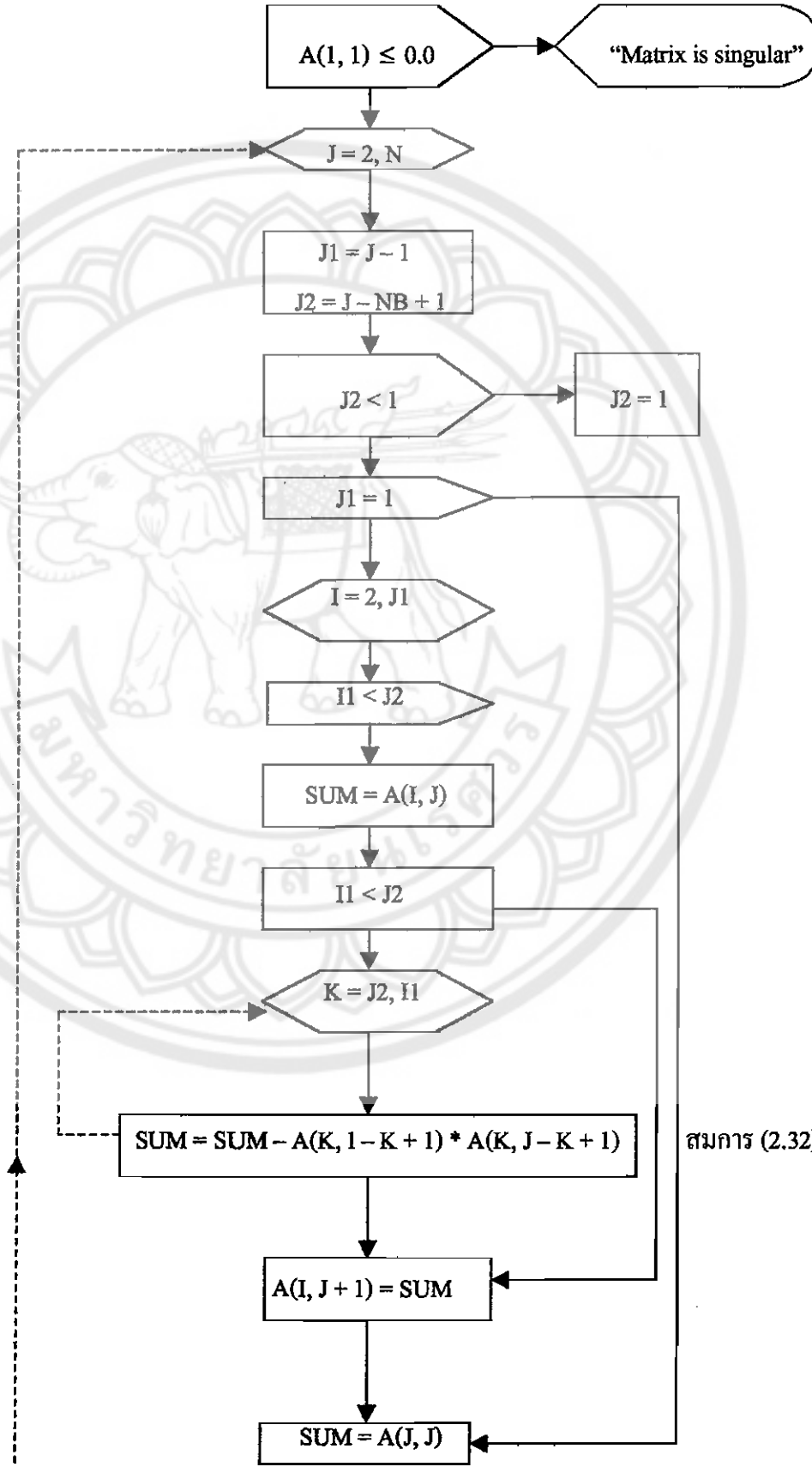
$$J = I + NB - 1 \qquad (1 < I \leq B) \qquad (3.5)$$

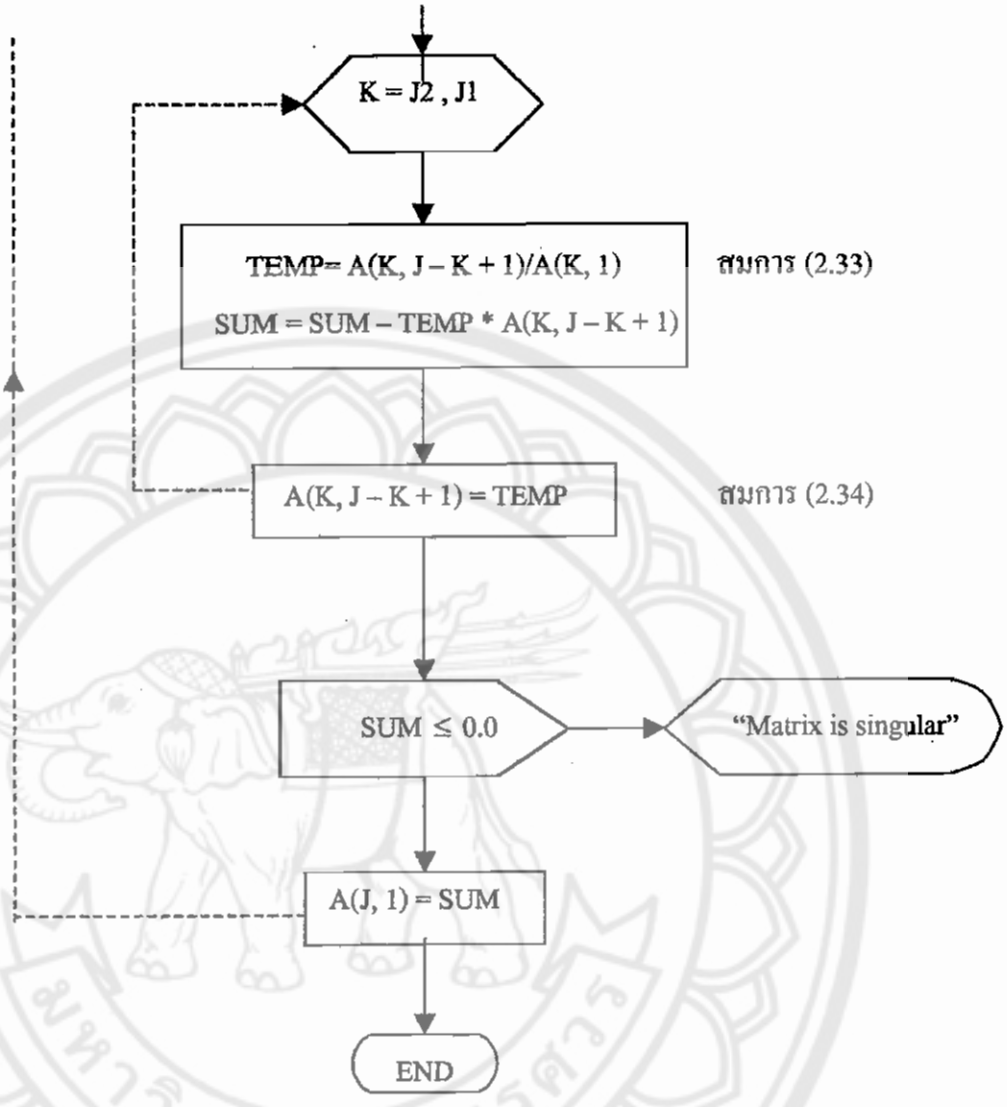
นอกจากนี้สำหรับแถว NB สุดท้าย (ยกเว้นแถว N)

$$J = N \qquad [(N - NB) < 1 < N] \qquad (3.6)$$

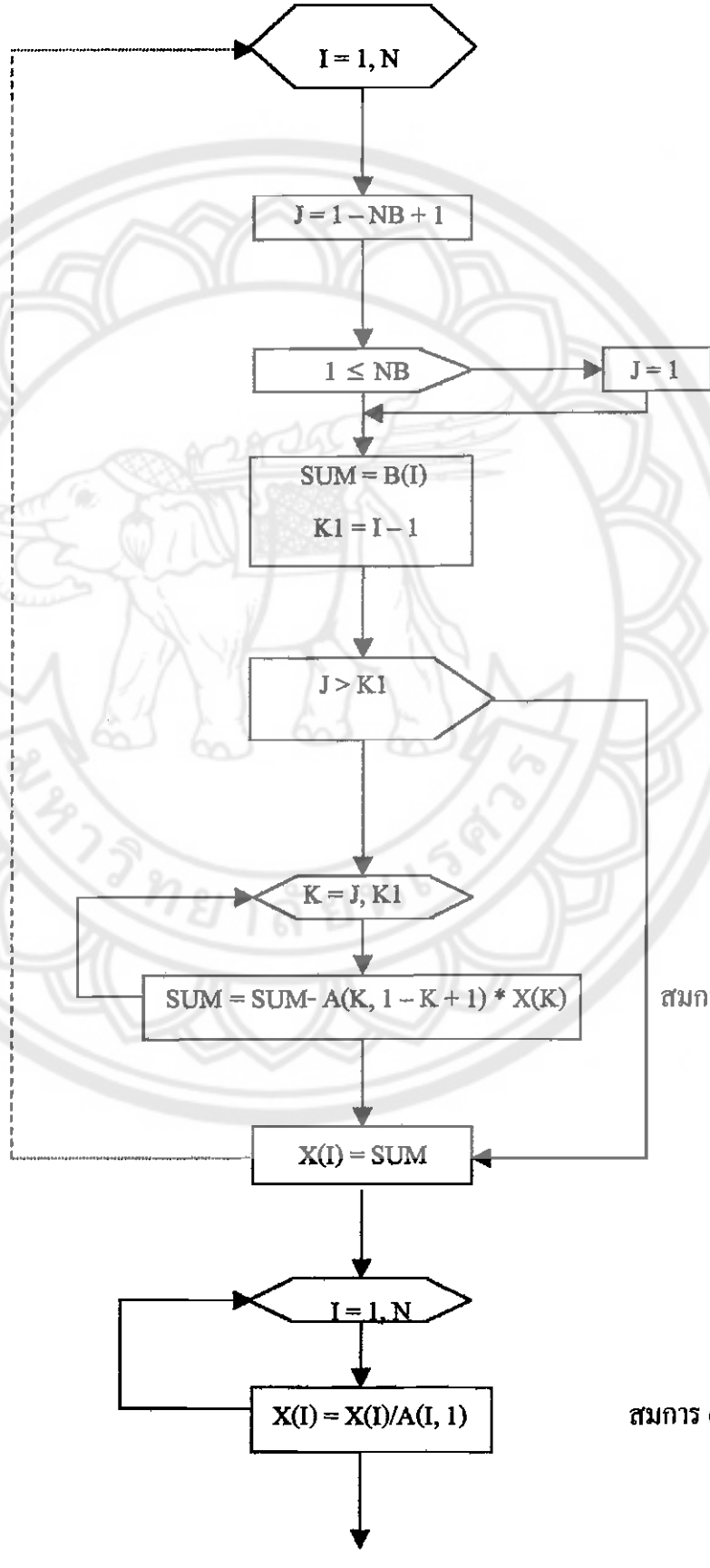
เลขประจำคอลัมน์ขององค์ประกอบในเมตริกซ์ $[\overline{U}]$ จะแสดงตำแหน่งจริงในอาเรย์สี่เหลี่ยมผืนผ้าเมื่อโปรแกรมทำการคำนวณค่าที่อยู่ระหว่าง z และ r จะเปลี่ยนไปอยู่ในค่าของ X และค่าสุดท้ายของ X จะถูกคำนวณโดยแทนค่ากลับ

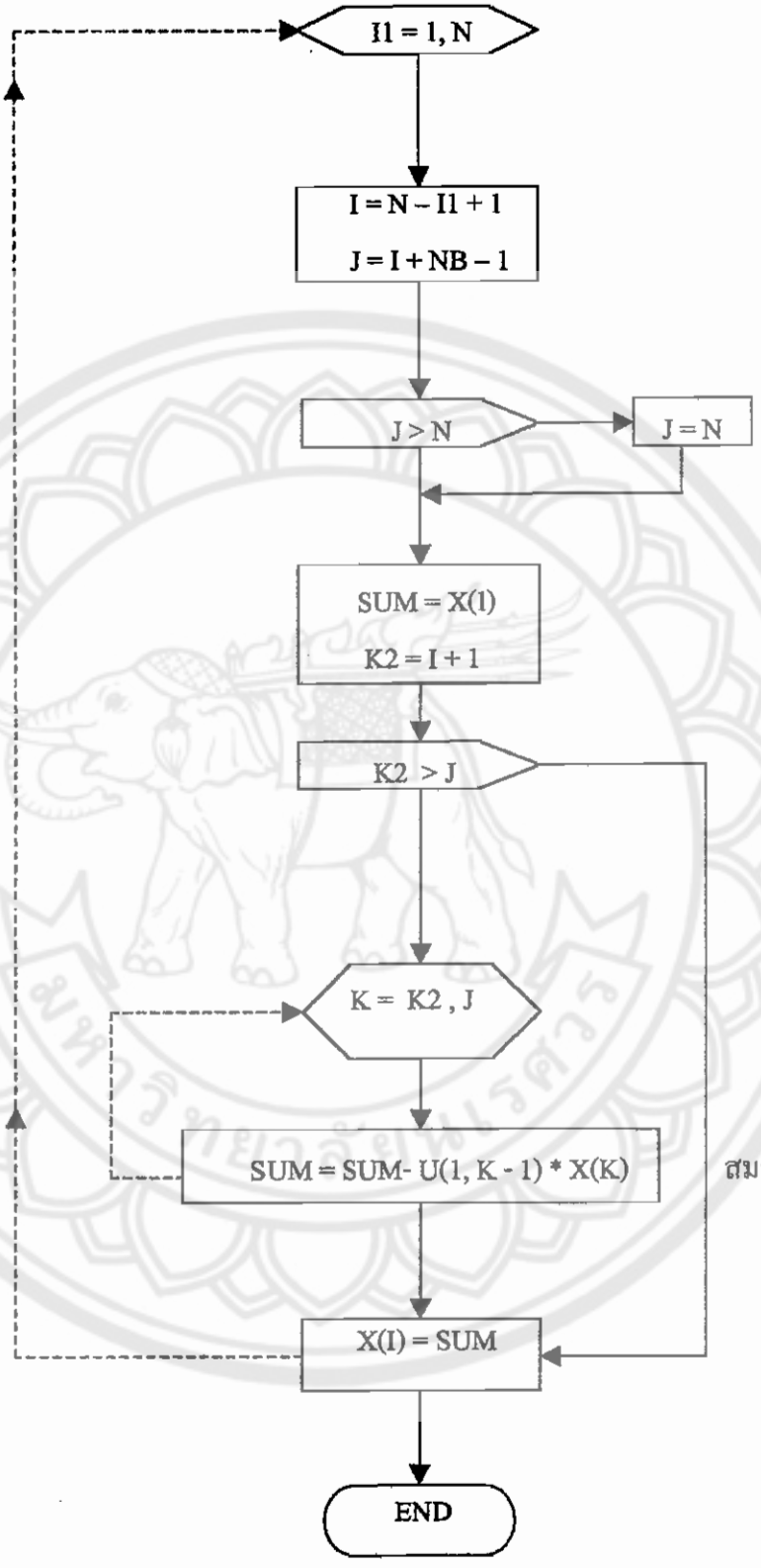
Flow Chart 12 : Subprogram BANFAC




K = J2 , J1
TEMP= A(K, J – K + 1)/A(K, 1)
สมการ (2.33)
SUM = SUM – TEMP * A(K, J – K + 1)
A(K, J – K + 1) = TEMP
สมการ (2.34)
SUM ≤ 0.0
"Matrix is singular"
A(J, 1) = SUM
END

Flow Chart 13 : Subprogram BANSOL




I1 = 1, N
I = N – I1 + 1
J = I + NB – 1
J > N
J = N
SUM = X(I)
K2 = I + 1
K2 > J
K = K2 , J
SUM = SUM- U(1, K - 1) * X(K)
สมการ (2.45)
X(I) = SUM
END