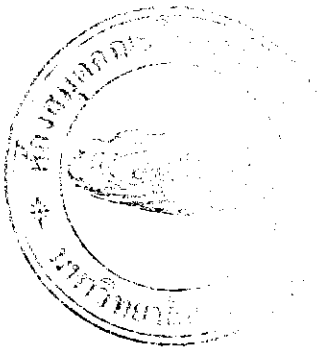


การพัฒนาโปรแกรมวิเคราะห์โครงข้อแข็งในระนาบด้วยภาษาพีเอชพี

DEVERLOPMENT OF PLANE FRAME ANALYSIS PROGRAM USING PHP LANGUAGE

นายคณาวุฒิ คีรีเกษตร รหัส 53360088

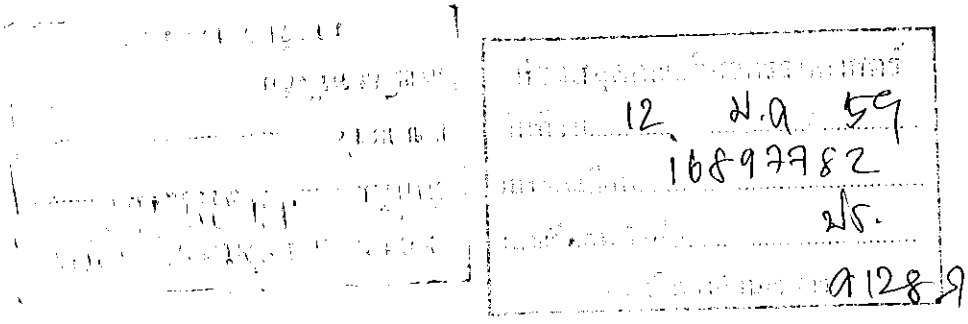
นายอนุชิต ช่างพินิจ รหัส 53360798

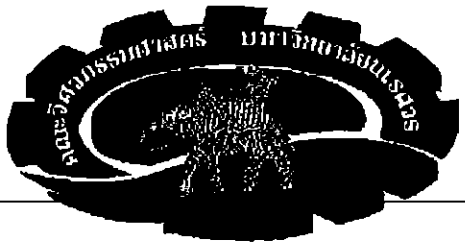
ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

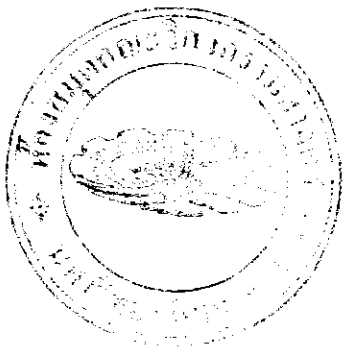
สาขาวิศวกรรมโยธา ภาควิชาวิศวกรรมโยธา

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2556







ใบรับรองปริญญานิพนธ์

ชื่อหัวข้อโครงงาน : การพัฒนาโปรแกรมวิเคราะห์โครงข้อแข็งในระนาบด้วยภาษาพีเอชพี

ผู้ดำเนินโครงงาน : นายคณาวุฒิ คีรีเกษตร รหัส 53360088

: นายอนุชิต ช่างพินิจ รหัส 53360798

ที่ปรึกษาโครงงาน : ผศ.ดร.สสิกรณ์ เหลืองวิชชเจริญ

สาขาวิชา : วิศวกรรมโยธา

ภาควิชา : วิศวกรรมโยธา คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมโยธา

คณะกรรมการสอบโครงงานวิศวกรรมศาสตร์

..... ประธานกรรมการ

(ผศ.ดร.สสิกรณ์ เหลืองวิชชเจริญ)

..... กรรมการ

(อาจารย์ภัคพงศ์ หอมเนียม)

..... กรรมการ

(อาจารย์บุญพล มีไชโย)

ชื่อหัวข้อโครงงาน	: การพัฒนาโปรแกรมวิเคราะห์โครงข้อแข็งในระนาบด้วยภาษาพีเอชพี
ผู้ดำเนินโครงงาน	: นายคณาวุฒิ คีรีเกษตร รหัส 53360088
	: นายอนุชิต ช่างพินิจ รหัส 53360798
ที่ปรึกษาโครงงาน	: ผศ.ดร.สสิกรณณ์ เหลืองวิชชเจริญ
สาขาวิชา	: วิศวกรรมโยธา
ภาควิชา	: วิศวกรรมโยธา
ปีการศึกษา	: 2556

.....

บทคัดย่อ

งานวิจัยนี้มุ่งพัฒนาโปรแกรมเคราะห์โครงข้อแข็งในระนาบด้วยภาษาพีเอชพี ซึ่งเป็นภาษาคอมพิวเตอร์สำหรับพัฒนาเว็บเพจแบบไดนามิก ที่ได้รับความนิยมมากในปัจจุบันและใช้กันอย่างแพร่หลาย โปรแกรมที่ได้พัฒนาขึ้นนี้ใช้หลักการวิเคราะห์โครงสร้างโดยวิธีการรวมสติฟเนสโดยตรง ในการคำนวณหาค่าต่างๆที่ต้องการ ผู้พัฒนาโปรแกรมได้ออกแบบโปรแกรมให้ใช้งานได้อย่างสะดวก เข้าใจง่าย แม่นยำ และอีกทั้งยังสามารถเข้าถึงโปรแกรมได้ง่ายทุกที่ทุกเวลาสามารถใช้งานได้กับทุก ระบบปฏิบัติการ เช่น ไอโอเอส, วินโดว์, แอนดรอยด์, ลีนุกซ์ ฯลฯ ทั้งนี้ผู้จัดทำได้เปิดให้ใช้งานฟรี โดยผู้ใช้สามารถป้อนพารามิเตอร์ทุกอย่างได้โดยง่าย ผลการคำนวณเมื่อเปรียบเทียบกับโปรแกรมวิเคราะห์โครงสร้าง เอสยูที สตรัคเจอร์ พบว่าค่าต่างๆที่ได้ถูกต้องตรงกันเป็นอย่างดี

Project Title	: Development of plane frame analysis program using php language
Name	: Mr.Kanawut Keereekaset ID.53360088
	: Mr.Anuchit Changpinit ID.53360798
Project advisor	: Assit.Prof.Dr.Sasikorn Leungvichcharoen
Major	: Civil Engineering
Department	: Civil Engineering
Academic year	: 2013

Abstract

This project proposes the method to develop of plane frame 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. The program developers design this program to used conveniently, easily understood, accuracy and can be use this program every time and everywhere with every operating system such as IOS, Windows, Android, Linux etc. All this, the program is free for user .Easy to use. 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 ชี้แนะแนวทางแก้ไขรวมถึงข้อคิดเห็นเสนอแนะต่างๆตลอดจนความดูแลเอาใจใส่ในการดำเนินโครงการมาโดยตลอด และขอขอบพระคุณคณะอาจารย์ประจำภาควิชาวิศวกรรมโยธา มหาวิทยาลัยนเรศวรทุกท่านที่ได้สอนวิชาความรู้เพื่อนำมาประยุกต์ใช้ในปริญญานิพนธ์ฉบับนี้

ขอขอบพระคุณคณะกรรมการทุกท่านที่มาร่วมการฟังบรรยายโครงงาน และได้ให้ข้อเสนอแนะที่เป็นประโยชน์ ซึ่งผู้จัดทำจะนำไปปรับปรุงและพัฒนาโครงงานให้สมบูรณ์

ขอขอบพระคุณเจ้าหน้าที่ และพนักงานภาควิชาวิศวกรรมโยธา ที่กรุณาให้ความช่วยเหลือร่วมมือตลอดระยะเวลาที่ทำโครงงานนี้เป็นอย่างดี

สุดท้ายนี้ผู้ดำเนินโครงงานขอกราบขอบคุณ บิดา มารดา ที่ได้ให้การดูแล อบรมสั่งสอนและให้กำลังใจด้วยดีเสมอมา ตลอดการดำเนินโครงงานจนสำเร็จการศึกษา

คณะผู้จัดทำโครงงานวิศวกรรม
นายคณาวุฒิ คีรีเกษตร
นายอนุชิต ช่างพินิจ
20 มีนาคม 2557

สารบัญ

เรื่อง	หน้า
ใบรับรองปริญญานิพนธ์	ก
บทคัดย่อภาษาไทย	ข
บทคัดย่อภาษาอังกฤษ	ค
กิตติกรรมประกาศ	ง
สารบัญ	จ
สารบัญตาราง	ช
สารบัญรูป	ซ
สารบัญสัญลักษณ์และอักษรย่อ	ญ
คำนิยามศัพท์	ฎ
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขอบเขตของโครงงาน	1
1.4 ขั้นตอนการดำเนินงาน	2
1.5 ผลที่คาดว่าจะได้รับ	2
1.6 แผนงานดำเนินงานตลอดโครงการ	3
1.7 รายละเอียดงบประมาณของโครงงาน	3
บทที่ 2 หลักการและทฤษฎี	4
2.1 วิธีการรวมสติฟเนสโดยตรง(Direct Stiffness Method)	4
2.2 การสังเคราะห์สติฟเนสของโครงสร้างข้อแข็งสองมิติ	4
2.3 โคออร์ดิเนตของการเปลี่ยนตำแหน่งอิสระ (Degree of Freedom)	4
2.4 สติฟเนตของชิ้นส่วนย่อยในระบบโคออร์ดิเนต (Local Stiffness Matrix)	5
2.5 ระบบโคออร์ดิเนตโกลบัล (Global Stiffness System)	6
2.6 เมตริกซ์แปลงสำหรับชิ้นส่วนโครงสร้างข้อแข็งสองมิติ	7
2.7 สติฟเนสของชิ้นส่วนย่อยในระบบโกลบัล (Global stiffness of Member)	8

สารบัญ (ต่อ)

เรื่อง	หน้า
2.8 สรุปขั้นตอนการวิเคราะห์โครงสร้างด้วยวิธีการรวมผลสติฟเนสโดยตรงของโครงสร้าง	9
บทที่ 3 วิธีการดำเนินโครงการ	10
3.1แผนผังขั้นตอนการทำงานของโปรแกรม	11
3.2 รายละเอียดของโปรแกรม	14
บทที่ 4 ผลการทดลองและวิเคราะห์	15
ตัวอย่างที่ 4.1 โครงข้อแข็ง 2 มิติ	15
ตัวอย่างที่ 4.2 โครงข้อแข็ง 2 มิติ	19
บทที่ 5 สรุปผลและข้อเสนอแนะ	31
5.1 สรุปผล	26
5.2 ข้อแนะนำ	26
บรรณานุกรม	ฐ
ภาคผนวก	ฑ

สารบัญรูป

รูปที่ หน้า

ตารางที่ 1.1 แผนการดำเนินงานตลอดโครงการ 3
ตารางที่ 4.1 แสดงการเปรียบเทียบค่า Member force ระหว่างโปรแกรม SUTStructor กับโปรแกรมวิเคราะห์โครงข้อแข็ง 2 มิติ(ตัวอย่างที่ 4.1) 23
ตารางที่ 4.2 แสดงการเปรียบเทียบค่า Displacement ระหว่างโปรแกรม SUTStructor กับโปรแกรมวิเคราะห์โครงข้อแข็ง 2 มิติ(ตัวอย่างที่ 4.1) 24
ตารางที่ 4.3 แสดงการเปรียบเทียบค่า Member force ระหว่างโปรแกรม SUTStructor กับโปรแกรมวิเคราะห์โครงข้อแข็ง 2 มิติ(ตัวอย่างที่ 4.2) 24
ตารางที่ 4.4 แสดงการเปรียบเทียบค่า Displacement ระหว่างโปรแกรม SUTStructor กับโปรแกรมวิเคราะห์โครงข้อแข็ง 2 มิติ(ตัวอย่างที่ 4.2) 25

สารบัญรูป

รูปที่	หน้า
2.1 แสดงดีกรีของความอิสระของแต่ละจุดต่อในระบบโคออร์ดิเนตประจำตัว	5
2.2 แสดงแรง (S) และเคลื่อนที่ (v) ในระบบโคออร์ดิเนตประจำตัว	6
2.3 แสดงแรง ($\bar{S}$) และการเคลื่อนที่ ($\bar{v}$) ในระบบโคออร์ดิเนตโกลบัล	7
3.1 ขั้นตอนการทำงานของโปรแกรมวิเคราะห์โครงข้อแข็งด้วยภาษา PHP	10
3.2 หน้าหลักของเว็บไซต์	13
3.3 การรับข้อมูลจำนวน Node และจำนวน Member	11
3.4 ตารางรับข้อมูลของ Node Information	11
3.5 ตารางรับข้อมูลของ Member Information	12
3.6 แสดงรูปโครงสร้างข้อแข็งเดิม	13
3.7 แสดงรูปตาราง Node Information, Member Information, Displacement, Member Force	13
3.8 แสดง Displacement(ซ้าย) และShear Force Diagram(ขวา)	14
3.9 รูปแสดง Axial Load Diagram(ซ้าย) และBending Moment Diagram(ขวา)	14
4.1 แสดงรูปโจทย์ตัวอย่างที่ 4.1	15
4.2 รูปโครงสร้างจากโปรแกรม SUTStructor	15
4.3 แสดง Displacement ของโครงสร้าง(ซ้าย) และShear Force Diagram(ขวา)	16
4.4 รูปแสดง Axial Load Diagram(ซ้าย) และBending Moment Diagram(ขวา)	16
4.5 แสดงตาราง Node และ Member ที่ป้อนค่าพารามิเตอร์ต่างๆในการคำนวณ จากโจทย์ตัวอย่างที่ 4.1	17
4.6 แสดงรูปโครงสร้างเดิมจากโปรแกรมวิเคราะห์โครงข้อแข็ง 2 มิติ	17
4.7 แสดงผลการคำนวณเป็นตาราง Node Information, Member Information, Displacement และ Member Force	18
4.8 แสดง Displacement(ซ้าย), และShear Force Diagram(ขวา)	18
4.9 แสดง Axial Load Diagram(ซ้าย), Bending Moment Diagram(ขวา)	18

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.10 แสดงรูปโจทย์ตัวอย่างที่ 4.2	19
4.11 รูปโครงสร้างจากโปรแกรม SUTStructor	19
4.12 แสดง Displacement ของโครงสร้าง(ซ้าย) และShear Force Diagram(ขวา)	20
4.13 รูปแสดง Axial Load Diagram(ซ้าย) และBending Moment Diagram(ขวา)	20
4.14 แสดงตาราง Node และ Member ที่ป้อนค่าพารามิเตอร์ต่างๆในการคำนวณ จากโจทย์ตัวอย่างที่ 4.2	21
4.15 แสดงรูปโครงสร้างเดิมจากโปรแกรมวิเคราะห์โครงข้อแข็ง 2 มิติ	21
4.16 แสดงผลการคำนวณเป็นตาราง Node Information, Member Information, Displacement และ Member Force	22
4.17 แสดง Displacement(ซ้าย), และShear Force Diagram(ขวา)	22
4.18 แสดง Axial Load Diagram(ซ้าย), Bending Moment Diagram(ขวา)	22

สารบัญสัญลักษณ์และอักษรย่อ

S	หมายถึง แรงภายในของระบบโคออร์ดิเนตประจำตัว
$\tilde{S}$	หมายถึง แรงภายในของระบบโคออร์ดิเนตโกลบัล
v	หมายถึง การเคลื่อนที่ของระบบโคออร์ดิเนตประจำตัว
$\tilde{v}$	หมายถึง การเคลื่อนที่ของระบบโคออร์ดิเนตโกลบัล
u	หมายถึง การเคลื่อนที่ของข้อต่อ
A	หมายถึง พื้นที่หน้าตัดของชิ้นส่วน
E	หมายถึง ค่าโมดูลัสของชิ้นส่วน
I	หมายถึง โมเมนต์อินเนอร์เชียของหน้าตัด
L	หมายถึง ความยาวของชิ้นส่วน
k	หมายถึง สติฟเนสของชิ้นส่วนย่อยในระบบโคออร์ดิเนตประจำตัว
$\tilde{k}$	หมายถึง สติฟเนสของชิ้นส่วนย่อยในระบบโคออร์ดิเนตโกลบัล
a	หมายถึง เมตริกซ์แปลง (Transformation Matrix) ของชิ้นส่วนโครงสร้างข้อแข็ง
θ	หมายถึง เป็นมุมวัดจากแนวระนาบไปยังชิ้นส่วน
u_i	หมายถึง การเคลื่อนที่ของข้อต่อที่ ดีกรี ของความอิสระของโครงสร้างที่ i
$\tilde{v}^m$	หมายถึง การเคลื่อนที่ที่ปลายของชิ้นส่วน m ณ จุดต่อเดียวกันและการเคลื่อนที่ในทิศเดียวกับ u_i
P_i	หมายถึง แรงภายนอกที่มากระทำ ณ ข้อต่อ ที่ดีกรีของความอิสระของโครงสร้างที่ i
$\tilde{S}^{m_r}$	หมายถึง แรงภายในที่เกิดขึ้นของชิ้นส่วน m_r ในทิศเดียวกับP_i ณ จุดต่อเดียวกัน
P	หมายถึง แรงที่มากระทำภายนอก
K	หมายถึง สติฟเนสของโครงสร้างในระบบโคออร์ดิเนตโกลบัล

คำนิยามศัพท์

ภาษา PHP	ภาษาคอมพิวเตอร์ในลักษณะเซิร์ฟเวอร์-ไซด์ สคริปต์
Structural Analysis	การวิเคราะห์โครงสร้าง
Fixed end support	ฐานรองรับแบบคงที่
Hinge support	ฐานรองรับแบบยึดหมุน
Roller support	ฐานรองรับแบบล้อเลื่อน
Point Load At Node	แรงกระทำที่จุดต่อ
Direct Stiffness Method	วิธีรวมสติฟเนสโดยตรง
Displacement	การเคลื่อนที่
Reaction	แรงปฏิกิริยาที่ฐานรองรับ
Global Coordinate System	ระบบโคออร์ดิเนตโกลบัล
Global stiffness of Member	สติฟเนสของชิ้นส่วนย่อยในระบบโกลบัล
Global Stiffness System	ระบบโคออร์ดิเนตโกลบัล
Global Stiffness of Space Frame Structure	สติฟเนสของโครงสร้างในระบบโคออร์ดิเนตโกลบัล
Local coordinate system	ระบบโคออร์ดิเนตประจำตัว
Local Stiffness Matrix	สติฟเนสของชิ้นส่วนย่อยในระบบโคออร์ดิเนตประจำตัว
Degree of Freedom	ดีกรีของความอิสระ
force component	แรงตามแกน
Transformation Matrix	เมตริกซ์แปลง
displacement transformation matrix	เมตริกซ์การเปลี่ยนตำแหน่ง
Compatibility Conditions	เงื่อนไขของความต่อเนื่อง
Equilibrium Condition	สภาวะสมดุล
Member	ชิ้นส่วน
node	จุดต่อ
Nodal displacement	การเคลื่อนที่ ณ จุดต่อ

คำนิยามศัพท์ (ต่อ)

Axial	แรงภายใน
Shear	แรงเฉือน
Moment	โมเมนต์
Transformation Matrix of Space Frame Member	เมตริกซ์แปลงสำหรับชิ้นส่วนโครงสร้างข้อแข็งสองมิติ
Direct Stiffness Method of Space Frame Structure	การวิเคราะห์โครงสร้างด้วยวิธีการรวมผลสติฟเนสโดยตรงของโครงสร้างข้อแข็ง

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของโครงการ

เนื่องจากการวิเคราะห์โครงสร้าง (Structural Analysis) มีความสำคัญมากในการคำนวณออกแบบโครงสร้างต่างๆทางด้านวิศวกรรม ถือว่าเป็นความรู้พื้นฐานที่มีความจำเป็นสำหรับวิศวกร ซึ่งหมายถึงความแข็งแรงและความปลอดภัยของโครงสร้างอาคารที่ออกแบบมา และในปัจจุบันได้มีพัฒนาเทคโนโลยีทางด้านคอมพิวเตอร์มาใช้ในการคำนวณออกแบบเพื่อความแม่นยำ และมีประสิทธิภาพที่มากขึ้น อีกทั้งยังมีโปรแกรมวิเคราะห์โครงสร้างต่างๆมากมาย โดยโปแกรมส่วนใหญ่ที่ใช้ทั่วไปนั้นจะต้องใช้ในระบบปฏิบัติการเดียวกันจึงจะสามารถใช้งานได้ เนื่องจากไม่สามารถใช้งานข้าม Platform ได้อย่างสมบูรณ์ คณะผู้จัดทำจึงมีความคิดที่จะพัฒนาโปรมแกรมโดยใช้ภาษาพีเอชพีมาร่วมกับการวิเคราะห์โครงข้อแข็ง โดยจะสามารถใช้งานได้กับทุกระบบปฏิบัติการ เช่น Windows, MacOS, IOS, Android, Linux ฯลฯ และช่วยลดปริมาณการซื้อซอฟแวร์หรือลดปัญหาการละเมิดลิขสิทธิ์ซอฟแวร์จากต่างประเทศได้ และเป็นประโยชน์ต่อผู้ที่สนใจจะศึกษาและต้องการใช้งานโปรแกรมวิเคราะห์โครงสร้างผ่านทางการใช้งานในรูปแบบเว็บเพจต่อไป

1.2 วัตถุประสงค์ของโครงการ

พัฒนาโปรแกรมการวิเคราะห์โครงสร้างข้อแข็งด้วยภาษาพีเอชพี เพื่อเผยแพร่ให้แก่ผู้ที่สนใจที่จะศึกษาสามารถใช้งานได้ในทุกระบบปฏิบัติการ และสามารถใช้งานได้ฟรีผ่านทางอินเตอร์เน็ต (Free Web Application)

1.3 ขอบเขตของโครงงาน

เขียนโปรแกรมวิเคราะห์โครงข้อแข็งด้วยภาษาพีเอชพี โดยวิธีการรวมสติฟเนสโดยตรงซึ่งมีชนิดของฐานรองรับเป็น แบบคงที่ (Fixed end), แบบจุดยึดหมุน (Hinge), และยึดหมุนเคลื่อนที่ได้ (Roller) สามารถวิเคราะห์โครงสร้างที่มีรูปแบบแรงเป็นแรงกระทำตามจุดยึดทั้งแนวราบและแนวดิ่ง(Point Load At Node) ลักษณะโครงสร้างมีการเชื่อต่อกันแบบโครงข้อแข็ง

1.4 ขั้นตอนการดำเนินงาน

1.4.1 คัดเลือกหัวข้อโครงงานที่น่าสนใจ

1.4.2 นำเสนอโครงงานกับอาจารย์ที่ปรึกษา

1.4.3 ศึกษาเนื้อหาเกี่ยวการเขียนโปรแกรมด้วยภาษาพีเอชพี

1.4.4 ศึกษาเนื้อหาทางทฤษฎีของการวิโครงสร้างข้อแข็งด้วยวิธีรวมสติฟเนสโดยตรง

1.4.5 วางแผนและเขียนขั้นตอนในการทำงาน

1.4.6 เขียนภาษาพีเอชพีให้ออกมาเป็นโปรแกรมวิเคราะห์โครงข้อแข็ง

1.4.7 ทดสอบโปรแกรมเพื่อหาข้อผิดพลาดและปรับปรุงแก้ไข

1.4.8 ให้อาจารย์ที่ปรึกษาทำงานตรวจสอบ

1.4.9 จัดทำเอกสารรายงานสรุปผลโครงการและจัดทำรูปเล่ม

1.5 ผลที่คาดว่าจะได้รับ

1.5.1 พัฒนาโปรแกรมวิเคราะห์โครงข้อแข็ง 2 มิติ ที่สามารถใช้งานได้ในทุกระบบปฏิบัติการ เช่น Windows, MacOS, IOS, Android, Linux ฯลฯ

1.5.2 สามารถเผยแพร่โปรแกรมให้กับผู้ที่ต้องการศึกษาและใช้งาน โดยใช้งานผ่านเว็บไซต์ฟรี

1.5.3 สะดวกในการทำงานทางด้านวิศวกรโยธา

1.5.4 ช่วยลดปริมาณการซื้อซอฟแวร์หรือลดปัญหาการละเมิดลิขสิทธิ์ซอฟแวร์จากต่างประเทศได้

1.5.5 ความเข้มแข็งให้แก่วงการวิศวกรรมโยธาของประเทศไทยได้อีกทางหนึ่ง

1.5.6 เข้าใจในทฤษฎีการวิเคราะห์โครงสร้างวิธี Direct Stiffness Method มากยิ่งขึ้น เพื่อเป็นประโยชน์ในการทำงานในอนาคต

1.5.7 ผู้จัดทำได้รับความรู้เกี่ยวกับภาษา PHP เพิ่มเติมซึ่งจะเป็นประโยชน์ในการประยุคใช้ในการทำงานในภายภาคหน้า

1.6 แผนงานดำเนินงานตลอดโครงการ

ตารางที่ 1.1 แผนการดำเนินงานตลอดโครงการ

เดือนกิจกรรม	ตุลาคม 1 2 3 4	พฤศจิกายน 1 2 3 4	ธันวาคม 1 2 3 4	มกราคม 1 2 3 4	กุมภาพันธ์ 1 2 3 4	มีนาคม 1 2 3 4
1. เสนอโครงการ						
2.ศึกษาการเขียนโปรแกรมด้วยภาษาพีเอชพีและการรวมสติฟเนสโดนตรง						
3.เขียนโปรแกรมด้วยภาษาพีเอชพี						
4.ตรวจสอบข้อผิดพลาดและปรับปรุงแก้ไข						
5.เขียนรายงานรูปเล่ม						

1.7 รายละเอียดงบประมาณของโครงงาน

1. ค่าสำเนาเอกสารข้อมูล	1000 บาท
2. ค่าจัดทำรูปเล่มและเผยแพร่	1000 บาท
รวมเป็นเงิน	2000 บาท(สองพันบาทถ้วน)

หมายเหตุ ขออนุมัติถัวเฉลี่ยทุกรายการ

บทที่ 2
หลักการและทฤษฎีเบื้องต้น

การวิเคราะห์โครงสร้างมักจะใช้แบบจำลองทางคณิตศาสตร์ของโครงสร้าง มาใช้เพื่อให้การวิเคราะห์โครงสร้างเป็นไปได้ โดยทั่วไป แบบจำลองทางคณิตศาสตร์ที่ใกล้เคียงกับโครงสร้างจริง ร่วมกับสมมุติฐาน และวิธีการวิเคราะห์ที่เหมาะสม จะให้ผลการวิเคราะห์ที่ใกล้เคียงกับพฤติกรรมของโครงสร้างของจริง

ในวิธีนี้ค่าการเคลื่อนที่ของโครงสร้าง (Displacement) จะใช้เป็นตัวไม่ทราบค่า เพื่อใช้หาตัวแปรต่างๆที่ต้องการ เช่น แรงปฏิกิริยา (Reaction) แรงภายใน (Member Force) เป็นต้น ความสัมพันธ์ของการเคลื่อนที่ของโครงสร้างกับส่วนของโครงสร้าง หาได้จากลักษณะความสัมพันธ์ของโครงสร้าง และอาศัยหลักการสมดุลของแรงกับการเคลื่อนที่ ทำให้คำนวณตัวแปรที่ไม่ทราบค่าได้ จากการแทนค่าของตัวไม่รู้ค่าลงในสมการความสมพันธ์ระหว่างแรงกับการเคลื่อนที่ภายในโครงสร้าง

2.1 วิธีการรวมสติฟเนสโดยตรง(Direct Stiffness Method)

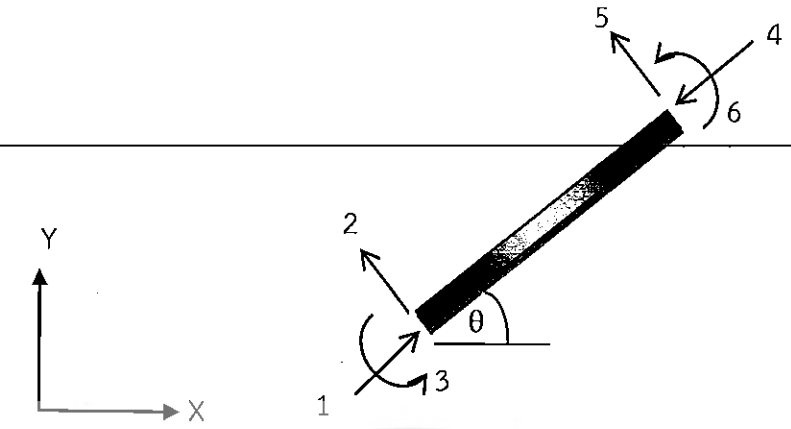
เป็นวิธีการสังเคราะห์ สติฟเนสของโครงสร้างทั้งระบบ จากสติฟเนสของชิ้นส่วนย่อยๆ โดยการเลือกระบบโคออร์ดิเนตที่เหมาะสม ซึ่งเป็นวิธีที่มีประสิทธิภาพ และเหมาะสมอย่างยิ่งสำหรับการวิเคราะห์โครงสร้างใหญ่ หรือ โครงสร้างที่ยุ่งยาก โดยอาศัยเครื่องคอมพิวเตอร์

2.2 การสังเคราะห์สติฟเนสของโครงสร้างข้อแข็งสองมิติ

สำหรับโครงสร้างที่กำหนดให้ จะใช้ระบบแกนคาร์ทีเซียน (X,Y) เพื่อใช้กำหนดเรขาคณิตของโครงสร้างและการเปลี่ยนตำแหน่งของข้อต่อทั้งระบบ เรียกว่า ระบบโคออร์ดิเนตโกลบัล (Global Coordinate System) การสังเคราะห์โครงสร้าง กระทำได้โดยการพิจารณาลักษณะของชิ้นส่วนย่อย ความต่อเนื่องของการเปลี่ยนตำแหน่ง และสภาวะสมดุลของข้อต่อ

2.3 โคออร์ดิเนตของการเปลี่ยนตำแหน่งอิสระ (Degree of Freedom)

สำหรับโครงข้อแข็งสองมิติ ซึ่งใน 1 จุดจะมี 3 ดีกรีของความอิสระ (Degree of freedom) ได้แก่ การเคลื่อนที่ในทิศทาง X, Y และการหมุนตามแนวแกนของการเคลื่อนที่ทั้ง 2 แกนดังรูป



รูปที่ 2.1 แสดงดีกรีของความอิสระของแต่ละจุดต่อในระบบโคออร์ดิเนตประจำตัว

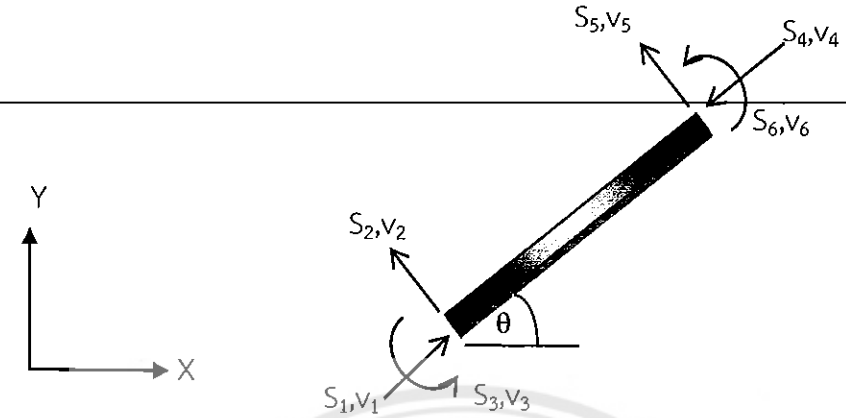
2.4 สติฟเนสของชิ้นส่วนย่อยในระบบโคออร์ดิเนต (Local Stiffness Matrix)

สติฟเนสของชิ้นส่วนย่อยในระบบโคออร์ดิเนตประจำตัว สำหรับชิ้นส่วนใดๆเลือกระบบโคออร์ดิเนตประจำตัว (Local coordinate System) ซึ่งผ่านแนวแกนของชิ้นส่วน และเขียนความสัมพันธ์ของแรงที่ปลายชิ้นส่วน { P } กับการเปลี่ยนตำแหน่งของปลายชิ้นส่วน { v }

$$S = k\,v \tag{1}$$

ซึ่งสำหรับชิ้นส่วนโครงข้อแข็งสองมิติ จะมีแรงและการเปลี่ยนตำแหน่งที่ปลายทั้งสอง อย่างล่ะ 6 องค์ประกอบ ดังแสดงในรูปที่

$$\begin{Bmatrix} S_1^1 \\ S_2^1 \\ S_3^1 \\ S_4^1 \\ S_5^1 \\ S_6^1 \end{Bmatrix} = \begin{bmatrix} k_{11}^1 & k_{12}^1 & k_{13}^1 & k_{14}^1 & k_{15}^1 & k_{16}^1 \\ k_{21}^1 & k_{22}^1 & k_{23}^1 & k_{24}^1 & k_{25}^1 & k_{26}^1 \\ k_{31}^1 & k_{32}^1 & k_{33}^1 & k_{34}^1 & k_{35}^1 & k_{36}^1 \\ k_{41}^1 & k_{42}^1 & k_{43}^1 & k_{44}^1 & k_{45}^1 & k_{46}^1 \\ k_{51}^1 & k_{52}^1 & k_{53}^1 & k_{54}^1 & k_{55}^1 & k_{56}^1 \\ k_{61}^1 & k_{62}^1 & k_{63}^1 & k_{64}^1 & k_{65}^1 & k_{66}^1 \end{bmatrix} \begin{Bmatrix} v_1^1 \\ v_2^1 \\ v_3^1 \\ v_4^1 \\ v_5^1 \\ v_6^1 \end{Bmatrix} \tag{2}$$



รูปที่ 2.2 แสดงแรง (S) และเคลื่อนที่ (v) ที่ปลายของชิ้นส่วนโครงข้อแข็งสองมิติ ในระบบโคออร์ดิเนตประจำตัว (Local coordinate system)

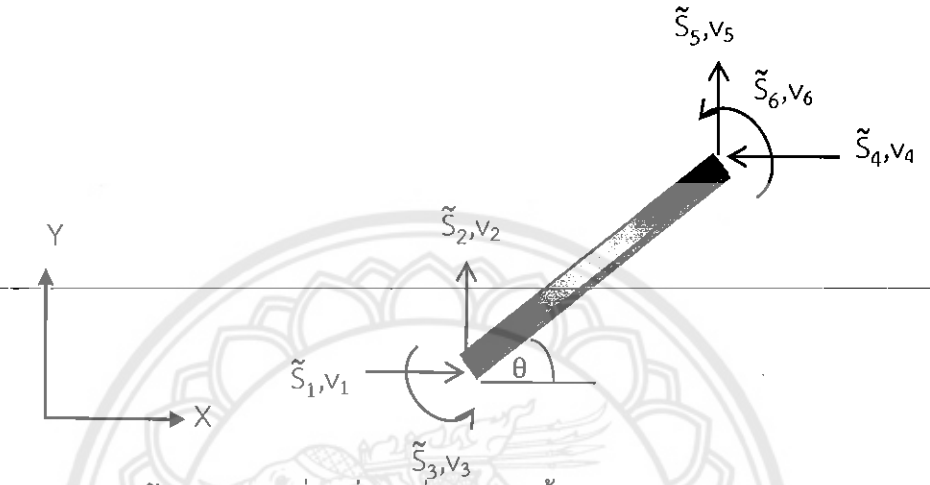
พึงสังเกตว่า การใช้ระบบโคออร์ดิเนตประจำตัวทำให้สามารถเขียนสติฟเนสของชิ้นส่วนต่างๆ ในรูปแบบเดียวกันโดยไม่ขึ้นกับทิศทางชิ้นส่วนนั้นๆทำให้ง่ายต่อการสร้าง (Generate) สติฟเนสประจำตัว โดยใช้โปรแกรมย่อย อันเดียวสำหรับชิ้นส่วนประเภท เดียวกัน

$$[K]=\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{array}\begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{bmatrix} \frac{AE}{L} & 0 & 0 & -\frac{AE}{L} & 0 & 0 \\ 0 & \frac{12EI}{L^3} & \frac{4EI}{L} & 0 & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{4EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{2EI}{L} \\ -\frac{AE}{L} & 0 & 0 & \frac{AE}{L} & 0 & 0 \\ 0 & -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & 0 & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{2EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{4EI}{L} \end{bmatrix} \end{array} \quad (3)$$

2.5 ระบบโคออร์ดิเนตโกลบัล (Global Coordinate System)

เนื่องจากระบบโคออร์ดิเนตประจำตัวของชิ้นส่วนแต่ละชิ้นส่วนหันไปในทิศทางต่างๆกัน เวกเตอร์ของแรงที่ปลายชิ้นส่วนแต่ละอัน ซึ่งแสดงเป็นส่วนของแรง (force component) ตามแกนของระบบโคออร์ดิเนตประจำตัว จึงไม่สามารถรวมกันได้โดยตรงในการพิจารณาสภาวะสมดุลของข้อต่อ เพื่อให้

สามารถทำการรวมเวคเตอร์ได้โดยตรง จำเป็นต้องนิยามแรงและการเปลี่ยนตำแหน่งของแต่ละชิ้นส่วนในระบบแกนร่วมกัน ในที่นี้ใช้ระบบโคออร์ดิเนตโกลบัล (x,y) รูปที่ 2.3 แสดงแรงและการเปลี่ยนตำแหน่งที่ปลายของชิ้นส่วนในระบบโคออร์ดิเนตทั้งสอง



รูปที่ 2.3 แสดงแรง ($\tilde{S}$) และการเคลื่อนที่ (v) ที่ปลายของชิ้นส่วนโครงข้อแข็งสองมิติในระบบโคออร์ดิเนตโกลบัล (Global coordinate system)

2.6 เมตริกซ์แปลงสำหรับชิ้นส่วนโครงสร้างข้อแข็งสองมิติ (Transformation Matrix of Space Frame Member)

สมการแสดงการเปลี่ยนแรงจากระบบโคออร์ดิเนตประจำตัวเป็นแรงของระบบโคออร์ดิเนตโกลบัลตามมุม θ_1

ในที่นี้ θ เป็นมุมวัดจากแกนราบไปยังชิ้นส่วน

$$\tilde{s}_1 = S_1\cos\theta_1 - S_2\sin\theta_1 \quad (4)$$

$$\tilde{s}_2 = S_1\sin\theta_1 - S_2\cos\theta_1 \quad (5)$$

$$\tilde{s}_3 = S_3 \quad (5)$$

$$\tilde{s}_4 = S_4\cos\theta_1 - S_5\sin\theta_1 \quad (6)$$

$$\tilde{s}_5 = S_4\sin\theta_1 + S_5\cos\theta_1 \quad (7)$$

$$\tilde{s}_6 = S_6 \quad (8)$$

เมื่อ $v = av$ (9)

และ $S = a\tilde{s}$ (10)

โดยที่ [a] มีชื่อว่าเมตริกซ์แปลงการเปลี่ยนตำแหน่ง (displacement transformation matrix) สำหรับชิ้นส่วน

$$[a]=\begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos\theta & \sin\theta & 0 \\ 0 & 0 & 0 & -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \qquad (11)$$

จากความสัมพันธ์ $v = a\tilde{v}$ (12)

จะได้

$$\begin{Bmatrix} v_1^1 \\ v_2^1 \\ v_3^1 \\ v_4^1 \\ v_5^1 \\ v_6^1 \end{Bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos\theta & \sin\theta & 0 \\ 0 & 0 & 0 & -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} \tilde{v}_1^1 \\ \tilde{v}_2^1 \\ \tilde{v}_3^1 \\ \tilde{v}_4^1 \\ \tilde{v}_5^1 \\ \tilde{v}_6^1 \end{Bmatrix} \qquad (13)$$

2.7 สติฟเนสของชิ้นส่วนย่อยในระบบโกลบัล (Global stiffness of Member)

จากความสัมพันธ์ของสมการที่ (1) กับสมการที่ (10) จะได้

$$S = k\ a\ \tilde{v} \qquad (14)$$

และเมื่อคูณ a^T เข้าในสมการที่ (12) จะได้ว่า

$$a^T S = a^T k\, a\, \tilde{v} \qquad (15)$$

จากความสัมพันธ์ $\tilde{s} = a^T s$ (16)

จะได้ $\tilde{s} = a^T k\, a\, \tilde{v}$ (17)

ดังนั้น $\tilde{s} = \tilde{k}\ \tilde{v}$ (18)

เมื่อ $\tilde{k}$ คือสติฟเนสเมตริกซ์ของชิ้นส่วนในระบบโคออร์ดิเนตโกลบัล

$$\tilde{k} = a^{T} k\, a \quad (19)$$

โดยสติฟเนสของโครงสร้างในระบบโคออร์ดิเนตโกลบัล (Global Stiffness of Space Frame Structure) คือ

$$P = K\, u \quad (20)$$

เมื่อ P คือแรงภายนอกที่กระทำ

u คือ ค่าการเคลื่อนที่

2.8 สรุปขั้นตอนการวิเคราะห์โครงสร้างด้วยวิธีการรวมผลสติฟเนสโดยตรงของโครงสร้าง (Direct Stiffness Method of Space Frame Structure)

2.11.1 แยกชิ้นส่วน (Member) ของแต่ละชิ้นส่วนของโครงสร้างออกมา กำหนด node และจำนวน Member เข้าไป

2.11.2 กำหนดโคออร์ดิเนตทั้งจุดเริ่มต้นและจุดปลายให้กับแต่ละชิ้นส่วนแล้วหาค่า k (Local Stiffness Matrix) ของแต่ละชิ้นส่วน

2.11.3 ทำการแปลงเมตริกซ์ (Transformation matrix) จาก k มาเป็น $\tilde{k}$ (Global stiffness matrix) โดยคูณการคูณเมตริกซ์ a (displacement transformation matrix)

2.11.4 ทำการรวมค่า $\tilde{k}$ ในแต่ละชิ้นส่วน จะได้ K (Global stiffness matrix of structure) คือโครงสร้างทั้งระบบ

2.11.5 คำนวณหาค่า u (Nodal displacement)

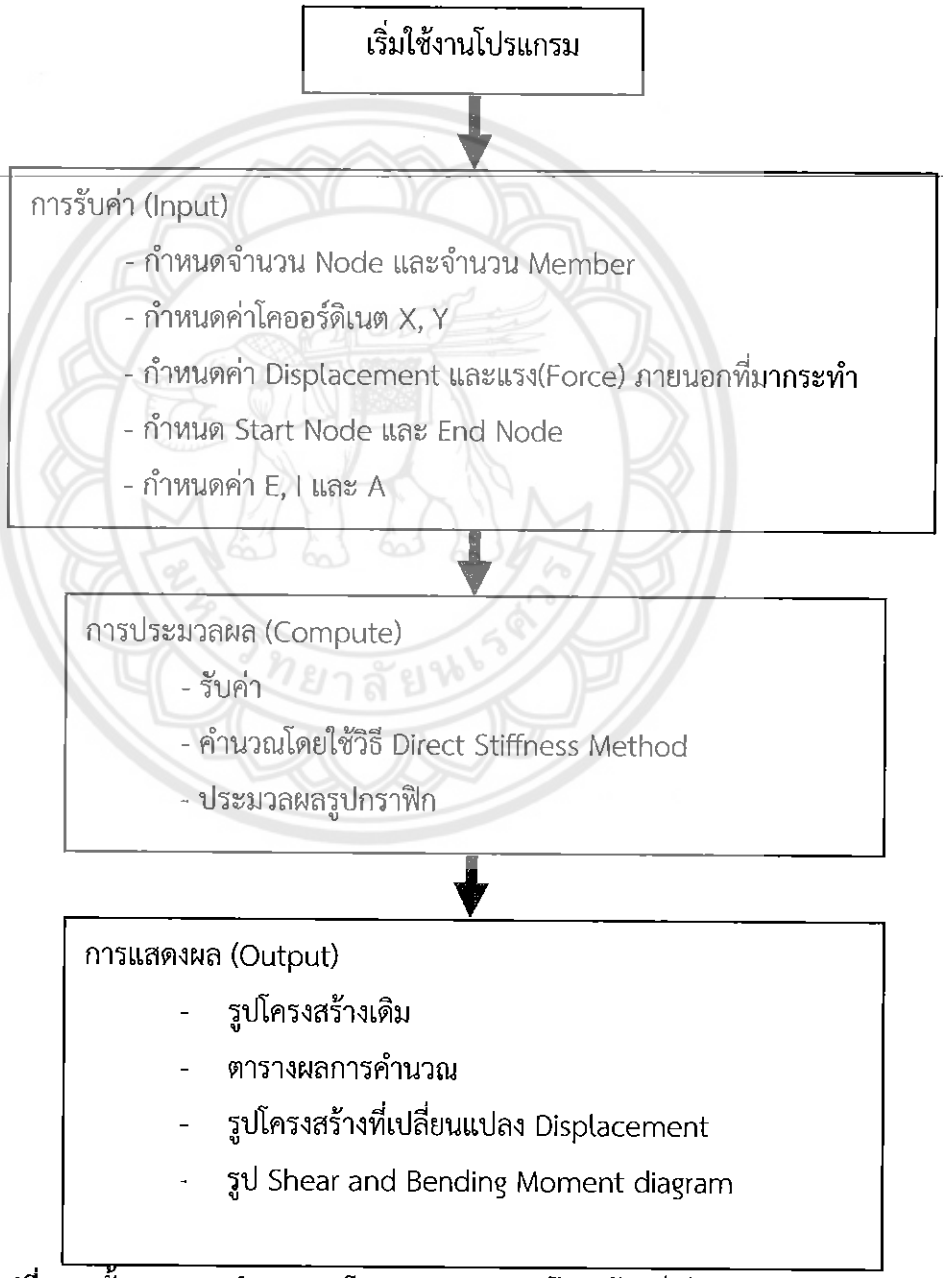
จาก $P = K\, u$

2.11.6 คำนวณหาค่า S ของแต่ละชิ้นส่วนจะได้ แรงภายใน (Axial), แรงเฉือน(Shear) และโมเมนต์(Moment) ที่ปลายทั้งหมดของ Member

2.11.7 ทำการแปลงเมตริกซ์ จาก S มาเป็น $\tilde{s}$ ของแต่ละชิ้นส่วน จะได้แรงปฏิกิริยาที่ฐานรองรับตามต้องการ

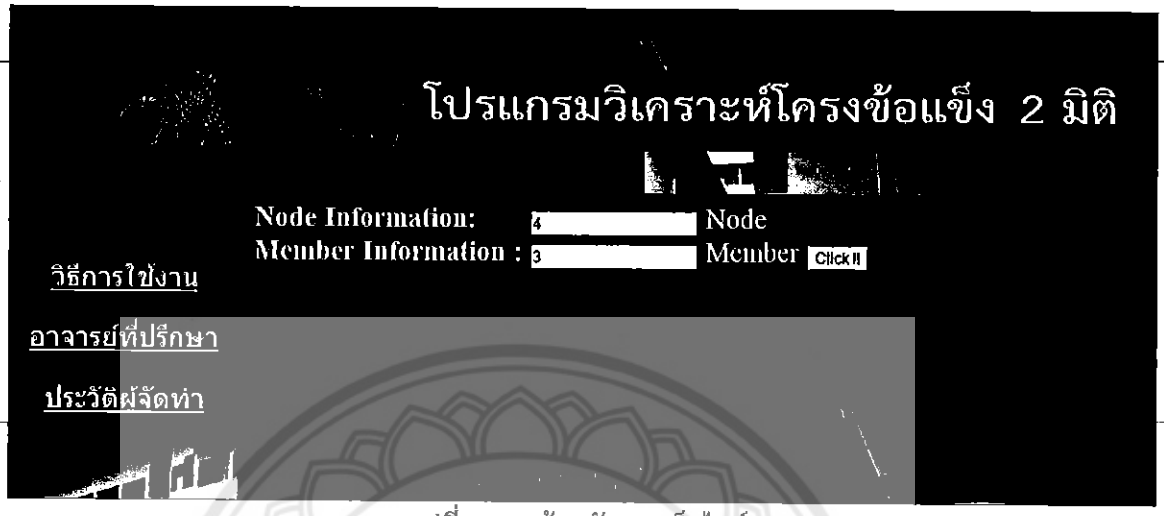
บทที่ 3
วิธีการดำเนินโครงการ

3.1 แผนผังขั้นตอนการทำงานของโปรแกรม



รูปที่ 3.1 ขั้นตอนการทำงานของโปรแกรมวิเคราะห์โครงข้อแข็งด้วยภาษา PHP

3.2 รายละเอียดของโปรแกรม



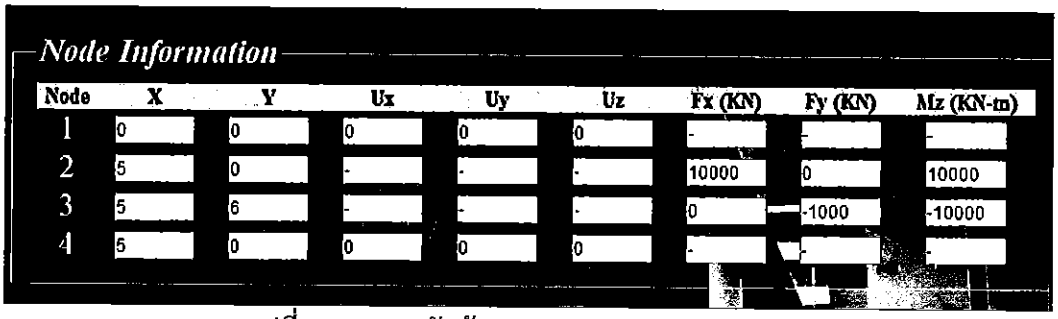
รูปที่ 3.2 หน้าหลักของเว็บไซต์

3.2.1 การนำเข้าข้อมูล (Input)

3.2.1.1 กำหนดจำนวน Node และจำนวน Member โดยใช้กำหนดจำนวน Node และ Member ในช่องดังกล่าวดังรูปที่ 3.3

รูปที่ 3.3 การรับข้อมูลจำนวน Node และจำนวน Member

3.2.1.2 กำหนด Node Information ดังรูปที่ 3.4



รูปที่ 3.4 ตารางรับข้อมูลของ Node Information

โดยกำหนดค่าโคออร์ดิเนต (X, Y) ของ Node ในช่อง X และY จากนั้นกำหนดค่า Displacement (Ux, Uy, Uz) ของ Node ในช่อง Ux, Uy และUz โดยกำหนดมีค่าเป็นศูนย์เมื่อตำแหน่งนั้นมีจุดรองรับตามแนวแกน ถัดไปกำหนดค่าแรงภายนอก (Force) ที่กระทำ ณ ตำแหน่ง Node ใดๆ ในช่อง Fx, Fy และ Mz โดยมีเงื่อนไขคือ

Fx เป็นบวก เมื่อแรงกระทำในแนวระนาบมีทิศไปทางด้านขวา

Fx เป็นลบ เมื่อแรงกระทำในแนวระนาบมีทิศไปทางด้านซ้าย

Fy เป็นบวก เมื่อแรงกระทำในแนวดิ่งมีทิศขึ้น

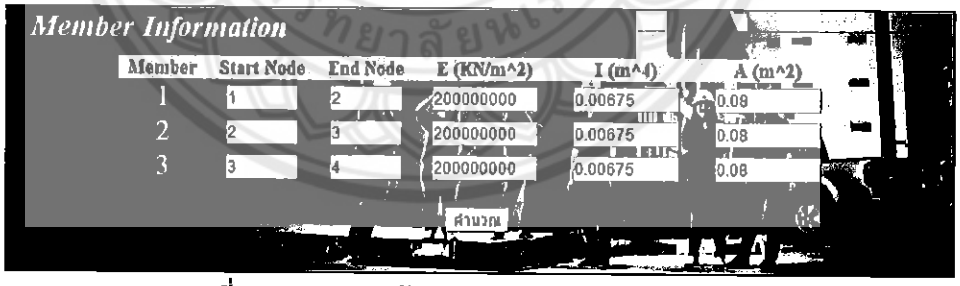
Fy เป็นลบ เมื่อแรงกระทำในแนวดิ่งมีทิศลง

Mz เป็นบวก เมื่อแรงหมุนกระทำมีทิศหมุนทวนเข็มนาฬิกา

Mz เป็นลบ เมื่อแรงหมุนกระทำมีทิศหมุนตามเข็มนาฬิกา

3.2.1.3 กำหนด Member Information โดย

- Start Node คือการกำหนดจุดเริ่มต้นของ Member นั้นๆว่ามีจุดเริ่มต้นที่ Node ใด
- End Node คือการกำหนดจุดสิ้นสุดของ Member นั้นๆว่ามีจุดสิ้นสุดที่ Node ใด
- E คือค่าพลังงานศักย์ของความยืดหยุ่น (Modulus of elasticity)
- I คือคาโมเมนตความเฉื่อย (Moment of inertia)
- A คือค่าพื้นที่หน้าตัด(Area) ของวัตถุนั้นๆ



รูปที่ 3.5 ตารางรับข้อมูลของ Member Information

3.3.2 การแสดงผล (Out put)

ในการแสดงผล (Out put) ของโปรแกรมจะแสดงผลดังนี้

3.3.2.1 รูปภาพโครงสร้างเดิม

3.3.2.2 ตาราง Node Information

3.3.2.3 ตาราง Member Information

3.3.2.4 ตาราง Displacement

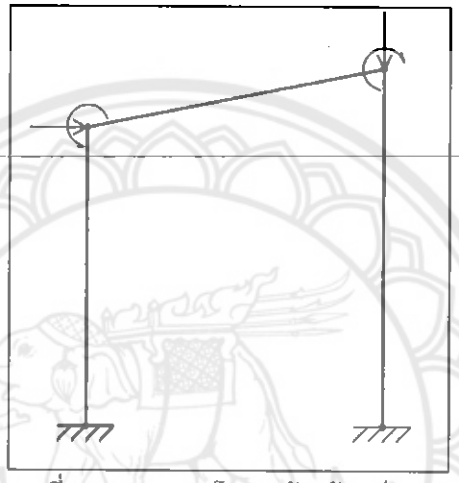
3.3.2.5 ตาราง Member Force

3.3.2.6 รูป Displacement

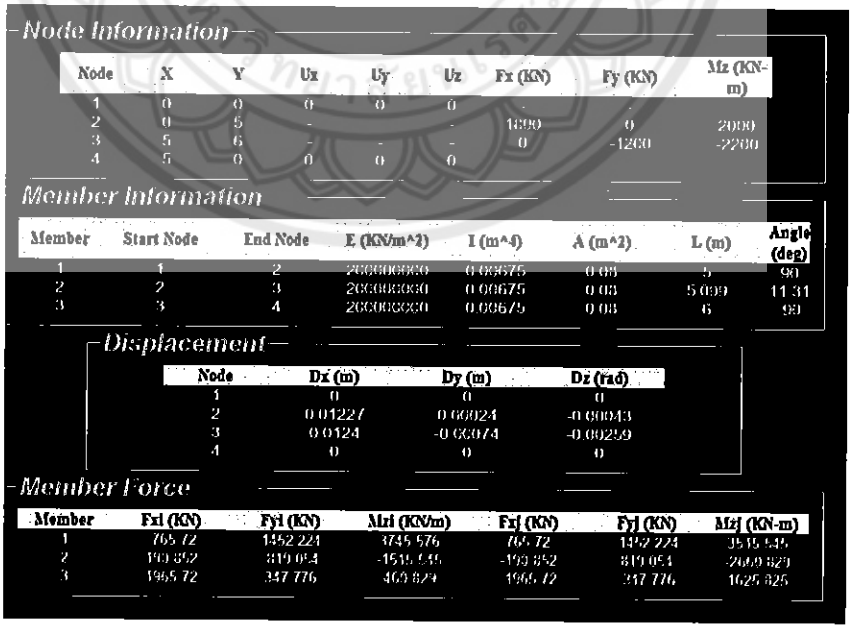
3.3.2.7 รูป Shear force diagram

3.3.2.8 รูป Axial load diagram

3.3.2.9 รูป Bending Moment diagram



รูปที่ 3.6 แสดงรูปโครงสร้างข้อแข็งเดิม



Node Information

Node	X	Y	Ux	Uy	Uz	Fx (KN)	Fy (KN)	Mz (KN-m)
1	0	0	0	0	0	-	-	
2	0	5	-	-	-	1000	0	2000
3	5	6	-	-	-	0	-1200	-2200
4	5	0	0	0	0			

Member Information

Member	Start Node	End Node	E (KN/m^2)	I (m^4)	A (m^2)	L (m)	Angle (deg)
1	1	2	2000000000	0.00675	0.03	5	90
2	2	3	2000000000	0.00675	0.03	5.099	11.31
3	3	4	2000000000	0.00675	0.03	6	90

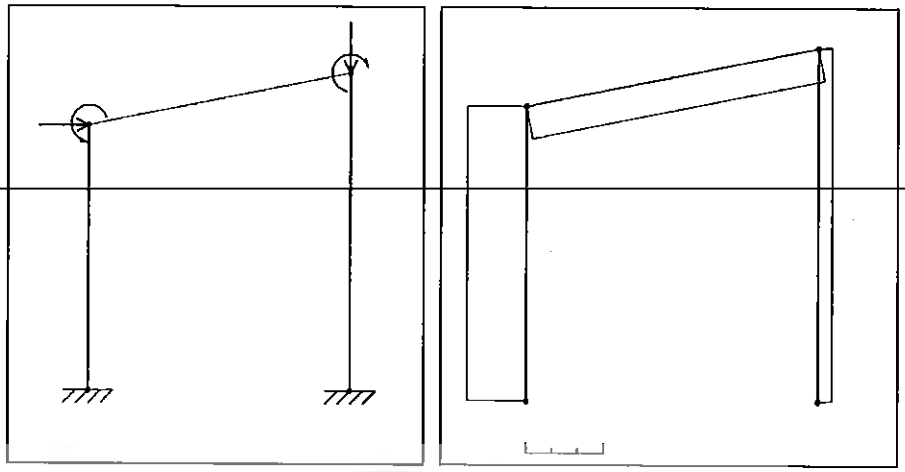
Displacement

Node	Dx (m)	Dy (m)	Dz (rad)
1	0	0	0
2	0.01227	0.00024	-0.00043
3	0.0124	-0.00074	-0.00259
4	0	0	0

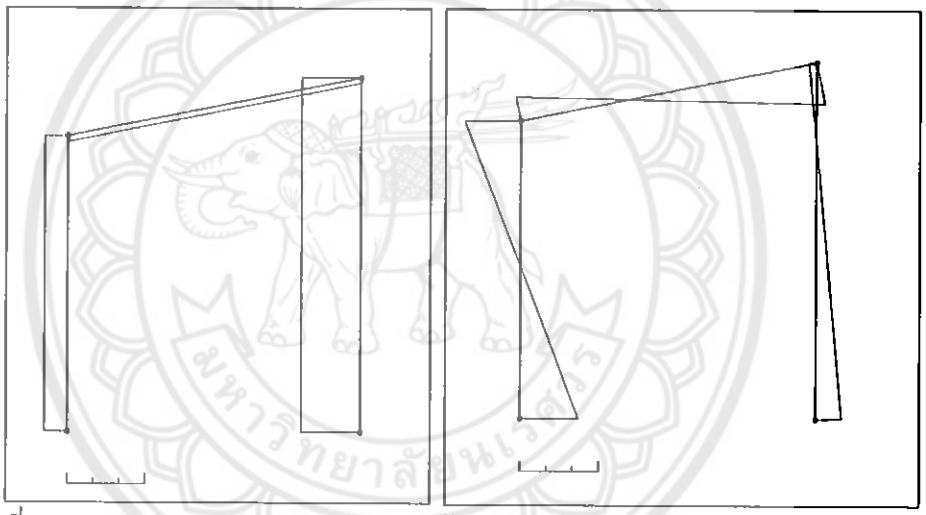
Member Force

Member	Fxi (KN)	Fyi (KN)	Mzi (KN/m)	Fxj (KN)	Fyj (KN)	Mzj (KN-m)
1	765.72	1452.224	3745.576	765.72	1452.224	3515.525
2	199.852	819.054	-1515.949	-199.852	819.054	-2669.829
3	1965.72	347.776	469.829	1965.72	347.776	1625.825

รูปที่ 3.7 แสดงรูปตาราง Node Information, Member Information, Displacement, Member Force



รูปที่ 3.8 แสดง Displacement(ซ้าย) และShear Force Diagram(ขวา) (scale 1 ช่อง เท่ากับ 50 KN)

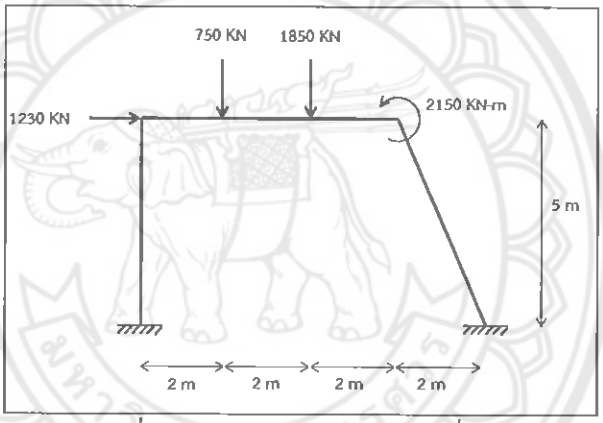


รูปที่ 3.9 รูปแสดง Axial Load Diagram(ซ้าย) และBending Moment Diagram(ขวา)
(scale 1 ช่อง เท่ากับ 50 KN)

บทที่ 4
ผลการทดลองและวิเคราะห์

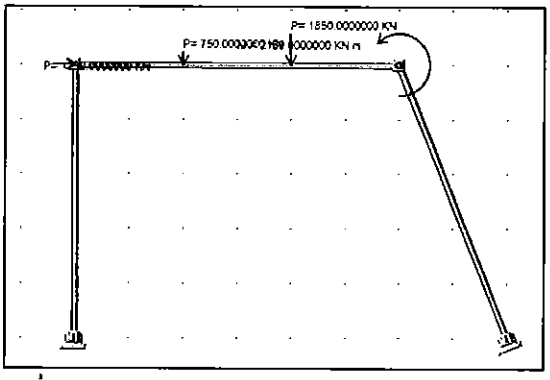
การทดลองใช้งานหน้าเว็บวิเคราะห์โครงสร้างข้อแข็งในแนวระนาบโดยนำมาวิเคราะห์เปรียบเทียบความถูกต้องกับโปรแกรม SUTStructor

ตัวอย่างที่ 4.1 โครงข้อแข็ง 2 มิติ มีแรงกระทำ ณ Node ต่างๆดังรูปที่ 4.1 กำหนดค่า E = 200000000 KN/m^2 , I = 6.75 x 10^{-4} m^4 และ A = 0.08 m^2 โดยให้แสดงการคำนวณในโปรแกรม SUTStructor และโปรแกรมวิเคราะห์โครงข้องแข็ง 2 มิติ

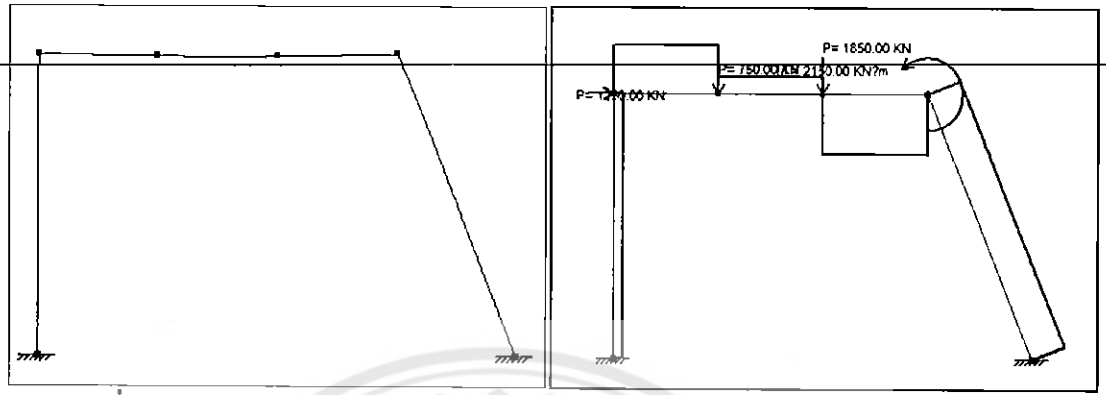


รูปที่ 4.1 แสดงรูปโจทย์ตัวอย่างที่ 4.1

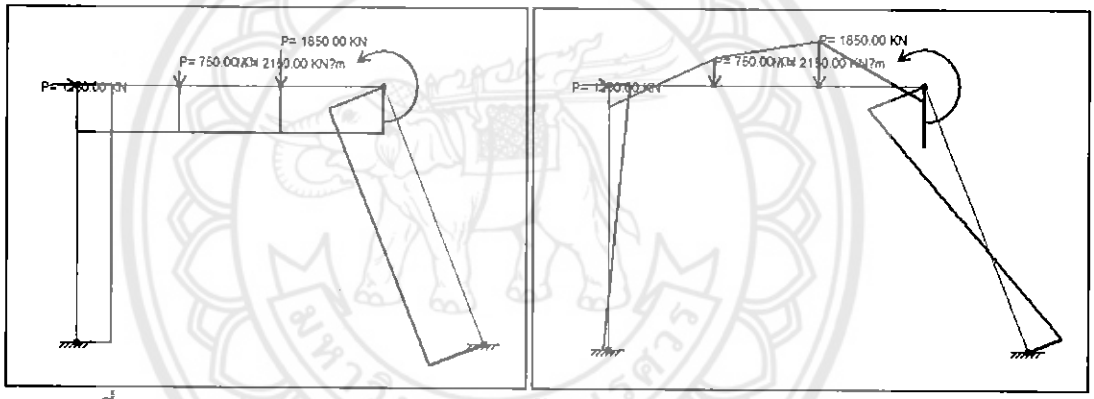
การคำนวณด้วยโปรแกรม SUTStructor จากโจทย์จะได้ รูปโครงสร้าง, แรงภายใน, Shear diagram, Axial load diagram, Bending moment diagram แสดงดังรูป 4.2 ถึง 4.4



รูปที่ 4.2 รูปโครงสร้างจากโปรแกรม SUTStructor

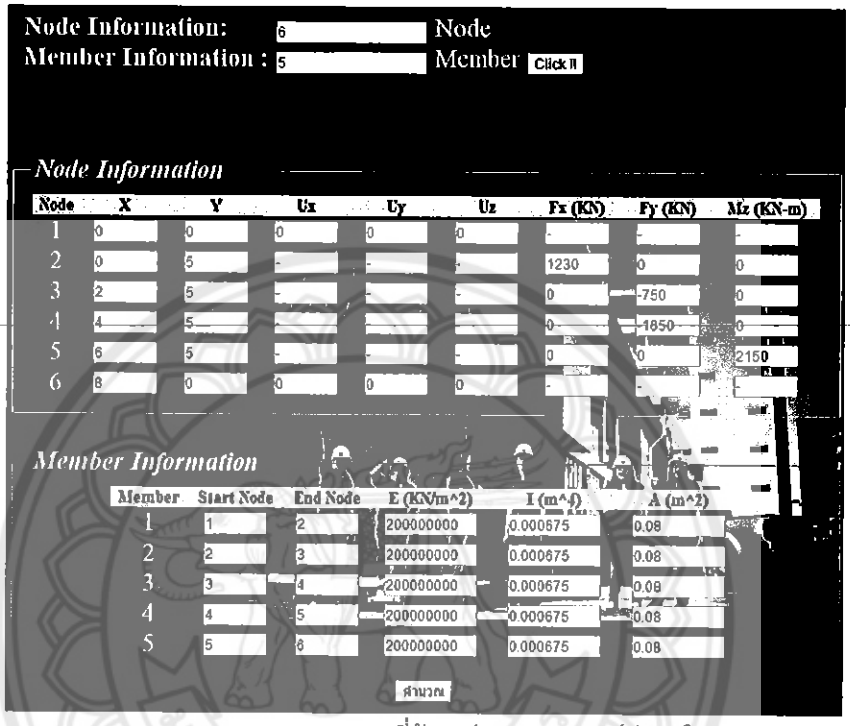


รูปที่ 4.3 แสดง Displacement ของโครงสร้าง(ซ้าย) และShear Force Diagram(ขวา)

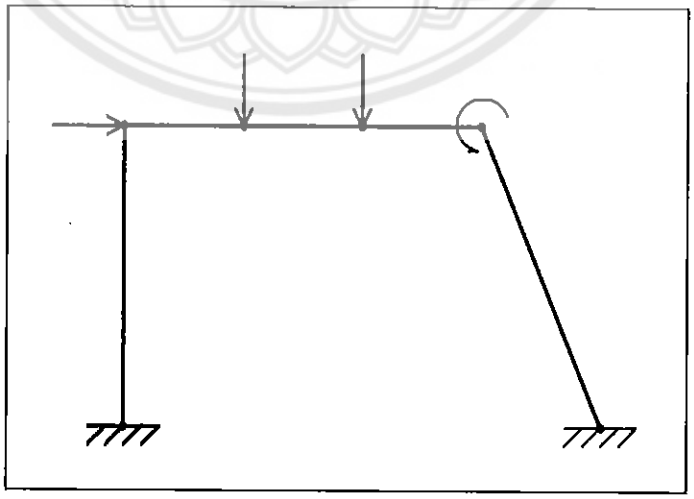


รูปที่ 4.4 รูปแสดง Axial Load Diagram(ซ้าย) และBending Moment Diagram(ขวา)

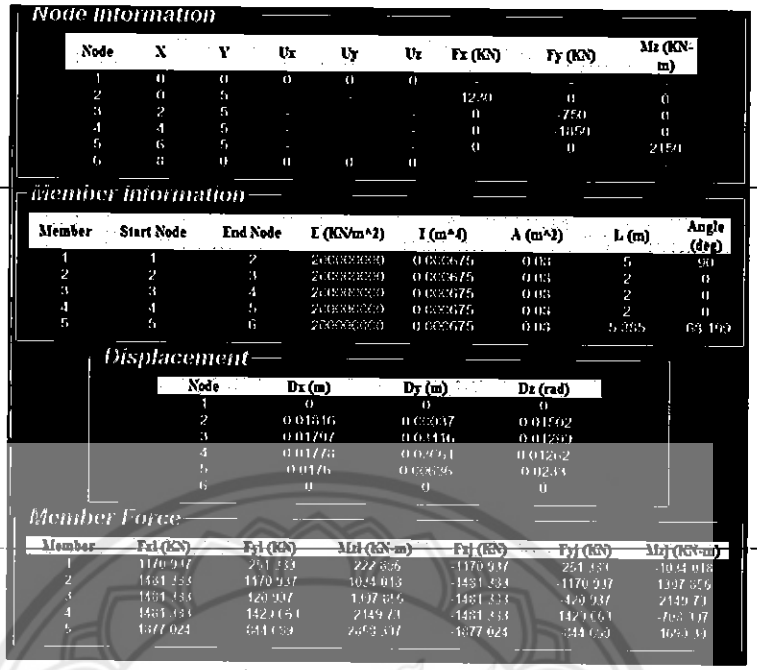
การคำนวณด้วยโปรแกรมวิเคราะห์โครงข้อแข็ง 2 มิติ จากโจทย์จะได้ รูปโครงสร้าง, แรงภายใน, Shear diagram, Axial load diagram, Bending moment diagram ดังรูปที่ 4.6 ถึง 4.9 โดยต้องใส่ค่าพารามิเตอร์ตามรูปที่ 4.5



รูปที่ 4.5 แสดงตาราง Node และ Member ที่ป้อนค่าพารามิเตอร์ต่างๆในการคำนวณ จากโจทย์ตัวอย่างที่ 4.1



รูปที่ 4.6 แสดงรูปโครงสร้างเดิมจากโปรแกรมวิเคราะห์โครงข้อแข็ง 2 มิติ



Node Information

Node	X	Y	Ux	Uy	Uz	Fx (KN)	Fy (KN)	Mz (KN-m)
1	0	0	0	0	0	-	-	-
2	0	5	-	-	-	12.49	0	0
3	2	5	-	-	-	0	-750	0
4	4	5	-	-	-	0	-1859	0
5	6	5	-	-	-	0	0	2150
6	8	0	0	0	0	-	-	-

Member Information

Member	Start Node	End Node	E (KN/m^2)	I (m^4)	A (m^2)	L (m)	Angle (deg)
1	1	2	200000000	0.0026675	0.05	5	90
2	2	3	200000000	0.0026675	0.05	2	0
3	3	4	200000000	0.0026675	0.05	2	0
4	4	5	200000000	0.0026675	0.05	2	0
5	5	6	200000000	0.0026675	0.05	5.385	63.169

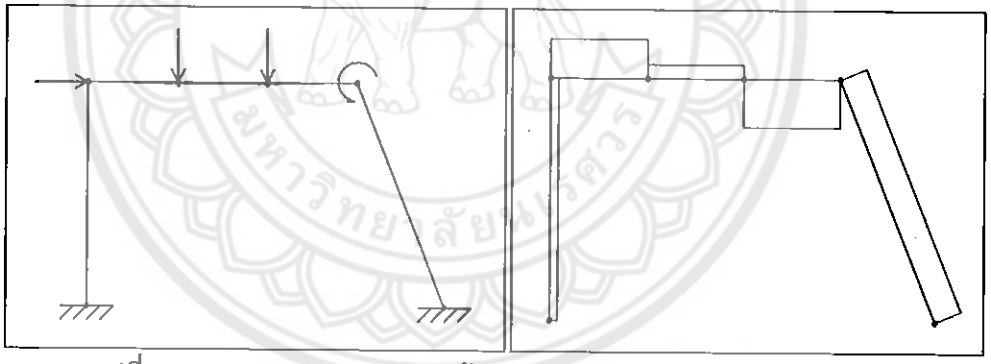
Displacement

Node	Dx (m)	Dy (m)	Dz (rad)
1	0	0	0
2	0.01616	0.03037	0.01562
3	0.01747	0.04136	0.01250
4	0.01776	0.04761	0.01202
5	0.01776	0.05096	0.01243
6	0	0	0

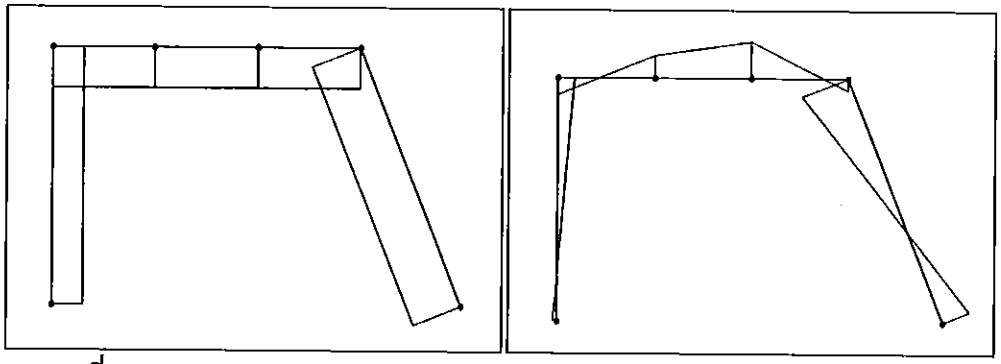
Member Force

Member	Fxi (KN)	Fyi (KN)	Mzi (KN-m)	Fxj (KN)	Fyj (KN)	Mzj (KN-m)
1	1170.937	251.533	222.886	-1170.937	251.533	-10.54.018
2	1481.533	1170.937	10.54.013	-1481.533	-1170.937	1397.856
3	1481.533	420.937	1397.856	-1481.533	-420.937	2149.73
4	1481.533	1421.061	2149.73	-1481.533	1421.061	-702.107
5	1877.024	844.059	2653.307	-1877.024	-844.059	1053.39

รูปที่ 4.7 แสดงผลการคำนวณเป็นตาราง Node Information, Member Information, Displacement และ Member Force

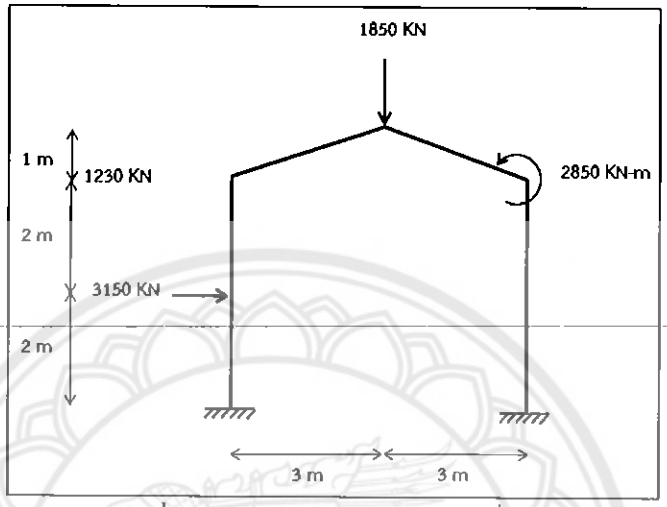


รูปที่ 4.8 แสดง Displacement(ซ้าย), และShear Force Diagram(ขวา)



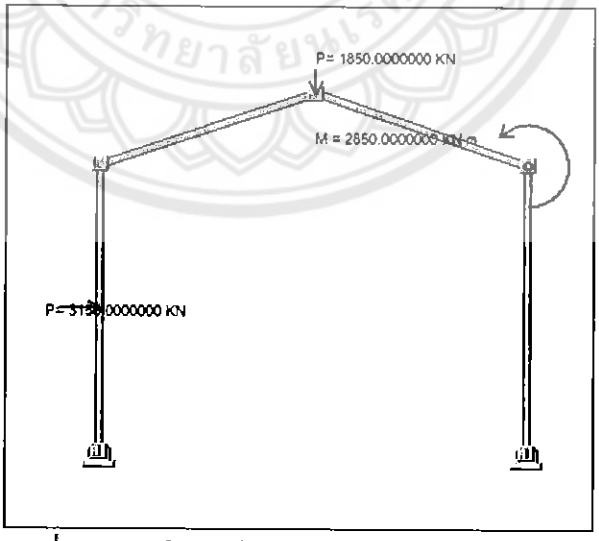
รูปที่ 4.9 แสดง Axial Load Diagram(ซ้าย), Bending Moment Diagram(ขวา)

ตัวอย่างที่ 4.2 โครงข้อแข็ง 2 มิติ มีแรงกระทำ ณ Node ต่างๆดังรูปที่ 4.12 กำหนดค่า E = 200000000 KN/m^2 , I = 6.75 x 10^{-4} m^4 และ A = 0.08 m^2 โดยให้แสดงการคำนวณในโปรแกรม SUTStructor และโปรแกรมวิเคราะห์โครงข้องแข็ง 2 มิติ

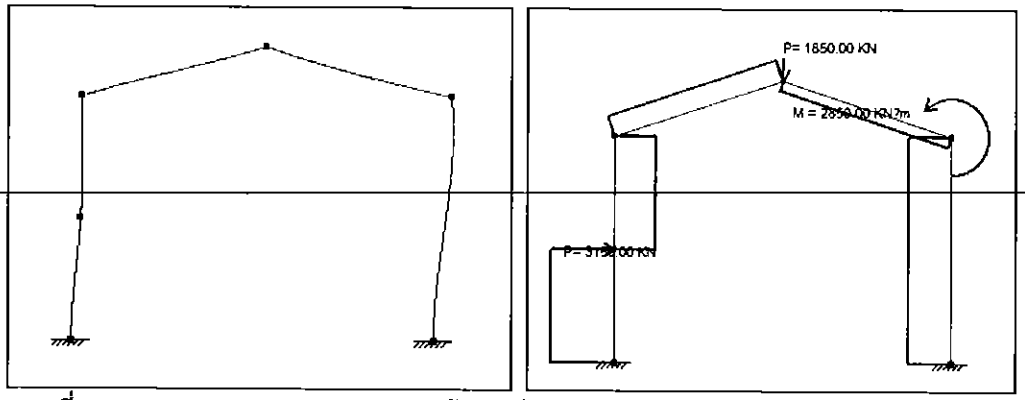


รูปที่ 4.10 แสดงรูปโจทย์ตัวอย่างที่ 4.2

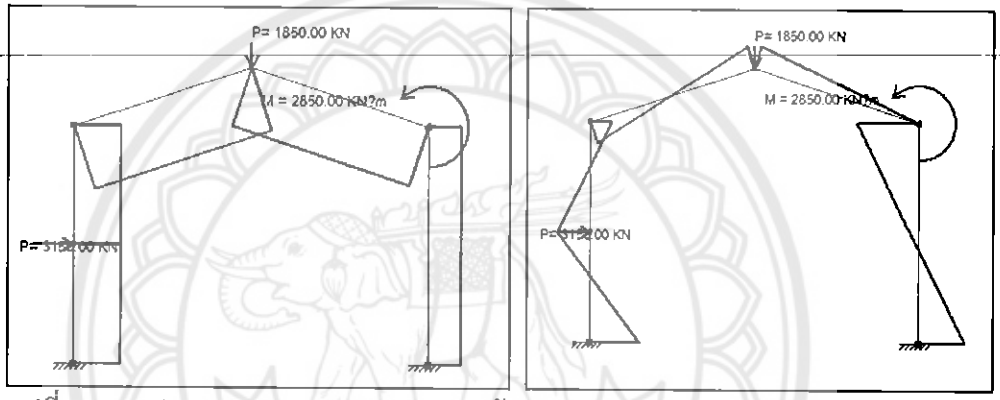
การคำนวณด้วยโปรแกรม SUTStructor จากโจทย์จะได้ รูปโครงสร้าง, แรงภายใน, Shear diagram, Axial load diagram, Bending moment diagram แสดงดังรูป 4.14 ถึง 4.19



รูปที่ 4.11 รูปโครงสร้างจากโปรแกรม SUTStructor

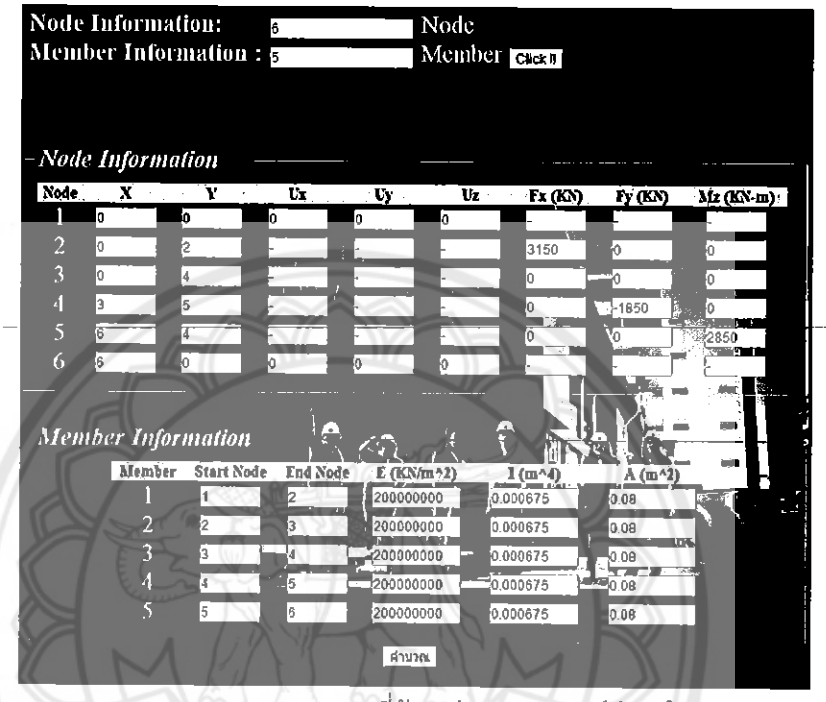


รูปที่ 4.12 แสดง Displacement ของโครงสร้าง(ซ้าย) และShear Force Diagram(ขวา)

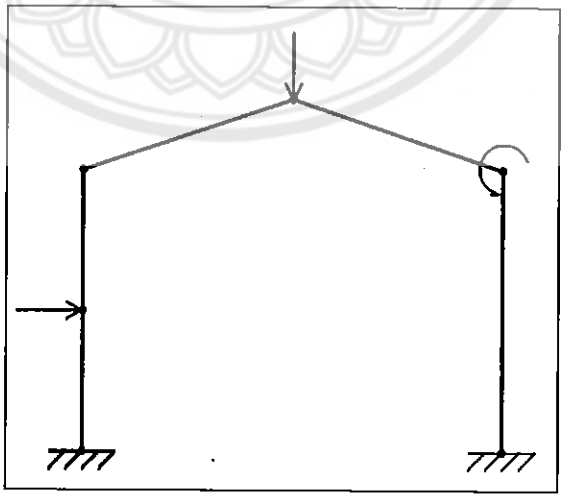


รูปที่ 4.13 รูปแสดง Axial Load Diagram(ซ้าย) และBending Moment Diagram(ขวา)

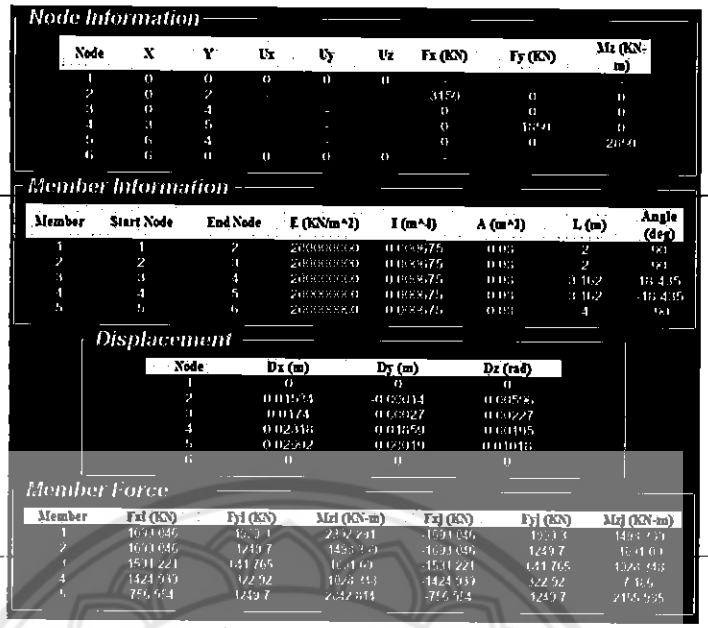
การคำนวณด้วยโปรแกรมวิเคราะห์โครงข้อแข็ง 2 มิติ จากโจทย์จะได้ รูปโครงสร้าง, แรงภายใน, Shear diagram, Axial load diagram, Bending moment diagram ดังรูปที่ 4.21 ถึง 4.24 โดยต้องใส่ค่าพารามิเตอร์ตามรูปที่ 4.20



รูปที่ 4.14 แสดงตาราง Node และ Member ที่ป้อนค่าพารามิเตอร์ต่างๆในการคำนวณ จากโจทย์ตัวอย่างที่ 4.2



รูปที่ 4.15 แสดงรูปโครงสร้างเดิมจากโปรแกรมวิเคราะห์โครงข้อแข็ง 2 มิติ



Node Information

Node	X	Y	Ux	Uy	Uz	Fx (KN)	Fy (KN)	Mz (KN-m)
1	0	0	0	0	0	-	-	-
2	0	2	-	-	-	-3150	0	0
3	0	4	-	-	-	0	0	0
4	3	5	-	-	-	0	1575	0
5	6	4	-	-	-	0	0	2835
6	6	0	0	0	0	-	-	-

Member Information

Member	Start Node	End Node	E (KN/m^2)	I (m^4)	A (m^2)	L (m)	Angle (deg)
1	1	2	2[illegible]	0.0[illegible]	0.03	2	90
2	2	3	2[illegible]	0.0[illegible]	0.03	2	90
3	3	4	2[illegible]	0.0[illegible]	0.03	3.162	18.435
4	4	5	2[illegible]	0.0[illegible]	0.03	3.162	-18.435
5	5	6	2[illegible]	0.0[illegible]	0.03	4	90

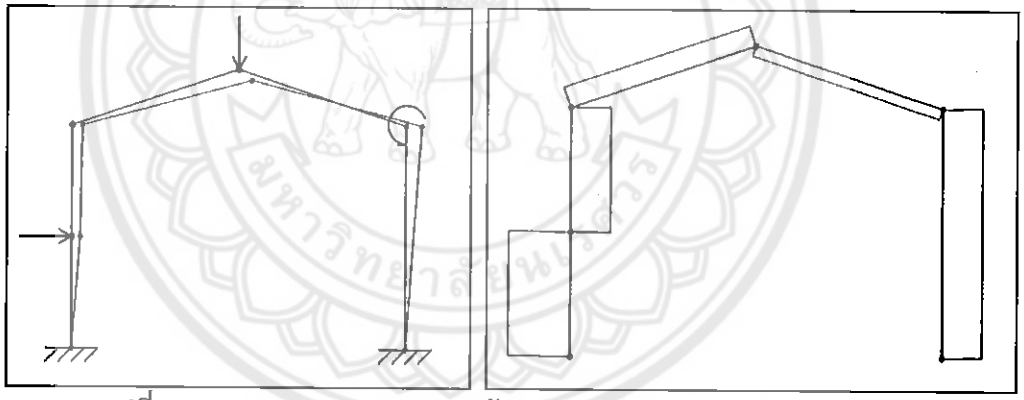
Displacement

Node	Dx (m)	Dy (m)	Dz (rad)
1	0	0	0
2	0.01574	-0.03014	0.03[illegible]
3	0.01174	0.03027	0.03027
4	0.02318	0.01859	0.01095
5	0.02642	0.02019	0.01016
6	0	0	0

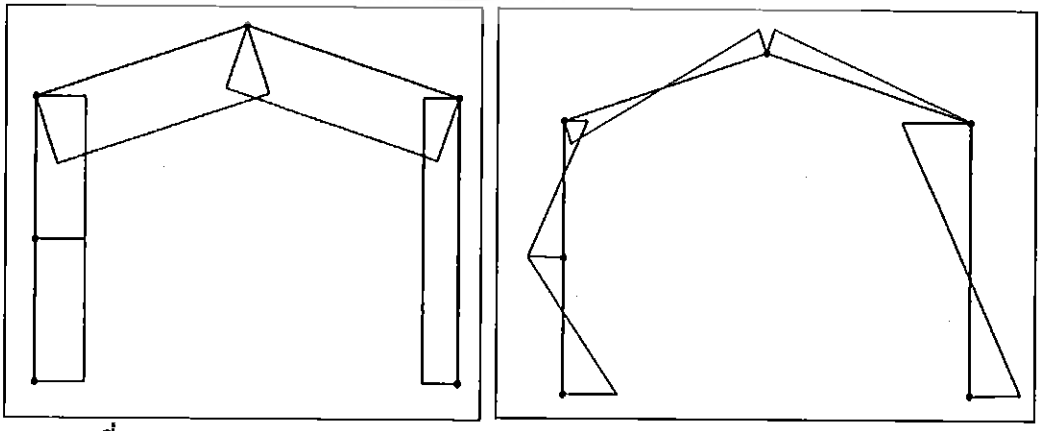
Member Force

Member	Fxi (KN)	Fyi (KN)	Mzi (KN-m)	Fxj (KN)	Fyj (KN)	Mzj (KN-m)
1	1693.036	1[illegible]	2[illegible]	-1693.036	1[illegible]	1[illegible]
2	1693.036	1249.7	1[illegible]	-1693.036	1249.7	1[illegible]
3	1531.221	141.765	1[illegible]	-1531.221	141.765	1[illegible]
4	1423.939	122.92	1[illegible]	-1423.939	122.92	2[illegible]
5	795.764	1249.7	2[illegible]	-795.764	1249.7	2[illegible]

รูปที่ 4.16 แสดงผลการคำนวณเป็นตาราง Node Information, Member Information, Displacement และ Member Force



รูปที่ 4.17 แสดง Displacement(ซ้าย), และShear Force Diagram(ขวา)



รูปที่ 4.18 แสดง Axial Load Diagram(ซ้าย), Bending Moment Diagram(ขวา)

จากตัวอย่างที่ 4.1 และ 4.2 เมื่อนำค่า การเคลื่อนที่ (Displacement) และแรงภายในแต่ละชิ้นส่วน (Member force) ที่ได้จากการการคำนวณของทั้งสองโปรแกรมมาหาผลต่างโดยนำค่าที่ได้จากโปรแกรม SUTStructor ลบด้วยค่าจากโปรแกรมวิเคราะห์โครงข้อแข็ง 2 มิติ จะเห็นได้ว่าไม่มีข้อแตกต่างแต่อย่างใด ซึ่งแสดงว่าโปรแกรมวิเคราะห์โครงข้อแข็ง 2 มิติ มีความแม่นจำแลถูกต้องเป็นอย่างดี ดังตารางที่ 4.1 ถึง 4.4

ตารางที่ 4.1 แสดงการเปรียบเทียบค่า Member force ระหว่างโปรแกรม SUTStructor กับโปรแกรมวิเคราะห์โครงข้อแข็ง 2 มิติ ๖ (ตัวอย่างที่ 4.1)

Member	โปรแกรมวิเคราะห์โครงข้อแข็ง 2 มิติ					
	Fxi (KN)	Fyi (KN)	Mzi (KN-m)	Fxj (KN)	Fyj (KN)	Mzj (KN-m)
1	1170.937	-251.383	-222.896	-1170.937	251.383	-1034.018
2	1481.383	1170.937	1034.018	-1481.383	-1429.063	-2858.397
3	1877.024	844.689	2858.397	-1877.024	-844.689	1690.39
Member	โปรแกรม SUTStructor					
	Fxi (KN)	Fyi (KN)	Mzi (KN-m)	Fxj (KN)	Fyj (KN)	Mzj (KN-m)
1	1170.937	-251.383	-222.896	-1170.937	251.383	-1034.018
2	1481.383	1170.937	1034.018	-1481.383	-1429.063	-2858.397
3	1877.024	844.689	2858.397	-1877.024	-844.689	1690.39
Member	[illegible]					
	Fxi (KN)	Fyi (KN)	Mzi (KN-m)	Fxj (KN)	Fyj (KN)	Mzj (KN-m)
1	0.000	0.000	0.000	0.000	0.000	0.000
2	0.000	0.000	0.000	0.000	0.000	0.000
3	0.000	0.000	0.000	0.000	0.000	0.000

ตารางที่ 4.2 แสดงการเปรียบเทียบค่า Displacement ระหว่างโปรแกรม SUTStructor กับโปรแกรมวิเคราะห์โครงข้อแข็ง 2 มิติ (ตัวอย่างที่ 4.1)

SUTStructor				โปรแกรมวิเคราะห์โครงสร้าง			ผลต่าง ระหว่าง 2 โปรแกรม		
Node	Dx(m)	Dy(m)	Dz(rad)	Dx(m)	Dy(m)	Dz(rad)	Dx(m)	Dy(m)	Dz(rad)
1	0	0	0	0	0	0	0	0	0
2	0.01816	-0.00037	-0.01502	0.01816	-.00037	-.01502	0	0	0
3	0.0176	0.00636	0.0233	0.0176	0.00636	0.0233	0	0	0
4	0	0	0	0	0	0	0	0	0

ตารางที่ 4.3 แสดงการเปรียบเทียบค่า Member force ระหว่างโปรแกรม SUTStructor กับโปรแกรมวิเคราะห์โครงข้อแข็ง 2 มิติ (ตัวอย่างที่ 4.2)

Member	โปรแกรมวิเคราะห์โครงข้อแข็ง 2 มิติ					
	Fxi (KN)	Fyi (KN)	Mzi (KN-m)	Fxj (KN)	Fyj (KN)	Mzj (KN-m)
1	1093.046	1900.300	2302.291	-1093.046	1249.700	-1001.09
2	1531.221	641.765	1001.09	-1531.221	-641.765	1028.348
3	1424.939	-322.920	-1028.348	-1424.939	322.920	7.186
4	756.654	1249.700	2155.985	-756.654	-1249.700	2842.814
Member	โปรแกรม SUTStructor					
	Fxi (KN)	Fyi (KN)	Mzi (KN-m)	Fxj (KN)	Fyj (KN)	Mzj (KN-m)
1	1093.046	1900.300	2302.291	-1093.046	1249.700	-1001.090
2	1531.221	641.765	1001.090	-1531.221	-641.765	1028.348
3	1424.939	-322.920	-1028.348	-1424.939	322.920	7.186
4	756.654	1249.700	2155.985	-756.654	-1249.700	2842.814
Member	ผลต่างระหว่างสองโปรแกรม					
	Fxi (KN)	Fyi (KN)	Mzi (KN-m)	Fxj (KN)	Fyj (KN)	Mzj (KN-m)
1	0.000	0.000	0.000	0.000	0.000	0.000
2	0.000	0.000	0.000	0.000	0.000	0.000
3	0.000	0.000	0.000	0.000	0.000	0.000
4	0.000	0.000	0.000	0.000	0.000	0.000

ตารางที่ 4.4 แสดงการเปรียบเทียบค่า Displacement ระหว่างโปรแกรม SUTStructor กับโปรแกรมวิเคราะห์โครงข้อแข็ง 2 มิติ (ตัวอย่างที่ 4.2)

	Dx(m)	Dy(m)	Dz(rad)	Dx(m)	Dy(m)	Dz(rad)	Dx(m)	Dy(m)	Dz(rad)
	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	0.01741	-0.00029	-0.00228	0.01741	-0.00029	-0.00228	0.00000	0.00000	0.00000
	0.02318	-0.01862	-0.00195	0.02318	-0.01862	-0.00195	0.00000	0.00000	0.00000
	0.02900	-0.00020	0.01018	0.02900	-0.00020	0.01018	0.00000	0.00000	0.00000
	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000

บทที่ 5
สรุปและข้อเสนอแนะ

5.1 สรุปผล

5.1.1 สามารถใช้บริการได้ฟรีแก่บุคคลทั่วไปแบบออนไลน์ ผ่านโดยใช้งานผ่าน Web browser ทั่วไป

5.1.2 สามารถใช้งานได้ในระบบปฏิบัติการทุกชนิด เช่น เช่น IOS, Windows, Android, Linux ฯลฯ

5.1.3 สามารถใช้งานได้ง่าย มีความถูกต้องตามทฤษฎีการวิเคราะห์โครงสร้างอย่างเป็นที่น่าพอใจ

5.2 ข้อเสนอแนะ

5.2.1 หากมีการพัฒนาโปรแกรมในลักษณะนี้มากยิ่งขึ้นจะช่วยให้วิศวกรสามารถทำงานได้อย่างสะดวก ลดการใช้งานโปรแกรมจากต่างประเทศที่มีราคาค่อนข้างสูง ซึ่งก่อให้เกิดประโยชน์แก่วงการวิศวกรรมโยธาได้พอสมควร

5.2.2 ทั้งนี้ควรมีการพัฒนาต่อไปเพื่อให้มีความสมบูรณ์ในการใช้งานมากยิ่งขึ้น รวมทั้งควรมีการพัฒนาโปรแกรมวิเคราะห์ชิ้นส่วนต่างๆในรูปแบบออนไลน์มากยิ่งขึ้นไป

บรรณานุกรม

[1] กิติศักดิ์ เจริญโภดานนท์, คู่มือเขียนเว็บดีคอมเมิร์ซด้วย PHP, พิมพ์ที่ ซัคเซลมีเดีย, กรุงเพทฯ, 2552

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

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

[4] บัญชา ปะสีละเตสัง, พัฒนาเว็บแอปพลิเคชั่นด้วย PHP ร่วมกับ MySQL และDreamweaver, พิมพ์ที่ บริษัท ซีเอ็ดยูเคชั่น จำกัด (มหาชน), กรุงเทพฯ, 2553

ภาพผนวก

ชื่อไฟล์	ความหมาย
Index.php	ส่วนของการรับข้อมูลต่างๆที่ใช้ในการคำนวณ
step2-table.php	ส่วนการคำนวณทั้งหมด และส่วนของการแสดงผลตาราง
step3.php	ส่วนของการวาดรูปโครงสร้างเดิม
DrawingDisplacement2.php	ส่วนของการวาดรูป Displacement
DrawingShear.php	ส่วนของการวาดรูป Shear diagram
DrawingAxail.php	ส่วนของการวาดรูป Axail load diagram
DrawingMoment.php	ส่วนของการวาดรูป Bending moment diagram

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=utf-8" />
<title>Untitled Document</title><style type="text/css">
#W{ width:1000px; margin:10px; padding:20px ; }
#H{ height:140px; background-color:#CCCC99;}
#M{ margin:0px; padding:0px;}
#C{ width:200px; height:600px; background-image:url(5.jpg); float:left;}
#S{ width:800px; height:600px; background-image:url(2.jpg); float:right}
#F{ height:100px; background-image:url(2.jpg); }
.style1 {color: #FF0000;font-size: 24px;}
.style2 {font-size: 24px}
.style3 {font-size: 24px; color: #FFFFFF; }
.style6 {font-size: 24px; color:#000000; }
.style11 {color: #000000; font-size: 20px; }
.style12 {color: #FFFF00;       font-style: italic;font-weight: bold;}
.style21 {color: #000000; font-weight: bold; font-size: 16px; }
.style25 {font-size: 24px; color: #FFFF33; }
.style29 {font-size: 24px; color: #FFFF00; }
.style30 {color:#FFFFFF}
</style></head><body bgcolor="#000000"><body><center><div id="W"><div class="style1" id="H"><p><img
src="main.png" width="1000" height="156" /></p></div>
            <div id="M"><div id="C">
<p class="style2"> </p>
<p class="style3"><u><a href="manual.php" class="style30">วิธีการใช้งาน</a> </span></span></u></p>
```

```
<p class="style3"><u><a href="Teacher.php"class="style30">อาจารย์ที่ปรึกษา</a></span></u></p>
<p class="style3"><u><a href="53360088.php"class="style30">ประวัติผู้จัดทำ</a></u></span> </p>
<p class="style2"> </p> </div> <!--c--><div class="style6" id="S">
<form method="post" action="index.php">
<p align="left"><span class="style11"><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</span><font size="3"> <input name="node" type="submit"  value="Click !!"/>
</font></p></form>

<? if($_POST[node]) {?>
        <form action="step2-table.php" method="post"><table width="100%">
        <tr><td> <fieldset style="border-style: ridge; border-color:#CCCCCC"><legend class="style12">Node
Information</legend>
        <table border="0" cellspacing="0" width="100%">
        <tr>
        <td align="center" bgcolor="#CCCCCC"><span class="style21">Node</span></td>
        <td align="center" bgcolor="#CCCCCC"><span class="style21">X</span></td>
        <td align="center" bgcolor="#CCCCCC"><span class="style21">Y</span></td>
        <td align="center" bgcolor="#CCCCCC"><span class="style21">Ux</span></td>
        <td align="center" bgcolor="#CCCCCC"><span class="style21">Uy</span></td>
        <td align="center" bgcolor="#CCCCCC"><span class="style21">Uz</span></td>
        <td align="center" bgcolor="#CCCCCC"><span class="style21">Fx (KN) </span></td>
        <td align="center" bgcolor="#CCCCCC"><span class="style21">Fy (KN)</span></td>
        <td align="center" bgcolor="#CCCCCC"><span class="style21">Mz (KN-m)</span></td>
        </tr>
                <? for($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="Uz[]" 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>
                <td align="center"><input name="Mz[]" type="text" size="5" /></td>
                </tr><? }?></table></fieldset></td></tr></table><br /><table width="100%">
```

```
<tr><td><fieldset style="border-style: ridge; border-color:#999999">
<legend class="style12">Member Information</legend><center>
<table border="0" cellspacing="0" width="80%">
<tr>
<td align="center" bgcolor="#CCCCCC"><span class="style21">Member</span></td>
<td align="center" bgcolor="#CCCCCC"><span class="style21">Start Node</span></td>
<td align="center" bgcolor="#CCCCCC"><span class="style21">End Node</span></td>
<td align="center" bgcolor="#CCCCCC"><span class="style21">E (KN/m^2) </span></td>
<td align="center" bgcolor="#CCCCCC"><span class="style21">I (m^4) </span></td>
<td align="center" bgcolor="#CCCCCC"><span class="style21">A (m^2) </span></td>
</tr>
<? for($m=1;$m<=$_POST[num_member];$m++) {?>
<tr>
<td align="center"><?=$m?></td>
<td align="center"><input name="start_node[]" type="text" size="5" /></td>
<td align="center"><input name="end_node[]" type="text" size="5" /></td>
<td align="center"><input name="E[]" type="text" size="10" /></td>
<td align="center"><input name="I[]" type="text" size="10" /></td>
<td align="center"><input name="A[]" type="text" size="10" /></td>
</tr><? }?></table></center></fieldset></td></tr></table>
<table width="100%"><tr>
<input name="num_node" type="hidden" value="<?=$_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> <p> </p>  <p> </p>
</div>  </div><!--ปิด M--> <div style="clear:both"></div></div><!--ปิด W -->
</center></body>  </html>
```

ชื่อ step2-table.php

```
<!---------- วาดรูปโครงสร้างเดิม ---------- -->
<div class="style6 style30" id="S">
<div align="center">
<p> </p><p><img src="step3.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]?>
        &Uz[<?=$n-1?>]=<?=$_POST[Uz][$n-1]?>
```

```
&Fx[<?=$n-1?>]=<?=$_POST[Fx][$n-1]?>
&Fy[<?=$n-1?>]=<?=$_POST[Fy][$n-1]?>
&Mz[<?=$n-1?>]=<?=$_POST[Mz][$n-1]; }
for($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" /></p>
<p class="style62">โครงสร้างเดิม</p> <p class="style62"> </p></div>
<!---------- Node Information ---------- -->
<table width="100%"><tr valign="top"> <td><fieldset class="style4" style="border-style: ridge; border-color:#FFFF99">
<legend class="style64 style30">Node Information</legend><center>
<table border="0" cellspacing="0" width="90%">
<tr>
<td width="10%" align="center" bgcolor="#CCCCCC"><strong>Node</strong></td>
<td width="10%" align="center" bgcolor="#CCCCCC"><strong>X</strong></td>
<td width="10%" align="center" bgcolor="#CCCCCC"><strong>Y</strong></td>
<td width="10%" align="center" bgcolor="#CCCCCC"><strong>Ux</strong></td>
<td width="10%" align="center" bgcolor="#CCCCCC"><strong>Uy</strong></td>
<td width="10%" align="center" bgcolor="#CCCCCC"><strong>Uz</strong></td>
<td width="10%" align="center" bgcolor="#CCCCCC"><strong>Fx (KN)</strong></td>
<td width="20%" align="center" bgcolor="#CCCCCC"><strong>Fy (KN)</strong></td>
<td width="20%" align="center" bgcolor="#CCCCCC"><strong>Mz (KN-m)</strong></td>
</tr>
<? for($n=1;$n<=$_POST[num_node];$n++) { $counter++;?>
<tr><td align="center" class="style39"><?=$n?></td>
<td align="center"><span class="style39"><?=$_POST[X][$n-1] ?></span>
<td align="center" class="style39"><?=$_POST[Y][$n-1]?></td>
<td align="center" class="style39"><?=$_POST[Ux][$n-1]?></td>
<td align="center" class="style39"><?=$_POST[Uy][$n-1]?></td>
<td align="center" class="style39"><?=$_POST[Uz][$n-1]?></td>
<td align="center" class="style39"><?=$_POST[Fx][$n-1]?></td>
<td align="center" class="style39"><?=$_POST[Fy][$n-1]?></td>
<td align="center" class="style39"><?=$_POST[Mz][$n-1]?></td>
</tr><? }?></table></center> </fieldset></td></tr></table>
<!---------- Table Member Information ---------- -->
<table width="100%">
<td><fieldset class="style4" style="border-style: ridge; border-color:#FFFF99"><legend class="style64">Member
Information</legend><center><table border="0" cellspacing="0" width="100%">
<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>E (KN/m^2)</strong></td>
<td width="14%" align="center" bgcolor="#CCCCCC"><strong>I (m^4)</strong></td>
<td width="14%" align="center" bgcolor="#CCCCCC"><strong>A (m^2)</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>
</tr>
<? for($m=1;$m<=$_POST[num_member];$m++) {?>
<tr>
<td align="center" class="style39"><?=$m?></td>
<td align="center" class="style39"><?=$_POST[start_node][$m-1]?></td>
<td align="center" class="style39"><?=$_POST[end_node][$m-1]?></td>
<td align="center" class="style39"><?=$_POST[E][$m-1]?></td>
<td align="center" class="style39"><?=$_POST[I][$m-1]?></td>
<td align="center" class="style39"><?=$_POST[A][$m-1]?></td>
<td align="center" class="style39"><?
//----------?????????---------- หาความยาวของ Member
$stNode[$m-1]=$_POST[start_node][$m-1];
$x1 =$_POST[X][$stNode[$m-1]-1];
 $y1=$_POST[Y][$stNode[$m-1]-1];
$enNode[$m-1]=$_POST[end_node][$m-1];
$x2=$_POST[X][$enNode[$m-1]-1];
$y2=$_POST[Y][$enNode[$m-1]-1];

$sum1=pow($y2-$y1,2);
$sum2=pow($x2-$x1,2);
$L[$m-1]=sqrt($sum1+$sum2);
$Delta1=$y2-$y1;
$Delta2=$x2-$x1;
if($Delta2==0){$Ang[$m-1]=pi()/2;}else{$Ang[$m-1]=atan($Delta1/$Delta2);}
if($Delta2==0 and $Delta1<0){$Ang[$m-1]=-pi()/2;}
DegX[$m-1] = 180/pi()*$Ang[$m-1];
echo  round($L[$m-1],3); ?>  
<td align="center" class="style39"><? echo round($DegX[$m-1],3)?></td>
</tr><? }?></table></center></fieldset></td>tr></table>
<?
//---------- Structure
$n3=3*$_POST[num_node];
```

```
for($i=1;$i<=n3;$i++) {for($j=1;$j<=n3;$j++){$kStructure[$i][$j]=0;}}
//---------------------------------------------------------------------- local
for($n=1;$n<=$_POST[num_member];$n++) {
$EI[$n-1]=($_POST[E][$n-1]*$_POST[I][$n-1]);
$EA[$n-1]=($_POST[E][$n-1]*$_POST[A][$n-1]);
        $klocal[$n-1][1][1] = $EA[$n-1]/$L[$n-1];
        $klocal[$n-1][1][2] = 0;
        $klocal[$n-1][1][3] = 0;
        $klocal[$n-1][1][4] = -$klocal[$n-1][1][1];
        $klocal[$n-1][1][5] = 0;
        $klocal[$n-1][1][6] = 0;


        $klocal[$n-1][2][1] = $klocal[$n-1][1][2] ;
        $klocal[$n-1][2][2] = 12*$EI[$n-1]/pow($L[$n-1],3);
        $klocal[$n-1][2][3] = 6*$EI[$n-1]/pow($L[$n-1],2);
        $klocal[$n-1][2][4] = 0;
        $klocal[$n-1][2][5] = -$klocal[$n-1][2][2];
        $klocal[$n-1][2][6] = $klocal[$n-1][2][3];

        $klocal[$n-1][3][1] = $klocal[$n-1][1][3];
        $klocal[$n-1][3][2] = $klocal[$n-1][2][3];
        $klocal[$n-1][3][3] = 4*$EI[$n-1]/$L[$n-1];
        $klocal[$n-1][3][4] = 0;
        $klocal[$n-1][3][5] = -$klocal[$n-1][2][3];
        $klocal[$n-1][3][6] = 2*$EI[$n-1]/$L[$n-1];

        $klocal[$n-1][4][1] = $klocal[$n-1][1][4];
        $klocal[$n-1][4][2] = $klocal[$n-1][2][4];
        $klocal[$n-1][4][3] = $klocal[$n-1][3][4];
        $klocal[$n-1][4][4] = $klocal[$n-1][1][1];
        $klocal[$n-1][4][5] = 0;
        $klocal[$n-1][4][6] = 0;

        $klocal[$n-1][5][1] = $klocal[$n-1][1][5];
        $klocal[$n-1][5][2] = $klocal[$n-1][2][5];
        $klocal[$n-1][5][3] = $klocal[$n-1][3][5];
        $klocal[$n-1][5][4] = $klocal[$n-1][4][5] ;
        $klocal[$n-1][5][5] = $klocal[$n-1][2][2];
```

```
$klocal[$n-1][5][6] = -$klocal[$n-1][2][3];
        $klocal[$n-1][6][1] = $klocal[$n-1][1][6];
        $klocal[$n-1][6][2] = $klocal[$n-1][2][6];
        $klocal[$n-1][6][3] = $klocal[$n-1][3][6];
        $klocal[$n-1][6][4] = $klocal[$n-1][4][6];
        $klocal[$n-1][6][5] = $klocal[$n-1][5][6];
        $klocal[$n-1][6][6] = $klocal[$n-1][3][3];
//-------------------------------------------------------
$sinX=sin($Ang[$n-1]); $cosX=cos($Ang[$n-1]);
        $mT[$n-1][1][1]=$cosX;  $mT[$n-1][1][2]=$sinX;  $mT[$n-1][1][3]=0;
        $mT[$n-1][1][4]=0;      $mT[$n-1][1][5]=0;      $mT[$n-1][1][6]=0;
        $mT[$n-1][2][1]=-$sinX; $mT[$n-1][2][2]=$cosX;  $mT[$n-1][2][3]=0;
        $mT[$n-1][2][4]=0;      $mT[$n-1][2][5]=0;      $mT[$n-1][2][6]=0;
        $mT[$n-1][3][1]=0;      $mT[$n-1][3][2]=0;      $mT[$n-1][3][3]=1;
        $mT[$n-1][3][4]=0;      $mT[$n-1][3][5]=0;      $mT[$n-1][3][6]=0;
        $mT[$n-1][4][1]=0;      $mT[$n-1][4][2]=0;      $mT[$n-1][4][3]=0;
        $mT[$n-1][4][4]=$cosX;  $mT[$n-1][4][5]=$sinX;  $mT[$n-1][4][6]=0;
        $mT[$n-1][5][1]=0;      $mT[$n-1][5][2]=0;      $mT[$n-1][5][3]=0;
        $mT[$n-1][5][4]=-$sinX; $mT[$n-1][5][5]=$cosX;  $mT[$n-1][5][6]=0;
        $mT[$n-1][6][1]=0;      $mT[$n-1][6][2]=0;      $mT[$n-1][6][3]=0;
        $mT[$n-1][6][4]=0;      $mT[$n-1][6][5]=0;      $mT[$n-1][6][6]=1;
//-------------------------------------------------------
 for($i=1;$i<=6;$i++) {
        for($j=1;$j<=6;$j++) {$sumK = 0;
                for($k=1;$k<=6;$k++) {$sumK = $sumK + $klocal[$n-1][$i][$k]*$mT[$n-1][$k][$j]; }
                        $kmA[$i][$j] = $sumK; }}
//-------------------------------------------------------
                                for($i=1;$i<=6;$i++) {
                                        for($j=1;$j<=6;$j++) {
                                                $sumK = 0;
                                                        for($k=1;$k<=6;$k++) {
                                                        $sumK = $sumK + $mT[$n-1][$k][$i]*$kmA[$k][$j];
                                                        }
                                                $kGlobal[$n-1][$i][$j] = $sumK;
                                        }
                                }
                                //-------------------------------------------------------
                                $dof[$n-1][1] = 3*$stNode[$n-1]-2;
                                $dof[$n-1][2] = 3*$stNode[$n-1]-1;
                                $dof[$n-1][3] = 3*$stNode[$n-1];
```

```
                $dof[$n-1][4] = 3*$enNode[$n-1]-2;
                $dof[$n-1][5] = 3*$enNode[$n-1]-1;
                $dof[$n-1][6] = 3*$enNode[$n-1];
                //-----------------------------------------------
                 for($i=1;$i<=6;$i++) {
                                for($j=1;$j<=6;$j++) {
$kStructure[$dof[$n-1][$i]][$dof[$n-1][$j]] = $kStructure[$dof[$n-1][$i]][$dof[$n-1][$j]]+ $kGlobal[$n-1][$i][$j];}}
?><? }?><?
//-----------------------------------------------
            $ifree = 0; $ifix = 0;
            for($n=1;$n<=$_POST[num_node];$n++) {
                $PStructure[3*$n-2] =$_POST[Fx][$n-1];
                $PStructure[3*$n-1] =$_POST[Fy][$n-1];
                $PStructure[3*$n] =$_POST[Mz][$n-1];
                $uStructure[3*$n-2] =$_POST[Ux][$n-1];
                $uStructure[3*$n-1] =$_POST[Uy][$n-1];
                $uStructure[3*$n] =$_POST[Uz][$n-1];
if($uStructure[3*$n-2]==="0"){$ifix=$ifix+1;$vfix[$ifix]=3*$n-2;}else{$ifree=$ifree+1;$vfree[$ifree]=3*$n-2;}
if($uStructure[3*$n-1]==="0"){$ifix=$ifix+1;$vfix[$ifix]=3*$n-1;}else{$ifree=$ifree+1;$vfree[$ifree]=3*$n-1;}
if($uStructure[3*$n]==="0"){$ifix=$ifix+1;$vfix[$ifix]=3*$n;}else{$ifree=$ifree+1;$vfree[$ifree]=3*$n;}}
//-----------------------------------------------
            for($i=1;$i<=$ifree;$i++) {
                for($j=1;$j<=$ifree;$j++) {
            $kReducedx[$i][$j] = $kStructure[$vfree[$i]][$vfree[$j]];
            $kReduced[$i][$j] = $kStructure[$vfree[$i]][$vfree[$j]];}
            $PReduced[$i]=$PStructure[$vfree[$i]];}
//-----------------------------------------------
            // Gauss - Forward Elimination
            for($k=1;$k<=$ifree-1;$k++)  {
                for($i=$k+1;$i<=$ifree;$i++) {
                        $c=$kReduced[$i][$k]/$kReduced[$k][$k];
                        for($j=$k;$j<=$ifree;$j++)   {
                        $kReduced[$i][$j]=$kReduced[$i][$j]-($c*$kReduced[$k][$j]); }
                        $PReduced[$i]=$PReduced[$i]-($c*$PReduced[$k]); } }
//-----------------------------------------------
            // Gauss - Back Substitution
            $u[$ifree]=$PReduced[$ifree]/$kReduced[$ifree][$ifree];
            for($k=$ifree-1;$k>=1;$k--) {
                        $sumk=0;
                for($j=$k+1;$j<=$ifree;$j++){
```

```
                                        $sumk=$sumk+($kReduced[$k][$j]*$u[$j]);          }
                                        $u[$k]=(1/$kReduced[$k][$k])*($PReduced[$k]-$sumk);}
                    //-----------------------------------------------------
                    // Displacement vector
                    for($i=1;$i<=$ifree;$i++) {
                                                $uStructure[$vfree[$i]]=$u[$i];
                                                }
                    //-----------------------------------------------------
                    // Force vector
                            for($i=1;$i<=$n3;$i++) {
                                                $sumK = 0;
                                                        for($k=1;$k<=$n3;$k++) {
                                                $sumK = $sumK + $kStructure[$i][$k]*$uStructure[$k];
                                                        }
                                                $CStructure[$i] = $sumK;}
                                        //-----------------------------------------------------
                                        // Member End Forces
                                        //-----------------------------------------------------
        for($n=1;$n<=$_POST[num_member];$n++) {
                                for($i=1;$i<=6;$i++) {
                                        $sumK = 0;
                                                for($k=1;$k<=6;$k++) {
                                        $sumK = $sumK + $kGlobal[$n-1][$i][$k]*$uStructure[$dof[$n-1][$k]];}
                                        $PMemberG[$n-1][$i]= $sumK;}
                                for($i=1;$i<=6;$i++) {
                                        $sumK = 0;
                                for($k=1;$k<=6;$k++) {
                                        $sumK = $sumK + $mT[$n-1][$i][$k]*$PMemberG[$n-1][$k];}
                                        $PMember[$n-1][$i]= $sumK; }}?>


<!----------------------------------------------------------------------<!-- แสดงผล Displacement -->
<table width="80%">
        <center>
                <tr valign="top">
        <td><fieldset class="style4" style="border-style: ridge; border-color:#FFFF99"><legend
class="style64">Displacement</legend>
                <center><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>Dx (m)</strong></td>
```

```
                    <td width="15%" align="center" bgcolor="#CCCCCC"><strong>Dy (m)</strong></td>
                    <td width="15%" align="center" bgcolor="#CCCCCC"><strong>Dz (rad)</strong></td>
                </tr>

            <? for($n=1;$n<=$_POST[num_node];$n++) { $counter++;?>
                <tr>
                    <td align="center" class="style39"><?=$n?></td>
                    <td align="center"><span class="style39"><?=round($uStructure[3*$n-2],5)?></span>
                    <td align="center" class="style39"><?=round($uStructure[3*$n-1],5)?></td>
                    <td align="center" class="style39"><?=round($uStructure[3*$n],5)?></td>
                </tr><? }?></table></center></fieldset></td></tr></center></table>
<!---------------------------------------------------------->
<table width="100%">
        <tr valign="top">
            <td>
                <fieldset class="style4" style="border-style: ridge; border-color:#FFFF99"><legend
class="style64">Member Force</legend>
        <table border="0" cellspacing="0" width="100%">
            <tr>
                <td width="12%" align="center" bgcolor="#CCCCCC"><strong>Member</strong></td>
                <td width="15%" align="center" bgcolor="#CCCCCC"><strong>Fxi (KN)</strong></td>
                <td width="15%" align="center" bgcolor="#CCCCCC"><strong>Fyi (KN)</strong></td>
                <td width="15%" align="center" bgcolor="#CCCCCC"><strong>Mzi (KN/m)</strong></td>
                                                    <td width="15%"
align="center" bgcolor="#CCCCCC"><strong>Fxj (KN)</strong></td>
                <td width="15%" align="center" bgcolor="#CCCCCC"><strong>Fyj (KN)</strong></td>
                <td width="15%" align="center" bgcolor="#CCCCCC"><strong>Mzj (KN-m)</strong></td>
            </tr>
            <? for($n=1;$n<=$_POST[num_member];$n++) {?>
                <? for($i=1;$i<=1;$i++) {?>
            <tr>
            <td align="center" class="style48"><?=$n?>
            </td><td align="center"><span class="style48"><?=round($PMember[$n-1][$i],3)?></span>
            <td align="center" class="style48"><?=round($PMember[$n-1][$i+1],3)?></td>
            <td align="center" class="style48"><?=round($PMember[$n-1][$i+2],3)?></td>
            <td align="center"><span class="style48"><?=round($PMember[$n-1][$i+3],3)?></span>
            <td align="center" class="style48"><?=round($PMember[$n-1][$i+4],3)?></td>
            <td align="center" class="style48"><?=round($PMember[$n-1][$i+5],3)?></td>
             </tr><? } ?> <? } ?> </table></fieldset>td></tr></table>
<!-- ----------------------------------------------------<!-- แสดงผล Local -->
```

```
<!----------DrawingDisplacement---->
<div align="center">
 <p><img src="DrawingDisplacement2.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]?>
                        &Uz[<?=$n-1?>]=<?=$_POST[Uz][$n-1]?>
                        &Fx[<?=$n-1?>]=<?=$_POST[Fx][$n-1]?>
                        &Fy[<?=$n-1?>]=<?=$_POST[Fy][$n-1]?>
                        &Mz[<?=$n-1?>]=<?=$_POST[Mz][$n-1]?>
                        &uX[<?=$n-1?>]=<?=$uStructure[3*$n-2]?>
                        &uY[<?=$n-1?>]=<?=$uStructure[3*$n-1]?>
                        &uZ[<?=$n-1?>]=<?=$uStructure[3*$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]; } ?>
" width="500" height="500" align="baseline" /></p>
 <p><span class="style63">Displacement</span>
 <td align="center"><!----------Shear Diagram----->
 <p><img src="DrawingShear.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]?>
                        &Uz[<?=$n-1?>]=<?=$_POST[Uz][$n-1]?>
                        &Fx[<?=$n-1?>]=<?=$_POST[Fx][$n-1]?>
                        &Fy[<?=$n-1?>]=<?=$_POST[Fy][$n-1]?>
                        &Mz[<?=$n-1?>]=<?=$_POST[Mz][$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]?>
                        &Ang[<?=$m-1?>]=<?=$Ang[$m-1]?>
                &M1[<?=$m-1?>]=<?=$PMember[$m-1][2]?>
                        &M2[<?=$m-1?>]=<?=$PMember[$m-1][5]?>
```

```
                                        <? } ?>
                                        " width="500" height="500" align="baseline" /></p>
                                         <p>
<span class="style63">Shear Force Diagram</span>
<!----------------------------------------------------------------Axial Diagram----->
          <p><img src="DrawingAxiaload.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]?>
                                        &Uz[<?=$n-1?>]=<?=$_POST[Uz][$n-1]?>
                                        &Fx[<?=$n-1?>]=<?=$_POST[Fx][$n-1]?>
                                        &Fy[<?=$n-1?>]=<?=$_POST[Fy][$n-1]?>
                                        &Mz[<?=$n-1?>]=<?=$_POST[Mz][$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]?>
                                        &Ang[<?=$m-1?>]=<?=$Ang[$m-1]?>
                                      &M1[<?=$m-1?>]=<?=$PMember[$m-1][1]?>
                                        &M2[<?=$m-1?>]=<?=$PMember[$m-1][4]?>
                                        <? } ?>
                                        " width="500" height="500" align="baseline" /></p>
                                         <p class="style63">Axial Load Diagram
<!----------------------------------------------------------------Bending Moment Diagram----->
 <p><img src="DrawingMoment.php?
          num_node=<?=$_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][$n-1]?>
                                        &Uz[<?=$n-1?>]=<?=$_POST[Uz][$n-1]?>
                                        &Fx[<?=$n-1?>]=<?=$_POST[Fx][$n-1]?>
                                        &Fy[<?=$n-1?>]=<?=$_POST[Fy][$n-1]?>
                                        &Mz[<?=$n-1?>]=<?=$_POST[Mz][$n-1];}?>
                    <?  for($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]?>
```

```
                                    &Ang[<?=$m-1?>]=<?=$Ang[$m-1]?>
                                    &M1[<?=$m-1?>]=<?=$PMember[$m-1][3]?>
                                    &M2[<?=$m-1?>]=<?=$PMember[$m-1][6]?>
                                    <? } ?>
" width="500" height="500" align="baseline" /></p><p>
<span class="style58"><span class="style63">Bending Moment Diagram </span><br />
</span>
```

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]; }

$dMin=0;
$dMax=0;

for ($b=0;$b<$n;$b++){
        vX[$b+1]=$_GET['X'][$b];
        $vY[$b+1]=$_GET['Y'][$b];
        $Ux[$b+1]=$_GET['Ux'][$b];
        $Uy[$b+1]=$_GET['Uy'][$b];
        $Uz[$b+1]=$_GET['Uz'][$b];
        $Fx[$b+1]=$_GET['Fx'][$b];
        $Fy[$b+1]=$_GET['Fy'][$b];
        $Mz[$b+1]=$_GET['Mz'][$b];
        if($vX[$b+1]>$dMax){$dMax=$vX[$b+1];}
        if($vY[$b+1]>$dMax){$dMax=$vY[$b+1];}
        if($vZ[$b+1]>$dMax){$dMax=$vZ[$b+1];}
        if($vX[$b+1]<$dMin){$dMin=$vX[$b+1];}
        if($vY[$b+1]<$dMin){$dMin=$vY[$b+1];}
        if($vZ[$b+1]<$dMin){$dMin=$vZ[$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, 238, 238, 209);
                                        $red = ImageColorAllocate($pic, 255, 0, 0);
                                        $blue = ImageColorAllocate($pic, 0, 0, 255);
                                        $Green = ImageColorAllocate($pic, 69, 139, 0);
                                        $Pink = ImageColorAllocate($pic, 225, 20, 147);
                                        ImageFill($pic,0,0,$white);
                                        ImageSetThickness($pic, 5);

                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;$b++){
                if($Ux[$b]=="0" && $Uy[$b]=="0"&&$Uz[$b]=="0") //Draw picture Fixed support
                                                                {
                                        $zx[1]=-20;$zy[1]=0;
                                        $zx[2]=20;$zy[2]=0;
                                        $zx[3]=-10;$zy[3]=0;
                                        $zx[4]=-20;$zy[4]=10;
                                        $zx[5]=0;$zy[5]=0;
                                        $zx[6]=-10;$zy[6]=10;
                                        $zx[7]=10;$zy[7]=0;
                                        $zx[8]=0;$zy[8]=10;
                                        $zx[9]=20;$zy[9]=0;
                                        $zx[10]=10;$zy[10]=10;
//-----------------------------------------------
        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,$zx[1],$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);

if($Ux[$b]=="0" && $Uy[$b]=="0"&&$Uz[$b]=="-") //pin{
            $zx[0]=0;$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" && $Uz[$b]=="-") // Roller Free X
{
        $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]=="-" && $Uz[$b]=="-") //Roller Free Y
                                                    {
$zx[0]=0;$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]=="-" &&$Ux[$b]=="-" ){ //No Support
        $x=toScreen($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)       //วาดรูปแรงในแกน Y น้อยกว่า 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, 5);
        ImageLine($pic,$zx[0],$zy[0],$zx[1],$zy[1],$Pink);
```

```
            ImageLine($pic,$zx[0],$zy[0],$zx[2],$zy[2],$Pink);
            ImageLine($pic,$zx[0],$zy[0],$zx[3],$zy[3],$Pink);
            }
if($Fy[$b] > 0)         //วาดรูปแรงในแกน Y มากกว่า 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, 5);
            ImageLine($pic,$zx[0],$zy[0],$zx[1],$zy[1],$Pink);
            ImageLine($pic,$zx[0],$zy[0],$zx[2],$zy[2],$Pink);
            ImageLine($pic,$zx[0],$zy[0],$zx[3],$zy[3],$Pink);
                                                                    }
if($Fx[$b] < 0)         //วาดรูปแรงในแกน X น้อยกว่า 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, 5);
            ImageLine($pic,$zx[0],$zy[0],$zx[1],$zy[1],$Pink);
            ImageLine($pic,$zx[0],$zy[0],$zx[2],$zy[2],$Pink);
            ImageLine($pic,$zx[0],$zy[0],$zx[3],$zy[3],$Pink);
}
if($Fx[$b] > 0)         //วาดรูปแรงในแนวแกน X มากกว่า 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, 5);
```

```
            ImageLine($pic,$zx[0],$zy[0],$zx[1],$zy[1],$Pink);
            ImageLine($pic,$zx[0],$zy[0],$zx[2],$zy[2],$Pink);
            ImageLine($pic,$zx[0],$zy[0],$zx[3],$zy[3],$Pink);
                                                                    }
if($Mz[$b] < 0)   //วาดรูปโมเมนต์ในทิศ ติดลบ ตามเข็มนาฬิกา
                            {
$zx[0]=0;$zy[0]=0;   $zx[1]=0;$zy[1]=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);
                                $zxR=$zx[0]+40*cos(pi()/9);
                                $zyR=$zy[0]-40*sin(pi()/9);
                                $zxL=$zx[0]+45*cos(pi()/4);
                                $zyL=$zy[0]-45*sin(pi()/4);
                                ImageLine($pic,$zxR,$zyR,$zxL,$zyL,$red);
                                ImageFilledArc($pic,$zx[0],$zy[0],80,80,100,340,$red,IMG_ARC_NOFILL);
                            }
                            if($Mz[$b] > 0)   //วาดรูปโมเมนต์ บวก ในทิศทวนเข็มนาฬิกา
                            {
                            $zx[0]=1;$zy[0]=1;
                            $zx[1]=0;$zy[1]=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);
                                $zxR=$zx[0]-40*sin(pi()/18);
                                $zyR=$zy[0]+40*cos(pi()/18);
                                $zxL=$zx[0]-45*sin(pi()/6);
                                $zyL=$zy[0]+45*cos(pi()/6);
                                ImageLine($pic,$zxR,$zyR,$zxL,$zyL,$red);
                                ImageFilledArc($pic,$zx[0],$zy[0],80,80,100,340,$red,IMG_ARC_NOFILL);
                            }
                                                                    }
```

```
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 DrawingDisplacement2.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]; }
$dMin=0;
$dMax=0;

for ($b=0;$b<$n;$b++)
        $vX[$b+1]=$_GET['X'][$b];
        $vY[$b+1]=$_GET['Y'][$b];
        $Ux[$b+1]=$_GET['Ux'][$b];
        $Uy[$b+1]=$_GET['Uy'][$b];
        $Uz[$b+1]=$_GET['Uz'][$b];
        $uX[$b+1]=$_GET['uX'][$b];
        $uY[$b+1]=$_GET['uY'][$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, 238, 238, 209);
        $red = ImageColorAllocate($pic, 255, 0, 0);
        $blue = ImageColorAllocate($pic, 0, 0, 255);
```

```
                $Green = ImageColorAllocate($pic, 69, 139, 0);
                $Pink = ImageColorAllocate($pic, 225, 20, 147);
                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);
        //-------------------------------------
        $xNew=toScreen($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,$blue);
                ImageFilledArc($pic,$xNew,$yNew,10,10,0,360,$red,IMG_ARC_EDGED);
                ImageFilledArc($pic,$x1New,$y1New,10,10,0,360,$red,IMG_ARC_EDGED);
        //-------------------------------------------------
        for ($b=1;$b<=$n;$b++){
                if($Ux[$b]=="0" && $Uy[$b]=="0"&&$Uz[$b]=="0") //Draw picture Fixed support
                                                                {
                        $zx[1]=-20;$zy[1]=0;
                        $zx[2]=20;$zy[2]=0;
                        $zx[3]=-10;$zy[3]=0;
                        $zx[4]=-20;$zy[4]=10;
                        $zx[5]=0;$zy[5]=0;
                        $zx[6]=-10;$zy[6]=10;
                        $zx[7]=10;$zy[7]=0;
                        $zx[8]=0;$zy[8]=10;
                        $zx[9]=20;$zy[9]=0;
                        $zx[10]=10;$zy[10]=10;
//-----------------------------------------
```

```
            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,$zx[1],$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);

if($Ux[$b]=="0" && $Uy[$b]=="0"&&$Uz[$b]=="-") //pin{
            $zx[0]=0;$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" && $Uz[$b]=="-") // Roller Free X
{
        $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]=="-" && $Uz[$b]=="-") //Roller Free Y
                                                        {
$zx[0]=0;$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]=="-" &&$Ux[$b]=="-" ){ //No Support
            $x=toScreen($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)       //วาดรูปแรงในแกน Y น้อยกว่า 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, 5);
```

```
            ImageLine($pic,$zx[0],$zy[0],$zx[1],$zy[1],$Pink);
            ImageLine($pic,$zx[0],$zy[0],$zx[2],$zy[2],$Pink);
            ImageLine($pic,$zx[0],$zy[0],$zx[3],$zy[3],$Pink);
            }
if($Fy[$b] > 0)         //วาดรูปแรงในแกน Y มากกว่า 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, 5);
            ImageLine($pic,$zx[0],$zy[0],$zx[1],$zy[1],$Pink);
            ImageLine($pic,$zx[0],$zy[0],$zx[2],$zy[2],$Pink);
            ImageLine($pic,$zx[0],$zy[0],$zx[3],$zy[3],$Pink);
}
if($Fx[$b] < 0)         //วาดรูปแรงในแกน X น้อยกว่า 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, 5);
            ImageLine($pic,$zx[0],$zy[0],$zx[1],$zy[1],$Pink);
            ImageLine($pic,$zx[0],$zy[0],$zx[2],$zy[2],$Pink);
            ImageLine($pic,$zx[0],$zy[0],$zx[3],$zy[3],$Pink);
}
if($Fx[$b] > 0)         //วาดรูปแรงในแนวแกน X มากกว่า 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, 5);
            ImageLine($pic,$zx[0],$zy[0],$zx[1],$zy[1],$Pink);
```

```
            ImageLine($pic,$zx[0],$zy[0],$zx[2],$zy[2],$Pink);
            ImageLine($pic,$zx[0],$zy[0],$zx[3],$zy[3],$Pink);
                                                                    }
if($Mz[$b] < 0)   //วาดรูปโมเมนต์ในทิศ ติดลบ ตามเข็มนาฬิกา
                        {
$zx[0]=0;$zy[0]=0;   $zx[1]=0;$zy[1]=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);
                            $zxR=$zx[0]+40*cos(pi()/9);
                            $zyR=$zy[0]-40*sin(pi()/9);
                            $zxL=$zx[0]+45*cos(pi()/4);
                            $zyL=$zy[0]-45*sin(pi()/4);
                            ImageLine($pic,$zxR,$zyR,$zxL,$zyL,$red);
                            ImageFilledArc($pic,$zx[0],$zy[0],80,80,100,340,$red,IMG_ARC_NOFILL);
                        }
                        if($Mz[$b] > 0)   //วาดรูปโมเมนต์ บวก ในทิศทวนเข็มนาฬิกา
                        {
                        $zx[0]=1;$zy[0]=1;
                        $zx[1]=0;$zy[1]=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);
                            $zxR=$zx[0]-40*sin(pi()/18);
                            $zyR=$zy[0]+40*cos(pi()/18);
                            $zxL=$zx[0]-45*sin(pi()/6);
                            $zyL=$zy[0]+45*cos(pi()/6);
                            ImageLine($pic,$zxR,$zyR,$zxL,$zyL,$red);
                            ImageFilledArc($pic,$zx[0],$zy[0],80,80,100,340,$red,IMG_ARC_NOFILL);
                        }
                                                                    }
                                                        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 Drawingshear.php

```
<?
$n=$_GET[num_node];
$m=$_GET[num_member];
for ($b=0;$b<$m;$b++){
        $Ang[$b+1]=$_GET['Ang'][$b];
        $M1[$b+1]=$_GET['M1'][$b];
        $M2[$b+1]=$_GET['M2'][$b];
        $vXYStart[$b+1]=$_GET['start_node'][$b];
        $vXYEnd[$b+1]=$_GET['end_node'][$b];

if($Ang[$b+1]==-90){$Ang[$b+1]=$Ang[$b+1]*-1;}}
$dMin=0;
$dMax=0;

for ($b=0;$b<$n;$b++)
$vX[$b+1]=$_GET['X'][$b];
        $vY[$b+1]=$_GET['Y'][$b];
        $Ux[$b+1]=$_GET['Ux'][$b];
        $Uy[$b+1]=$_GET['Uy'][$b];
        $Uz[$b+1]=$_GET['Uz'][$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, 238, 238, 209);
            $red = ImageColorAllocate($pic, 255, 0, 0);
            $blue = ImageColorAllocate($pic, 0, 0, 255);
            $Green = ImageColorAllocate($pic, 69, 139, 0);
            $Pink = ImageColorAllocate($pic, 225, 20, 147);
            ImageFill($pic,0,0,$white);
            ImageSetThickness($pic, 2);
            //-------------------------------------------------------
// scale plot moment
$xMbx = 0;
for ($b=1;$b<=$m;$b++)
            {
            $v1=$vXYStart[$b];
            $v2=$vXYEnd[$b];
            $xMax=max(abs($M1[$b]),abs($M2[$b]));
            if($xMbx<$xMax){$xMbx=$xMax;}
            }
$xMx= 1/$xMbx;
//-------------------------------------------------------
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);
//---------------------------------------
$xNew=toScreen($vX[$v1]-$xMx*$M1[$b]*sin($Ang[$b]),$dMin,$dMax,900,200);
$yNew=toScreen($vY[$v1]+$xMx*$M1[$b]*cos($Ang[$b]),$dMin,$dMax,-900,900);
$x1New=toScreen($vX[$v2]+$xMx*$M2[$b]*sin($Ang[$b]),$dMin,$dMax,900,200);
$y1New=toScreen($vY[$v2]-$xMx*$M2[$b]*cos($Ang[$b]),$dMin,$dMax,-900,900);
//---------------------------------------
            Imageline ($pic,$xNew,$yNew,$x1New,$y1New,$blue);
            ImageFilledArc($pic,$xNew,$yNew,10,10,0,360,$red,IMG_ARC_EDGED);
            ImageFilledArc($pic,$x1New,$y1New,10,10,0,360,$red,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;
} ?>
```

ชื่อ DrawingAxail.php

```
<?
$n=$_GET[num_node];
$m=$_GET[num_member];
for ($b=0;$b<$m;$b++){
        $Ang[$b+1]=$_GET['Ang'][$b];
        $M1[$b+1]=$_GET['M1'][$b];
        $M2[$b+1]=$_GET['M2'][$b];
        $vXYStart[$b+1]=$_GET['start_node'][$b];
        $vXYEnd[$b+1]=$_GET['end_node'][$b];

if($Ang[$b+1]==-90){$Ang[$b+1]=$Ang[$b+1]*-1;}}
$dMin=0;
$dMax=0;

for ($b=0;$b<$n;$b++)
$vX[$b+1]=$_GET['X'][$b];
        $vY[$b+1]=$_GET['Y'][$b];
        $Ux[$b+1]=$_GET['Ux'][$b];
        $Uy[$b+1]=$_GET['Uy'][$b];
        $Uz[$b+1]=$_GET['Uz'][$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, 238, 238, 209);
$red = ImageColorAllocate($pic, 255, 0, 0);
$blue = ImageColorAllocate($pic, 0, 0, 255);
$Green = ImageColorAllocate($pic, 69, 139, 0);
$Pink = ImageColorAllocate($pic, 225, 20, 147);
ImageFill($pic,0,0,$white);
ImageSetThickness($pic, 2);
//-------------------------------------------------------
// scale plot moment
$xMbx = 0;
for ($b=1;$b<=$m;$b++)
{
$v1=$vXYStart[$b];
$v2=$vXYEnd[$b];
$xMax=max(abs($M1[$b]),abs($M2[$b]));
if($xMbx<$xMax){$xMbx=$xMax;}
}
$xMx= 1/$xMbx;
//-------------------------------------------------------
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);
//---------------------------------------
$xNew=toScreen($vX[$v1]-$xMx*-1*$M1[$b]*sin($Ang[$b]),$dMin,$dMax,900,200);
$yNew=toScreen($vY[$v1]+$xMx*-1*$M1[$b]*cos($Ang[$b]),$dMin,$dMax,-900,900);
$x1New=toScreen($vX[$v2]+$xMx*-1*$M2[$b]*sin($Ang[$b]),$dMin,$dMax,900,200);
$y1New=toScreen($vY[$v2]-$xMx*-1*$M2[$b]*cos($Ang[$b]),$dMin,$dMax,-900,900);
//---------------------------------------
Imageline ($pic,$x,$y,$xNew,$yNew,$red);
Imageline ($pic,$xNew,$yNew,$x1New,$y1New,$red);
Imageline ($pic,$x1New,$y1New,$x1,$y1,$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 DrawingMoment.php

```
<?
$n=$_GET[num_node];
$m=$_GET[num_member];
for ($b=0;$b<$m;$b++){
        $Ang[$b+1]=$_GET['Ang'][$b];
        $M1[$b+1]=$_GET['M1'][$b];
        $M2[$b+1]=$_GET['M2'][$b];
        $vXYStart[$b+1]=$_GET['start_node'][$b];
        $vXYEnd[$b+1]=$_GET['end_node'][$b];
}
$dMin=0;
$dMax=0;

for ($b=0;$b<$n;$b++)
$vX[$b+1]=$_GET['X'][$b];
        $vY[$b+1]=$_GET['Y'][$b];
        $Ux[$b+1]=$_GET['Ux'][$b];
        $Uy[$b+1]=$_GET['Uy'][$b];
        $Uz[$b+1]=$_GET['Uz'][$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, 238, 238, 209);
```

```
                $red = ImageColorAllocate($pic, 255, 0, 0);
                $blue = ImageColorAllocate($pic, 0, 0, 255);
                $Green = ImageColorAllocate($pic, 69, 139, 0);
                $Pink = ImageColorAllocate($pic, 225, 20, 147);
                ImageFill($pic,0,0,$white);
                ImageSetThickness($pic, 2);
                //-------------------------------------------------------
        // scale plot moment
        $xMbx = 0;
        for ($b=1;$b<=$m;$b++)
                {
                $v1=$vXYStart[$b];
                $v2=$vXYEnd[$b];
                $xMax=max(abs($M1[$b]),abs($M2[$b]));
                if($xMbx<$xMax){$xMbx=$xMax;}
                }
        $xMx= 1/$xMbx;
        //-------------------------------------------------------
        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);
        //-------------------------------------
                $xNew=toScreen($vX[$v1]+$xMx*$M1[$b]*sin($Ang[$b]),$dMin,$dMax,900,200);
                $yNew=toScreen($vY[$v1]-$xMx*$M1[$b]*cos($Ang[$b]),$dMin,$dMax,-900,900);
                $x1New=toScreen($vX[$v2]-$xMx*$M2[$b]*sin($Ang[$b]),$dMin,$dMax,900,200);
                $y1New=toScreen($vY[$v2]+$xMx*$M2[$b]*cos($Ang[$b]),$dMin,$dMax,-900,900);
        //-------------------------------------
                Imageline ($pic,$x,$y,$xNew,$yNew,$red);
                Imageline ($pic,$xNew,$yNew,$x1New,$y1New,$red);
                Imageline ($pic,$x1New,$y1New,$x1,$y1,$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;
} ?>
```

ประวัติผู้เขียน

ชื่อ : นายคณาวุฒิ คีรีเกษตร

เกิดวันที่ : 2 มีนาคม 2535

สถานที่เกิด : อำเภอขุนยวม จังหวัดแม่ฮ่องสอน

ที่อยู่ปัจจุบัน 814 ม.1 ตำบลขุนยวม อำเภอขุนยวม จังหวัดแม่ฮ่องสอน 58140

ประวัติการศึกษา - มัธยมศึกษาตอนต้น โรงเรียนขุนยวมวิทยา

- มัธยมศึกษาตอนปลาย โรงเรียนขุนยวมวิทยา
- ปัจจุบันกำลังศึกษา ณ คณะวิศวกรรมศาสตร์ สาขาวิศวกรรมโยธา มหาวิทยาลัยนเรศวร

.....

ชื่อ : นายอนุชิต ช่างพินิจ

เกิดวันที่ : 10 พฤศจิกายน 2534

สถานที่เกิด : อำเภอเมือง จังหวัดพิษณุโลก

ที่อยู่ปัจจุบัน : 82/11 ถนนศรีธรรมไตรปิฎก ตำบลในเมือง อำเภอเมือง จังหวัดพิษณุโลก 65000

ประวัติการศึกษา - มัธยมศึกษาตอนต้น โรงเรียนธีรธาดา พิษณุโลก

- มัธยมศึกษาตอนปลาย โรงเรียนเฉลิมขัญสตรี
- ปัจจุบันกำลังศึกษา ณ คณะวิศวกรรมศาสตร์ สาขาวิศวกรรมโยธา มหาวิทยาลัยนเรศวร