บทที่ 3
การเขียนโปรแกรม

3.1บทนำ

จากที่กล่าวมาแล้วว่า ในส่วนของโปรแกรมเราได้แบ่งเป็น 2 ทฤษฎีหลัก ๆ ดังนั้นในด้านการเขียนโปรแกรมจึงจะขอแยกการอธิบายเป็น 2 หัวข้อ คือ

1. การเขียนโปรแกรมแก้ปัญหาคานต่อเนื่องโดยใช้วิธีการรวมสติฟเนสโดยตรง
2. การเขียนโปรแกรมแก้สมการ K * u = P โดยใช้ modified Cholesky method

3.2 การเขียนโปรแกรมวิเคราะห์คานต่อเนื่อง

ในการเขียนโปรแกรมโดยใช้ Visual BASIC 6.0 จะทำการเขียนโปรแกรมเชิงวัตถุ (Object Oriented Programming) นั่นคือ จะเขียนโปรแกรมตามปุ่มคำสั่งที่จะตอบสนองต่อเหตุการณ์ต่าง ๆ ดังนั้น Flow Chart ในโปรแกรมนี้จะแบ่งตามปุ่มคำสั่งที่จะตอบสนองกับผู้ใช้

3.2.1 สัญลักษณ์สำคัญที่ใช้ใน Program

สัญลักษณ์	คำนิยาม
A	ระยะทางที่แรงกระทำบนชิ้นส่วน
AC ( )	Combined joint loads A_C (ทิศทางตาม global coordinate)
AE ( )	Equivalent joint loads A_E (ทิศทางตาม global coordinate)
AJ ( )	แรงภายนอกกระทำที่ Joint (ทิศทางตาม global coordinate)
AM ( )	แรงภายในที่ปลายของชิ้นส่วนแต่ละชิ้น
AMD ( )	แรงกระทำที่ปลายของชิ้นส่วนที่มีการยึดรั้งเนื่องจากแรงกระทำภายนอก (ทิศทางตาม local coordinate)
AML ( )	ค่าของแรงยึดปลายของชิ้นส่วนอันเนื่องมาจากกรณีที่มีน้ำหนักบรรทุกกระทำบนชิ้นส่วน หรือ เนื่องจากสาเหตุอื่น ๆ ทุกชนิด โดยถือว่าปลายทั้งสองข้างของชิ้นส่วนเป็นแบบยึดตรึงแน่น

สัญลักษณ์	คำนิยาม
AR ( )	แรงปฏิกิริยาในแนวต่าง ๆ
D	ระยะทางที่ moment กระทำบนชิ้นส่วน
DF ( )	การเคลื่อนที่ของจุดต่ออิสระ (ทิศทางตาม global coordinate)
DJ ( )	การเคลื่อนที่ของจุดต่อ (ทิศทางตาม global coordinate)
E	ค่า modulus of elasticity
EL ( )	ความยาวของชิ้นส่วน
I	หมายเลขประจำชิ้นส่วน
ID	Displacement indexes สำหรับจุดต่อ
IM	Displacement indexes สำหรับชิ้นส่วน
IR, IC	หมายเลขประจำแถวและคอลัมน์ตามลำดับ
J, K	หมายเลขประจำจุดต่อ
JRL ( )	รายการจุดยึดรั้ง
LML ( )	รายการของชิ้นส่วนที่มีแรงกระทำ
LN	หมายเลขของชุดแรงกระทำ
M	จำนวนชิ้นส่วนทั้งหมด
MD	จำนวนคู่ลำดับการเคลื่อนที่ของชิ้นส่วน (MD = 4)
MMT	ค่าของ moment load
N	จำนวนดีกรีอิสระ
NB	ความกว้างครึ่งแบนของเมตริกซ์สติฟเนส สำหรับคาน NB = 4
ND	จำนวนคู่ลำดับการเคลื่อนที่ของทุกจุดต่อ
NDJ	จำนวนการเคลื่อนที่ของจุดต่อทั้งหมด
NJ	จำนวนจุดต่อทั้งหมด
NLJ	จำนวนแรงกระทำที่จุดต่อทั้งหมด
NML	จำนวน moment load
NO	หมายเลขของโครงสร้างที่ตั้งไว้ (Structural No.)
NPL	จำนวน Point load
NR	จำนวนแรงยึดรั้งที่ support
NRJ	จำนวนจุดยึดรั้งทั้งหมด

QA
76.73
.B3
ก321 ก
2549

- 4 ก.ค. 2543
4340241

25
สำนักหอสมุด

คำนิยาม	สัญลักษณ์
NTL	จำนวน Triangular load
NUL	จำนวน Uniform load
P	ค่าของ Point load
RTT ( )	ค่าของการเคลื่อนที่รอบแกน Z มีหน่วยเป็นเรเดียน
SFF ( )	เมตริกซ์สติฟเนสของการเคลื่อนที่ที่จุดต่อิสระ
SM ( )	เมตริกซ์สติฟเนสของชิ้นส่วนสำหรับแกน member – oriented
TYP	ชนิดของ Triangular load
Uy ( )	การเคลื่อนที่ในแนวแกน Y
WT	ค่าของ Triangular load
WU	ค่าของ Uniform load
ZI ( )	Moment of inertia รอบแกน Z

3.2.2 ภาพรวมของโปรแกรม

โปรแกรมวิเคราะห์โครงสร้างนี้ประกอบไปด้วยโปรแกรมย่อยทั้งหมด 13 โปรแกรมแบ่งออกได้เป็น 3 กลุ่มใหญ่คือ

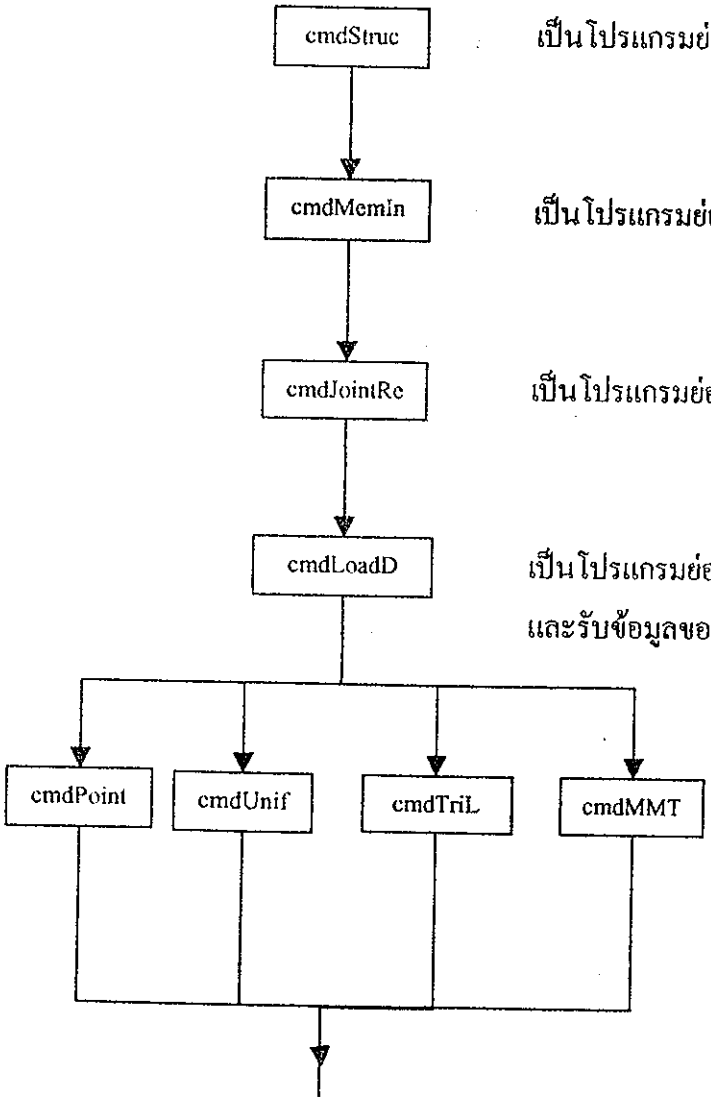
กลุ่มที่1 มีหน้าที่รับข้อมูล ได้แก่ Subprogram cmdStruc, cmdMemIn, cmdJointRe, cmdLoadD cmdPoint, cmdUnif, cmdTriL, cmdMMT

กลุ่มที่ 2 มีหน้าที่วิเคราะห์ผล ได้แก่ Subprogram cmdResult, BANFAC, BANSOL

กลุ่มที่ 3 มีหน้าที่ปลีกย่อยอื่น ๆ ได้แก่ Subprogram cmdPrint, cmdQuit

ลักษณะและลำดับของแต่ละโปรแกรมย่อยดังแสดงใน Flow chart 1

Flow Chart 1 : Main Program



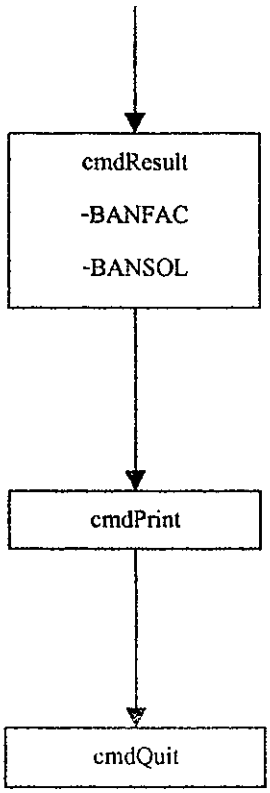
เป็นโปรแกรมย่อยรับข้อมูลของทั้งโครงสร้าง

เป็นโปรแกรมย่อยรับข้อมูลของชิ้นส่วนย่อย

เป็นโปรแกรมย่อยที่จะรับค่าสภาพยึดครั้งที่จุดต่อ

เป็นโปรแกรมย่อยที่จะรับค่าจำนวนแรงกระทำแบบต่าง และรับข้อมูลของแรงกระทำและการทรุดตัวที่จุดต่อ

เป็นโปรแกรมย่อยที่จะรับค่าของแรงแบบต่าง ๆ ที่กระทำระหว่างจุดต่อ



เป็นโปรแกรมย่อยที่ใช้ในการวิเคราะห์คานและแสดงผลลัพธ์ทางจอภาพ โดยภายในโปรแกรมจะมีการเรียกใช้โปรแกรมย่อยอีก 2 โปรแกรมคือ โปรแกรมย่อย BANFAC และโปรแกรมย่อย BANSOL

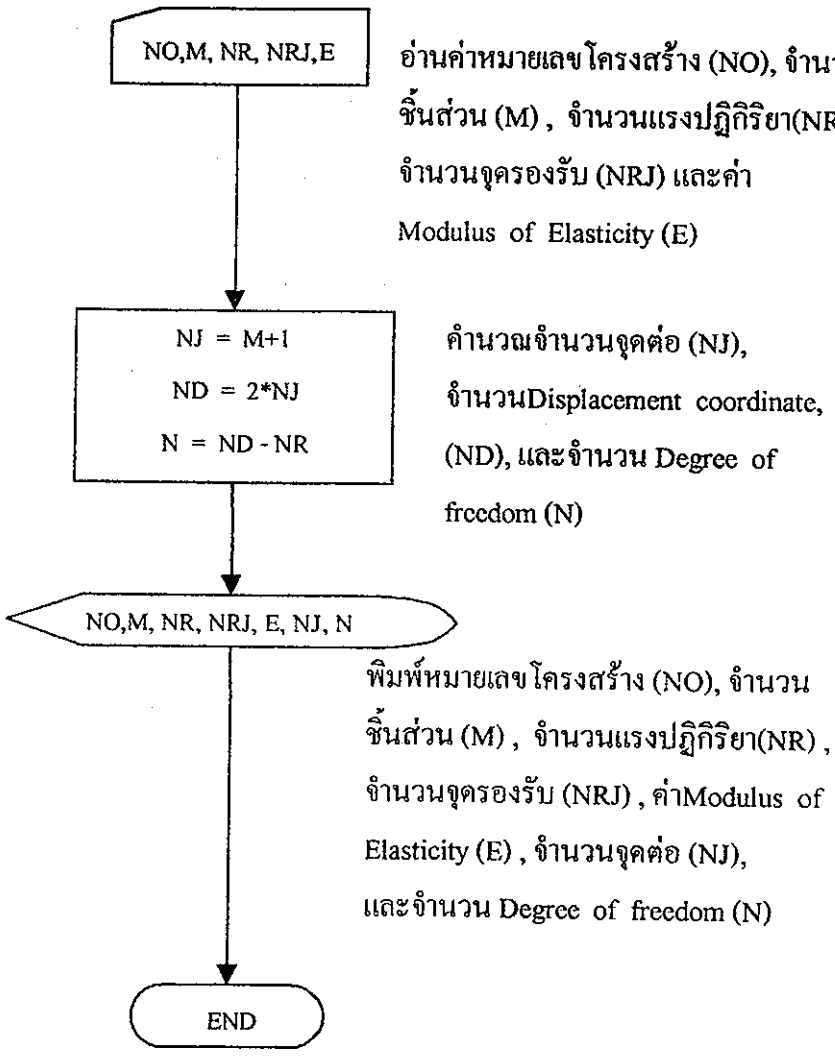
เป็นโปรแกรมย่อยที่ใช้ในการสั่งให้เครื่องพิมพ์ทำการพิมพ์คำตอบที่ได้ออกมา

เป็นโปรแกรมย่อยเพื่อที่เราจะออกจากตัวโปรแกรม

3.2.3 Subprogram cmdStruc

ทำหน้าที่รับค่าข้อมูลของโครงสร้าง นั่นคือจำนวนชิ้นส่วน (M) , จำนวนแรงปฏิกิริยา (NR) , จำนวนจุดรองรับ (NRJ) และค่า Modulus of Elasticity (E) จากนั้นนำค่า M และมาคำนวณหาค่า จำนวน degree of freedom (N) แล้วพิมพ์ทั้งหมดออกทางหน้าจอ ดังแสดงใน Flow Chart 2

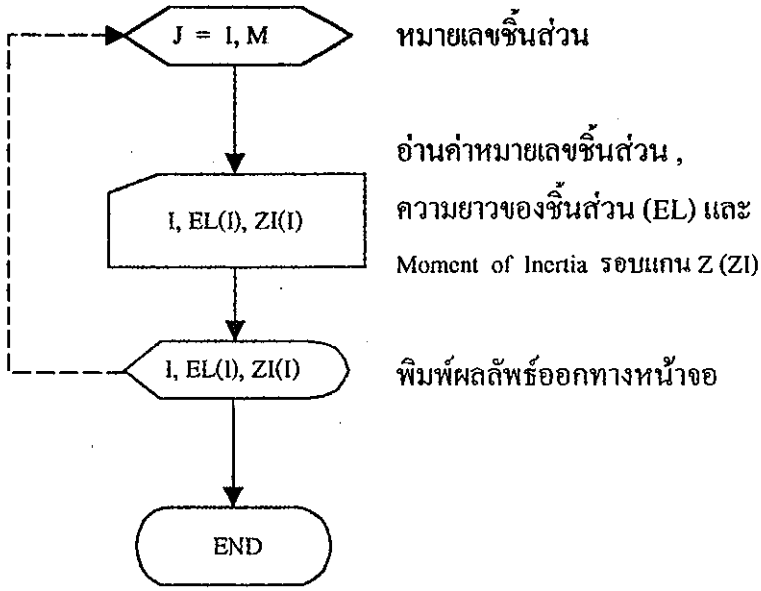
Flow Chart 2 : Subprogram cmdStruc



3.2.4 Subprogram cmdMenIn

ทำหน้าที่รับค่าข้อมูลของชิ้นส่วน นั้นคือ หมายเลขชิ้นส่วน, ความยาวของชิ้นส่วน (EL) และ Moment of Inertia รอบแกน Z (ZI) จากนั้นจะพิมพ์ผลลัพธ์ออกทางหน้าจอ ดังแสดงใน Flow Chart 3

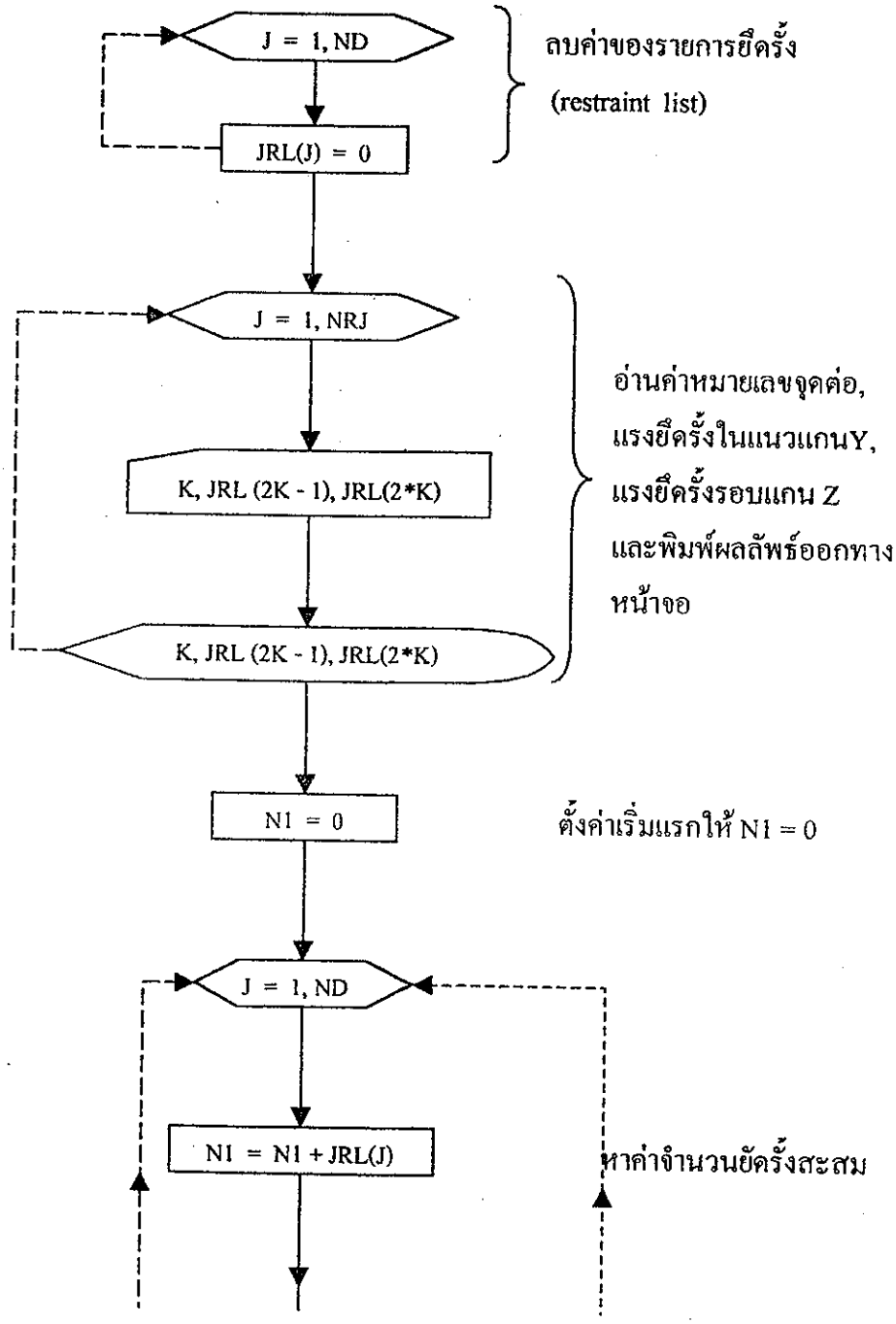
Flow Chart 3 : Subprogram cmdMenIn

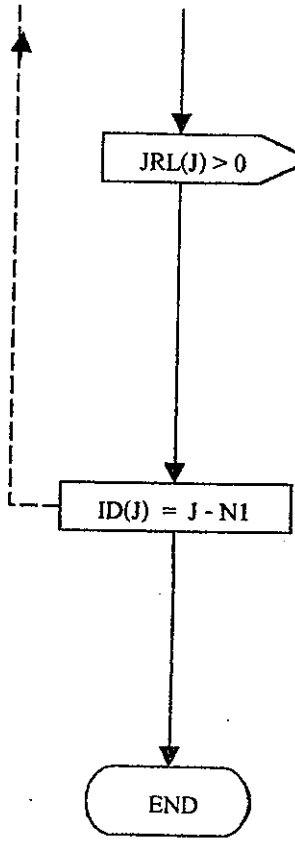


3.2.5 Subprogram cmdJointRe

ทำหน้าที่รับค่าสภาพยึดรั้งของจุดต่อ และพิมพ์ค่าการยึดรั้งออกทางหน้าจอ จากนั้นจะทำการตั้งค่า Joint displacement index เพื่อแสดงหมายเลขที่มีการยึดรั้ง และหมายเลขที่ไม่มีการยึดรั้ง ดังแสดงใน Flow Chart 4

Flow Chart 4 : Subprogram cmdJointRe





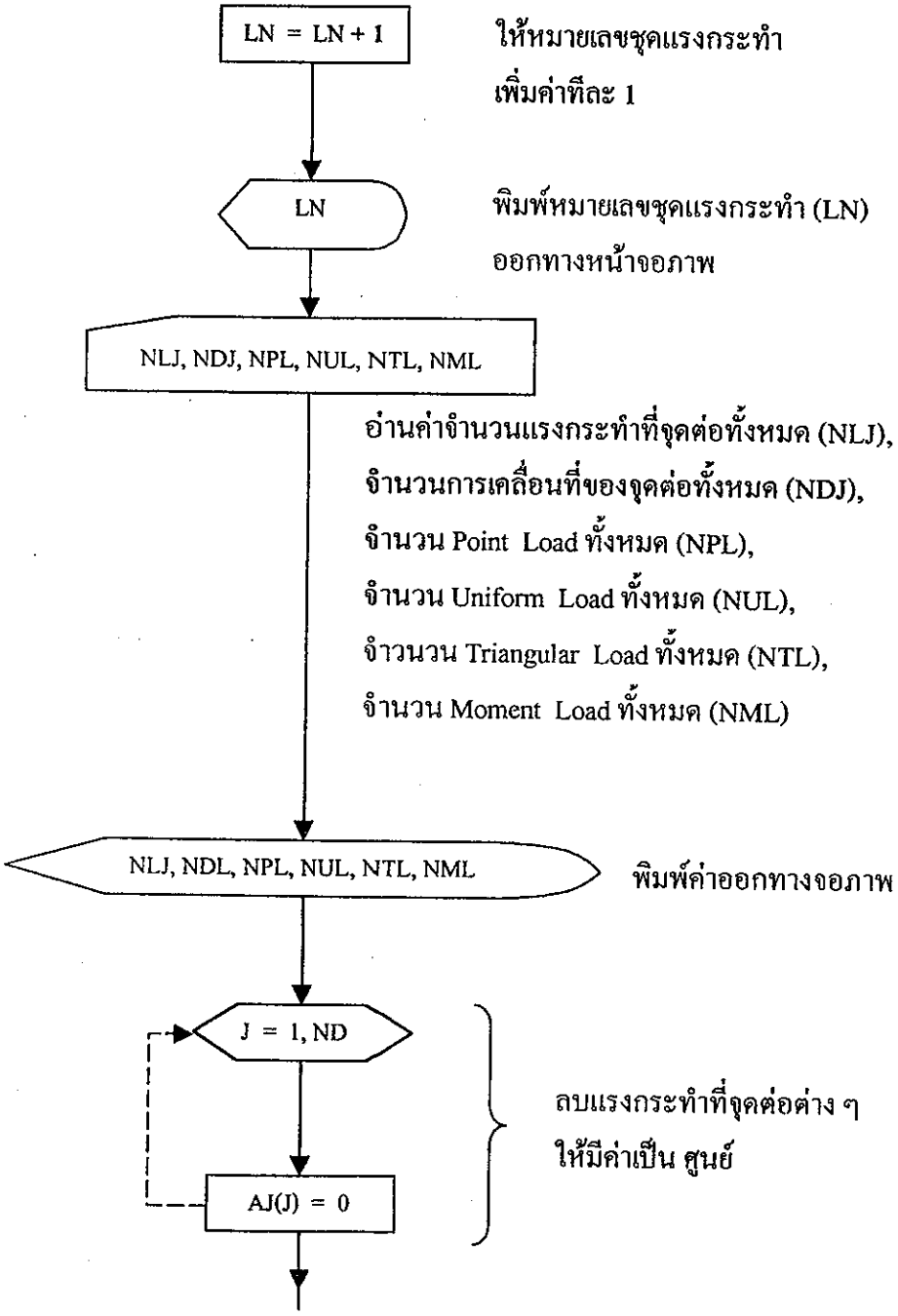
สำหรับกรณีมีการยึดรั้ง ตั้งค่า joint displacement index ตลอดการยึดรั้งที่ฐานรองรับ

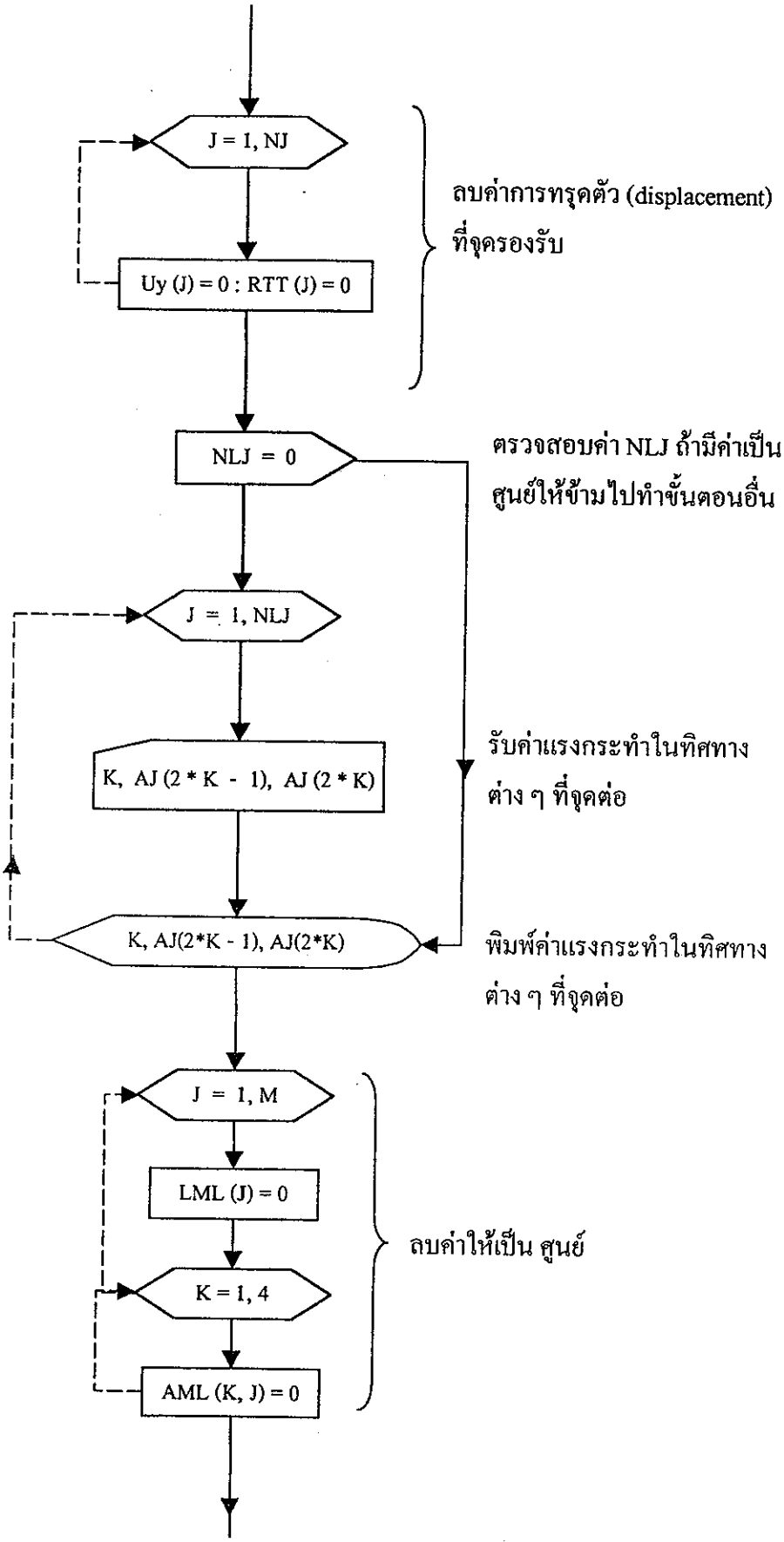
สำหรับกรณีไม่มีการยึดรั้ง ตั้งค่า joint displacement index ตลอด degree of freedom

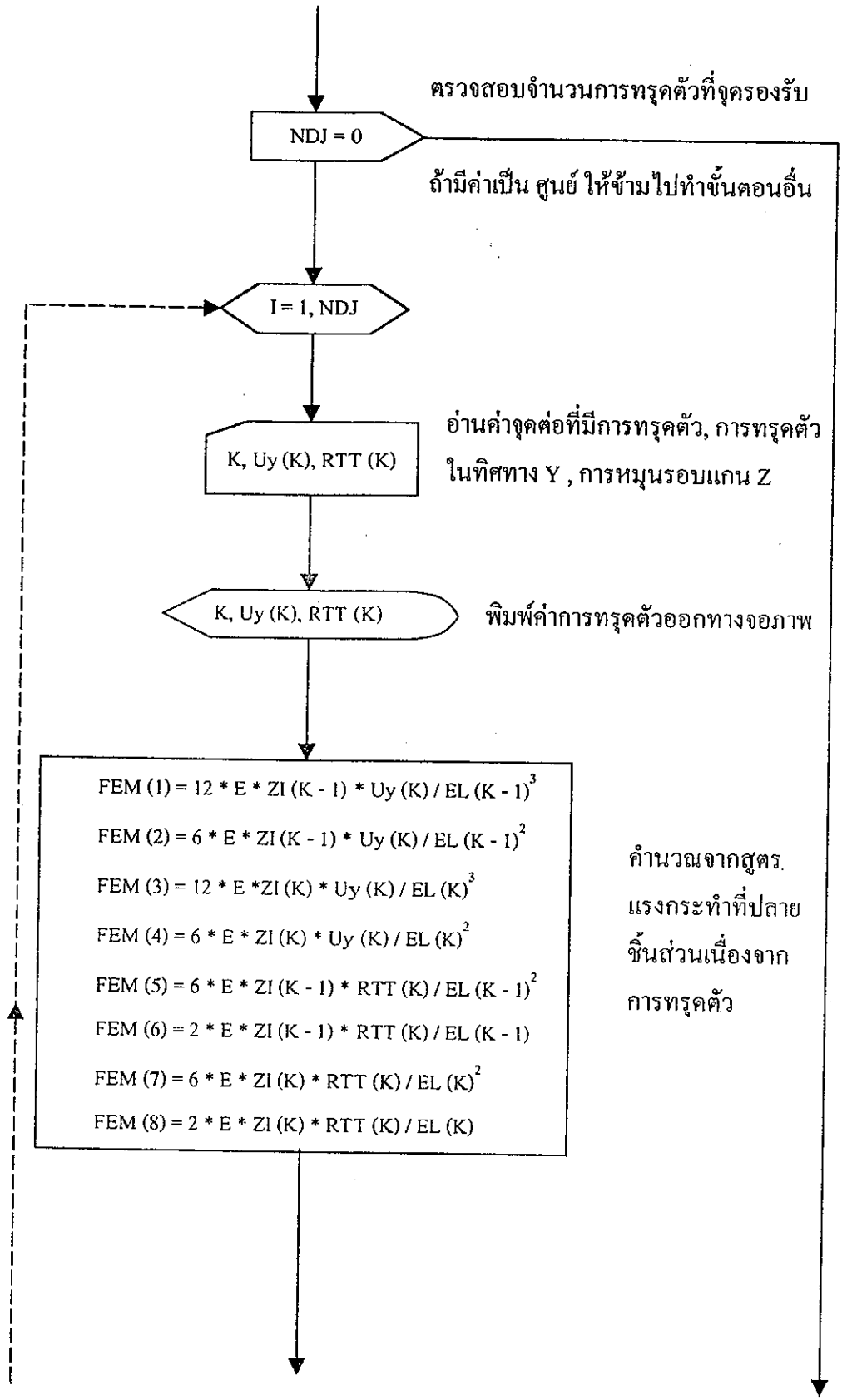
3.2.6 Subprogram cmdLoadD

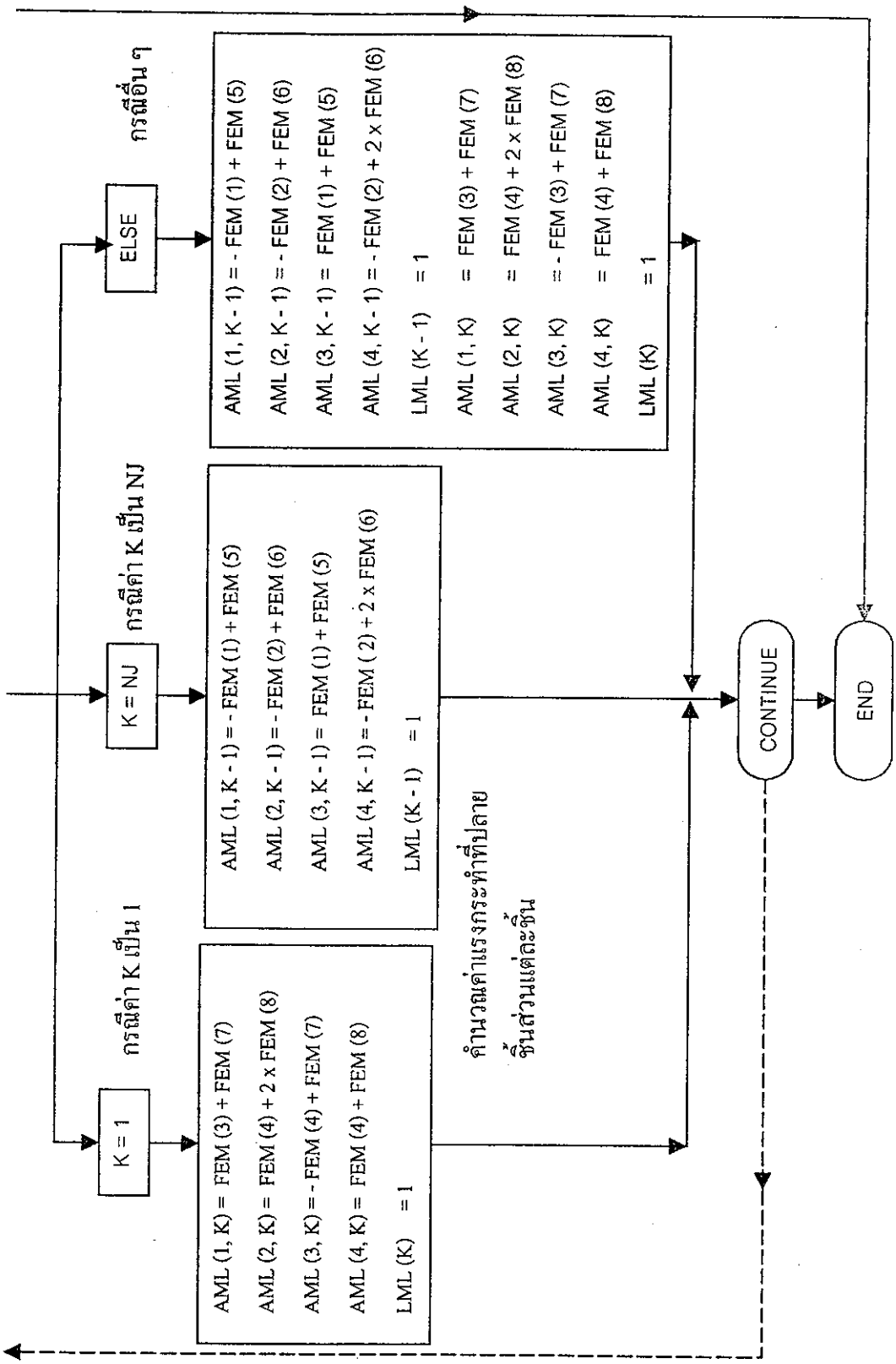
ทำหน้าที่รับค่าจำนวนแรงกระทำต่าง ๆ ที่กระทำบนโครงสร้าง, คำนวณแรงกระทำที่ปลายของชิ้นส่วนที่มีการยึดรั้งเนื่องจากแรงกระทำที่จุดต่อและการทรุดตัวที่จุดต่าง ๆ ดังแสดงใน Flow Chart 5

Flowchart 5 : Subprogram cmdLoadD




J = 1, NJ
Uy (J) = 0 : RTT (J) = 0
ลบค่าการทรุดตัว (displacement)
ที่จุดรองรับ
NLJ = 0
ตรวจสอบค่า NLJ ถ้ามีค่าเป็น
ศูนย์ให้ข้ามไปทำขั้นตอนอื่น
J = 1, NLJ
K, AJ (2 * K - 1), AJ (2 * K)
รับค่าแรงกระทำในทิศทาง
ต่าง ๆ ที่จุดต่อ
K, AJ(2*K - 1), AJ(2*K)
พิมพ์ค่าแรงกระทำในทิศทาง
ต่าง ๆ ที่จุดต่อ
J = 1, M
LML (J) = 0
K = 1, 4
AML (K, J) = 0
ลบค่าให้เป็น ศูนย์

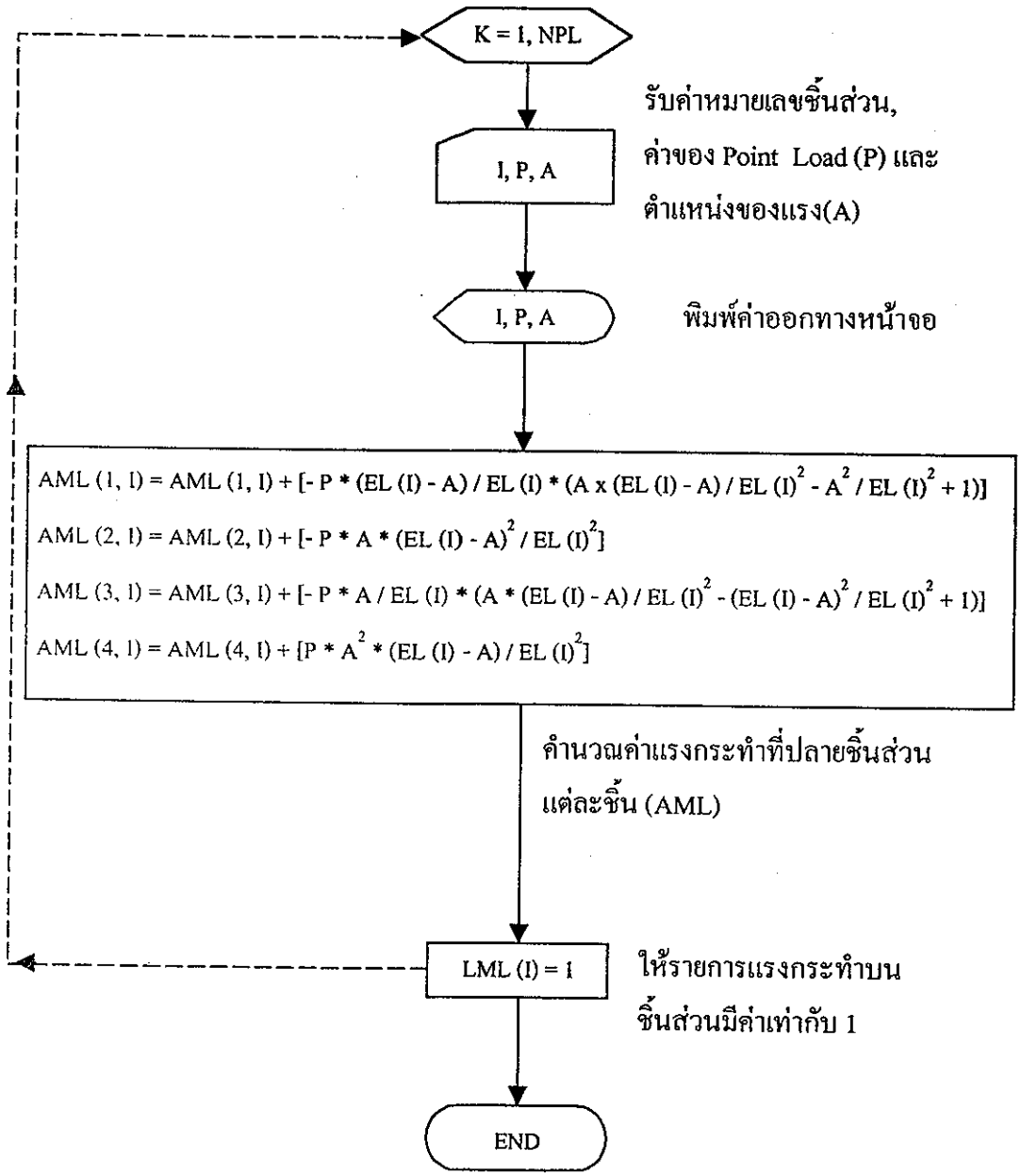

ตรวจสอบจำนวนการทรุดตัวที่จุดรองรับ
NDJ = 0
ถ้ามีค่าเป็น ศูนย์ ให้ข้ามไปทำขั้นตอนอื่น
I = 1, NDJ
K, Uy (K), RTT (K)
อ่านค่าจุดต่อที่มีการทรุดตัว, การทรุดตัว ในทิศทาง Y , การหมุนรอบแกน Z
K, Uy (K), RTT (K)
พิมพ์ค่าการทรุดตัวออกทางจอภาพ
FEM (1) = 12 * E * ZI (K - 1) * Uy (K) / EL (K - 1)3
FEM (2) = 6 * E * ZI (K - 1) * Uy (K) / EL (K - 1)2
FEM (3) = 12 * E *ZI (K) * Uy (K) / EL (K)3
FEM (4) = 6 * E * ZI (K) * Uy (K) / EL (K)2
FEM (5) = 6 * E * ZI (K - 1) * RTT (K) / EL (K - 1)2
FEM (6) = 2 * E * ZI (K - 1) * RTT (K) / EL (K - 1)
FEM (7) = 6 * E * ZI (K) * RTT (K) / EL (K)2
FEM (8) = 2 * E * ZI (K) * RTT (K) / EL (K)
คำนวณจากสูตร แรงกระทำที่ปลายชิ้นส่วนเนื่องจากการทรุดตัว


K = 1
กรณีค่า K เป็น 1
AML (1, K) = FEM (3) + FEM (7)
AML (2, K) = FEM (4) + 2 x FEM (8)
AML (3, K) = - FEM (4) + FEM (7)
AML (4, K) = FEM (4) + FEM (8)
LML (K) = 1
K = NJ
กรณีค่า K เป็น NJ
AML (1, K - 1) = - FEM (1) + FEM (5)
AML (2, K - 1) = - FEM (2) + FEM (6)
AML (3, K - 1) = FEM (1) + FEM (5)
AML (4, K - 1) = - FEM (2) + 2 x FEM (6)
LML (K - 1) = 1
ELSE
กรณีอื่น ๆ
AML (1, K - 1) = - FEM (1) + FEM (5)
AML (2, K - 1) = - FEM (2) + FEM (6)
AML (3, K - 1) = FEM (1) + FEM (5)
AML (4, K - 1) = - FEM (2) + 2 x FEM (6)
LML (K - 1) = 1
AML (1, K) = FEM (3) + FEM (7)
AML (2, K) = FEM (4) + 2 x FEM (8)
AML (3, K) = - FEM (3) + FEM (7)
AML (4, K) = FEM (4) + FEM (8)
LML (K) = 1
คำนวณค่าแรงกระทำที่ปลาย
ชิ้นส่วนแต่ละชิ้น
CONTINUE
END

3.2.7 Subprogram cmdPoint

ทำหน้าที่รับค่าหมายเลขชิ้นส่วนที่มีแรง point load กระทำ, ค่าของ Point Load, และตำแหน่งของแรงกระทำ พิมพ์ผลลัพธ์ออกทางหน้าจอ โดยจะนำมาคำนวณเป็นค่าแรงกระทำที่ปลายชิ้นส่วนแต่ละชิ้น ดังแสดงใน Flow Chart 6

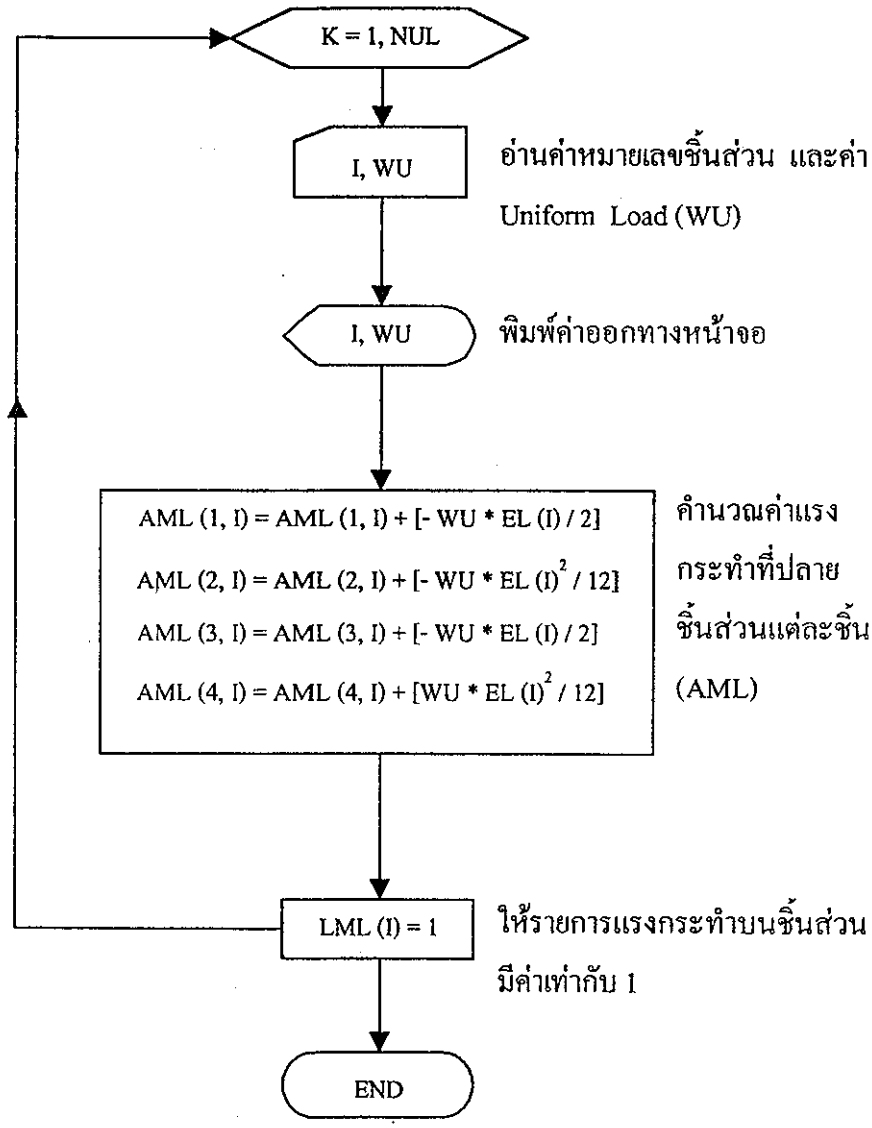
Flowchart 6 : Subprogram cmdPoint



3.2.8 Subprogram cmdUnif

ทำหน้าที่รับค่าหมายเลขชิ้นส่วน และค่าแรง Uniform Load จากนั้นพิมพ์ผลลัพธ์ออกทางหน้าจอ โดยจะนำค่ามาคำนวณหาแรงกระทำที่ปลายของชิ้นส่วนแต่ละชิ้น ดังแสดงใน Flow Chart 7

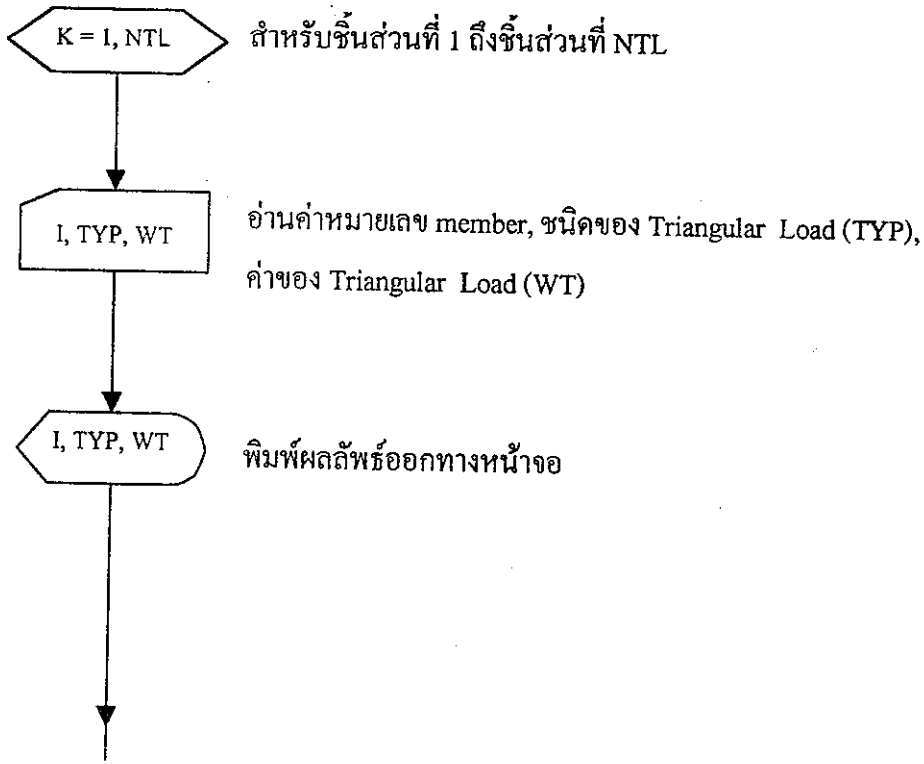
Flow Chart 7 : Subprogram cmdUnif

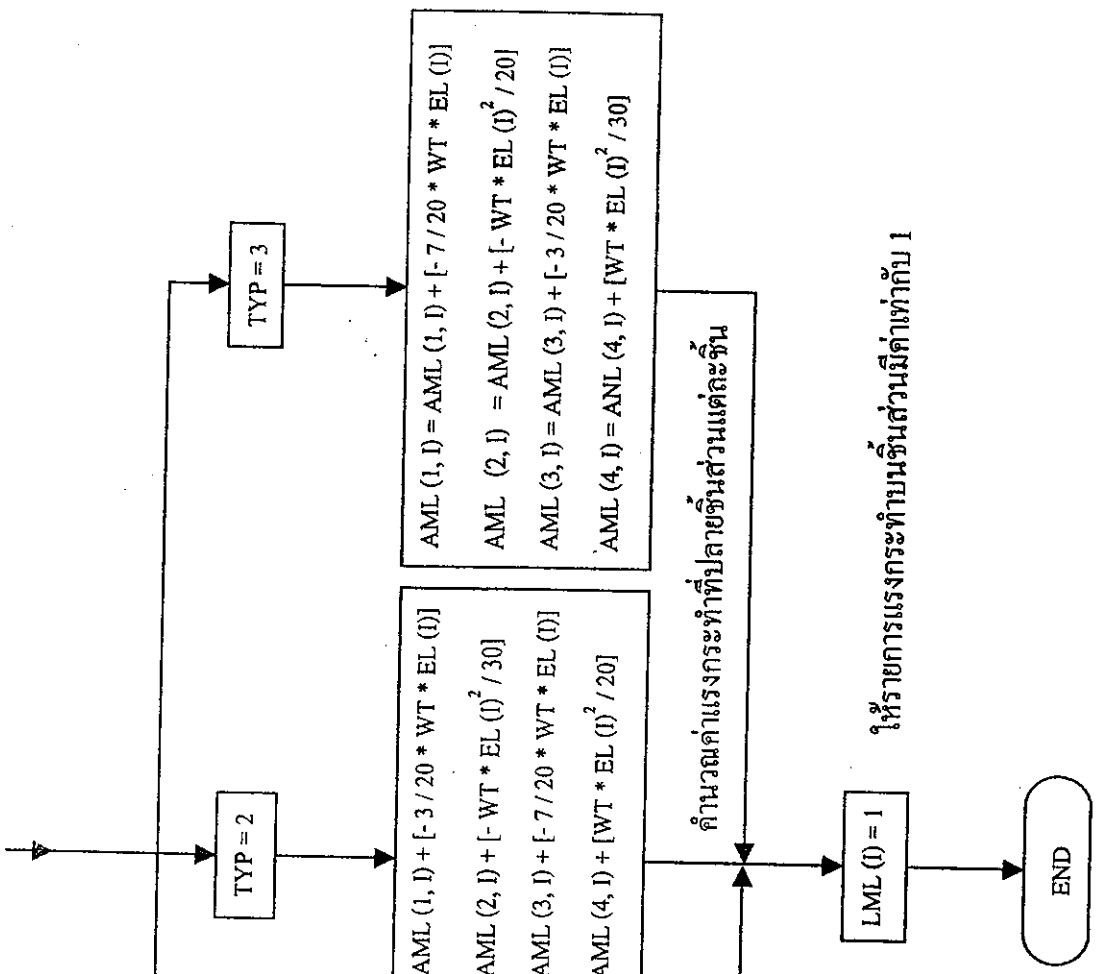


3.2.9 Subprogram cmdTril

ทำหน้าที่รับค่าหมายเลขชิ้นส่วน และค่าแรง Triangular Load จากนั้นพิมพ์ผลลัพธ์ออกทางหน้าจอ โดยจะนำค่ามาคำนวณหาแรงกระทำที่ปลายของชิ้นส่วนแต่ละชิ้น ดังแสดงใน Flow Chart 8

Flow Chart 8 : Subprogram cmdTriL

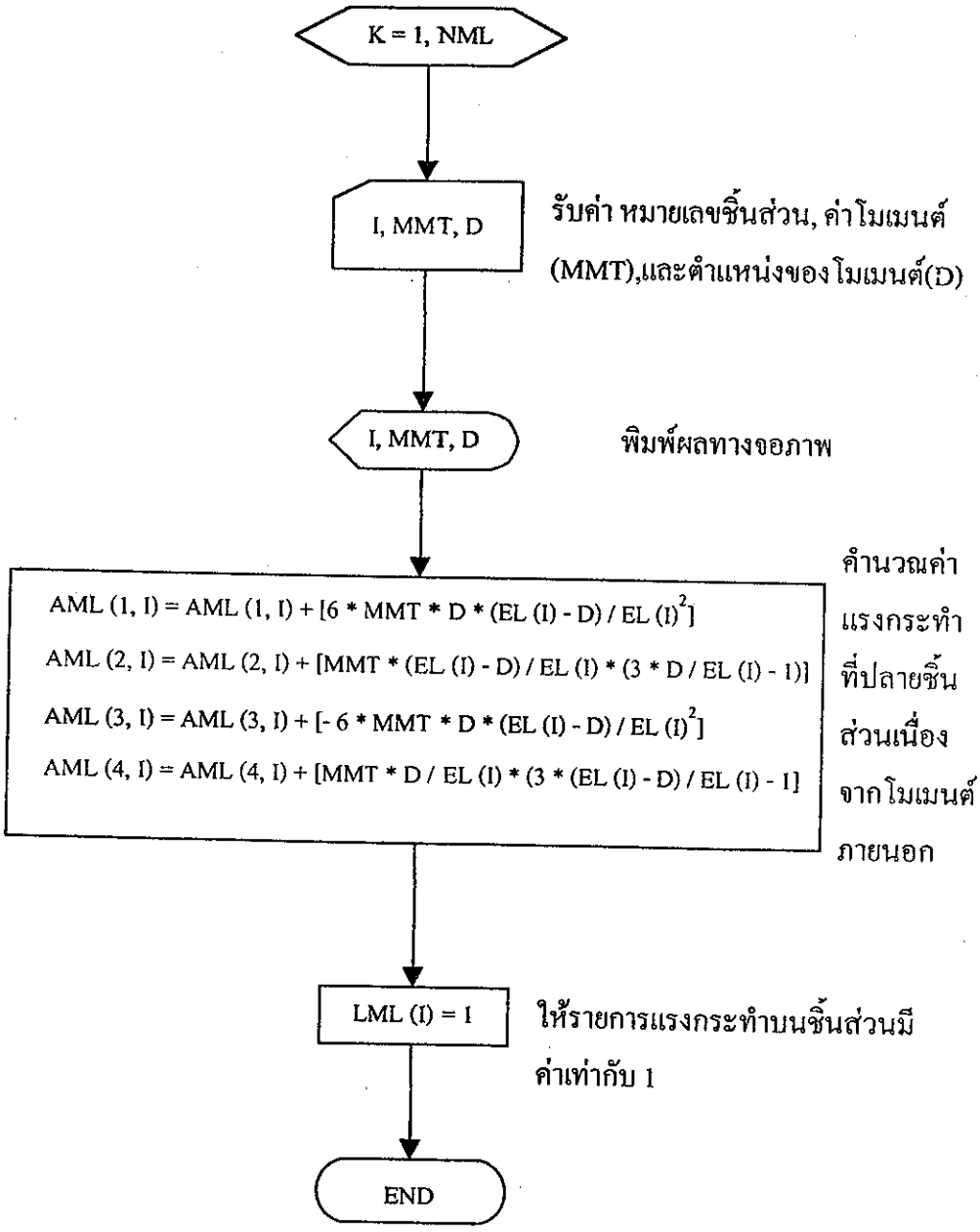



TYP = 1
TYP = 2
TYP = 3
AML (1, I) = AML (1, I) + [- WT * EL (I) / 4]
AML (2, I) = AML (2, I) + [- 5 * WT * EL (I)² / 96]
AML (3, I) = AML (3, I) + [- WT * EL (I) / 4]
AML (4, I) = AML (4, I) + [5 * WT * EL (I)² / 96]
AML (1, I) = AML (1, I) + [- 3 / 20 * WT * EL (I)]
AML (2, I) = AML (2, I) + [- WT * EL (I)² / 30]
AML (3, I) = AML (3, I) + [- 7 / 20 * WT * EL (I)]
AML (4, I) = AML (4, I) + [WT * EL (I)² / 20]
AML (1, I) = AML (1, I) + [- 7 / 20 * WT * EL (I)]
AML (2, I) = AML (2, I) + [- WT * EL (I)² / 20]
AML (3, I) = AML (3, I) + [- 3 / 20 * WT * EL (I)]
AML (4, I) = ANL (4, I) + [WT * EL (I)² / 30]
คำนวณค่าแรงกระทำที่ปลายชิ้นส่วนแต่ละชิ้น
LML (I) = 1
ให้รายการแรงกระทำยกชิ้นส่วนมีค่าเท่ากับ 1
END

3.2.10 Subprogram cmdMMT

ทำหน้าที่รับค่าหมายเลขชิ้นส่วน ค่าแรง Moment Load และตำแหน่งของแรง Moment จากนั้นพิมพ์ผลลัพธ์ออกทางหน้าจอ โดยจะนำค่ามาคำนวณหาแรงกระทำที่ปลายของชิ้นส่วนแต่ละชิ้น ดังแสดงใน Flow Chart 9

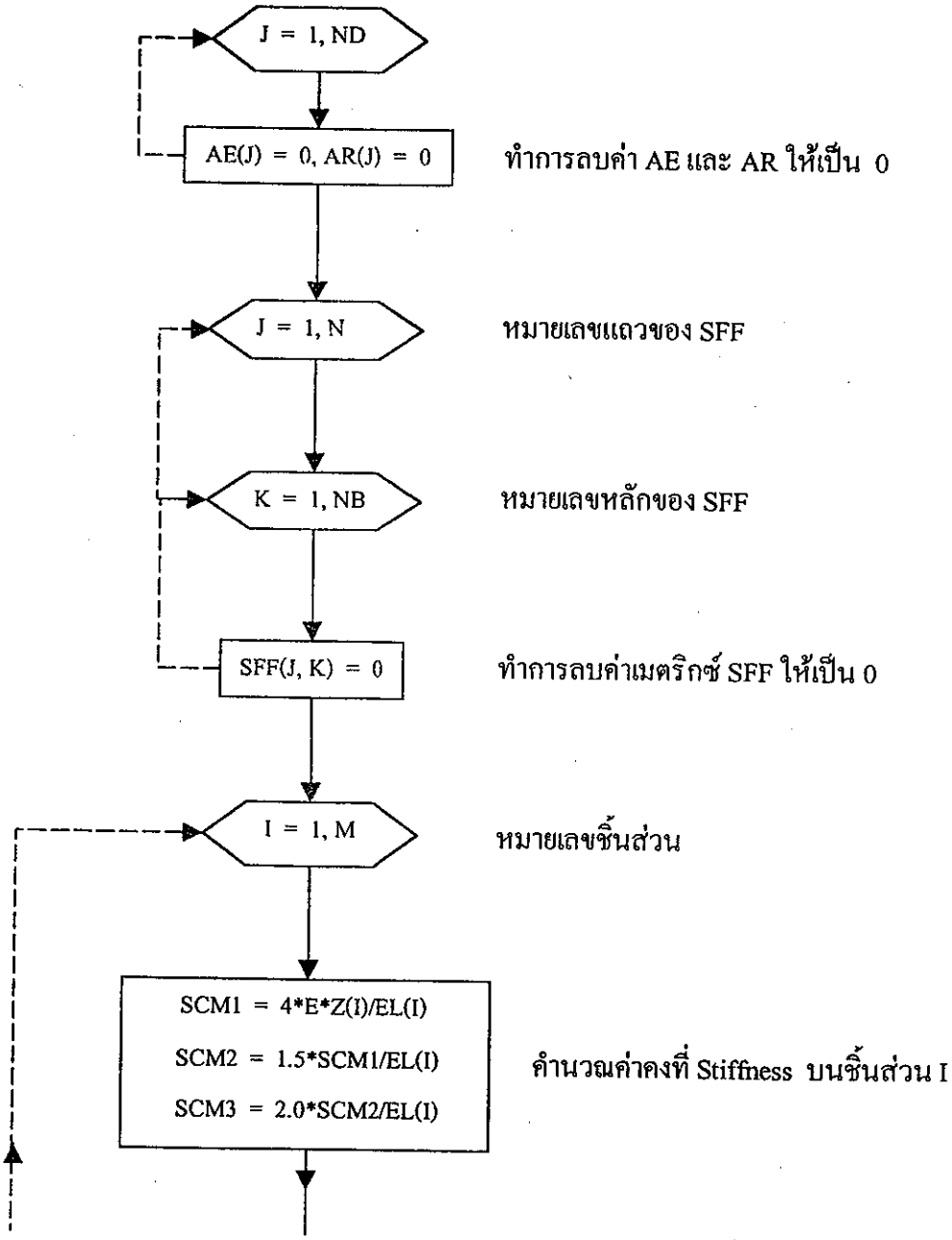
Flowchart 9 : Subprogram cmdMMT

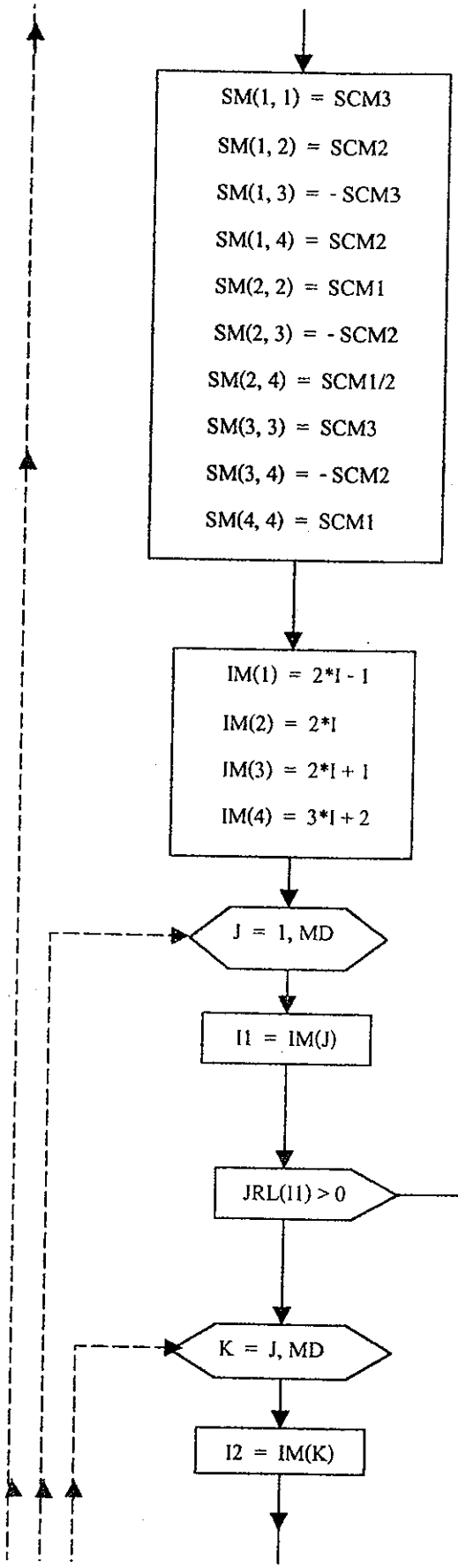


3.2.11Subprogram cmdResult

ทำหน้าที่วิเคราะห์คำตอบของโครงสร้าง โดยภายในโปรแกรมจะมีการเรียกใช้ Subprogram BANFAC และSubprogram BANSOL จากนั้นจะทำการคำนวณหาค่า Joint Displacement, Member – end Action และ Support Reaction โดยมีขั้นตอนดังแสดงใน Flow Chart 10

Flow Chart 10 : Subprogram cmdResult





เติมส่วนสามเหลี่ยมบนของเมตริกซ์สติฟเนส SM โดยใช้ค่าคงที่สติฟเนสที่ได้คำนวณไว้แล้ว

เติมค่าใน member displacement indexes

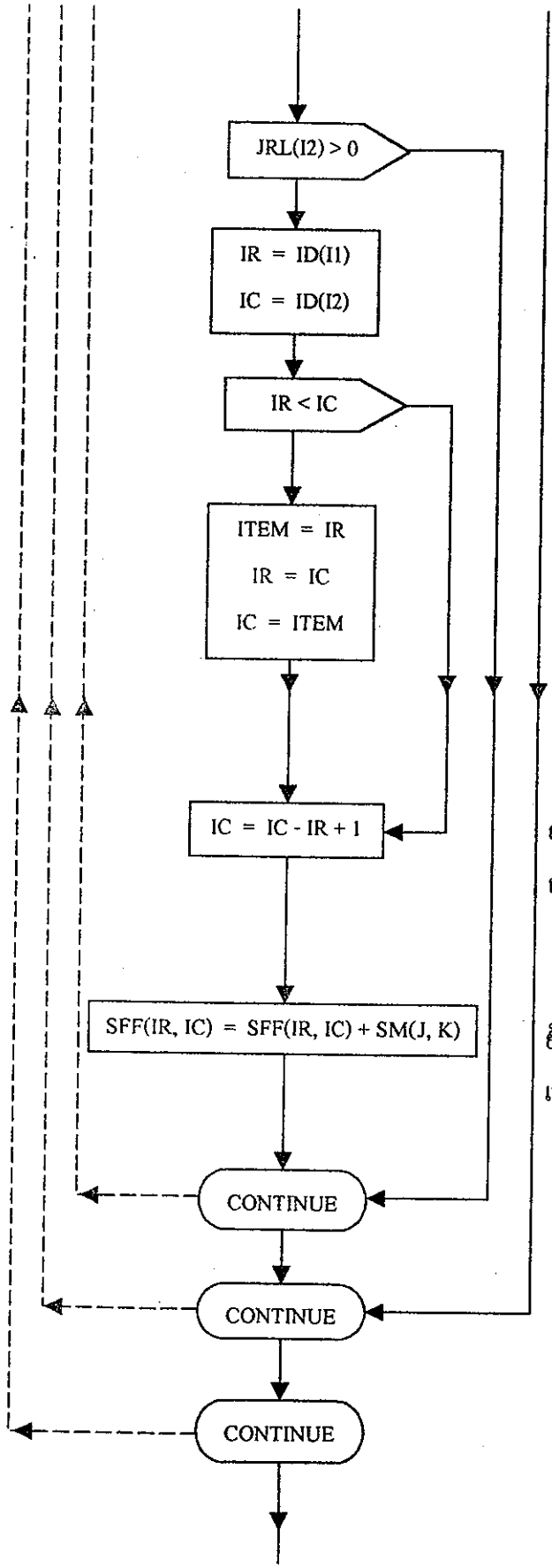
หมายเลขแถวของ SM

ตั้งค่าแถวชั้วคราว I1 สำหรับเมตริกซ์ SFF ให้เท่ากับ IM (I)

ตรวจสอบสำหรับการยึดรั้งถ้าเท่ากับ 1 ให้ข้ามไปทำขั้นตอนอื่น

หมายเลขของคอลัมน์ SM

ตั้งคอลัมน์ชั่วคราว I2 สำหรับเมตริกซ์ SFF เท่ากับ IM (k)



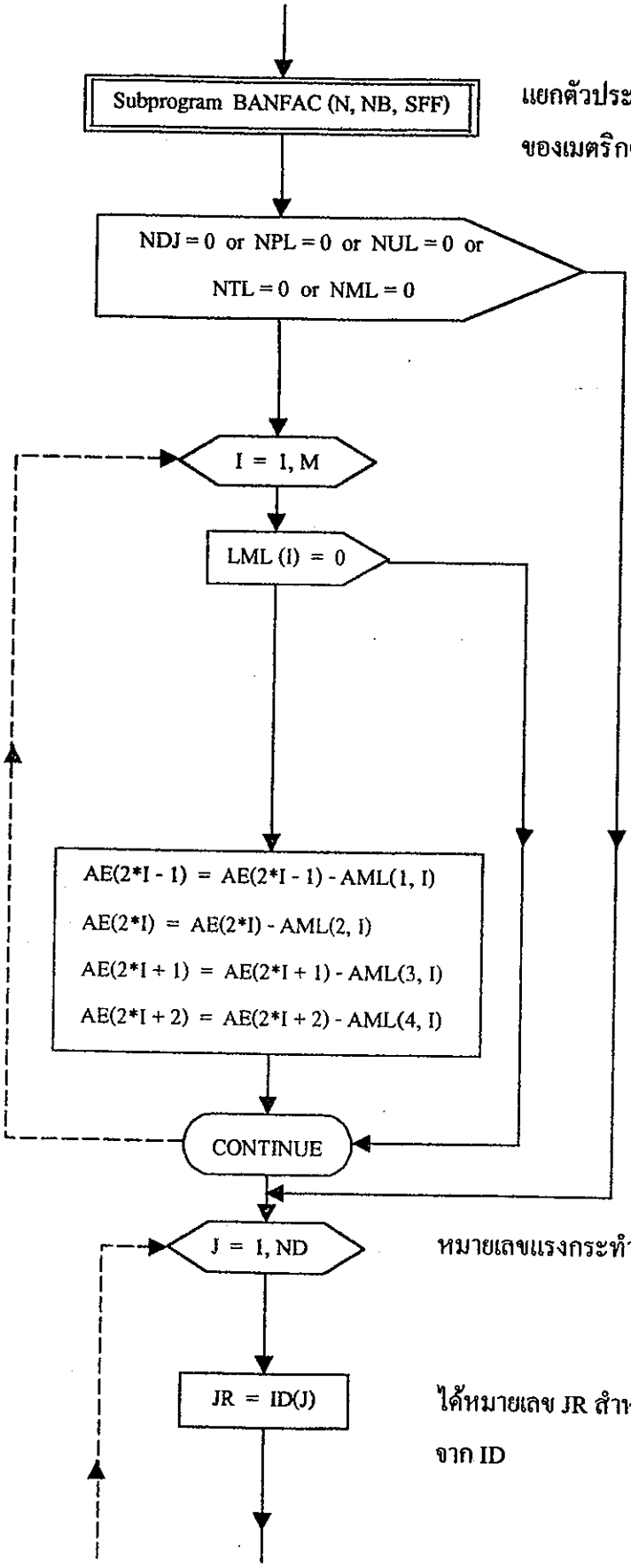
ตรวจสอบค่ายึดรั้งที่สอดคล้องกับ I2 ได้รับค่าแถวจริงและคอลัมน์จริง (IR และ IC) จาก ID

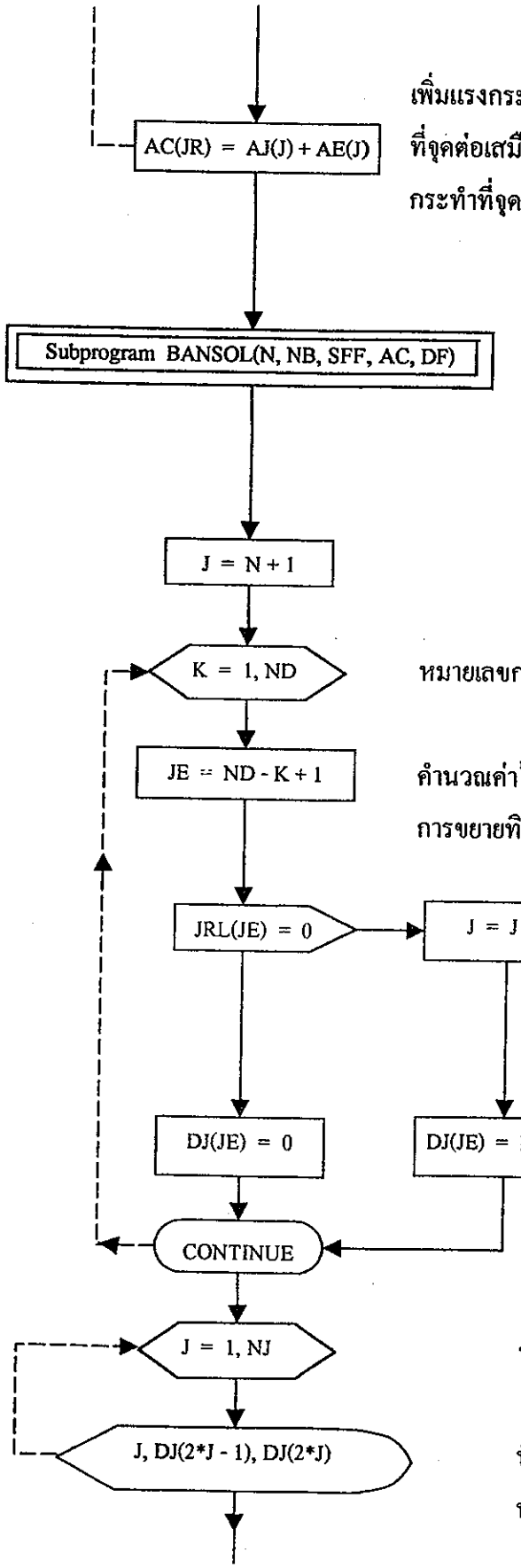
ตรวจสอบตำแหน่งสามเหลี่ยมบน ถ้าใช่ให้ข้ามไป

สำหรับตำแหน่งสามเหลี่ยมล่างให้เปลี่ยนเป็นตำแหน่งสามเหลี่ยมบน

ยกเลิกหมายเลขคอลัมน์สำหรับการเก็บbandส่วนบนของเมตริกซ์ SFFเป็นอาเรย์สี่เหลี่ยมผืนผ้า

ย้ายสมาชิกของ SM ไปตำแหน่งที่เหมาะสมในเมตริกซ์ SFF โดยให้เป็น IR และ IC


Subprogram BANFAC (N, NB, SFF)
แยกตัวประกอบของ band ส่วนบน ของเมตริกซ์ SFF
NDJ = 0 or NPL = 0 or NUL = 0 or NTL = 0 or NML = 0
ถ้าไม่มีแรงกระทำบนชิ้นส่วนให้ข้ามไปทำส่วนอื่น
I = 1, M
หมายเลขประจำชิ้นส่วน
LML (I) = 0
ถ้าไม่มีแรงกระทำให้ข้ามไปทำขั้นตอนอื่น
AE(2*I - 1) = AE(2*I - 1) - AML(1, I)
AE(2*I) = AE(2*I) - AML(2, I)
AE(2*I + 1) = AE(2*I + 1) - AML(3, I)
AE(2*I + 2) = AE(2*I + 2) - AML(4, I)
ย้ายค่าลบของแรงกระทำที่ปลายชิ้นส่วน, AML ไปยังตำแหน่งที่เหมาะสมในแรงกระทำที่จุดต่อเสมือนในทิศทาง AE
CONTINUE
J = 1, ND
หมายเลขแรงกระทำที่จุดต่อ
JR = ID(J)
ได้หมายเลข JR สำหรับการจัดเรียงใหม่จาก ID



เพิ่มแรงกระทำที่จุดต่อจริง AJ และแรงกระทำที่จุดต่อเสมือน AE ที่ได้รับจากการรวมแรงกระทำที่จุดต่อในการจัดเรียงรูปแบบใหม่

แก้สมการสำหรับ free joint displacement DF ใช้การแยกตัวประกอบเมตริกซ์ SFF และสมาชิก N ตัวแรกในทิศทาง AC

หมายเลขการทรุดตัวที่จุดต่อ

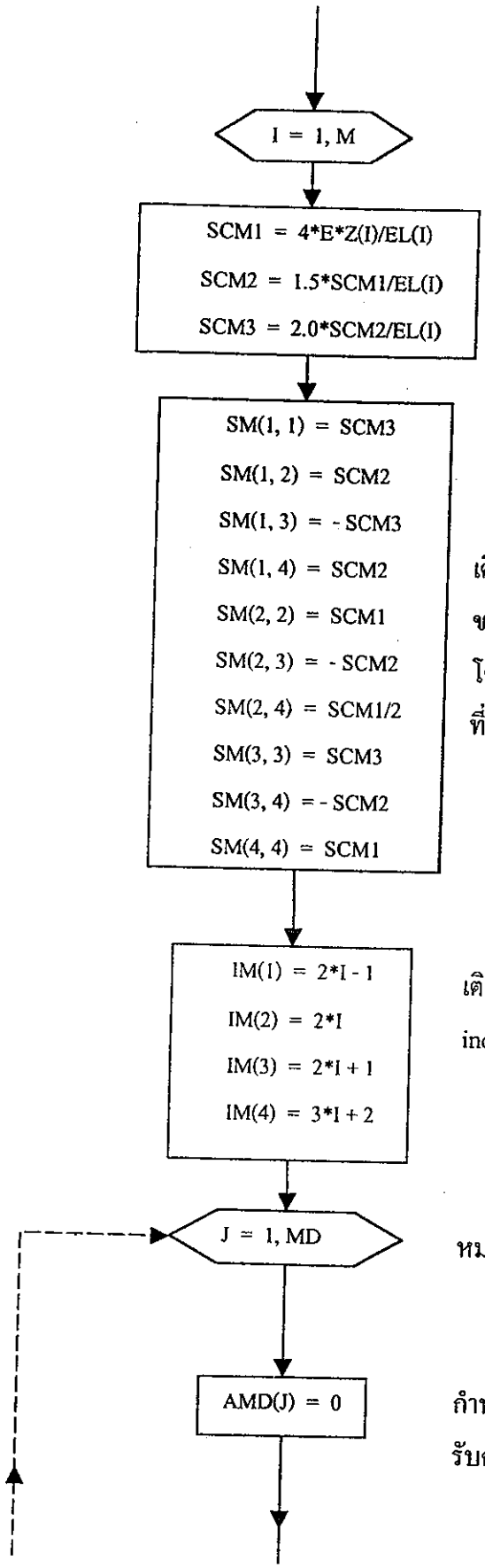
คำนวณค่าได้มาของหมายเลข JE สำหรับการขยายทิศทางการทรุดตัว

สำหรับกรณีไม่มีจุดยึดรั้ง ลดค่า J ลง 1 เพื่อบอกตำแหน่งสมาชิกของ DF

ย้ายสมาชิกของ DF ไปยังตำแหน่งที่เหมาะสมใน DJ

หมายเลขประจำจุดต่อ

พิมพ์หมายเลขจุดต่อ และการทรุดตัวที่จุดต่อออกทางหน้าจอ



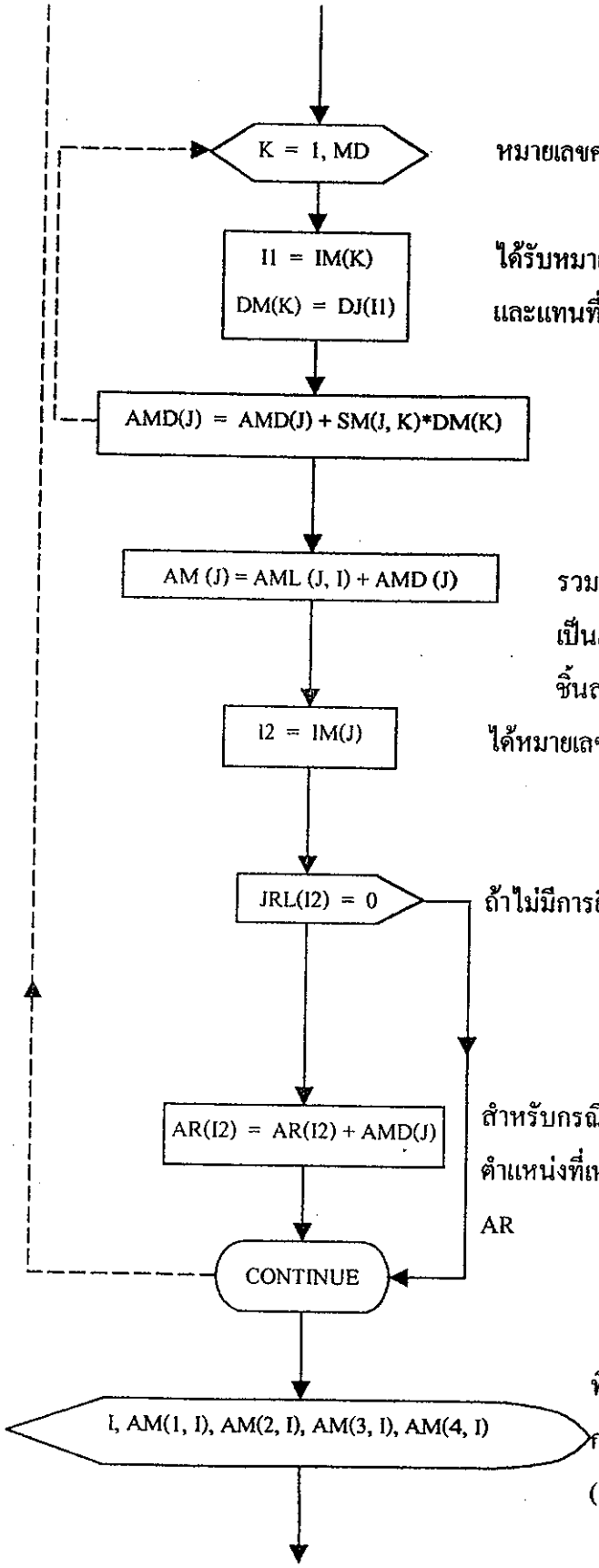
คำนวณค่าคงที่ Stiffness บน
ชิ้นส่วน I

เติมส่วนสามเหลี่ยมบน
ของเมตริกซ์สติฟเนส SM
โดยใช้ค่าคงที่สติฟเนส
ที่ได้คำนวณไว้แล้ว

เติมค่าใน member displacement
indexes

หมายเลขแถว SM

กำหนดให้ AMD (J) เป็นศูนย์เพื่อเตรียม
รับคำตอบที่จะสะสมไปเรื่อย ๆ



หมายเลขคอลัมน์ของ SM

ได้รับหมายเลขการทรุดตัว I1 จากทิศทาง IM และแทนที่สมาชิก DJ ใน DM

คำนวณ AMD (J) โดยผลของเมตริกซ์ทิศทาง SM และ DM

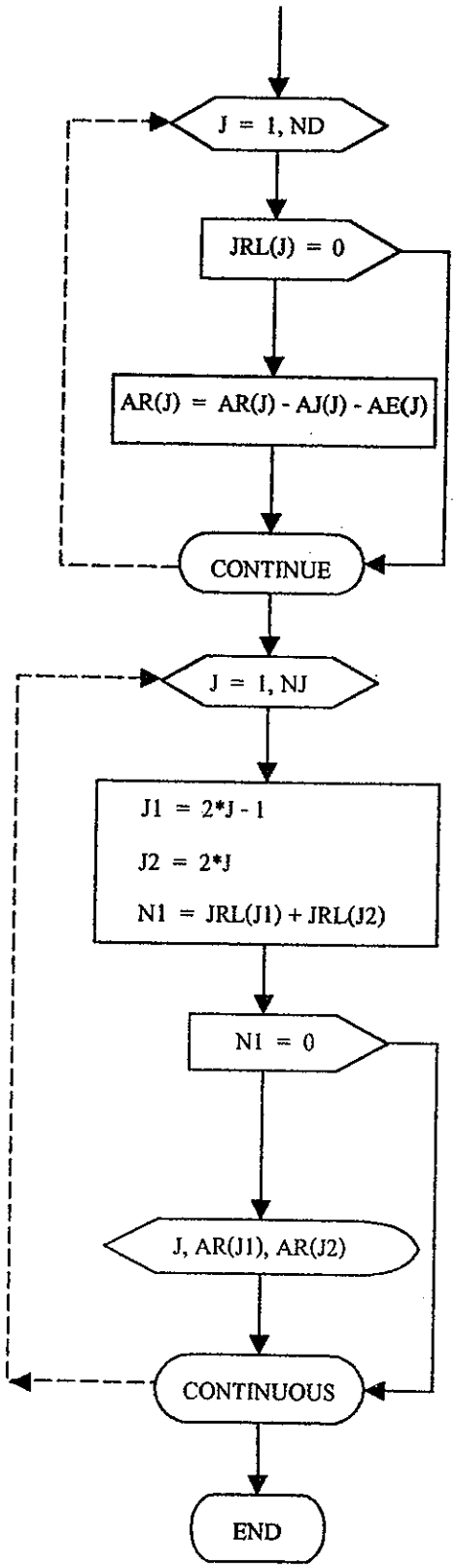
รวมค่า AML (J, I) และ AMD (J) เป็นแรงกระทำภายในที่ปลายชิ้นส่วน

ได้หมายเลขการทรุดตัว I2 จากทิศทาง IM

ถ้าไม่มีการยึดรั้งเกิดขึ้นให้ข้ามไป

สำหรับกรณีมีการยึดรั้ง ย้าย AMD (J) ไปยังตำแหน่งที่เหมาะสมในทิศทางแรงปฏิกิริยา AR

พิมพ์หมายเลขชิ้นส่วนและแรงกระทำภายในที่ปลายชิ้นส่วน (Final member end - action)



หมายเลขของแรงปฏิกิริยาที่ฐานรองรับ

ถ้าไม่มีการยึดรั้งเกิดขึ้นให้ข้ามไป

กรณีมีการยึดรั้ง

หมายเลขจุดต่อ

ตั้งหมายเลข J1 และ J2 แล้ว
รวมค่าJRL ได้ N1

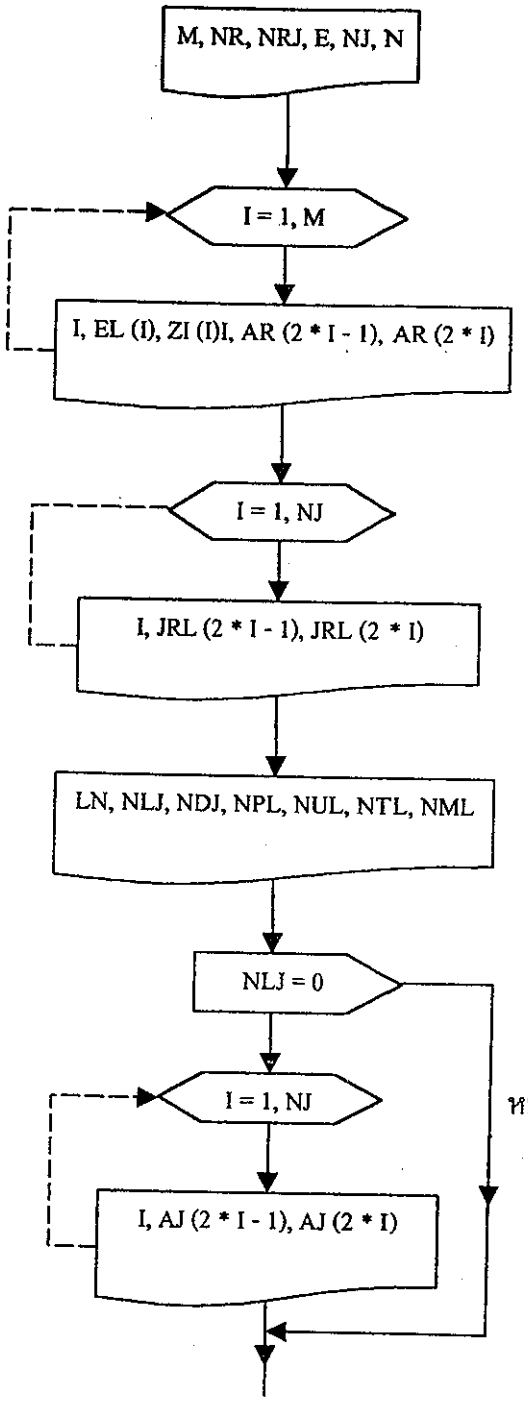
ถ้า N1 = 0, ไม่มีการเกิดการยึดรั้งขึ้น
ที่จุดต่อ J ดังนั้นไม่ต้องพิมพ์ค่าแรง
ปฏิกิริยา

พิมพ์หมายเลขจุดต่อ 'และแรงที่ฐานรองรับ
ที่มีการยึดรั้งเกิดขึ้น

3.2.12 Subprogram cmdPrint

ทำหน้าที่สั่งให้เครื่อง print พิมพ์ข้อมูล และผลลัพธ์ออกมาทางกระดาษ โดยมีขั้นตอนดังแสดงใน Flow Chart 11

Flow Chart 11 : Subprogram cmdPrint



พิมพ์ค่า Structural parameter ออกทางเครื่องพิมพ์

หมายเลขชิ้นส่วน

พิมพ์คุณสมบัติของแต่ละชิ้นส่วน ออกทางเครื่องพิมพ์

หมายเลขจุดต่อ

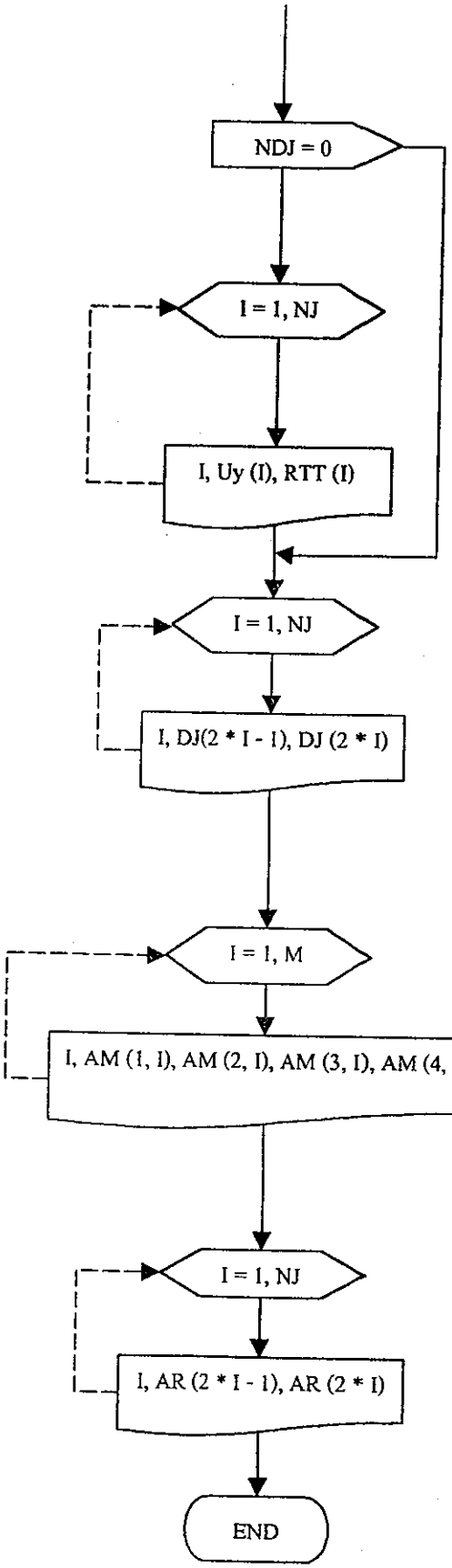
พิมพ์ค่าการยึดครั้งที่จุดต่อต่าง ๆ ออกทางเครื่องพิมพ์

พิมพ์ค่าจำนวนแรงกระทำต่าง ๆ ออกทางเครื่องพิมพ์

ถ้าแรงกระทำที่จุดต่อไม่มีให้ข้ามไปทำขั้นตอนอื่น

หมายเลขจุดต่อ

กรณีมีแรงกระทำที่จุดต่อให้พิมพ์หมายเลขจุดต่อ และแรงที่กระทำที่จุดต่อออกทางเครื่องพิมพ์



ถ้าการทรุดตัวที่จุดต่อไม่มีให้ข้ามไปทำขั้นตอนอื่น

หมายเลขจุดต่อ

กรณีมีการทรุดที่จุดต่อให้พิมพ์หมายเลขจุดต่อ และการทรุดตัวที่จุดต่อออกทางเครื่องพิมพ์

หมายเลขจุดต่อ

พิมพ์หมายเลขจุดต่อและการทรุดตัวเนื่องจากแรงที่มากระทำในทิศทางต่าง ๆ ออกทางเครื่องพิมพ์

กำหนดหมายเลขชิ้นส่วน

พิมพ์หมายเลขชิ้นส่วน และแรงกระทำภายในที่ปลายของแต่ละชิ้นส่วนออกทางเครื่องพิมพ์

กำหนดหมายเลขจุดต่อ

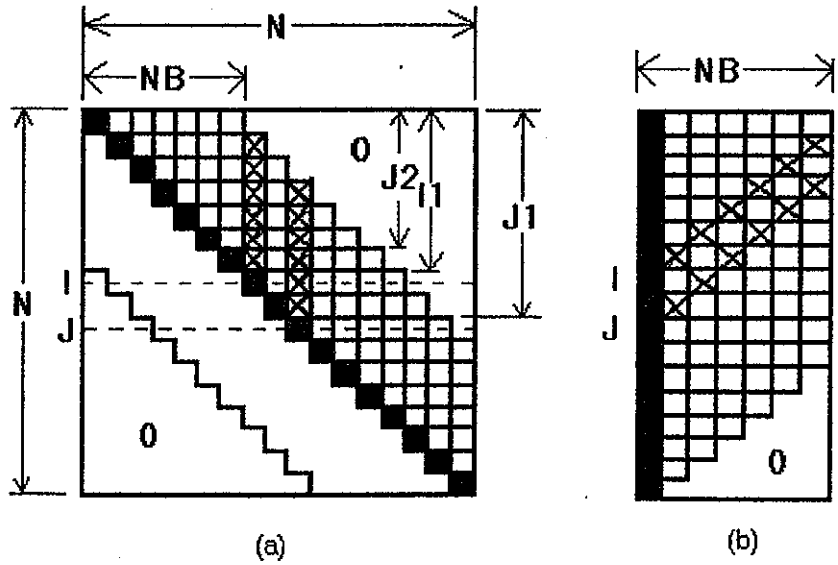
พิมพ์หมายเลขจุดต่อ และแรงปฏิกิริยาที่ฐานรองรับออกทางเครื่องพิมพ์

3.3 การเขียนโปรแกรมแก้สมการ K * u = P

3.3.1 Subprogram BANFAC

เนื่องจากเมตริกซ์สติฟเนสที่ได้มีลักษณะสมมาตรและเป็น banded matrix ดังนั้นเราจึงทำการเขียนโปรแกรมเพื่อที่จะนำ banded matrix มาแยกเป็นสมการ 2.20 แล้วทำการหาค่าของเมตริกซ์ $[\overline{U}]$ โดยโปรแกรมนี้จะถูกเรียกใช้ใน Subprogram cmdResult ซึ่งวิธีการแยกตัวประกอบแบบ banded matrix จะมีประสิทธิภาพดีกว่าแบบเติมเป็นอาเรย์ เพราะว่าจะไม่มีการคำนวณในส่วนที่อยู่นอก band รูป 3.1a แสดงรูปร่างโดยทั่วไปของ banded symmetric matrix สัญลักษณ์ NB ที่แสดงในรูปแทนความกว้างครึ่ง band (semi – band width) และ N คือ ขนาดของเมตริกซ์ มีเพียงส่วนบนของ band (รวมทั้งส่วนที่อยู่ในแนวทแยงมุม) ที่เราจะทำการเก็บข้อมูลไว้แสดงด้วยสี่เหลี่ยมจัตุรัสเล็ก ๆ ดังรูป 3.1a

รูปแบบการเก็บ band ครึ่งบนตามแนวทแยงมุมของ matrix แสดงในรูป 3.1b ในการจัดเรียงแบบที่จะจัดเรียงเป็นอาร์เรย์สี่เหลี่ยมผืนผ้า โดยมีค่าตามแนวทแยงมุมอยู่ในคอลัมน์แรก



รูป3.1 Banded matrix : (a) usual form of storage and
(b) upper band stroed as a rectangular array

ดังแสดงใน Flow Chart 1 แสดงขั้นตอนของ Subprogram ที่องค์ประกอบ band ส่วนบนของเมตริกซ์สมมาตรถูกเก็บไว้เป็นอาเรย์สี่เหลี่ยมผืนผ้า ชื่อของโปรแกรมย่อยคือ

BANFAC (N, NB, A)

ค่าตัวแปรในวงเล็บมีความหมายดังนี้

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 เทอมแรกที่ไม่ใช้ศูนย์ในคอลัมน์นั้นจะมีหมายเลขแถว

$$J2 = J - NB + 1 \quad (NB \leq J \leq N) \tag{3.1}$$

นอกจากนี้สำหรับ NB คอลัมน์แรก (ยกเว้นคอลัมน์ 1)

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

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 และทำการแก้สมการหาตัวไม่ทราบค่าในระบบเดิมของสมาการ ชื่อของโปรแกรมย่อยนี้คือ

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

โดยสัญลักษณ์ที่เพิ่มขึ้นมามีคำนิยามดังนี้

U หมายถึง เมตริกซ์ที่ได้มาจาก Subprogram BANFAC

B และ X หมายถึง เวกเตอร์จริง (real vectors) ของเทอมที่มีค่าคงที่และตัวไม่ทราบค่าตามลำดับ

ในFlow Chart 2 แสดงขั้นตอนของ Subprogram BANSOL ในการแทนค่าไปข้างหน้าและแทนค่ากลับ (ดูสมการ 2.23 และ 2.25) สัญลักษณ์ J จะถูกใช้วิเคราะห์เทอมที่ไม่เป็นศูนย์ที่

สรุปไว้ในการคำนวณ ในการแทนค่าไปข้างหน้าเลขประจำแถวสำหรับค่าที่ไม่เป็นศูนย์ ค่าแรกในคอลัมน์ I คือ

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

นอกจากนี้สำหรับคอลัมน์ NB แรก (ยกเว้นคอลัมน์ 1)

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

คล้ายคลึงกับการแทนค่าย้อนกลับเลขประจำคอลัมน์สำหรับส่วนที่ไม่เป็นศูนย์ สุดท้ายในแถว I ของเมตริกซ์ $[\overline{U}]$ คือ

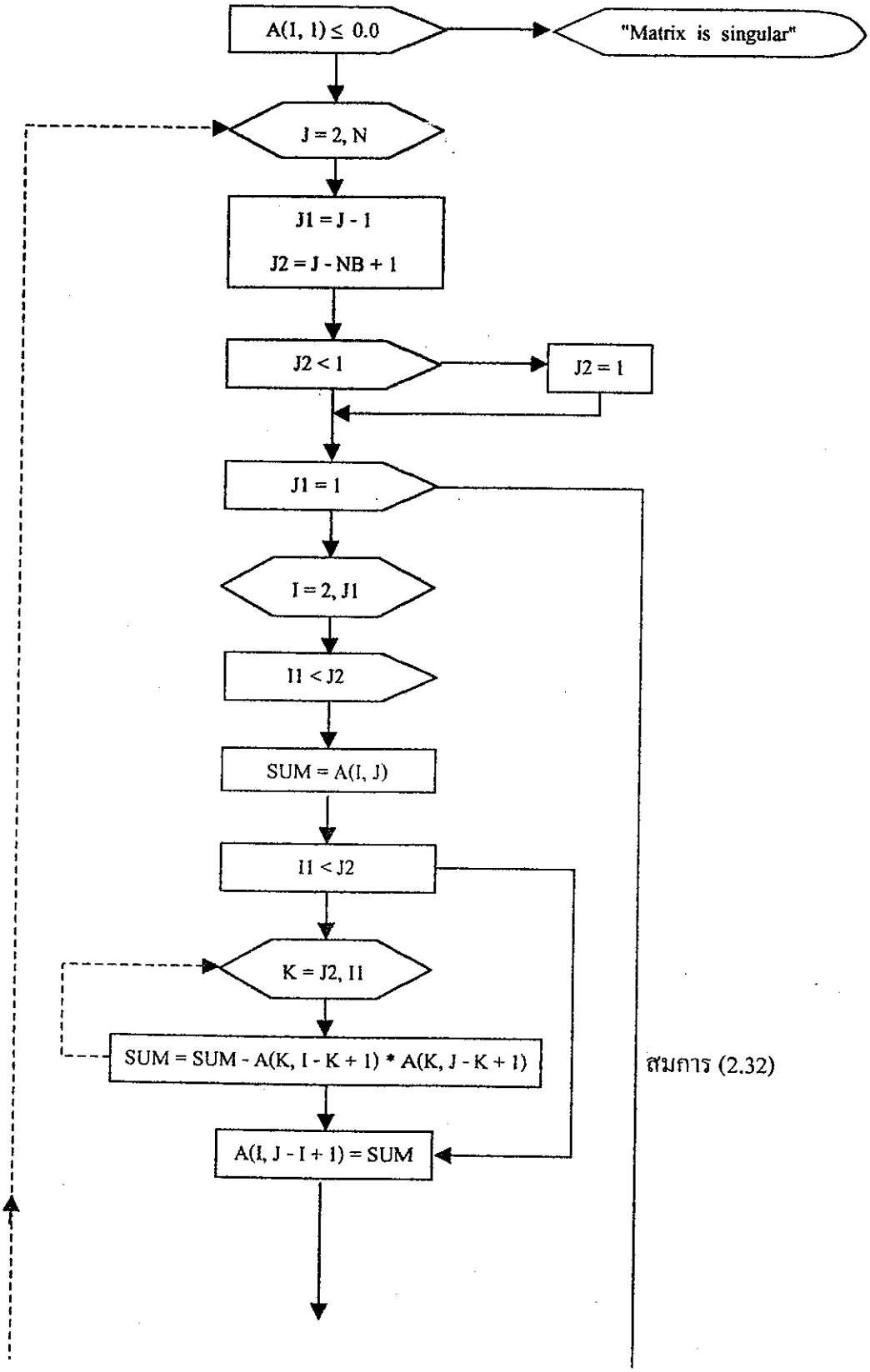
$$J = I + NB - 1 \qquad [I \leq 1 \leq (N - NB)] \qquad (3.5)$$

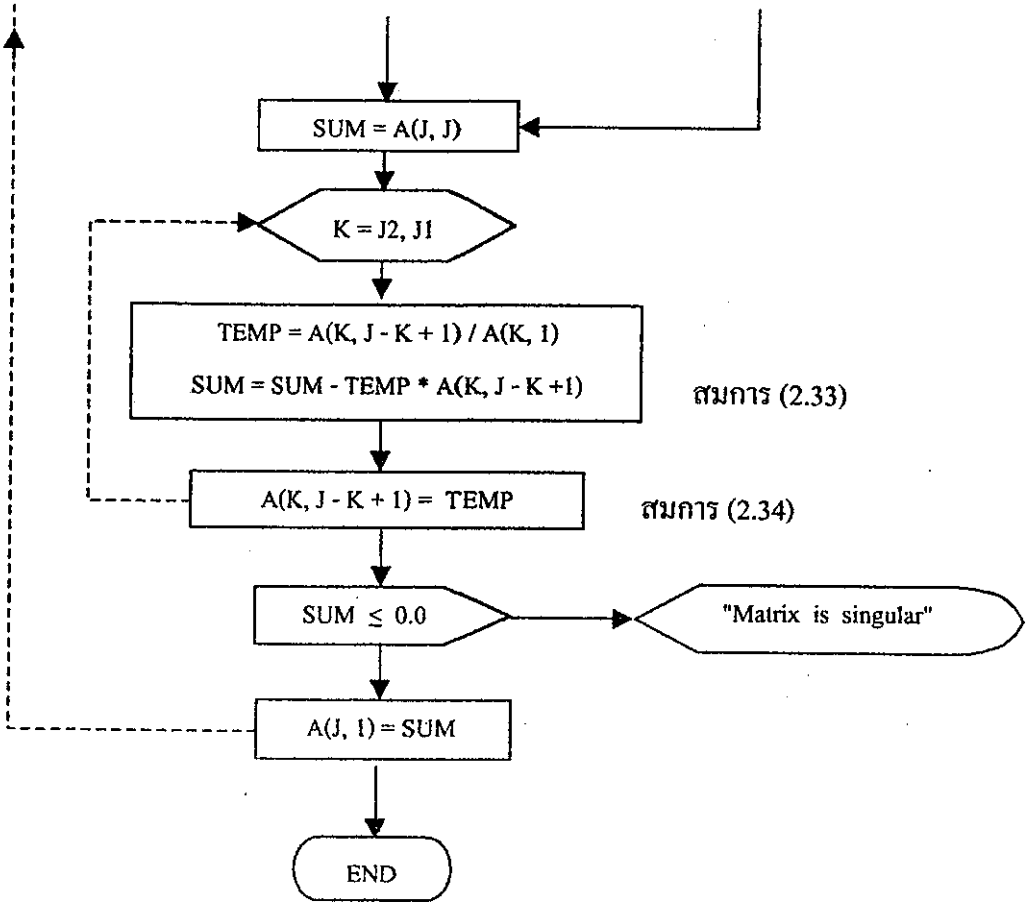
นอกจากนี้ สำหรับแถว NB สุดท้าย (ยกเว้นแถว N)

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

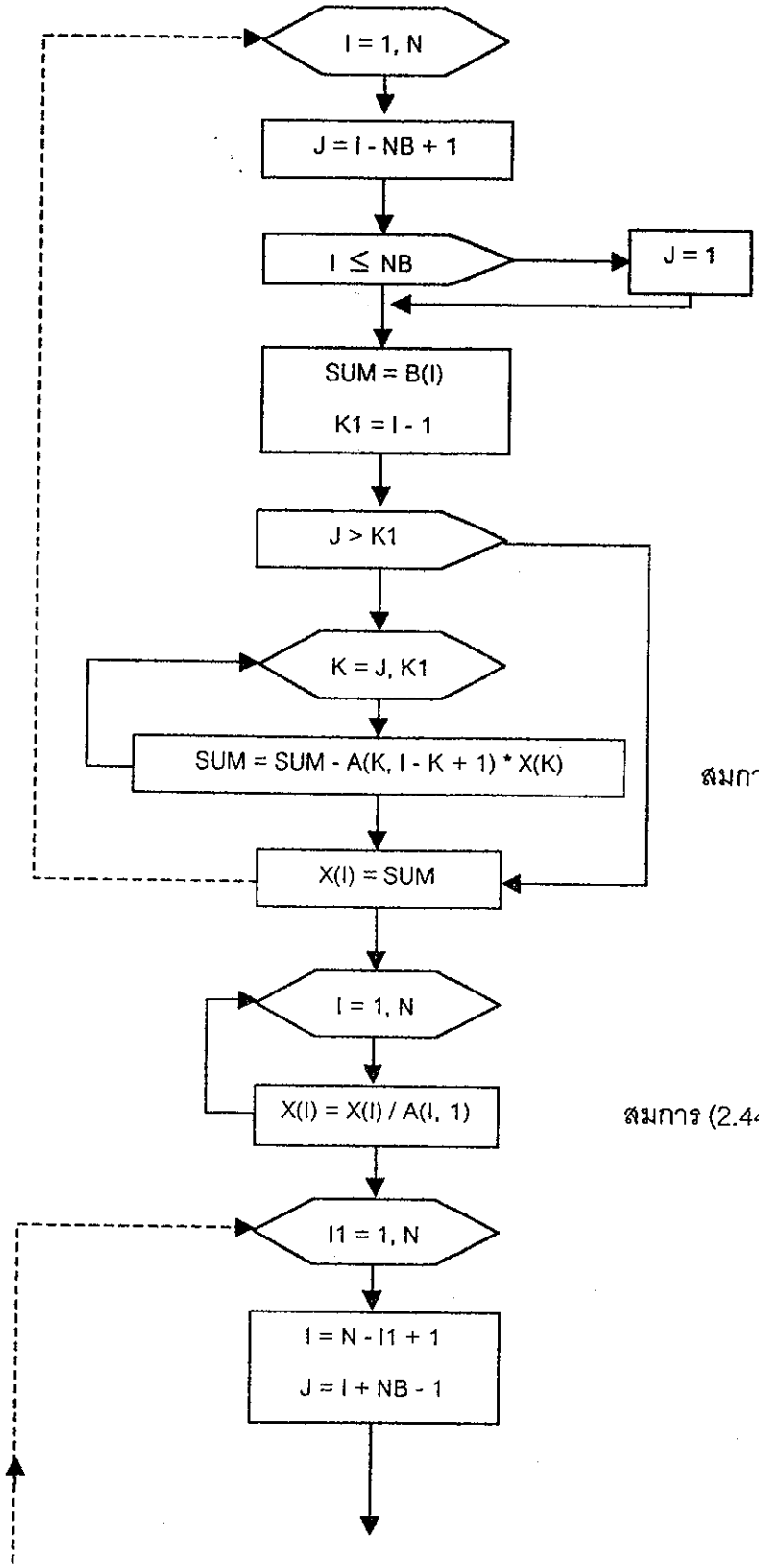
เลขประจำคอลัมน์ขององค์ประกอบในเมตริกซ์ $[\overline{U}]$ จะแสดงตำแหน่งจริงในอาเรย์สี่เหลี่ยมผืนผ้าเมื่อโปรแกรมทำการคำนวณค่าที่อยู่ระหว่าง z และ r จะเปลี่ยนไปอยู่ในค่าของ X และค่าสุดท้ายของ X จะถูกคำนวณโดยแทนค่ากลับ

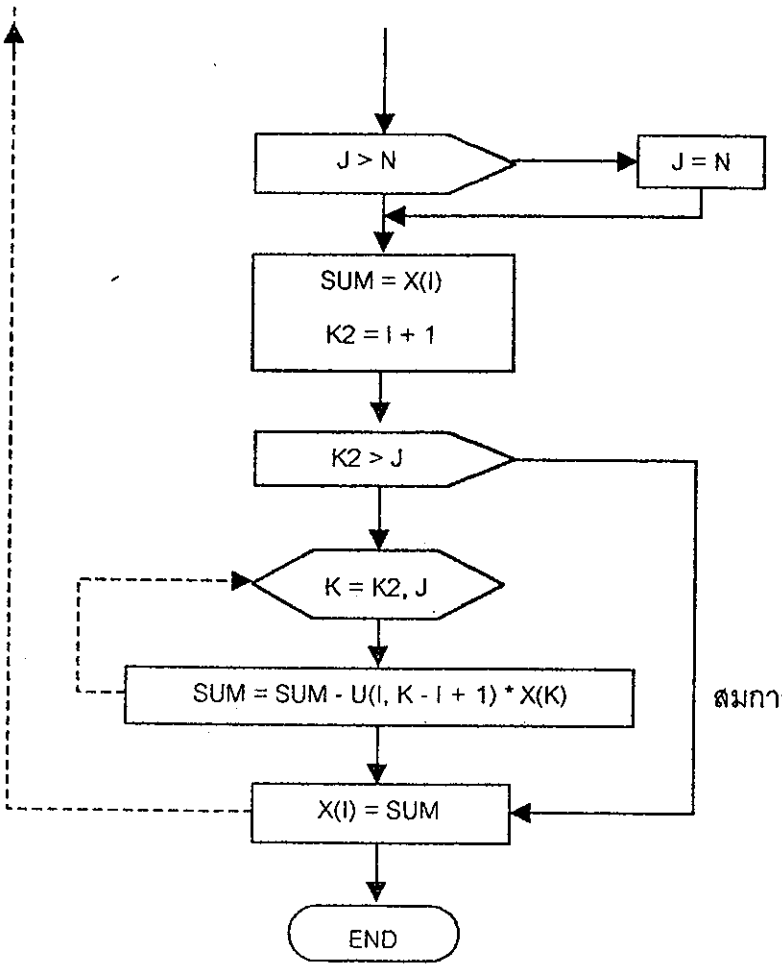
Flow Chart 12 : Subprogram BANFAC




SUM = A(J, J)
K = J2, J1
TEMP = A(K, J - K + 1) / A(K, 1)
SUM = SUM - TEMP * A(K, J - K +1)
สมการ (2.33)
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





สมการ (2.45)