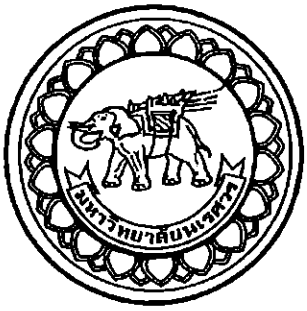


การพัฒนาโปรแกรมวิเคราะห์โครงข้อหมุนในระนาบด้วยภาษาพีเอชพี

DEVELOPMENT OF PLANE TRUSS ANALYSIS PROGRAM USING PHP LANGUAGE

นายพงษ์เพชร เดชศิริ รหัส 52364056

นางสาวสนธยา ตาเห็น รหัส 52364216

นายอาคุณย์ หารสาริกิจ รหัส 52364391

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ - 7 ส.ค. 2556
เลขทะเบียน 1634062
เลขเรียกหนังสือ ....
มหาวิทยาลัยนเรศวร QV 161 ก

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโยธา ภาควิชาวิศวกรรมโยธา

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2555

ใบรับรองปริญญานิพนธ์

ชื่อหัวข้อโครงงาน การพัฒนาโปรแกรมวิเคราะห์โครงข้อหมุนในระนาบด้วยภาษาพีเอชพี

ผู้ดำเนินโครงงาน นายพงษ์เพชร เดชศิริ รหัส 52364056

นางสาวสนธยา ตาเห็น รหัส 52364216

นายอาดุลย์ หารสาริกิจ รหัส 52364391

ที่ปรึกษาโครงงาน ผศ.ดร.สสิกรณ์ เหลืองวิชชเจริญ

สาขาวิชา วิศวกรรมโยธา

ภาควิชา วิศวกรรมโยธา

ปีการศึกษา 2555

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมโยธา

.....ประธานกรรมการ

(ผศ.ดร.สสิกรณ์ เหลืองวิชชเจริญ)

.....กรรมการ

(อาจารย์บุญพล มีไชโย)

.....กรรมการ

(อาจารย์ภัคพงศ์ หอมเนียม)

ชื่อหัวข้อโครงงาน การพัฒนาโปรแกรมวิเคราะห์โครงข้อหมุนในระนาบด้วยภาษาพีเอชพี

ผู้ดำเนินโครงงาน นายพงษ์เพชร เดชศิริ รหัส 52364056

นางสาวสนธยา ตาเห็น รหัส 52364216

นายอาดุลย์ หารสาริกิจ รหัส 52364391

ที่ปรึกษาโครงงาน ผศ.ดร.สสิกรณ์ เหลืองวิชชเจริญ

สาขาวิชา วิศวกรรมโยธา

ภาควิชา วิศวกรรมโยธา

ปีการศึกษา 2555

บทคัดย่อ

โครงงานนี้ได้เสนอแนวทางการพัฒนาโปรแกรมวิเคราะห์โครงข้อหมุนในระนาบด้วยภาษาพีเอชพีซึ่งเป็นภาษาคอมพิวเตอร์สำหรับพัฒนาเว็บเพจแบบไดนามิก ที่ได้รับความนิยมมากในปัจจุบัน โปรแกรมที่พัฒนาขึ้นนี้ใช้หลักการของวิธีสติฟเนสโดยตรงการวิเคราะห์หาคำตอบ ผู้พัฒนาได้ออกแบบการใช้งานโปรแกรมให้สะดวก โดยผู้ใช้งานสามารถป้อนค่าพารามิเตอร์ทุกอย่างได้ง่าย ผลการคำนวณเมื่อเปรียบเทียบกับโปรแกรมวิเคราะห์โครงสร้าง SUT-STRUCTOR พบว่าให้คำตอบถูกต้องตรงกันเป็นอย่างดี รายละเอียดโปรแกรมในส่วนของการคำนวณและการแสดงผลในรูปแบบกราฟฟิก ได้ถูกเสนอไว้ในรายงานฉบับนี้

Project title	DEVELOPMENT OF PLANE TRUSS ANALYSIS PROGRAM USING PHP LANGUAGE
Name	Mr. Pongpatch Dechsiri ID.52364056
	Miss. Sontaya Tahen ID.52364216
	Mr. Adun Hansarikit ID.52364391
Project advisor	Assit. Prof. Dr.Sasikorn Leungvichcharoen
Major	Civil Engineering
Department	Civil Engineering
Academic year	2012

Abstract

This project proposes the method to develop of plane truss analysis program using PHP Language, a very popular computer language for developing dynamic web pages. The method of direct stiffness method is used in the development of this program. This program is user-friendly all parameters can be easy to inputted. Numerical results of this program are in good agreement with the results of SUT-Structor structural analysis program. Parts of this program, i.e., the algorithms of computation and graphical output, are presented in this report.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้ สำเร็จลุล่วงไปได้ด้วยดีด้วยความช่วยเหลือของ อาจารย์สสิกรณ์ เหลืองวิชชเจริญ อาจารย์ที่ปรึกษาโครงงาน ได้ให้คำปรึกษาแนะนำ ชี้แนะแนวทางแก้ไขรวมถึงข้อคิดเห็นเสนอแนะต่างตลอดจนความดูแลเอาใจใส่ในการดำเนินโครงการมาโดยตลอด และขอขอบพระคุณคณะอาจารย์ประจำภาควิชาวิศวกรรมโยธา มหาวิทยาลัยนเรศวรทุกท่านที่ได้สอนวิชาความรู้เพื่อนำมาประยุกต์ใช้ในปริญญานิพนธ์ฉบับนี้

ขอขอบพระคุณคณะกรรมการทุกท่านที่มาร่วมการฟังบรรยายโครงงาน และได้ให้ข้อเสนอแนะที่เป็นประโยชน์ ซึ่งคณะผู้จัดทำจะนำไปปรับปรุงและพัฒนาโครงงานให้สมบูรณ์

ขอขอบพระคุณเจ้าหน้าที่ และพนักงานภาควิชาวิศวกรรมโยธา ที่กรุณาให้ความช่วยเหลือร่วมมือตลอดระยะเวลาการทำโครงงานนี้เป็นอย่างดียิ่ง

สุดท้ายนี้ผู้ดำเนินโครงงานขอกราบขอบพระคุณ บิดา มารดา ที่ได้ให้การดูแล อบรมสั่งสอนและให้กำลังใจด้วยดีเสมอมา ตลอดการดำเนินโครงงานจนสำเร็จการศึกษา

คณะผู้จัดทำโครงงานวิศวกรรม

นายพงษ์เพชร เดชศิริ

นางสาวสนธยา ตาเห็น

นายอาดุลย์ หารสาริกิจ

22 มีนาคม 2555

สารบัญ

หน้า

ใบรับรองปริญญานิพนธ์.....ก

บทคัดย่อภาษาไทย.....ข

บทคัดย่อภาษาอังกฤษ.....ค

กิตติกรรมประกาศ.....ง

สารบัญ.....จ-ฉ

สารบัญรูป.....ช-ซ

บทที่ 1 บทนำ

1.1 ความเป็นมาและความสำคัญของโครงงาน 1-1

1.2 วัตถุประสงค์ของโครงงาน 1-1

1.3 ขอบเขตของโครงงาน 1-1

1.4 ขั้นตอนการดำเนินงาน 1-2

1.5 ผลที่คาดว่าจะได้รับ 1-2

1.6 แผนการดำเนินงาน 1-2

1.7 รายละเอียดงบประมาณตลอดโครงการ 1-3

บทที่ 2 หลักการและทฤษฎีเบื้องต้น

2.1 ทฤษฎีการวิเคราะห์โครงสร้าง 2-1

2.2 การวิเคราะห์ด้วยวิธีการรวมสติฟเนสโดยตรง 2-1

2.3 ขั้นตอนการคำนวณโดยใช้วิธี Direct Stiffness Method 2-5

บทที่ 3 วิธีการดำเนินโครงงาน

3.1 แผนผังขั้นตอนการทำงานของโปรแกรม 3-1

3.2 รายละเอียดของโปรแกรม 3-2

สารบัญ (ต่อ)

หน้า

บทที่ 4 ผลการทดลองและวิเคราะห์

4.1 ตัวอย่างการใช้งานหน้าเว็บวิเคราะห์โครงข้อหมุนในระนาบ 4-1

4.2 วิเคราะห์ผลของโปรแกรม 4-7

บทที่ 5 บทสรุปและข้อเสนอแนะ

5.1 สรุปผลโครงงาน 5-1

5.2 ข้อเสนอแนะ 5-1

บรรณานุกรม

ภาคผนวก

ประวัติผู้เขียน

สารบัญรูป

รูปที่	หน้า
รูปที่ 2.1 แสดงค่าแรงและการเปลี่ยนตำแหน่งที่ปลายชิ้นส่วนในระบบโคออร์ดิเนตประจำตัวชิ้นส่วนโครงข้อหมุนในระนาบ (X, Y)	2-2
รูปที่ 2.2 แสดงค่าแรงและการเปลี่ยนตำแหน่งที่ปลายชิ้นส่วนในระบบโคออร์ดิเนตโกลบัล $(\overline{X}, \overline{Y})$ ของชิ้นส่วนโครงข้อหมุนในระนาบ	2-3
รูปที่ 2.3 แสดงมุม $\emptyset_x$, $\emptyset_y$ ของชิ้นส่วนโครงข้อหมุนในระนาบ	2-4
รูปที่ 3.1 ขั้นตอนการทำงานของโปรแกรมวิเคราะห์โครงข้อหมุนด้วยภาษาพีเอชพี	3-1
รูปที่ 3.2 การรับค่าจำนวน Node และ Member	3-2
รูปที่ 3.3 ตารางรับค่า Node	3-2
รูปที่ 3.4 ตารางการรับค่า Member	3-3
รูปที่ 3.5 การแสดงผลของตาราง Node	3-4
รูปที่ 3.6 การแสดงผลของตาราง Member	3-4
รูปที่ 3.7 แสดงโครงข้อหมุนที่จะทำการวิเคราะห์	3-5
รูปที่ 3.8 แสดงรูปโครงข้อหมุนและการเปลี่ยนแปลงที่เกิดขึ้น	3-5
รูปที่ 4.1 โจทย์ตัวอย่างที่ 1 (รูปที่ได้จากโปรแกรม SUT-Structor)	4-1
รูปที่ 4.2 การแสดงผลการคำนวณโจทย์ตัวอย่างที่ 1 (รูปที่ได้จากโปรแกรม SUT-Structor)	4-1
รูปที่ 4.3 รูปโครงสร้างภายหลังการเปลี่ยนแปลงของตัวอย่างที่ 1 (รูปที่ได้จากโปรแกรม SUT-Structor)	4-2
รูปที่ 4.4 แสดงตารางเพื่อป้อนค่าพารามิเตอร์ต่างๆในการคำนวณ (ตัวอย่างที่ 1)	4-3
รูปที่ 4.5 แสดงรูปโครงสร้างและแรงกระทำ (ตัวอย่างที่ 1)	4-3
รูปที่ 4.6 แสดงผลการคำนวณ (ตัวอย่างที่ 1)	4-4
รูปที่ 4.7 แสดงรูปโครงสร้างและโครงสร้างภายหลังการเปลี่ยนแปลง (ตัวอย่างที่ 1)	4-4
รูปที่ 4.8 โจทย์ตัวอย่างที่ 2 (รูปที่ได้จากโปรแกรม SUT-Structor)	4-5
รูปที่ 4.9 การแสดงผลการคำนวณโจทย์ตัวอย่างที่ 2 (รูปที่ได้จากโปรแกรม SUT-Structor)	4-5
รูปที่ 4.10 รูปโครงสร้างภายหลังการเปลี่ยนแปลง โจทย์ตัวอย่างที่ 2 (รูปที่ได้จากโปรแกรม SUT-Structor)	4-5
รูปที่ 4.11 แสดงตารางเพื่อป้อนค่าพารามิเตอร์ต่างๆในการคำนวณ (ตัวอย่างที่ 2)	4-6

สารบัญรูป (ต่อ)

รูปที่ 4.12 แสดงรูปโครงสร้างและแรงกระทำ (ตัวอย่างที่ 2) 4-6

รูปที่ 4.13 แสดงผลการคำนวณ (ตัวอย่างที่ 2) 4-7

รูปที่ 4.14 แสดงรูปโครงสร้างและโครงสร้างภายหลังการเปลี่ยนแปลง (ตัวอย่างที่ 2) 4-7

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของโครงงาน

การวิเคราะห์โครงสร้างถือว่าเป็นความรู้พื้นฐานที่มีความจำเป็นสำหรับวิศวกรในการคำนวณออกแบบเพื่อความมั่นคงปลอดภัยของโครงสร้าง ในปัจจุบันวิทยาการด้านคอมพิวเตอร์มีการพัฒนาและก้าวหน้าไปมาก ซึ่งการพัฒนาทางเทคโนโลยีนี้ส่งผลทำให้การคำนวณมีความแม่นยำและมีประสิทธิภาพมากยิ่งขึ้น เพื่อส่งเสริมการใช้วิทยาการความรู้ทางเทคโนโลยีมาผนวกกับความรู้ทางด้านวิศวกรรม คณะผู้จัดทำจึงมีความคิดในการพัฒนาโปรแกรมสำหรับการวิเคราะห์โครงข้อหมุนในระนาบขึ้นเพื่อจะเป็นประโยชน์ในการทำงานและการศึกษา

โปรแกรมภาษาพีเอชพีเป็นหนึ่งในภาษาสำหรับการพัฒนาเว็บที่มีผู้ใช้จากทั่วโลกเป็นจำนวนมาก โดยเฉพาะอย่างยิ่งในประเทศไทยนั้น พีเอชพีเป็นภาษาที่ได้รับความนิยมสูงสุด เนื่องจากไม่มีกฎเกณฑ์ หรือโครงสร้างทางภาษาที่ซับซ้อน ผู้ศึกษาจึงสามารถเรียนรู้ได้ในระยะเวลาอันสั้น คณะผู้จัดทำมีความคิดในการพัฒนาโปรแกรมภาษาพีเอชพีร่วมกับการวิเคราะห์โครงข้อหมุนในระนาบ เพื่อที่จะเป็นประโยชน์ต่อผู้สนใจที่จะศึกษาการวิเคราะห์โครงสร้างได้ใช้งานในรูปของเว็บเพจต่อไป

1.2 วัตถุประสงค์ของโครงงาน

เพื่อการพัฒนาโปรแกรมการวิเคราะห์โครงข้อหมุนในระนาบด้วยภาษาพีเอชพี และเผยแพร่ให้แก่ผู้สนใจศึกษาและประกอบการทำงานได้ใช้งานฟรีทางอินเตอร์เน็ต (Free Web Application) สำหรับต่อไป

1.3 ขอบเขตของโครงงาน

เขียนโปรแกรมวิเคราะห์โครงข้อหมุนในระนาบด้วยภาษาพีเอชพีโดยวิธีรวมสติฟเนสโดยตรง ซึ่งมีชนิดของฐานรองรับเป็นแบบยึดหมุนหรือแบบล้อเลื่อน (Hinge or Roller support) สามารถวิเคราะห์โครงสร้างที่มีรูปแรงประเภทแรงกระทำในแนวราบและแนวดิ่งแบบจุด (Concentrate Loads, Point Loads) ลักษณะของโครงสร้างมีการเชื่อมต่อกันแบบโครงข้อหมุน (Hinge joint)

1.4 ขั้นตอนการดำเนินงาน

1. คัดเลือกหัวข้อโครงงานที่น่าสนใจ
2. เสนอโครงงานกับอาจารย์ที่ปรึกษา(Research proposal)
3. ศึกษาเนื้อหาทางทฤษฎีของการวิเคราะห์โครงข้อหมุนแบบ 2 มิติด้วยวิธีสติฟเนสโดยตรง
4. ศึกษาเนื้อหาของภาษาพีเอชพี
5. กำหนดและเขียนขั้นตอนในการทำงาน
6. เขียนภาษาพีเอชพีให้ออกมาเป็นโปรแกรมวิเคราะห์โครงข้อหมุนในระนาบ
7. ทดสอบโปรแกรมและตรวจสอบความถูกต้องของโปรแกรมเพื่อปรับปรุงแก้ไข
8. ทำการตรวจสอบโดยอาจารย์ที่ปรึกษา
9. จัดทำเอกสารรายงานสรุปผลโครงการและจัดทำรูปเล่ม

1.5 ผลที่คาดว่าจะได้รับ

ได้โปรแกรมที่สามารถช่วยในการวิเคราะห์โครงข้อหมุนในระนาบที่มีประสิทธิภาพการคำนวณที่แม่นยำ และสามารถเผยแพร่ให้กับผู้ที่ต้องการใช้งานและศึกษาได้ใช้งานต่อไป

1.6 แผนการดำเนินงาน

เดือน / กิจกรรม	ตุลาคม				พฤศจิกายน				ธันวาคม				มกราคม				กุมภาพันธ์			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1. การนำเสนอโครงงาน																				
2. ตรวจดูสถานที่ทำโครงงาน																				
3. ติดต่อข้อมูลจากสำนักงานที่เกี่ยวข้อง																				
4. วิเคราะห์ปัญหาที่เกิดขึ้น																				
5. เขียนโครงการ																				

1.7 รายละเอียดงบประมาณตลอดโครงการ

1.	ค่าสำเนาเอกสารข้อมูล	1000 บาท
2.	ค่าวัสดุคอมพิวเตอร์	500 บาท
3.	ค่าจัดรูปเล่มและเผยแพร่	1500 บาท
	รวมเป็นเงิน	3000 บาท (สามพันบาทถ้วน)

หมายเหตุ ขออนุมัติถัวเฉลี่ยทุกรายการ

บทที่ 2

หลักการและทฤษฎีเบื้องต้น

2.1 ทฤษฎีการวิเคราะห์โครงสร้าง

การวิเคราะห์โครงสร้างที่ใหญ่หรือโครงสร้างที่ยุ่งยากส่วนมากจะมีค่าที่เราไม่ทราบหลายค่า (Redundant) ทำให้เป็นโครงสร้างแบบอินดีเทอร์มิเนตที่ต้องอาศัยการแก้สมการหลายชั้นตาม ปรกติการวิเคราะห์โครงสร้างแบบอินดีเทอร์มิเนตทำได้ 2 วิธี

1. Force Method หรือ Flexibility Method หรือ Compatibility Method

ในวิธีนี้ทำได้โดยการพยายามทำโครงสร้างให้เป็นแบบดีเทอร์มิเนตเสียก่อน ซึ่งทำได้โดยการกำหนดทิศทางของแรงไม่ทราบค่าแล้วหาการเคลื่อนที่ที่เกิดขึ้น ณ จุดที่แรงไม่ทราบค่านี้กระทำซึ่งการเคลื่อนที่ดังกล่าวจะมีพจน์ของแรงไม่ทราบค่าอยู่ด้วยจากนั้นจึงสร้างสมการโดยใช้เงื่อนไข ความต่อเนื่อง (Compatibility Condition) ซึ่งจำนวนสมการที่ได้จะเท่ากับจำนวนของแรงเมื่อแก้สมการ ก็จะได้คำตอบที่สมบูรณ์

2. Displacement Method หรือ Stiffness Method หรือ Equilibrium Method

ในวิธีนี้ค่าการเคลื่อนที่ของโครงสร้างจะใช้เป็นตัวไม่ทราบค่า ความสัมพันธ์ของการเคลื่อนที่ของโครงสร้างกับการเคลื่อนที่ของส่วนโครงสร้าง หาได้จากลักษณะสัมพันธ์ของโครงสร้าง และอาศัยหลักการสมดุลของแรงกับการเคลื่อนที่ ทำให้คำนวณหาตัวไม่ทราบค่าได้ ค่าของแรงภายในส่วนโครงสร้างต่างๆได้จากการแทนค่าของตัวไม่รู้ค่าลงในสมการของความสัมพันธ์ระหว่างแรงและการเคลื่อนที่ภายในโครงสร้าง

2.2 การวิเคราะห์ด้วยวิธีการรวมสติฟเนสโดยตรง

วิธีการรวมสติฟเนสโดยตรง (The Direct Stiffness Method)[2] ใช้กับโครงสร้างทั่วๆไปไม่ว่าจะเป็นโครงข้อหมุน โครงข้อแข็งระนาบ โครงข้อแข็งสามมิติ หรืออื่นๆ เพียงแต่เมตริกซ์ที่เกี่ยวข้อง (เช่น สติฟเนสเมตริกซ์ เมตริกซ์แปลงการเปลี่ยนตำแหน่ง) เปลี่ยนไปสำหรับปัญหาหนึ่งๆ นอกจากสำหรับโครงสร้างที่กำหนดให้ ตั้งระบบแกนคาร์ทีเซียน(x , y) เพื่อกำหนดเรขาคณิตของโครงสร้างทั้งระบบและระยะเปลี่ยนตำแหน่งของข้อต่อ ระบบโคออร์ดิเนตนี้เรียกว่า ระบบโคออร์ดิเนตโกลบัล การสังเคราะห์สติฟเนสของโครงสร้างกระทำได้โดยการพิจารณาลักษณะสมบัติของชิ้นส่วนย่อย ความต่อเนื่องของการเปลี่ยนแปลงตำแหน่ง และสภาวะสมดุลของข้อต่อ

2.2.1 สติฟเนสของชิ้นส่วนย่อยในระบบโคออร์ดิเนตประจำตัว (Local Coordinate System)

สำหรับชิ้นส่วนใดๆ สติฟเนสของชิ้นส่วนย่อยในระบบโคออร์ดิเนตประจำตัว (Local Coordinate System) ซึ่งผ่านแนวแกนของชิ้นนั้น (ตามรูปที่ 2.1) ความสัมพันธ์ของแรงที่ปลายชิ้นส่วน $\{S\}$ กับการเปลี่ยนตำแหน่งของปลายชิ้นส่วน $\{v\}$ เป็นตามสมการ

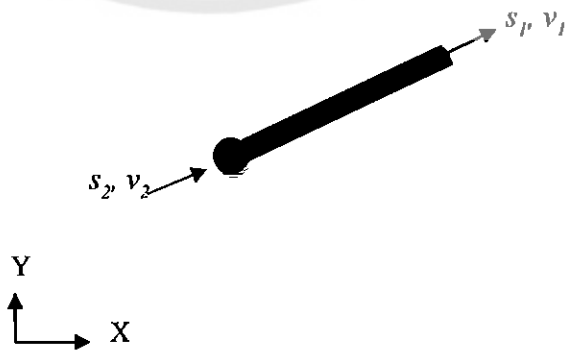
$$\{S\} = [k]\{v\} \qquad \text{--- (1)}$$

เมื่อ $[k]$ หมายถึง สติฟเนสของชิ้นส่วนย่อยในระบบโคออร์ดิเนตประจำตัว สำหรับชิ้นส่วนของโครงสร้างข้อหมุนใดๆ (ดังแสดงในรูปที่ 2.1) จะได้ว่า

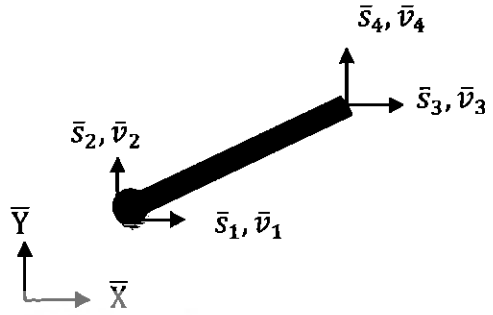
$$[k] = \frac{EA}{L}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \qquad \text{--- (2)}$$

2.2.2 ระบบโคออร์ดิเนตโกลบัล (Global Coordinate System)

เนื่องจากระบบโคออร์ดิเนตประจำตัวของชิ้นส่วนแต่ละอันหันในทิศทางต่างๆกัน เวกเตอร์ของแรงที่ปลายชิ้นส่วนแต่ละอัน จึงไม่สามารถรวมกันได้โดยตรง ในการพิจารณาสภาวะสมดุลข้อต่อเพื่อให้สามารถทำการรวมเวกเตอร์ได้โดยตรง จำเป็นต้องนิยามแรงและการเปลี่ยนตำแหน่งของแต่ละชิ้นส่วนในระบบโคออร์ดิเนตร่วมกัน ในที่นี้ใช้ระบบโคออร์ดิเนตโกลบัล แสดงในรูปที่ 2.1 และ รูปที่ 2.2 พึงระลึกว่า ชิ้นส่วนที่แสดงเป็นชิ้นส่วนเดียวกันแรงและการเปลี่ยนตำแหน่งใดๆ จึงเป็นแรงและการเปลี่ยนตำแหน่งเดียวกัน เพียงแต่เขียนแสดงในระบบโคออร์ดิเนตเป็นแรงและการเปลี่ยนตำแหน่งเดียวกัน เพียงแต่เขียนแสดงในระบบโคออร์ดิเนตที่ต่างกันเท่านั้น



รูปที่ 2.1 แสดงค่าแรงและการเปลี่ยนตำแหน่งที่ปลายชิ้นส่วนในระบบโคออร์ดิเนตประจำตัวชิ้นส่วนโครงข้อหมุนในระนาบ (X, Y)



รูปที่ 2.2 แสดงค่าแรงและการเปลี่ยนตำแหน่งที่ปลายชิ้นส่วนในระบบโคออร์ดิเนตโกลบัล $(\bar{X}, \bar{Y})$ ของชิ้นส่วนโครงข้อหมุนในระนาบ

2.2.3 เมตริกซ์แปลงสำหรับชิ้นส่วนโครงข้อหมุน (Truss Element Transformation Matrix)

จากการพิสูจน์ทางคณิตศาสตร์สามารถเขียนความสัมพันธ์ระหว่าง แรง หรือการเปลี่ยนตำแหน่งในระบบโคออร์ดิเนตประจำตัวกับระบบโคออร์ดิเนตโกลบัลของชิ้นส่วนย่อยใดๆได้ดังนี้

$$\{v\} = [a]\{\bar{v}\} \qquad \text{--- (3)}$$

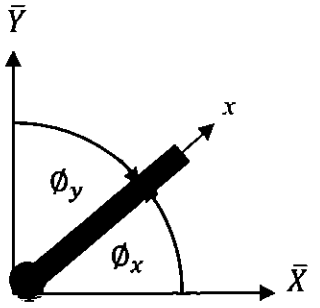
$$\{S\} = [a]\{\bar{S}\} \qquad \text{--- (4)}$$

เมื่อ $\{\bar{S}\}$ และ $\{\bar{v}\}$ หมายถึง แรง และการเปลี่ยนตำแหน่งที่จุดต่อ (Node) ของชิ้นส่วนย่อยในระบบโคออร์ดิเนตโกลบัล และ [a] หมายถึงเมตริกซ์แปลงของชิ้นส่วนย่อย (Element Transformation Matrix)

สำหรับชิ้นส่วนโครงข้อหมุนใน 2 มิติเฉพาะการจัดเรียงลำดับของแรงและการเคลื่อนที่ ณ จุดต่อเป็นเป็นดังรูปที่ 2.1 และรูปที่ 2.2 จะได้ว่า

$$[a] = \begin{bmatrix} cos\emptyset_x & cos\emptyset_y & 0 & 0 \\ 0 & 0 & cos\emptyset_x & cos\emptyset_y \end{bmatrix} \qquad \text{--- (5)}$$

โดย $\emptyset_x$ และ $\emptyset_y$ หมายถึงมุมที่วัดจากทิศทางตามยาวของชิ้นส่วนไปยังแกน $\bar{x}$ และแกน $\bar{y}$ ตามลำดับ ดังแสดงในรูปที่ 2.3



รูปที่ 2.3 แสดงมุม$\emptyset_x$, $\emptyset_y$ ของชิ้นส่วนโครงข้อหมุนในระนาบ

2.2.4 สติฟเนสของชิ้นส่วนย่อยในระบบโคออร์ดิเนตโกลบัล

แทนสมการ (3) ลงใน (1) จะได้ว่า

$$\{S\} = [k]\ [a]\ [\bar{v}] \qquad \text{--- (6)}$$

จากนั้นแทน(4) ลงในสมการ (6) แล้วใช้คุณสมบัติเชิงตั้งฉากในสมการ (5) ได้ความสัมพันธ์ระหว่าง แรง และ การเคลื่อนที่ที่ปลายในระบบโคออร์ดิเนตโกลบัล ดังนี้

$$\{\bar{S}\} = [a]^T\ [k]\ [a]\ [\bar{v}] \qquad \text{--- (7a)}$$

หรือเขียนใหม่ได้เป็น

$$\{S\} = [\bar{k}]\ \{\bar{S}\} \qquad \text{--- (7b)}$$

เมื่อ $[\bar{k}]$ หมายถึง สติฟเนตของชิ้นส่วนในระบบโคออร์ดิเนตโกลบัล

$$[\bar{k}] = [a]^T\ [k]\ [a] \qquad \text{--- (8)}$$

สำหรับโครงข้อหมุนใน 2 มิติ จากสมการที่ (5) และ (8) จะได้ว่า

$$[\bar{k}] = \frac{EA}{L}\begin{bmatrix} cos\emptyset_x & 0 \\ cos\emptyset_y & 0 \\ 0 & cos\emptyset_x \\ 0 & cos\emptyset_y \end{bmatrix}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} cos\emptyset_x & cos\emptyset_y & 0 & 0 \\ 0 & 0 & cos\emptyset_x & cos\emptyset_y \end{bmatrix} \quad \text{- -(9a)}$$

$$[\bar{k}] = \frac{EA}{L}\begin{bmatrix} cos^2\emptyset_x & cos\emptyset_x cos\emptyset_y & -cos^2\emptyset_x & -cos\emptyset_x cos\emptyset_y \\ cos\emptyset_x cos\emptyset_y & cos^2\emptyset_y & -cos\emptyset_x cos\emptyset_y & -cos^2\emptyset_y \\ -cos^2\emptyset_x & -cos\emptyset_x cos\emptyset_y & cos^2\emptyset_x & cos\emptyset_x cos\emptyset_y \\ -cos\emptyset_x cos\emptyset_y & -cos^2\emptyset_y & cos\emptyset_x cos\emptyset_y & cos^2\emptyset_y \end{bmatrix} \text{--- (9b)}$$

2.3 ขั้นตอนการคำนวณโดยใช้วิธี Direct Stiffness Method

1. กำหนดเมตริกซ์ของการเคลื่อนที่ที่จุดต่อซึ่งเป็นตัวไม่รู้ค่า $\{U^*\}$

2. จากสมการที่ (2) หาสติฟเนสในโคออร์ดิเนตประจำตัว $[k]$ ของแต่ละ Member

ดังนี้ $[k] = \frac{EA}{L}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$

3. จากสมการที่ (5) คำนวณหาค่าการเปลี่ยนตำแหน่ง Transformation Matrix ของแต่ละ Member สำหรับโครงข้อหมุนในระนาบ

$$[a] = \begin{bmatrix} cos\emptyset_x & cos\emptyset_y & 0 & 0 \\ 0 & 0 & cos\emptyset_x & cos\emptyset_y \end{bmatrix}$$

4. หาสติฟเนสในโคออร์ดิเนตโกลบัล $[\bar{k}]$ ของแต่ละ Member จากสมการที่ (8)

$$[\bar{k}] = [a]^T [k] [a]$$

5. คำนวณหา $[\bar{K}^*]$ ของโครงสร้าง

โดยที่ $[\bar{K}^*] = \sum_{m=1}^{member} [\bar{K}^m]$

เมื่อ $[\bar{K}^m]$ หมายถึง สติฟเนสในโคออร์ดิเนตโกลบัลของชิ้นส่วนย่อย

6. หา $\{U^*\}$ จากสมการ $\{U^*\} = [K^*]^{-1} \{P^*\}$

เมื่อ $\{P^*\}$ คือ แรงภายนอกกระทำ ณ ข้อต่อของโครงสร้างในทิศทางและตำแหน่งเดียวกับการเคลื่อนที่ $\{U^*\}$

7. คำนวณหาแรงภายในส่วนโครงสร้าง จากสมการ $\{S\} = [k]\ [a]\ [\bar{v}]$

เมื่อได้ $\{\bar{v}\}$ ได้จาก $\{U^*\}$ ในทิศทางและตำแหน่งที่ถูกต้องของชิ้นส่วนย่อย

8. คำนวณหาแรงที่ฐานรองรับ (Support Reactions) และที่ปลายชิ้นส่วนในโครงสร้างจากสมการ

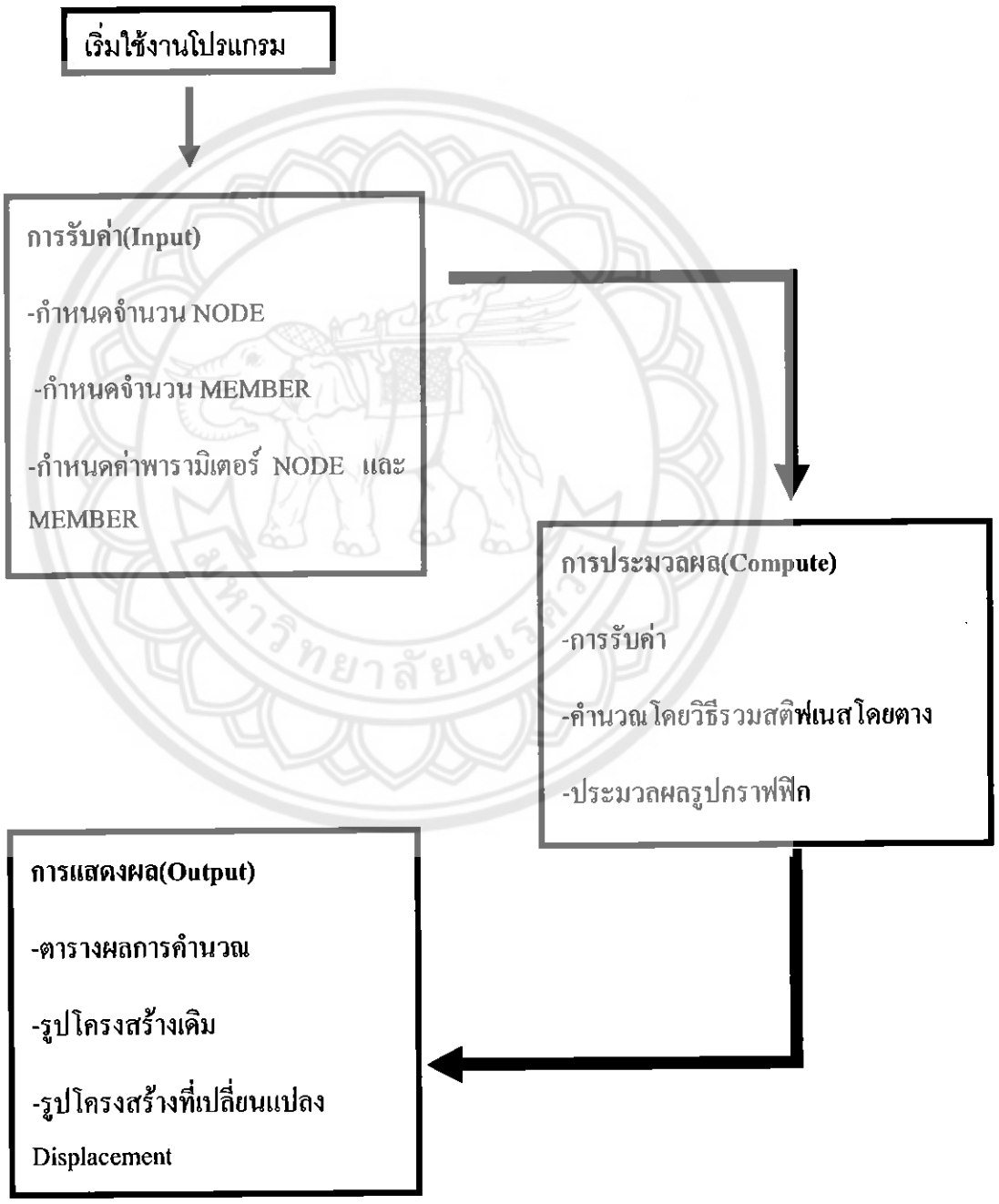
$\{\bar{S}\} = [a]^T \{S\}$

จากขั้นตอนข้างต้นสามารถนำไปใช้ในการคำนวณเพื่อวิเคราะห์โครงข้อหมุน 2 มิติโดยโปรแกรมภาษาพีเอชพี

บทที่ 3

วิธีการดำเนินโครงงาน

3.1 แผนผังขั้นตอนการทำงานของโปรแกรม

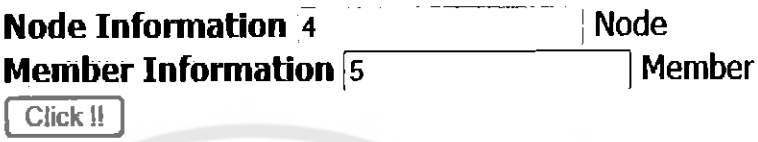


รูปที่ 3.1 ขั้นตอนการทำงานของโปรแกรมวิเคราะห์โครงข้อหมุนด้วยภาษาพีเอชพี

3.2 รายละเอียดของโปรแกรม

การนำเข้าข้อมูล (Input)

ในขั้นต้นนั้นโปรแกรมจะรับค่าจำนวน Node และ จำนวน Member จากนั้นก็จะปรากฏตารางของ Node และ Member แล้วทำการป้อนข้อมูลค่าดังต่อไปนี้



รูปที่ 3.2 การรับค่าจำนวน Node และ Member

1. ตาราง Node

- ค่าโคออร์ดิเนต (x,y) ของ Node ในช่อง X และY โดยเริ่มต้นที่ Node **แรกมีค่าเป็น (0, 0)**
- ค่า Displacement (ux,uy) ของ Node ในช่อง Ux และ Uy โดยกำหนดให้ **มีค่าเป็นศูนย์เมื่อ** ตำแหน่งนั้นมีจุดที่รองรับตามแกน
- ค่าแรงภายนอก (Force) ที่กระทำ ณ Node ใดๆ ในช่อง Fx และ Fy **โดยมีเงื่อนไข คือ**

Fx เป็นบวกเมื่อมีแรงกระทำแนวราบมีทิศทางไปทางขวา

Fx เป็นลบเมื่อมีแรงกระทำแนวราบมีทิศทางไปทางซ้าย

Fy เป็นบวกเมื่อมีแรงกระทำแนวดิ่งมีทิศทางขึ้น

Fy เป็นลบเมื่อมีแรงกระทำแนวดิ่งมีทิศทางลง

Node Information

Node	X(m)	Y(m)	Ux(m)	Uy(m)	Fx(kg)	Fy(kg)
1	0	0	0	0		
2	0	6	0			
3	4	3			10000	-2250
4	8	0				1500

รูปที่ 3.3 ตารางรับค่า Node

ในช่องที่ว่าง หมายถึง ค่าที่จะได้มาจากการคำนวณและนำกลับมาเติมในตาราง

2. ตาราง Member

– Start Node จะเป็นตัวกำหนดว่า Member นั้นๆมีจุดเริ่มต้นที่ Node ใด

– End Node จะเป็นตัวกำหนดว่า Member นั้นๆมีจุดสิ้นสุดที่ Node ใด

– EA คือค่าโมดูลัสคูณด้วยพื้นที่ของวัสดุโครงสร้าง

Member Information

Member	Start Node	End Node	EA(kg)
1	2	3	1000000
2	3	4	1000000
3	1	2	1000000
4	1	3	1000000
5	1	4	1000000

ต่อไป

รูปที่ 3.4 ตารางการรับค่า Member

การประมวลผล(Compute)

ในการคำนวณของโปรแกรม จะทำการรับค่าจากตารางข้อมูล โดยมีลักษณะการคำนวณแบบแยกชิ้นส่วนแต่ละ Member โดยเริ่มต้นคำนวณที่

1. ความยาว L = (Y2-Y1)/(X2-X1)
2. มุม(องศา) = Arctan (Y2-Y1)/(X2-X1)

   โดยที่ X1, Y1 มาจากการดึงค่าโคออร์ดิเนตที่ Start Node ของ Member ใดๆ

   X2, Y2 มาจากการดึงค่าโคออร์ดิเนตที่ End Node ของ Member ใดๆ
3. โปรแกรมจะดึงค่ามุมแล้วสร้างสติฟเนสเมตริกซ์ของชิ้นส่วน (member stiffness matrix)
4. โปรแกรมจะดึงค่าพารามิเตอร์ข้างต้นเพื่อเข้าสู่กระบวนการ Direct Stiffness
5. โปรแกรมทำการดึงค่าโคออร์ดิเนตโกลบัลของแต่ละชิ้นส่วนเพื่อสร้างเมตริกซ์ของโครงสร้าง(รวมเมตริกซ์)
6. โปรแกรมทำการวิเคราะห์ค่า Ux, Uy (displacement)
7. โปรแกรมทำคำสั่งลดค่าในเมตริกซ์เพื่อให้ได้เมกตริกซ์ดังรูป
8. เมื่อทำการลดค่าแล้วโปรแกรมก็จะทำการแทนค่ากลับเพื่อที่จะหาค่า displacement ของโครงสร้าง
9. เมื่อได้ค่า displacement แล้ว โปรแกรมจะนำค่าที่ได้ไปหา แรงที่กระทำของแต่ละ Node จากนั้นค่าของแรงที่กระทำในแต่ละ Node จะถูกนำไปหาค่าแรงภายใน

การแสดงผล(Out put)

ในการแสดงของโปรแกรมจะแยกการแสดงผลออกเป็น ตารางค่าการคำนวณ รูปของโครงสร้างเดิมและโครงสร้างใหม่ที่มีการเปลี่ยนแปลงของ Displacement

ในส่วนของตารางจะแสดงผลดังรูปตัวอย่างต่อไปนี้

Node Information

Node	X(m)	Y(m)	Ux(m)	Uy(m)	Fx(kg)	Fy(kg)
1	0	0	0	0	-5500	750
2	0	6	0	-0.02025	-4500	0
3	4	3	0.03884	-0.01533	10000	-2250
4	8	0	0.016	-0.02496	-0	1500

รูปที่ 3.5 การแสดงผลของตาราง Node

ตาราง Node จะแสดงค่า Ux,Uy และ Fx,Fy ใหม่ที่ได้จากการคำนวณ

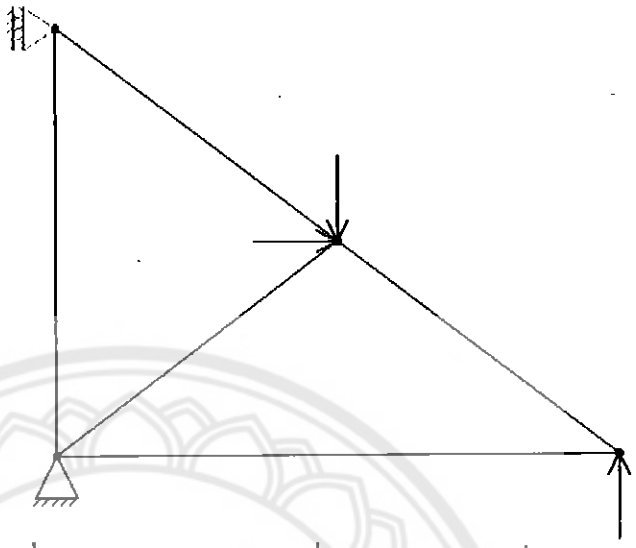
Member Information

Member	Start Node	End Node	EA(kg)	L(m)	Angle(deg)	F(kg)
1	2	3	1000000	5	-36.8698	5625
2	3	4	1000000	5	-36.8698	-2500
3	1	2	1000000	6	90	-3375
4	1	3	1000000	5	36.8699	4375
5	1	4	1000000	8	0	2000

รูปที่ 3.6 การแสดงผลของตาราง Member

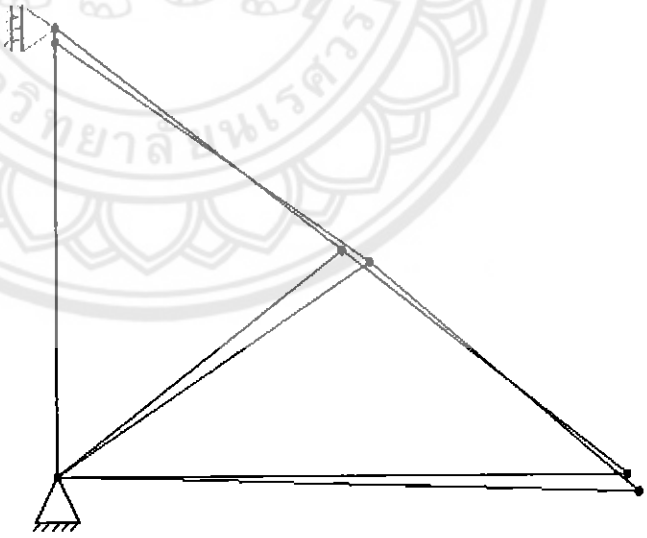
ตาราง Member จะแสดงค่า ความยาว มุม (องศา) และแรงภายในของแต่ละ Member

การแสดงรูปของโครงสร้าง



รูปที่ 3.7 แสดงโครงข้อหมุนที่จะทำการวิเคราะห์

รูปโครงสร้างเดิมจะเป็นการรับค่าจากการป้อนข้อมูลลงในตารางการ**รับค่าของ Node และ** Member จากนั้นจะประมวลผลออกมาเป็นรูปที่มีรายละเอียดแสดงโครงสร้าง **จุดรองรับและแรง** ภายนอก (Fx,Fy)



รูปที่ 3.8 แสดงรูปโครงข้อหมุนและการเปลี่ยนแปลงที่เกิดขึ้น

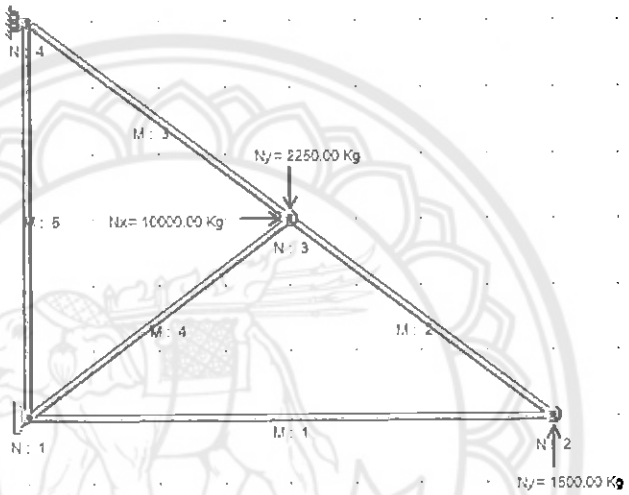
รูปโครงสร้างใหม่นี้จะแสดงเป็นโครงสร้างเดิมโดยที่มีแนวเส้นการเปลี่ยนของ Displacement

บทที่ 4

ผลการทดลองและวิเคราะห์

4.1 ตัวอย่างการใช้งานหน้าเว็บวิเคราะห์โครงข้อหมุนในระนาบ

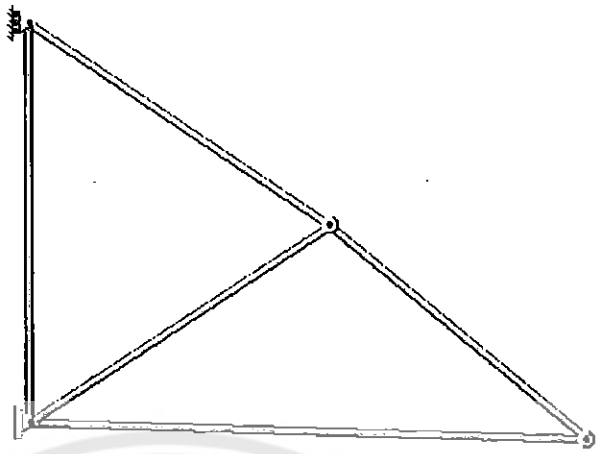
ตัวอย่างที่ 1 รูปตัวอย่างการคำนวณด้วยโปรแกรม SUT-Structor โดยการกำหนดพารามิเตอร์และลักษณะของโครงสร้างด้วยกราฟฟิกกำหนดดังรูป



รูปที่ 4.1 โจทย์ตัวอย่างที่ 1 (รูปที่ได้จากโปรแกรม SUT-Structor)

Member	Fx.i	Fy.i	Mz.i	Fx.j	Fy.j	Mz.j
1	-2000.00	0.00	0.00	2000.00	0.00	0.00
2	2500.00	0.00	0.00	-2500.00	0.00	0.00
3	-5625.00	0.00	0.00	5625.00	0.00	0.00
4	-4375.00	0.00	0.00	4375.00	0.00	0.00
5	3375.00	0.00	0.00	-3375.00	0.00	0.00

รูปที่ 4.2 การแสดงผลการคำนวณโจทย์ตัวอย่างที่ 1 (รูปที่ได้จากโปรแกรม SUT-Structor)



รูปที่ 4.3 รูปโครงสร้างภายหลังการเปลี่ยนแปลงของตัวอย่างที่ 1 (รูปที่ได้จากโปรแกรม SUT-Structor)

ป้อนค่าพารามิเตอร์ที่ใช้ในการคำนวณการวิเคราะห์โครงข้อหมุนด้วยภาษาพีเอชพี

1.จำนวน Node จำนวน Member

2.กด Click!!

3.การป้อนข้อมูลในตาราง Node

3.1 ป้อนข้อมูลพิกัด (X,Y) ของแต่ละ Node

3.2 กำหนดค่า Ux,Uy(Displacement)โดยมีเงื่อนไขคือ

เมื่อจุดรองรับเป็น Hinge support ให้ Ux และ Uy เป็น 0

เมื่อจุดรองรับเป็น Roller support ให้ Ux หรือ Uy เป็น 0 เพียงค่าเดียว

3.3 กำหนดค่า Fx,Fy ที่กระทำในแต่ละ Node (ในที่นี้โปรแกรมไม่กำหนดหน่วยแรงโดยผู้ทำสามารถกำหนดเวยความเข้าใจของตนเอง)

4.การป้อนข้อมูลในตาราง Member

4.1 กำหนดจุด Start Node และ End Node ของแต่ละ Member

4.2 กำหนดค่า EA

5. กดคำนวณ

ดังแสดงในรูปตัวอย่างต่อไปนี้

Node Information

Node	X(m)	Y(m)	Ux(m)	Uy(m)	Fx(kg)	Fy(kg)
1	0	0	0	0		
2	0	6	0			
3	4	3			10000	-2250
4	8	0				1500

Member Information

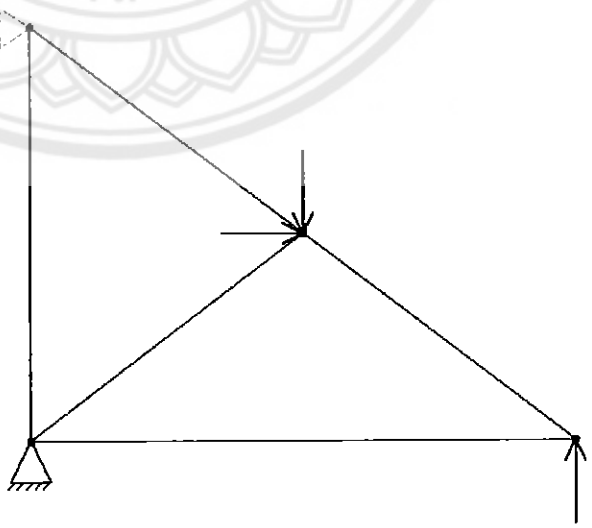
Member	Start Node	End Node	EA(kg)
1	2	3	1000000
2	3	4	1000000
3	1	2	1000000
4	1	3	1000000
5	1	4	1000000

ต่อไป

รูปที่ 4.4 แสดงตารางเพื่อป้อนค่าพารามิเตอร์ต่างๆในการคำนวณ **(ตัวอย่างที่ 1)**

การแสดงผล

โปรแกรมจะทำการประมวลผลและแสดงข้อมูลที่นำเข้า ผลการคำนวณ แสดงรูปของโครงสร้าง ดังแสดงในรูป



รูปที่ 4.5 แสดงรูปโครงสร้างและแรงกระทำ (ตัวอย่างที่ 1)

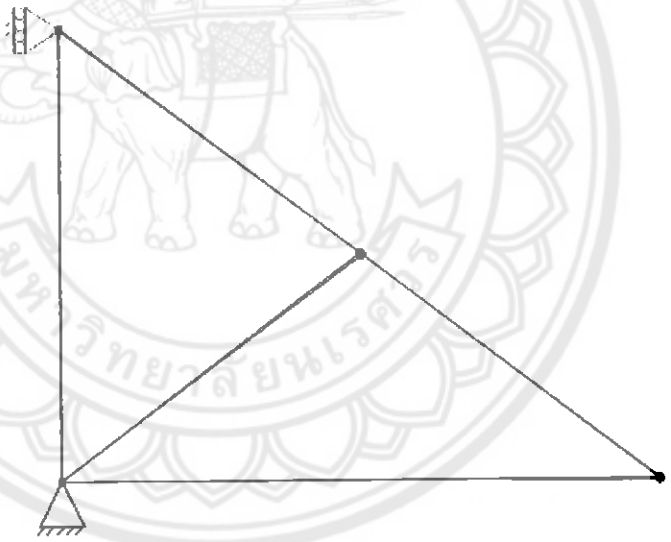
Node Information

Node	X(m)	Y(m)	Ux(m)	Uy(m)	Fx(kg)	Fy(kg)
1	0	0	0	0	-5500	750
2	0	6	0	-0.02025	-4500	0
3	4	3	0.03884	-0.01533	10000	-2250
4	8	0	0.016	-0.02496	-0	1500

Member Information

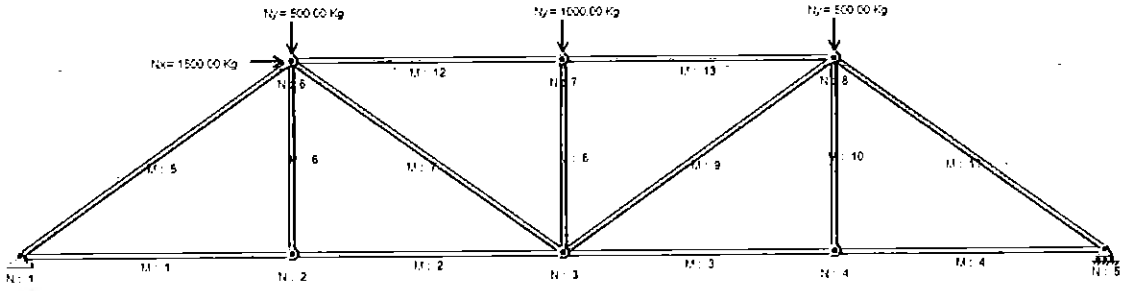
Member	Start Node	End Node	EA(kg)	L(m)	Angle(deg)	F(kg)
1	2	3	1000000	5	-36.8698	5625
2	3	4	1000000	5	-36.8698	-2500
3	1	2	1000000	6	90	-3375
4	1	3	1000000	5	36.8699	4375
5	1	4	1000000	8	0	2000

รูปที่ 4.6 แสดงผลการคำนวณ (ตัวอย่างที่ 1)



รูปที่ 4.7 แสดงรูปโครงสร้างและโครงสร้างภายหลังการเปลี่ยนแปลง (ตัวอย่างที่ 1)

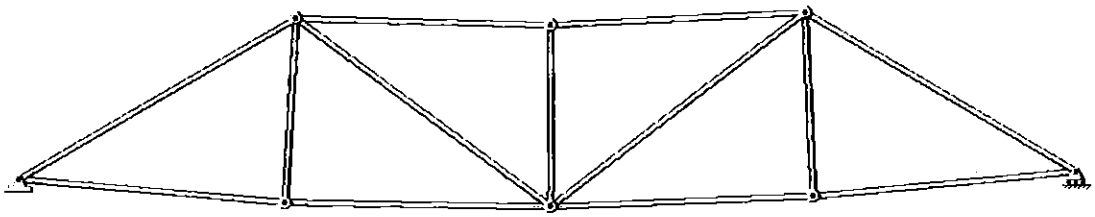
ตัวอย่างที่ 2 รูปตัวอย่างการคำนวณด้วยโปรแกรม SUT-Structor โดยการกำหนดพารามิเตอร์และลักษณะของโครงสร้างด้วยกราฟฟิกกำหนดดังรูป



รูปที่ 4.8 โจทย์ตัวอย่างที่ 2 (รูปที่ได้จากโปรแกรม SUT-Structor)

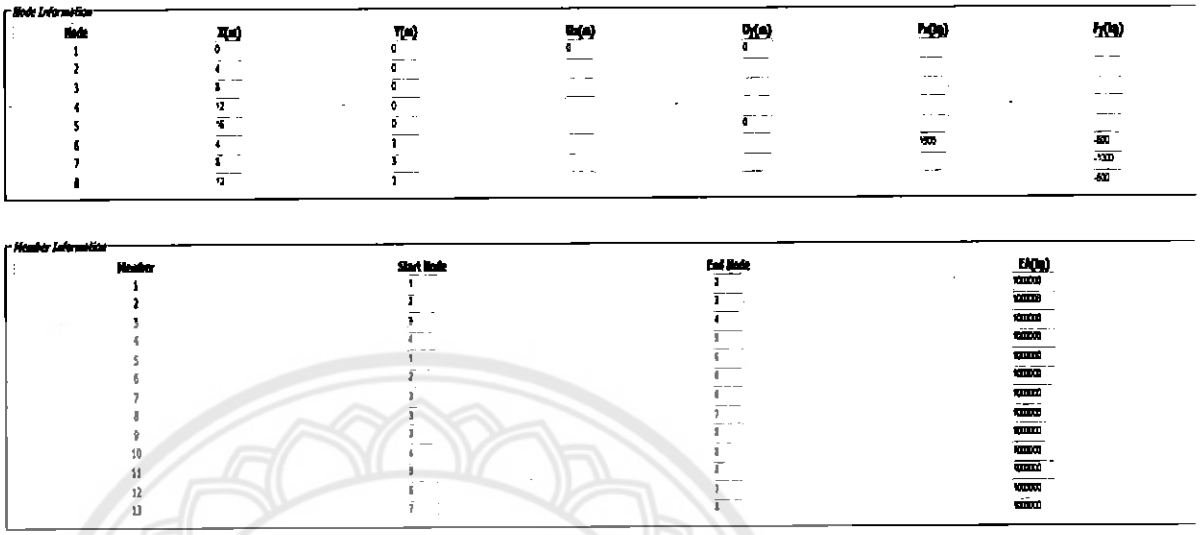
Member	Fx.i	Fy.i	Mz.i	Fx.j	Fy.j	Mz.j
1	-2458.33	0.00	0.00	2458.33	0.00	0.00
2	-2458.33	0.00	0.00	2458.33	0.00	0.00
3	-1708.33	0.00	0.00	1708.33	0.00	0.00
4	-1708.33	0.00	0.00	1708.33	0.00	0.00
5	1197.92	0.00	0.00	-1197.92	0.00	0.00
6	0.00	0.00	0.00	0.00	0.00	0.00
7	-364.58	0.00	0.00	364.58	0.00	0.00
8	1000.00	0.00	0.00	-1000.00	0.00	0.00
9	-1302.08	0.00	0.00	1302.08	0.00	0.00
10	0.00	0.00	0.00	0.00	0.00	0.00
11	2135.42	0.00	0.00	-2135.42	0.00	0.00
12	2750.00	0.00	0.00	-2750.00	0.00	0.00
13	2750.00	0.00	0.00	-2750.00	0.00	0.00

รูปที่ 4.9 การแสดงผลการคำนวณ โจทย์ตัวอย่างที่ 2 (รูปที่ได้จากโปรแกรม SUT-Structor)



รูปที่ 4.10 รูปโครงสร้างภายหลังการเปลี่ยนแปลง โจทย์ตัวอย่างที่ 2
(รูปที่ได้จากโปรแกรม SUT-Structor)

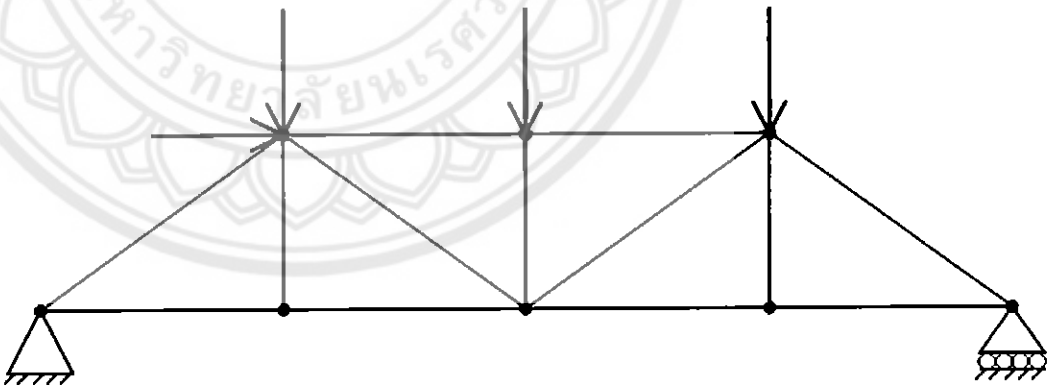
การคำนวณวิเคราะห์โครงข้อหมุนในระนาบด้วยภาษาพีเอชพีของตัวอย่างที่ 2



รูปที่ 4.11 แสดงตารางเพื่อป้อนค่าพารามิเตอร์ต่างๆในการคำนวณ (ตัวอย่างที่ 2)

การแสดงผลของหน้าเว็บ

โปรแกรมจะทำการประมวลผลและแสดงข้อมูลที่นำเข้า ผลการคำนวณ แสดงรูปของโครงสร้าง ดังแสดงในรูป



รูปที่ 4.12 แสดงรูปโครงสร้างและแรงกระทำของ (ตัวอย่างที่ 2)

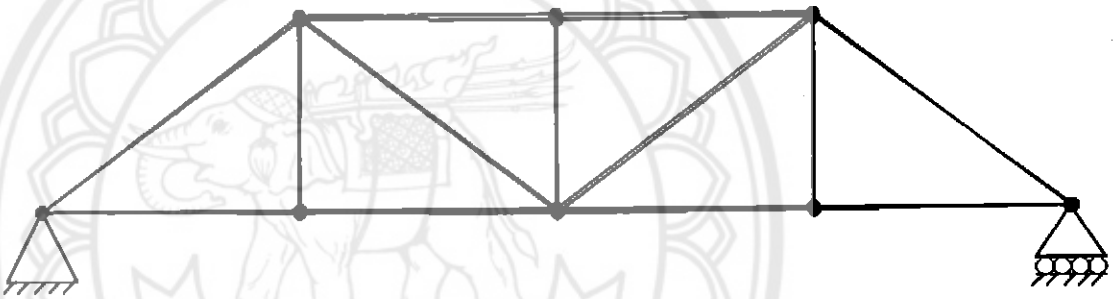
Node Information

Node	X(m)	Y(m)	Ux(m)	Uy(m)	Fx(kg)	Fy(kg)
1	0	0	0	0	-1500	718.75
2	4	0	0.00983	-0.05278	-0	0
3	8	0	0.01967	-0.07239	-0	0
4	12	0	0.0265	-0.04878	0	0
5	16	0	0.03333	0	-0	1281.25
6	4	3	0.0321	-0.05278	1500	-500
7	8	3	0.0211	-0.07539	-0	-1000
8	12	3	0.0101	-0.04878	-0	-500

Member Information

Member	Start Node	End Node	EA(kg)	L(m)	Angle(deg)	F(kg)
1	1	2	1000000	4	0	2458.3333
2	2	3	1000000	4	0	2458.3333
3	3	4	1000000	4	0	1708.3333
4	4	5	1000000	4	0	1708.3333
5	1	6	1000000	5	36.8699	-1197.9167
6	2	6	1000000	3	90	0
7	3	6	1000000	5	-36.8698	-364.5833
8	3	7	1000000	3	90	-1000
9	3	8	1000000	5	36.8699	1302.0833
10	4	8	1000000	3	90	0
11	5	8	1000000	5	-36.8698	2135.4167
12	6	7	1000000	4	0	-2750
13	7	8	1000000	4	0	-2750

รูปที่ 4.13 แสดงผลการคำนวณ (ตัวอย่างที่ 2)



รูปที่ 4.14 แสดงรูปโครงสร้างและโครงสร้างภายหลังการเปลี่ยนแปลง (ตัวอย่างที่ 2)

4.2 วิเคราะห์ผลของโปรแกรม

เมื่อเปรียบเทียบผลการวิเคราะห์ของหน้าเว็บที่เขียนขึ้น กับการวิเคราะห์ของโปรแกรม SUT-Structor พบว่าการแสดงผลทั้งหมดสามารถแสดงออกหน้าเว็บเพจได้ และให้ผลการคำนวณดังแสดงในตารางมีค่าที่ตรงกัน ในการแสดงผลรูปนั้นในส่วนของโครงสร้างเดิมแสดงผลได้ตรงกัน และรูปที่มีการเปลี่ยนแปลง Displacement โปรแกรมวิเคราะห์โครงข้อหมุนในระนาบด้วยภาษาพีเอชพีจะแสดงการเปลี่ยนแปลงของ Displacement ได้ไม่ชัดเจนเท่ากับ โปรแกรม SUT-Structor ซึ่งจากการพิจารณาในการคำนวณจะเห็นว่าค่าของ EA มีผลต่อความชัดเจนต่อการแสดงผลของการเปลี่ยนแปลง Displacement

บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 สรุปผลโครงงาน

- โปรแกรมสามารถใช้งานกับเบราว์เซอร์ Internet Exploler, Google Chrome, Mozilla Firefox, Opera, Safari ได้
- สามารถทำงานบนโทรศัพท์มือถือและแท็บเล็ตพีซี (Tablet PC) ได้ (อุปกรณ์ต้องรองรับภาษาไทยได้)
- โปรแกรมสามารถคำนวณวิเคราะห์โครงข้อหมุนในระนาบซึ่งได้ผลลัพธ์ถูกต้องตามหลักทฤษฎีการวิเคราะห์

5.2 ข้อเสนอแนะ

หากมีการพัฒนาโปรแกรมในลักษณะนี้มากยิ่งขึ้นจะช่วยให้วิศวกรสามารถทำงานได้อย่างสะดวก ลดการใช้งานโปรแกรมจากต่างประเทศที่มีราคาค่อนข้างสูง ซึ่งก่อให้เกิดประโยชน์แก่วงการวิศวกรรมโยธาได้พอสมควร

ทั้งนี้ควรมีการพัฒนาโปรแกรมวิเคราะห์โครงสร้างแบบอื่นๆ เช่น โครงข้อหมุน โครงข้อแข็ง 3 มิติ รวมทั้งโปรแกรมช่วยในการออกแบบชิ้นส่วนโครงสร้างชนิดต่างๆ ในแบบออนไลน์นี้ให้มากยิ่งขึ้น

บรรณานุกรม

[1] กิตติศักดิ์ เจริญโภคานนท์, คู่มือเขียนเว็บอีคอมเมิร์ซด้วย PHP, พิมพ์ที่ ซัคเซสมีเดีย, กรุงเทพฯ, 2552

[2] ปณิธาน ลักคุณะประสิทธิ์, การวิเคราะห์โครงสร้าง, พิมพ์ที่ โรงพิมพ์ จุฬาลงกรณ์ มหาวิทยาลัย, กรุงเทพฯ, 2537

[3] สมศักดิ์ โชคชัยชุติกุล, อินไซท์ PHP5, พิมพ์ที่ บริษัท โปรวิชั่น จำกัด, กรุงเทพฯ, 2547

[4] ณัฐภัชร ณ เขากต, เทคนิคการสร้างกราฟสวยใน PHP ด้วย JPGraph, พิมพ์ที่ สำนักพิมพ์ วิดดี้กรุ๊ป, กรุงเทพฯ, 2550

[5] อภิเชษฐ์ ไชยสังการ, ปิยะเมศธ์ ยอดเณร และบุญญฤทธิ์ ศรีสง่า (2555), การพัฒนาโปรแกรมวิเคราะห์คานต่อเนื่องด้วยภาษาพีเอชพี, ปริญญานิพนธ์, ภาควิชาวิศวกรรมโยธา คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

[6] เจษฎา วิษณุคุณากร, พงษ์เทพ เครือคำอ้าย, นัฐพล โพธิ์แสง (2555), การวิเคราะห์โครงข้อหมุน 2 มิติด้วยโปรแกรม SCILAB, ปริญญานิพนธ์, ภาควิชาวิศวกรรมโยธา คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ภาคผนวก

ชื่อไฟล์	ความหมาย
index.php	เป็นส่วนของการรับข้อมูลต่างๆ ที่ใช้ในการคำนวณ
step2-table.php	เป็นส่วนของการคำนวณทั้งหมด ตั้งแต่การหาค่ามุม ความยาว การคำนวณโดยวิธีการรวมสติฟเนสโดยตรง ตารางแสดงผลจากการคำนวณ
step3.php	เป็นส่วนของการแสดงรูปโครงสร้าง Truss ที่ได้จากข้อมูลข้างต้น
step4.php	เป็นส่วนของการแสดงรูปโครงสร้าง Truss ที่มีการเปลี่ยนแปลงซึ่งได้จากการคำนวณ

ชื่อ index.php

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head> .
<meta http-equiv="Content-Type" content="text/html; charset=windows-874" />
<title>Untitled Document</title>
</head>
<body bgcolor="#FFFFCC">
<form action="" method="post" enctype="multipart/form-data" name="form2" id="form2">
 <label>
 <div align="center">
<label>
 <input type="image" name="imageField" src="12345.jpg" />
</label>
</div>
</label>
</form>
<br />
<form id="form1" name="form1" method="post" action="">
<div align="center"><font color="#1F14EB" size="6"><strong>การวิเคราะห์โครงข้อหมุนแบบ 2 มิติ </strong></font> </div>
</form>
<br />
<form method="post" action="index.php">
<font size="3">          // การรับค่าจากการ input ข้อมูลจำนวน node และ member
<strong>Node Information </strong> <input name="num_node" type="text" value="<?=$_POST[num_node]?>" /> Node <br>
<strong>Member Information </strong> <input name="num_member" type="text" value="<?=$_POST[num_member]?>"  /> Member<br>
```

```
</font>
 <input name="node" type="submit"  value="Click !!"/>
</form>
<br />
<br />
<? if($_POST[node]) {?>
<form action="step2-table.php" method="post">          //การส่งค่าข้อมูลไปยัง step2-tabel.php
<table width="100%">
<tr>
<td>
<fieldset style="border-style: ridge; border-color: red">          //การสร้างตารางรับค่าต่างๆที่ได้จากการ input จำนวน node
<legend><i><b>Node Information</b></i></legend>
<table border="0" cellspacing="0" width="80%">
<tr>
<td align="center" bgcolor="#CCCCCC"><strong>Node</strong></td>
<td align="center" bgcolor="#CCCCCC"><strong>X(m)</strong></td>
<td align="center" bgcolor="#CCCCCC"><strong>Y(m)</strong></td>
<td align="center" bgcolor="#CCCCCC"><strong>Ux(m)</strong></td>
<td align="center" bgcolor="#CCCCCC"><strong>Uy(m)</strong></td>
<td align="center" bgcolor="#CCCCCC"><strong>Fx(kg)</strong></td>
<td align="center" bgcolor="#CCCCCC"><strong>Fy(kg)</strong></td>
</tr>
<? for($n=1;$n<=$_POST[num_node];$n++) {?>
<tr>
<td align="center"><?=$n?></td>
<td align="center"><input name="X[]" type="text" size="5" /></td>
<td align="center"><input name="Y[]" type="text" size="5" /></td>
<td align="center"><input name="Ux[]" type="text" size="5" /></td>
<td align="center"><input name="Uy[]" type="text" size="5" /></td>
<td align="center"><input name="Fx[]" type="text" size="5" /></td>
<td align="center"><input name="Fy[]" type="text" size="5" /></td>
</tr>
<? }?>
</table>
</fieldset>
</td>
</tr>
</table>
<br/>
<br />
<table width="100%">
<tr>
```

```
<td>
<fieldset style="border-style: ridge; border-color: red">        //การสร้างตารางรับค่าต่างๆที่ได้จากการ input จำนวน member
<legend><i><b>Member Information</b></i></legend>
<table border="0" cellspacing="0" width="80%">
<tr>
<td align="center" bgcolor="#CCCCCC"><strong>Member</strong></td>
<td align="center" bgcolor="#CCCCCC"><strong>Start Node</strong></td>
<td align="center" bgcolor="#CCCCCC"><strong>End Node</strong></td>
<td align="center" bgcolor="#CCCCCC"><strong>EA(kg)</strong></td>
</tr>
<? for($m=1;$m<=$_POST[num_member];$m++) {?>
<tr>
<td align="center"><?=$m?></td>
<td align="center"><input name="start_node[]" type="text" size="5" /></td>
<td align="center"><input name="end_node[]" type="text" size="5" /></td>
<td align="center"><input name="EA[]" type="text" size="10" /></td>
</tr>
<? }?>
</table>
</fieldset>
</td>
</tr>
</table>
<table width="100%">
<tr>
<input name="num_node" type="hidden" value="<?=$_POST[num_node]?>" />
<input name="num_member" type="hidden" value="<?=$_POST[num_member]?>" />
<td align="center"><input name="cal" type="submit" value="คำนวณ" /></td>
</tr> </table> <? }?>
</form>
</body>
</html>
```

ชื่อ step2- table.php

```
<?                                        //การวิเคราะห์โครงสร้าง truss โดยวิธีการรวมสติฟเนส
for($m=1;$m<=$_POST[num_member];$m++)  {
for($n=1;$n<=$_POST[start_node][$m-1];$n++) {
        $x1 =$_POST[X][$n-1];
        $y1=$_POST[Y][$n-1];
 }
 for($n=1;$n<=$_POST[end_node][$m-1];$n++) {
```

```
        $x2=$_POST[X][$n-1];
        $y2=$_POST[Y][$n-1];
 }
        $sum1=$y2-$y1;
        $sum2=$x2-$x1;
if($sum2==0){
        $sum3=90;
        $cos=0;
         $sin=1;
}else{
         $sum3=atan($sum1/$sum2);          //การหาค่ามุม
         $cos=cos(($sum3));
         $sin=sin(($sum3));
}
         $coss[$m]=$cos;
         $sinn[$m]=$sin;
        $stn=$_POST[start_node][$m-1];
        $ux1=$_POST[Ux][$stn-1];
if($ux1=="0"){
}else{
         $ux1=2*$_POST[start_node][$m-1]-1;
}
         $uy1=$_POST[Uy][$stn-1];
if($uy1=="0"){
}else{
         $uy1=2*$_POST[start_node][$m-1];
}
        $edn=$_POST[end_node][$m-1];
        $ux2=$_POST[Ux][$edn-1];
if($ux2=="0"){
}else{
         $ux2=2*$_POST[end_node][$m-1]-1;
}
$uy2=$_POST[Uy][$edn-1];
if($uy2=="0"){
}else{
         $uy2=2*$_POST[end_node][$m-1];
}

 for($n=1;$n<=$_POST[start_node][$m-1];$n++) {
        $xx1 =$_POST[X][$n-1];
        $yy1=$_POST[Y][$n-1];
```

```
}
for($n=1;$n<=$_POST[end_node][$m-1];$n++) {
            $xx2=$_POST[X][$n-1];
            $yy2=$_POST[Y][$n-1];
}
$sum11=pow($yy2-$yy1,2);
$sum22=pow($xx2-$xx1,2);
$l=sqrt($sum11+$sum22);        // การหาค่าความยาว member

$ea=$_POST[EA][$m-1];
$eal=$ea/$l;

            $d[1]=2*$_POST[start_node][$m-1]-1;
            $d[2]=2*$_POST[start_node][$m-1];
            $d[3]=2*$_POST[end_node][$m-1]-1;
            $d[4]=2*$_POST[end_node][$m-1];

for($i=1;$i<=4;$i++) {
for($j=1;$j<=4;$j++) {
if($i==1 and $j==1){
            $kf[$m-1][$i][$j]=$eal*(pow($cos,2));
                        }
if($i==1 and $j==2){
            $kf[$m-1][$i][$j]=$eal*($sin*$cos);
                        }
if($i==1 and $j==3){
            $kf[$m-1][$i][$j]=$eal*(-(pow($cos,2)));
                        }
if($i==1 and $j==4){
            $kf[$m-1][$i][$j]=$eal*(-($cos*$sin));
                        }
if($i==2 and $j==1){
            $kf[$m-1][$i][$j]=$eal*($cos*$sin);
                        }
if($i==2 and $j==2){
            $kf[$m-1][$i][$j]=$eal*(pow($sin,2));
                        }
if($i==2 and $j==3){
            $kf[$m-1][$i][$j]=$eal*(-($cos*$sin));
                        }
if($i==2 and $j==4){
            $kf[$m-1][$i][$j]=$eal*(-(pow($sin,2)));
```

```
                                }
if($i==3 and $j==1){
            $kf[$m-1][$i][$j]=$eal*(-(pow($cos,2)));
                        }
if($i==3 and $j==2){
            $kf[$m-1][$i][$j]=$eal*(-($cos*$sin));
                        }
if($i==3 and $j==3){
            $kf[$m-1][$i][$j]=$eal*(pow($cos,2));
                        }
if($i==3 and $j==4){
            $kf[$m-1][$i][$j]=$eal*($cos*$sin);
                        }
if($i==4 and $j==1){
            $kf[$m-1][$i][$j]=$eal*(-($cos*$sin));
                        }
if($i==4 and $j==2){
            $kf[$m-1][$i][$j]=$eal*(-(pow($sin,2)));
                        }
if($i==4 and $j==3){
            $kf[$m-1][$i][$j]=$eal*($cos*$sin);
                        }
if($i==4 and $j==4){
            $kf[$m-1][$i][$j]=$eal*(pow($sin,2));
                        }
            $kg[$d[$i]][$d[$j]]=$kg[$d[$i]][$d[$j]]+$kf[$m-1][$i][$j];
            }
}
}
for($m=1;$m<=$_POST[num_node];$m++) {
            $ug[2*$m-1]=$_POST[Ux][$m-1];
            $ug[2*$m]=$_POST[Uy][$m-1];
            $fg[2*$m-1]=$_POST[Fx][$m-1];
            $fg[2*$m]=$_POST[Fy][$m-1];
}
for($m=1;$m<=2*$_POST[num_node];$m++) {
            if($ug[$m]==""){
                                $ai=$ai+1;
                                $vai[$ai]=$m;
             }else{
                                $bi=$bi+1;
                                $vbi[$bi]=$m;
```

```
}
}

 for($i=1;$i<=$ai;$i++) {
for($j=1;$j<=$ai;$j++) {
                                    $kr[$i][$j]=$kg[$vai[$i]][$vai[$j]];

                                   }
                                  $fr[$i]=$fg[$vai[$i]];
                                   $fu[$i]=$ug[$vai[$i]];
 }
for($k=1;$k<=$ai-1;$k++) {
for($i=$k+1;$i<=$ai;$i++) {
                        $c=$kr[$i][$k]/$kr[$k][$k];
for($j=$k;$j<=$ai;$j++) {
                        $kr[$i][$j]=$kr[$i][$j]-($c*$kr[$k][$j]);
                           }
                        $fr[$i]=$fr[$i]-($c*$fr[$k]);
            }
            }
                        $uf[$ai][$ai]=$fr[$ai]/$kr[$ai][$ai];
                        $u[$ai]=$uf[$ai][$ai];
for($k=$ai-1;$k>=1;$k--) {
                         $sumk=0;
for($j=$k+1;$j<=$ai;$j++) {
                        $sumk=$sumk+($kr[$k][$j]*$u[$j]);
                         }
                         $u[$k]=(1/$kr[$k][$k])*($fr[$k]-$sumk);
            }
for($i=1;$i<=$ai;$i++) {
                        $ug[$vai[$i]]=$u[$i];
            }
for($i=1;$i<=2*$_POST[num_node];$i++) {
                        $f[$i]=0;
for($j=1;$j<=2*$_POST[num_node];$j++) {
                        $f[$i]=$f[$i]+($kg[$i][$j]*$ug[$j]);
            }
                        $uu[$i]=$ug[$i];
            }

for($k=1;$k<=$_POST[num_member];$k++) {
                        $d[1]=2*$_POST[start_node][$k-1]-1;
```

```
                    $d[2]=2*$_POST[start_node][$k-1];
                    $d[3]=2*$_POST[end_node][$k-1]-1;
                    $d[4]=2*$_POST[end_node][$k-1];
for($i=1;$i<=2;$i++) {
                    $fd[$i]=0;
for($j=1;$j<=4;$j++) {
                    $fd[$i]=$fd[$i]+($kf[$k-1][$i][$j]*$uu[$d[$j]]);
          }
          }
                $nf[$k]=-((($fd[1]*$coss[$k])+($fd[2]*$sinn[$k]));
          }
?>
<style type="text/css">
<!--
.style1 {color: #FFFF80}
body {
          background-color: #FFFFCC;
}
-->
</style>

<div align="center"><img src="step3.php?        // การส่งค่าไปสร้างรูป truss ที่step3.php
          num_node=<?=$_POST[num_node];?>&num_member=<?=$_POST[num_member];
for($n=1;$n<=$_POST[num_node];$n++) { ?>
          &X[<?=$n-1?>]=<?=$_POST[X][$n-1]?>
          &Y[<?=$n-1?>]=<?=$_POST[Y][$n-1]?>
          &Ux[<?=$n-1?>]=<?=$_POST[Ux][$n-1]?>
          &Uy[<?=$n-1?>]=<?=$_POST[Uy][$n-1]?>
          &Fx[<?=$n-1?>]=<?=$_POST[Fx][$n-1]?>
          &Fy[<?=$n-1?>]=<?=$_POST[Fy][$n-1];   }
for($m=1;$m<=$_POST[num_member];$m++) {?>
          &start_node[<?=$m-1?>]=<?=$_POST[start_node][$m-1]?>
          &end_node[<?=$m-1?>]=<?=$_POST[end_node][$m-1]?>
          &ea[<?=$m-1?>]=<?=$_POST[EA][$m-1]; } ?>
" width="500" height="500" align="baseline" />
</div>
<table width="100%">
<tr valign="top">
<td>
<fieldset style="border-style: ridge; border-color: red">
<legend><i><b>Node Information</b></i></legend>
<table border="0" cellspacing="0" width="80%">
```

```
<tr>
<td width="12%" align="center" bgcolor="#CCCCCC"><strong>Node</strong></td>
<td width="15%" align="center" bgcolor="#CCCCCC"><strong>X(m)</strong></td>
<td width="15%" align="center" bgcolor="#CCCCCC"><strong>Y(m)</strong></td>
<td width="15%" align="center" bgcolor="#CCCCCC"><strong>Ux(m)</strong></td>
<td width="14%" align="center" bgcolor="#CCCCCC"><strong>Uy(m)</strong></td>
<td width="14%" align="center" bgcolor="#CCCCCC"><strong>Fx(kg)</strong></td>
<td width="15%" align="center" bgcolor="#CCCCCC"><strong>Fy(kg)</strong></td>
</tr>

<? for($n=1;$n<=$_POST[num_node];$n++) { $counter++;?>
<tr>
<td align="center"><?=$n?></td>
<td align="center"><?=$_POST[X][$n-1]?>
 <label></label></td>
<td align="center"><?=$_POST[Y][$n-1]?></td>
<td align="center"><?$nn=$n-1;echo ceil(100000*$ug[$nn+$counter])/100000;?></td>
<td align="center"><?echo ceil(100000*$ug[$nn+$counter+1])/100000;?></td>
<td align="center"><?echo ceil(10000* $f[$nn+$counter])/10000;?></td>
<td align="center"><?echo ceil(10000*$f[$nn+$counter+1])/10000;?></td>
</tr>
<? }?>
</table>
</fieldset>
</td>
</tr>
</table>
<span class="style1"><br/>
<br />
</span>
<table width="100%">
<tr align="top">
<td>
<fieldset style="border-style: ridge; border-color: red">
<legend><i><b>Member Information</b></i></legend>
<table border="0" cellspacing="0" width="80%">
<tr>
<td width="11%" align="center" bgcolor="#CCCCCC"><strong>Member</strong></td>
<td width="16%" align="center" bgcolor="#CCCCCC"><strong>Start Node</strong></td>
<td width="15%" align="center" bgcolor="#CCCCCC"><strong>End Node</strong></td>
<td width="14%" align="center" bgcolor="#CCCCCC"><strong>EA(kg)</strong></td>
<td width="15%" align="center" bgcolor="#CCCCCC"><strong>L(m)</strong></td>
```

```
<td width="14%" align="center" bgcolor="#CCCCCC"><strong>Angle(deg)</strong></td>
<td width="15%" align="center" bgcolor="#CCCCCC"><strong>F(kg)</strong></td>
</tr>
<? for($m=1;$m<=$_POST[num_member];$m++) {?>
<tr>
<td align="center"><?=$m?></td>
<td align="center"><?=$_POST[start_node][$m-1]?></td>
<td align="center"><?=$_POST[end_node][$m-1]?></td>
<td align="center"><?=$_POST[EA][$m-1]?></td>
<td align="center"><?
  for($n=1;$n<=$_POST[start_node][$m-1];$n++) {
            $x1 =$_POST[X][$n-1];
            $y1=$_POST[Y][$n-1];
  }
  for($n=1;$n<=$_POST[end_node][$m-1];$n++) {
            $x2=$_POST[X][$n-1];
            $y2=$_POST[Y][$n-1];
  }
$sum1=pow($y2-$y1,2);
$sum2=pow($x2-$x1,2);
$sum3=sqrt($sum1+$sum2);
 echo $sum3;
?> </td>
<td align="center"><?

  for($n=1;$n<=$_POST[start_node][$m-1];$n++) {
            $x1 =$_POST[X][$n-1];
            $y1=$_POST[Y][$n-1];
  }
  for($n=1;$n<=$_POST[end_node][$m-1];$n++) {
            $x2=$_POST[X][$n-1];
            $y2=$_POST[Y][$n-1];
  }
            $sum1=$y2-$y1;
            $sum2=$x2-$x1;
 if($sum2==0){
            $sum3=90;
            echo $sum3;
 }else{
            $sum3=atan($sum1/$sum2);
            echo ceil(10000*rad2deg($sum3))/10000;
 }
```

```
?>
</td><td align="center"><? echo $nf[$m] ?></td></tr><? }?>
</table>
</fieldset>
</td>
</tr>
</table>
<? if($_POST[X]) {?>
<input name="num_node" type="hidden" value="<?=$_POST[num_node]?>" />
<input name="num_member" type="hidden" value="<?=$_POST[num_member]?>" />
<td align="center"> </td>
<? }?>
</form>
<div align="center"><img src="step4.php?          // การส่งค่าไปสร้างรูป truss ที่step3.php
        num_node=<?=$_POST[num_node];?>&num_member=<?=$_POST[num_member];
for($n=1;$n<=$_POST[num_node];$n++) { ?>
        &X[<?=$n-1?>]=<?=$_POST[X][$n-1]?>
        &Y[<?=$n-1?>]=<?=$_POST[Y][$n-1]?>
        &Ux[<?=$n-1?>]=<?=$_POST[Ux][$n-1]?>
        &Uy[<?=$n-1?>]=<?=$_POST[Uy][$n-1]?>
        &Fx[<?=$n-1?>]=<?=$_POST[Fx][$n-1]?>
        &Fy[<?=$n-1?>]=<?=$_POST[Fy][$n-1]?>
        &uX[<?=$n?>]=<?=$ug[2*$n-1]?>
        &uY[<?=$n?>]=<?=$ug[2*$n];  }
for($m=1;$m<=$_POST[num_member];$m++)  {?>
        &start_node[<?=$m-1?>]=<?=$_POST[start_node][$m-1]?>
        &end_node[<?=$m-1?>]=<?=$_POST[end_node][$m-1]?>
        &ea[<?=$m-1?>]=<?=$_POST[EA][$m-1];  } ?>
" width="500" height="500" align="baseline" />
</div>
```

ชื่อ step3.php

```
<?
$n=$_GET[num_node];
$m=$_GET[num_member];
for ($b=0;$b<$m;$b++)
        {
$vXYStart[$b+1]=$_GET['start_node'][$b];
$vXYEnd[$b+1]=$_GET['end_node'][$b];
$vEA[$b+1]=$_GET['ea'][$b];
        }
```

```
$dMin=0;
$dMax=0;
for ($b=0;$b<$n;$b++)          //การสร้างรูปโครง truss
            {
            $vX[$b+1]=$_GET['X'][$b];
            $vY[$b+1]=$_GET['Y'][$b];
            $Ux[$b+1]=$_GET['Ux'][$b];
            $Uy[$b+1]=$_GET['Uy'][$b];
            $Fx[$b+1]=$_GET['Fx'][$b];
            $Fy[$b+1]=$_GET['Fy'][$b];
            if($vX[$b+1]>$dMax){$dMax=$vX[$b+1];}
            if($vY[$b+1]>$dMax){$dMax=$vY[$b+1];}
            if($vX[$b+1]<$dMin){$dMin=$vX[$b+1];}
            if($vY[$b+1]<$dMin){$dMin=$vY[$b+1];}

            }
            $dMin=$dMin-1;
            $dMax=$dMax+1;
            $iMag=0.003*($dMax-$dMin);
$pic = ImageCreateTrueColor(1100,1100);
$black = ImageColorAllocate($pic, 0, 0, 0);
$white = ImageColorAllocate($pic, 255, 255, 255);
$red = ImageColorAllocate($pic, 255, 0, 0);
ImageFill($pic,0,0,$white);
ImageSetThickness($pic, 2);
for ($b=1;$b<=$m;$b++)
            {
            $v1=$vXYStart[$b];
            $v2=$vXYEnd[$b];
            $x=toScreen($vX[$v1],$dMin,$dMax,900,200);
            $y=toScreen($vY[$v1],$dMin,$dMax,-900,900);
            $x1=toScreen($vX[$v2],$dMin,$dMax,900,200);
            $y1=toScreen($vY[$v2],$dMin,$dMax,-900,900);

ImageLine($pic,$x,$y,$x1,$y1,$black);
ImageFilledArc($pic,$x,$y,10,10,0,360,$black,IMG_ARC_EDGED);
ImageFilledArc($pic,$x1,$y1,10,10,0,360,$black,IMG_ARC_EDGED);
}
for ($b=1;$b<=$n;$b++)          // การสร้างรูป support
            {
            if($Ux[$b]=="0" && $Uy[$b]=="0")
```

```
                {
                $zx[0]=0;$zy[0]=0;
                $zx[1]=-10;$zy[1]=20;
                $zx[2]=10,$zy[2]=20;
                $zx[3]=-8;$zy[3]=20;
                $zx[4]=-11;$zy[4]=23;
                $zx[5]=-4;$zy[5]=20;
                $zx[6]=-7;$zy[6]=23;
                $zx[7]=1;$zy[7]=20;
                $zx[8]=-2;$zy[8]=23;
                $zx[9]=5;$zy[9]=20;
                $zx[10]=2;$zy[10]=23;
                $zx[11]=9;$zy[11]=20;
                $zx[12]=6;$zy[12]=23;
for($i=0;$i<=12;$i++){
                $xtemp=$iMag*($zx[$i])+$vX[$b];
                $ytemp=-$iMag*($zy[$i])-$vY[$b];
                $zx[$i]=toScreen($xtemp,$dMin,$dMax,900,200);
                $zy[$i]=toScreen($ytemp,$dMin,$dMax,-900,900);}
                ImageLine($pic,$zx[0],$zy[0],$zx[1],$zy[1],$red);
                ImageLine($pic,$zx[0],$zy[0],$zx[2],$zy[2],$red);
                ImageLine($pic,$zx[1],$zy[1],$zx[2],$zy[2],$red);
                ImageLine($pic,$zx[3],$zy[3],$zx[4],$zy[4],$red);
                ImageLine($pic,$zx[5],$zy[5],$zx[6],$zy[6],$red);
                ImageLine($pic,$zx[7],$zy[7],$zx[8],$zy[8],$red);
                ImageLine($pic,$zx[9],$zy[9],$zx[10],$zy[10],$red);
                ImageLine($pic,$zx[11],$zy[11],$zx[12],$zy[12],$red);
                }
if($Ux[$b]=="" && $Uy[$b]=="0")
                {
                $zx[0]=0;$zy[0]=0;
                $zx[1]=-10;$zy[1]=-20;
                $zx[2]=10;$zy[2]=-20;
                $zx[3]=-8;$zy[3]=-20;
                $zx[4]=-11;$zy[4]=-23;
                $zx[5]=-4;$zy[5]=-20;
                $zx[6]=-7;$zy[6]=-23;
                $zx[7]=1;$zy[7]=-20;
                $zx[8]=-2;$zy[8]=-23;
                $zx[9]=5;$zy[9]=-20;
                $zx[10]=2;$zy[10]=-23;
                $zx[11]=9;$zy[11]=-20;
```

```
            $zx[12]=6;$zy[12]=-23;
            $zx[13]=-10;$zy[13]=-15;
            $zx[14]=10;$zy[14]=-15;
            $zx[15]=-7.5;$zy[15]=-17.5;
            $zx[16]=-2.5;$zy[16]=-17.5;
            $zx[17]=2.5;$zy[17]=-17.5;
            $zx[18]=7.5;$zy[18]=-17.5;

for($i=0;$i<=18;$i++){
            $xtemp=$iMag*($zx[$i])+$vX[$b];
            $ytemp=$iMag*($zy[$i])+$vY[$b];
            $zx[$i]=toScreen($xtemp,$dMin,$dMax,900,200);
            $zy[$i]=toScreen($ytemp,$dMin,$dMax,-900,900);}
            ImageLine($pic,$zx[0],$zy[0],$zx[13],$zy[13],$red);
            ImageLine($pic,$zx[0],$zy[0],$zx[14],$zy[14],$red);
            ImageLine($pic,$zx[13],$zy[13],$zx[14],$zy[14],$red);
            ImageLine($pic,$zx[1],$zy[1],$zx[2],$zy[2],$red);
            ImageLine($pic,$zx[3],$zy[3],$zx[4],$zy[4],$red);
            ImageLine($pic,$zx[5],$zy[5],$zx[6],$zy[6],$red);
            ImageLine($pic,$zx[7],$zy[7],$zx[8],$zy[8],$red);
            ImageLine($pic,$zx[9],$zy[9],$zx[10],$zy[10],$red);
            ImageLine($pic,$zx[11],$zy[11],$zx[12],$zy[12],$red);
            ImageArc($pic,$zx[15],$zy[15],15,15,0,360,$red);
            ImageArc($pic,$zx[16],$zy[16],15,15,0,360,$red);
            ImageArc($pic,$zx[17],$zy[17],15,15,0,360,$red);
            ImageArc($pic,$zx[18],$zy[18],15,15,0,360,$red);
            }
if($Ux[$b]=="0" && $Uy[$b]="")
            {
            $zx[0]=0;$zy[0]=0;
            $zx[1]=-20;$zy[1]=-10;
            $zx[2]=-20;$zy[2]=10;
            $zx[3]=-20;$zy[3]=-8;
            $zx[4]=-23;$zy[4]=-11;
            $zx[5]=-20;$zy[5]=-4;
            $zx[6]=-23;$zy[6]=-7;
            $zx[7]=-20;$zy[7]=1;
            $zx[8]=-23;$zy[8]=-2;
            $zx[9]=-20;$zy[9]=5;
            $zx[10]=-23;$zy[10]=2;
            $zx[11]=-20;$zy[11]=9;
            $zx[12]=-23;$zy[12]=6;
```

```
            $zx[13]=-15;$zy[13]=-10;
            $zx[14]=-15;$zy[14]=10;
            $zx[15]=-17.5;$zy[15]=-7.5;
            $zx[16]=-17.5;$zy[16]=-2.5;
            $zx[17]=-17.5;$zy[17]=2.5;
            $zx[18]=-17.5;$zy[18]=7.5;
for($i=0;$i<=18;$i++){
            $xtemp=$iMag*($zx[$i])+$vX[$b];
            $ytemp=$iMag*($zy[$i])+$vY[$b];
            $zx[$i]=toScreen($xtemp,$dMin,$dMax,900,200);
            $zy[$i]=toScreen($ytemp,$dMin,$dMax,-900,900);}
            ImageSetThickness($pic, 1);
            ImageLine($pic,$zx[0],$zy[0],$zx[13],$zy[13],$red);
            ImageLine($pic,$zx[0],$zy[0],$zx[14],$zy[14],$red);
            ImageLine($pic,$zx[13],$zy[13],$zx[14],$zy[14],$red);
            ImageLine($pic,$zx[1],$zy[1],$zx[2],$zy[2],$red);
            ImageLine($pic,$zx[3],$zy[3],$zx[4],$zy[4],$red);
            ImageLine($pic,$zx[5],$zy[5],$zx[6],$zy[6],$red);
            ImageLine($pic,$zx[7],$zy[7],$zx[8],$zy[8],$red);
            ImageLine($pic,$zx[9],$zy[9],$zx[10],$zy[10],$red);
            ImageLine($pic,$zx[11],$zy[11],$zx[12],$zy[12],$red);
            ImageArc($pic,$zx[15],$zy[15],15,15,0,360,$red);
            ImageArc($pic,$zx[16],$zy[16],15,15,0,360,$red);
            ImageArc($pic,$zx[17],$zy[17],15,15,0,360,$red);
            ImageArc($pic,$zx[18],$zy[18],15,15,0,360,$red);
                    }

if($Ux[$b]=="" && $Uy[$b]==""){
            $x=toScreen($vX[$b],$dMin,$dMax,900,200);
            $y=toScreen($vY[$b],$dMin,$dMax,-900,900);
            ImageFilledArc($pic,$x,$y,10,10,0,360,$black,IMG_ARC_EDGED);
                    }
if($Fy[$b] < 0)        // การสร้างทิศทางของแรง(ลูกศร)
                    {
            $zx[0]=0;$zy[0]=0;
            $zx[1]=-5;$zy[1]=10;
            $zx[2]=5;$zy[2]=10;
            $zx[3]=0;$zy[3]=40;
for($i=0;$i<=3;$i++){
            $xtemp=$iMag*($zx[$i])+$vX[$b];
            $ytemp=$iMag*($zy[$i])+$vY[$b];
            $zx[$i]=toScreen($xtemp,$dMin,$dMax,900,200);
```

```
            $zy[$i]=toScreen($ytemp,$dMin,$dMax,-900,900);      }
            ImageSetThickness($pic, 3);
            ImageLine($pic,$zx[0],$zy[0],$zx[1],$zy[1],$red);
            ImageLine($pic,$zx[0],$zy[0],$zx[2],$zy[2],$red);
            ImageLine($pic,$zx[0],$zy[0],$zx[3],$zy[3],$red);

                        }
if($Fy[$b] > "0")
                        {
            $zx[0]=0;$zy[0]=0;
            $zx[1]=-5;$zy[1]=-10;
            $zx[2]=5;$zy[2]=-10;
            $zx[3]=0;$zy[3]=-40;
for($i=0;$i<=3;$i++){
            $xtemp=$iMag*($zx[$i])+$vX[$b];
            $ytemp=$iMag*($zy[$i])+$vY[$b];
            $zx[$i]=toScreen($xtemp,$dMin,$dMax,900,200);
            $zy[$i]=toScreen($ytemp,$dMin,$dMax,-900,900);      }
            ImageSetThickness($pic, 3);
            ImageLine($pic,$zx[0],$zy[0],$zx[1],$zy[1],$red);
            ImageLine($pic,$zx[0],$zy[0],$zx[2],$zy[2],$red);
            ImageLine($pic,$zx[0],$zy[0],$zx[3],$zy[3],$red);
                        }
if($Fx[$b] < 0)
                        {
            $zx[0]=0;$zy[0]=0;
            $zx[1]=10;$zy[1]=-5;
            $zx[2]=10;$zy[2]=5;
            $zx[3]=40;$zy[3]=0;
for($i=0;$i<=3;$i++){
            $xtemp=$iMag*($zx[$i])+$vX[$b];
            $ytemp=$iMag*($zy[$i])+$vY[$b];
            $zx[$i]=toScreen($xtemp,$dMin,$dMax,900,200);
            $zy[$i]=toScreen($ytemp,$dMin,$dMax,-900,900);}
            ImageSetThickness($pic, 3);
            ImageLine($pic,$zx[0],$zy[0],$zx[1],$zy[1],$red);
            ImageLine($pic,$zx[0],$zy[0],$zx[2],$zy[2],$red);
            ImageLine($pic,$zx[0],$zy[0],$zx[3],$zy[3],$red);
                        }
if($Fx[$b] > "0")
                        {
            $zx[0]=0;$zy[0]=0;
```

```
                $zx[1]=-10;$zy[1]=-5;
                        $zx[2]=-10;$zy[2]=5;
                        $zx[3]=-40;$zy[3]=0;
                        for($i=0;$i<=3;$i++){
                        $xtemp=$iMag*($zx[$i])+$vX[$b];
                        $ytemp=$iMag*($zy[$i])+$vY[$b];
                        $zx[$i]=toScreen($xtemp,$dMin,$dMax,900,200);
                        $zy[$i]=toScreen($ytemp,$dMin,$dMax,-900,900);}
                        ImageSetThickness($pic, 3);
                        ImageLine($pic,$zx[0],$zy[0],$zx[1],$zy[1],$red);
                        ImageLine($pic,$zx[0],$zy[0],$zx[2],$zy[2],$red);
                        ImageLine($pic,$zx[0],$zy[0],$zx[3],$zy[3],$red);
                        }


        }
header("Content-type: image/PNG");
ImagePNG($pic);
ImageDestroy($pic);
function toScreen($xy,$xyMin,$xyMax,$xyWidth,$xyOrigin){
$temp = $xyOrigin+$xy*$xyWidth/($xyMax-$xyMin);
return $temp;
} ?>
```

ชื่อ step4.php

```
<?
$n=$_GET[num_node];
$m=$_GET[num_member];
for ($b=0;$b<$m;$b++)
        {
$vXYStart[$b+1]=$_GET['start_node'][$b];
$vXYEnd[$b+1]=$_GET['end_node'][$b];
$vEA[$b+1]=$_GET['ea'][$b];
        }
$dMin=0;
$dMax=0;
for ($b=0;$b<$n;$b++)        // การสร้างรูปโครง truss และการเปลี่ยนแปลงที่ได้จากการคำนวณ
        {
        $vX[$b+1]=$_GET['X'][$b];
        $vY[$b+1]=$_GET['Y'][$b];
        $uX[$b+1]=$_GET['uX'][$b+1];
```

```
            $uY[$b+1]=$_GET['uY'][$b+1];
            $Ux[$b+1]=$_GET['Ux'][$b];
            $Uy[$b+1]=$_GET['Uy'][$b];
            $Fx[$b+1]=$_GET['Fx'][$b];
            $Fy[$b+1]=$_GET['Fy'][$b];
            if($vX[$b+1]>$dMax){$dMax=$vX[$b+1];}
            if($vY[$b+1]>$dMax){$dMax=$vY[$b+1];}
            if($vX[$b+1]<$dMin){$dMin=$vX[$b+1];}
            if($vY[$b+1]<$dMin){$dMin=$vY[$b+1];}
            if($uX[$b+1]>$dMax){$dMax=$uX[$b+1];}
            if($uY[$b+1]>$dMax){$dMax=$uY[$b+1];}
            if($uX[$b+1]<$dMin){$dMin=$uX[$b+1];}
            if($uY[$b+1]<$dMin){$dMin=$uY[$b+1];}

            }
            $dMin=$dMin-1;
            $dMax=$dMax+1;
            $iMag=0.003*($dMax-$dMin);
$pic = ImageCreateTrueColor(1100,1100);
$black = ImageColorAllocate($pic, 0, 0, 0);
$white = ImageColorAllocate($pic, 255, 255, 255);
$red = ImageColorAllocate($pic, 255, 0, 0);
ImageFill($pic,0,0,$white);
ImageSetThickness($pic, 2);
for ($b=1;$b<=$m;$b++)
            {
                    $v1=$vXYStart[$b];
                    $v2=$vXYEnd[$b];
                    $x=toScreen($vX[$v1],$dMin,$dMax,900,200);
                    $y=toScreen($vY[$v1],$dMin,$dMax,-900,900);
                    $x1=toScreen($vX[$v2],$dMin,$dMax,900,200);
                    $y1=toScreen($vY[$v2],$dMin,$dMax,-900,900);
                    $xNew=toScreen($vX[$v1]+$uX[$v1],$dMin,$dMax,900,200);
                    $yNew=toScreen($vY[$v1]+$uY[$v1],$dMin,$dMax,-900,900);
                    $x1New=toScreen($vX[$v2]+$uX[$v2],$dMin,$dMax,900,200);
                    $y1New=toScreen($vY[$v2]+$uY[$v2],$dMin,$dMax,-900,900);
                    Imageline ($pic,$xNew,$yNew,$x1New,$y1New,$red);
                    ImageFilledArc($pic,$xNew,$yNew,10,10,0,360,$red,IMG_ARC_EDGED);
                    ImageFilledArc($pic,$x1New,$y1New,10,10,0,360,$red,IMG_ARC_EDGED);
                    ImageLine($pic,$x,$y,$x1,$y1,$black);
                    ImageFilledArc($pic,$x,$y,10,10,0,360,$black,IMG_ARC_EDGED);
                    ImageFilledArc($pic,$x1,$y1,10,10,0,360,$black,IMG_ARC_EDGED);
```

```
}
for ($b=1;$b<=$n;$b++)
        {
        if($Ux[$b]=="0" && $Uy[$b]=="0")        // การสร้างรูป support
                {
                $zx[0]=0;$zy[0]=0;
                $zx[1]=-10;$zy[1]=20;
                $zx[2]=10;$zy[2]=20;
                $zx[3]=-8;$zy[3]=20;
                $zx[4]=-11;$zy[4]=23;
                $zx[5]=-4;$zy[5]=20;
                $zx[6]=-7;$zy[6]=23;
                $zx[7]=1;$zy[7]=20;
                $zx[8]=-2;$zy[8]=23;
                $zx[9]=5;$zy[9]=20;
                $zx[10]=2;$zy[10]=23;
                $zx[11]=9;$zy[11]=20;
                $zx[12]=6;$zy[12]=23;
        for($i=0,$i<=12;$i++){
                $xtemp=$iMag*($zx[$i])+$vX[$b];
                $ytemp=-$iMag*($zy[$i])-$vY[$b];
                $zx[$i]=toScreen($xtemp,$dMin,$dMax,900,200);
                $zy[$i]=toScreen($ytemp,$dMin,$dMax,-900,900);}
                ImageLine($pic,$zx[0],$zy[0],$zx[1],$zy[1],$black);
                ImageLine($pic,$zx[0],$zy[0],$zx[2],$zy[2],$black);
                ImageLine($pic,$zx[1],$zy[1],$zx[2],$zy[2],$black);
                ImageLine($pic,$zx[3],$zy[3],$zx[4],$zy[4],$black);
                ImageLine($pic,$zx[5],$zy[5],$zx[6],$zy[6],$black);
                ImageLine($pic,$zx[7],$zy[7],$zx[8],$zy[8],$black);
                ImageLine($pic,$zx[9],$zy[9],$zx[10],$zy[10],$black);
                ImageLine($pic,$zx[11],$zy[11],$zx[12],$zy[12],$black);
                }
        if($Ux[$b]=="" && $Uy[$b]=="0")
                {
                $zx[0]=0;$zy[0]=0;
                $zx[1]=-10;$zy[1]=-20;
                $zx[2]=10;$zy[2]=-20;
                $zx[3]=-8;$zy[3]=-20;
                $zx[4]=-11;$zy[4]=-23;
                $zx[5]=-4;$zy[5]=-20;
                $zx[6]=-7;$zy[6]=-23;
                $zx[7]=1;$zy[7]=-20;
```

```
$zx[8]=-2;$zy[8]=-23;
$zx[9]=5;$zy[9]=-20;
$zx[10]=2;$zy[10]=-23;
$zx[11]=9;$zy[11]=-20;
$zx[12]=6;$zy[12]=-23;
$zx[13]=-10;$zy[13]=-15;
$zx[14]=10;$zy[14]=-15;
$zx[15]=-7.5;$zy[15]=-17.5;
$zx[16]=-2.5;$zy[16]=-17.5;
$zx[17]=2.5;$zy[17]=-17.5;
$zx[18]=7.5;$zy[18]=-17.5;
for($i=0;$i<=18;$i++){
$xtemp=$iMag*($zx[$i])+$vX[$b];
$ytemp=$iMag*($zy[$i])+$vY[$b];
$zx[$i]=toScreen($xtemp,$dMin,$dMax,900,200);
$zy[$i]=toScreen($ytemp,$dMin,$dMax,-900,900);}
ImageLine($pic,$zx[0],$zy[0],$zx[13],$zy[13],$black);
ImageLine($pic,$zx[0],$zy[0],$zx[14],$zy[14],$black);
ImageLine($pic,$zx[13],$zy[13],$zx[14],$zy[14],$black);
ImageLine($pic,$zx[1],$zy[1],$zx[2],$zy[2],$black);
ImageLine($pic,$zx[3],$zy[3],$zx[4],$zy[4],$black);
ImageLine($pic,$zx[5],$zy[5],$zx[6],$zy[6],$black);
ImageLine($pic,$zx[7],$zy[7],$zx[8],$zy[8],$black);
ImageLine($pic,$zx[9],$zy[9],$zx[10],$zy[10],$black);
ImageLine($pic,$zx[11],$zy[11],$zx[12],$zy[12],$black);
ImageArc($pic,$zx[15],$zy[15],15,15,0,360,$black);
ImageArc($pic,$zx[16],$zy[16],15,15,0,360,$black);
ImageArc($pic,$zx[17],$zy[17],15,15,0,360,$black);
ImageArc($pic,$zx[18],$zy[18],15,15,0,360,$black);
}
if($Ux[$b]=="0" && $Uy[$b]=="")
{
$zx[0]=0;$zy[0]=0;
$zx[1]=-20;$zy[1]=-10;
$zx[2]=-20;$zy[2]=10;
$zx[3]=-20;$zy[3]=-8;
$zx[4]=-23;$zy[4]=-11;
$zx[5]=-20;$zy[5]=-4;
$zx[6]=-23;$zy[6]=-7;
$zx[7]=-20;$zy[7]=1;
$zx[8]=-23;$zy[8]=-2;
$zx[9]=-20;$zy[9]=5;
```

```
                $zx[10]=-23;$zy[10]=2;
                $zx[11]=-20;$zy[11]=9;
                $zx[12]=-23;$zy[12]=6;
                $zx[13]=-15;$zy[13]=-10;
                $zx[14]=-15;$zy[14]=10;
                $zx[15]=-17.5;$zy[15]=-7.5;
                $zx[16]=-17.5;$zy[16]=-2.5;
                $zx[17]=-17.5;$zy[17]=2.5;
                $zx[18]=-17.5;$zy[18]=7.5;
        for($i=0;$i<=18;$i++){
                $xtemp=$iMag*($zx[$i])+$vX[$b];
                $ytemp=$iMag*($zy[$i])+$vY[$b];
                $zx[$i]=toScreen($xtemp,$dMin,$dMax,900,200);
                $zy[$i]=toScreen($ytemp,$dMin,$dMax,-900,900);}
                ImageSetThickness($pic, 1);
                ImageLine($pic,$zx[0],$zy[0],$zx[13],$zy[13],$black);
                ImageLine($pic,$zx[0],$zy[0],$zx[14],$zy[14],$black);
                ImageLine($pic,$zx[13],$zy[13],$zx[14],$zy[14],$black);
                ImageLine($pic,$zx[1],$zy[1],$zx[2],$zy[2],$black);
                ImageLine($pic,$zx[3],$zy[3],$zx[4],$zy[4],$black);
                ImageLine($pic,$zx[5],$zy[5],$zx[6],$zy[6],$black);
                ImageLine($pic,$zx[7],$zy[7],$zx[8],$zy[8],$black);
                ImageLine($pic,$zx[9],$zy[9],$zx[10],$zy[10],$black);
                ImageLine($pic,$zx[11],$zy[11],$zx[12],$zy[12],$black);
                ImageArc($pic,$zx[15],$zy[15],15,15,0,360,$black);
                ImageArc($pic,$zx[16],$zy[16],15,15,0,360,$black);
                ImageArc($pic,$zx[17],$zy[17],15,15,0,360,$black);
                ImageArc($pic,$zx[18],$zy[18],15,15,0,360,$black);
                        }
        if($Ux[$b]=="" && $Uy[$b]==""){
                $x=toScreen($vX[$b],$dMin,$dMax,900,200);
                $y=toScreen($vY[$b],$dMin,$dMax,-900,900);
                ImageFilledArc($pic,$x,$y,10,10,0,360,$black,IMG_ARC_EDGED);
}
}
header("Content-type: image/PNG");
ImagePNG($pic);
ImageDestroy($pic);
function toScreen($xy,$xyMin,$xyMax,$xyWidth,$xyOrigin){
$temp = $xyOrigin+$xy*$xyWidth/($xyMax-$xyMin);
return $temp;} ?>
```