สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การหาผลเฉลยของระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่ง โดยใช้คอมพิวเตอร์

SOLVING FOR SYSTEM OF FIRST ORDER LINEAR ORDINARY DIFFERENTIAL EQUATIONS BY COMPUTER

T043020

กมล ทิมอ่อน
เปรมวุฒิ วิทวัสวงศ์
เอกนัย ตีรณธาดา

เลขหมู่.....
เลขทะเบียน 43020
วัน, เดือน, ปี 26 มิ.ย. 2545

.b.....
.i.....

ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต
ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2544

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SOLVING FOR SYSTEM OF FIRST ORDER LINEAR ORDINARY DIFFERENTIAL EQUATIONS BY COMPUTER

KAMOL TIMON

PREMWUT WITTAWATWONG

EKANAI TEERANATHADA

A SPECIAL PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF BACHELOR OF SCIENCE

DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE

FACULTY OF SCIENCE

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2001

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปัญหาพิเศษ	การหาผลเฉลยของระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่งโดยใช้คอมพิวเตอร์		
ชื่อนักศึกษา	นายกมล	ทิมอ่อน	41051001
	นายเปรมวุฒิ	วิทวัสวงศ์	41051030
	นายเอกนัย	ตีรณธาดา	41051065
ภาควิชา	คณิตศาสตร์และวิทยาการคอมพิวเตอร์		
สาขาวิชา	คณิตศาสตร์ประยุกต์		
ปีการศึกษา	2544		
อาจารย์ที่ปรึกษา	รองศาสตราจารย์ผ่องพรรณ รัตนธนาวันต์		
	ผู้ช่วยศาสตราจารย์สุนทร สุชาติเวชภูมิ		

ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้นับปัญหาพิเศษฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต สาขาคณิตศาสตร์ประยุกต์ ประจำปีการศึกษา 2544

คณะกรรมการ	ลายมือชื่อ
ประธานกรรมการ รองศาสตราจารย์ภัคคินี ชิตสกุล	
กรรมการ อาจารย์พรชัย ชัยสนิท	
กรรมการและที่ปรึกษา รองศาสตราจารย์ผ่องพรรณ รัตนธนาวันต์	
กรรมการและที่ปรึกษา ผู้ช่วยศาสตราจารย์สุนทร สุชาติเวชภูมิ	

(ผู้ช่วยศาสตราจารย์ไพรบูลย์ พันธรักษ์พงษ์)
หัวหน้าภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์

ลิขสิทธิ์ของภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปัญหาพิเศษ	การหาผลเฉลยของระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่งโดยใช้คอมพิวเตอร์	
ชื่อนักศึกษา	นายกมล ทิมอ่อน	41051001
	นายเปรมวุฒิ วิทวัสวงศ์	41051030
	นายเอกนัย ตีรณธาดา	41051065
ปริญญา	วิทยาศาสตรบัณฑิต	
ภาควิชา	คณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์	
สาขาวิชา	คณิตศาสตร์ประยุกต์	
ปีการศึกษา	2544	
อาจารย์ที่ปรึกษา	รองศาสตราจารย์ผ่องพรรณ รัตนธนาวันต์	
	ผู้ช่วยศาสตราจารย์สุนทร สุชาติเวชภูมิ	

บทคัดย่อ

ปัญหาพิเศษฉบับนี้ มีวัตถุประสงค์เพื่อศึกษาและสร้างโปรแกรมที่สามารถหาผลเฉลยของระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่งโดยใช้คอมพิวเตอร์ โดยใช้เมทริกซ์กับการประยุกต์ของระบบสมการเชิงอนุพันธ์เชิงเส้น โดยได้นำซอฟต์แวร์วิชวลเบซิก 6 มาใช้ในการสร้างอินเตอเฟซเพื่อรับค่าอินพุต จากนั้นก็นำค่าอินพุตที่ได้ไปประมวลผลที่ซอฟต์แวร์แมททิเมติกา ที่ได้ทำการเชื่อมระหว่างซอฟต์แวร์ทั้งสองไว้แล้ว และยังได้นำวิธีแมกซ์ซิมโปเซอร์มาใช้ในการหาสมการลักษณะเฉพาะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Special Project Title	Solving for System of First Order Linear Ordinary Differential Equations by Computer		
Student	Mr.Kamol	Timon	41051001
	Mr.Premwut	Wittawatwong	41051030
	Mr.Ekanai	Teeranathada	41051065
Degree	Bachelor's Degree of Science		
Department	Mathematics and Computer Science , Faculty of Science		
Programme	Applied Mathematics		
Academic Year	2001		
Special Project Advisor	Associate Professor Pongpan Rattanathanawan		
	Assistant Professor Sunthorn Suchatvejapoom		

ABSTRACT

This project has concerning in studying and designing a program that can be solving for system of first order linear ordinary differential equations by computer. This program uses the method of matrix with the applying of linear ordinary differential equation system. This program is created for benefit , comfortable and the fast of students who take this course. This program uses VISUAL BASIC 6.0 for create input's interface and then brings the input values to calculate at MATHEMATICA. The link of both is ready.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ในการทำปัญหาพิเศษเรื่อง การหาผลเฉลยของระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับที่หนึ่งโดยใช้คอมพิวเตอร์ สามารถสำเร็จลุล่วงไปได้ด้วยดีนั้น ทางคณะผู้จัดทำต้องขอขอบพระคุณ

รองศาสตราจารย์ผ่องพรรณ รัตนธนาวันต์

ผู้ช่วยศาสตราจารย์สุนทร สุชาติเวชภูมิ

อาจารย์ที่ปรึกษาปัญหาพิเศษฉบับนี้ที่กรุณาให้คำแนะนำและเป็นที่ปรึกษาในการแก้ปัญหาต่างๆ รวมทั้งเป็นผู้ตรวจสอบความถูกต้อง ของปัญหาพิเศษฉบับนี้ด้วยความเอาใจใส่เป็นอย่างยิ่ง

ขอขอบพระคุณเจ้าหน้าที่ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ ที่ให้ความสะดวกในการใช้ห้องปฏิบัติการคอมพิวเตอร์ และให้ความสะดวกในการเบิกอุปกรณ์ต่างที่ใช้ในการจัดทำปัญหาพิเศษ

คณะผู้จัดทำขอกราบขอบพระคุณอาจารย์ทุกท่านที่ประสาทวิชาความรู้ทั้งในภาคทฤษฎีและภาคปฏิบัติแก่ผู้จัดทำจนกระทั่งปัญหาพิเศษฉบับนี้สัมฤทธิ์ผลได้ด้วยทุกประการ

คณะผู้จัดทำ

มีนาคม 2545

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I

บทคัดย่อภาษาอังกฤษ.....II

กิตติกรรมประกาศ.....III

สารบัญ.....IV

สารบัญรูป.....VI

บทที่ 1 บทนำ.....1

1.1 ความเป็นมาและความสำคัญของปัญหา.....1

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....1

1.3 สมมุติฐานของการศึกษา.....1

1.4 ทฤษฎีหรือแนวทางความคิดที่ใช้ในการศึกษา.....2

1.5 ขอบเขตการศึกษา.....2

1.6 ขั้นตอนการศึกษา.....2

1.7 ข้อตกลงเบื้องต้น.....2

บทที่ 2 งานวิจัยที่เกี่ยวข้อง.....3

2.1 ค่าเจาะจงและเวคเตอร์เจาะจง.....3

2.2 การหาสมการลักษณะและค่ากำหนดของเมทริกซ์ โดยวิธีแมกซ์ซิมโปเซอร์.....6

2.3 ค่าเจาะจงซ้ำ.....8

2.3.1 ค่าเจาะจงซ้ำ 2 ค่า.....8

2.3.2 การหาผลเฉลยแบบที่ 2.....10

2.3.3 ค่าเจาะจงซ้ำ 3 ค่า.....11

2.4 การประยุกต์ในระบบเชิงเส้น.....12

2.5 ระบบสมการแบบไม่เอกพันธ์.....22

2.6 โปรแกรมแมททิเมติกา.....24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

2.6.1 การคำนวณแมททิเมติกา..... 24
2.6.1.1 หน้าจออินเตอร์เฟซ.....24
2.6.2 คำสั่งเกี่ยวพีชคณิตเชิงเส้น..... 26
2.6.2.1 การสร้างเมทริกซ์.....26
2.6.2.2 คำสั่งที่เกี่ยวกับเมทริกซ์.....26
2.6.2.3 คำสั่งหาค่าเจาะจงและเวกเตอร์เจาะจง.....27

บทที่ 3 วิธีดำเนินการวิจัย.....28
3.1 ระบบงาน..... 28
3.2 ขั้นตอนการดำเนินงาน.....28
3.3 ขั้นตอนต่างๆในการใช้งานโปรแกรม.....30
3.4 วิธีการทำงานของโปรแกรม..... 37

บทที่ 4 การอภิปรายผล.....43
4.1 ส่งเสริมให้นักศึกษาเกิดความเข้าใจใน
เรื่องการหาผลเฉลยระบบสมการเชิงอนุพันธ์สามัญเชิงเส้น.....43
4.2 ด้านการใช้งานและความเข้าใจ.....43
4.3 ข้อเสนอแนะที่ควรแก้ไข.....43

บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ.....45
5.1 สรุปลการวิจัย.....45
5.2 ข้อเสนอแนะ.....45

ภาคผนวก ก.....46
ภาคผนวก ข.....50
บรรณานุกรม.....93

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ **หน้า**

3.1 แสดง System Flow Diagram.....29
3.2 แสดงชื่อโปรแกรม.....30
3.3 ระบุไดเรกเทอรี่โดยใช้คีย์บอร์ด.....30
3.4 ระบุได้เรกเทอรี่โดยใช้ฟอร์มไดเรกเทอรี่.....31
3.5 หน้าจอรับค่าอินพุตแบบ 3x3 แบบไม่เอกพันธ์.....31
3.6 หน้าจอรับค่าอินพุตแบบ 2x2 แบบไม่เอกพันธ์.....32
3.7 หน้าจอรับค่าอินพุตแบบ 2x2 แบบเอกพันธ์.....32
3.8 หน้าจอรับค่าอินพุตแบบ 3x3 แบบเอกพันธ์.....33
3.9 แสดงค่าเอาท์พุตด้วยวิธี det(A-mI) ของระบบสมการแบบเอกพันธ์ ขนาด 3x3.....33
3.10 แสดงค่าเอาท์พุตด้วยวิธีแมกซ์ซิมโปเซอร์ของระบบสมการแบบเอกพันธ์ ขนาด 3x3.....34
3.11 แสดงค่าเอาท์พุตด้วยวิธี det(A-mI) ของระบบสมการแบบไม่เอกพันธ์ ขนาด 3x3.....35
3.12 แสดงค่าเอาท์พุตด้วยวิธี det(A-mI) ของระบบสมการแบบไม่เอกพันธ์ ขนาด 2x2.....36
ผ. 1 แสดงหน้าจอแรกของการติดตั้ง.....46
ผ. 2 แสดงหน้าจอระบุไดเรกเทอรี่.....47
ผ. 3 แสดงขั้นตอนการติดตั้ง.....48
ผ. 4 เสร็จสิ้นการติดตั้ง..... 49

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เนื่องจากการหาผลเฉลยของระบบสมการเชิงอนุพันธ์สามารถหาได้หลายวิธี ซึ่งในแต่ละวิธีก็จะมีวิธีการที่แตกต่างกันออกไปคณะผู้จัดทำมีความสนใจและประสงค์ที่จะศึกษาวิธีการหาผลเฉลยของระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่งและเล็งเห็นว่า ปัจจุบันวิวัฒนาการด้านคอมพิวเตอร์ได้ก้าวหน้าไปมากจึงมีแนวความคิดว่า ผู้ที่กำลังศึกษาในหัวข้อเรื่อง ระบบสมการเชิงอนุพันธ์สามัญเชิงเส้น น่าจะนำความรู้ทางด้านคอมพิวเตอร์มาประยุกต์ใช้ ดังนั้นคณะผู้จัดทำจึงทำการศึกษาเฉพาะหัวข้อเรื่องดังกล่าวและได้จัดทำโปรแกรมสำเร็จรูปช่วยในการหาผลเฉลยของระบบสมการด้วยวิธีเมทริกซ์

โปรแกรมที่จัดทำนี้สามารถนำไปใช้ในการเรียนการสอนวิชาพีชคณิตเชิงเส้นโดยจะเน้นเฉพาะหัวข้อ “การประยุกต์ในระบบสมการเชิงอนุพันธ์เชิงเส้น” เท่านั้น

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

1) เพื่อศึกษาการหาผลเฉลยของในระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นโดยวิธีเมทริกซ์

2) เพื่อสร้างโปรแกรมสำเร็จรูปสำหรับการหาผลเฉลยของระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่ง

1.3 สมมุติฐานของการศึกษา

ได้ศึกษาหัวข้อเรื่องระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่งได้อย่างละเอียดโดยเฉพาะวิธีเมทริกซ์และได้โปรแกรมที่สามารถหาผลเฉลยของระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่ง ซึ่งผลเฉลยจะอยู่ในรูปเมทริกซ์

1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในการศึกษา

การหาผลเฉลยของระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่งได้ใช้วิธีเมทริกซ์และได้นำวิธีของแมกซิมโปเชอร์มาหาสมการลักษณะเฉพาะ

1.5 ขั้นตอนการศึกษา

1) ศึกษาหัวข้อที่มีประโยชน์และสมการนำมาเสนอเป็นหัวข้อวิจัย

2) ศึกษาเนื้อหาที่เกี่ยวกับระบบสมการเชิงอนุพันธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ทำการศึกษาที่มาและขั้นตอนของแต่ละวิธีที่จะนำมาใช้

4) ทำการศึกษาและเลือกซอฟต์แวร์ที่เหมาะสมสำหรับโปรแกรมนี้

5) เขียนโปรแกรม

6) ตรวจสอบและแก้ไขโปรแกรมที่สร้างขึ้นให้มีความถูกต้อง

1.6 ขอบเขตการศึกษา

โปรแกรมที่จัดทำขึ้นนี้จะเน้นในส่วนของการหาผลเฉลยของระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่งที่ใช้วิธีเมทริกซ์และโดยเฉพาะค่าเจาะจงและเวกเตอร์เจาะจงที่เป็นจำนวนจริงเท่านั้น

1.7 ข้อตกลงเบื้องต้น

เนื่องจากโปรแกรมที่จัดทำขึ้นนี้จะเรียกใช้ซอฟต์แวร์แมททิเมติกา ดังนั้นคอมพิวเตอร์ที่ใช้ในการติดตั้งโปรแกรมนี้จึงควรมีซอฟต์แวร์แมททิเมติกาอยู่ด้วย คุณสมบัติของเครื่องคอมพิวเตอร์ที่เหมาะสมกับการใช้งานโปรแกรมแมททิเมติกา

1) มีซีพียูตั้งแต่ Pentium II 200 เมกกะเฮิร์ต ขึ้นไป

2) มีหน่วยความจำไม่น้อยกว่า 32 เมกกะไบต์

3) มีวินโดว์ 98 หรือ ME

4) มีพื้นที่ว่างในฮาร์ดดิสก์มากกว่า 200 เมกกะไบต์ขึ้นไป

5) มีซีดี-รอม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ความรู้พื้นฐานและทฤษฎีที่เกี่ยวข้อง

2.1 ค่าเจาะจงและเวกเตอร์เจาะจง

นิยาม ถ้า A เป็นเมทริกซ์ขนาด n x n แล้วเวกเตอร์ X ที่ไม่เป็นเวกเตอร์ศูนย์ใน R^n จะเรียกว่า เวกเตอร์เจาะจง A ถ้า AX เป็นผลคูณสเกลาร์กับ X กล่าวคือ $AX = \lambda X$ สำหรับบางสเกลาร์ λ เรียกสเกลาร์ λ ว่าค่าเจาะจงของ A และเรียกเวกเตอร์ X ว่าเป็นเวกเตอร์เจาะจงที่สมนัยกับ λ

<u>ตัวอย่าง 2.1.1</u> เวกเตอร์ $X = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$ เป็นเวกเตอร์เจาะจงของ $A = \begin{bmatrix} 3 & 0 \\ 8 & -1 \end{bmatrix}$

ที่สมนัยกับค่าเจาะจงของ $\lambda = 3$

เมื่อ $$AX = \begin{bmatrix} 3 & 0 \\ 8 & -1 \end{bmatrix}\begin{bmatrix} 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 3 \\ 6 \end{bmatrix} = 3X$$ #

การหาค่าเจาะจงและเวกเตอร์เจาะจงของเมทริกซ์ขนาด n x n

จากนิยาม $AX = \lambda X$

เขียนใหม่ $AX = \lambda IX$

หรือ $$(A - \lambda I)X = 0 \qquad (2.11)$$

สำหรับ λ ที่เป็นค่าเจาะจง ต้องมีผลเฉลยที่ไม่เป็นศูนย์ของระบบสมการนั้น สมการ (2.11) เราจะมีผลเฉลยที่ไม่เป็นศูนย์ได้ก็ต่อเมื่อ

$$\det(A - \lambda I) = 0 \quad \text{หรือ} \quad |A - \lambda I| = 0 \qquad (2.12)$$

เรียกสมการ (2.12) ว่าเป็น **สมการลักษณะเฉพาะ** ของเมทริกซ์ A สเกลาร์ λ ที่สอดคล้องกับสมการนี้คือค่าเจาะจงของ A เมื่อกระจายตัวกำหนด $\det(A - \lambda I)$ จะเป็นพหุนามใน λ เราเรียกว่า **พหุนามลักษณะเฉพาะ** ของ A

<u>ตัวอย่าง 2.1.2</u> จงหาค่าเจาะจงของเมทริกซ์ $A = \begin{bmatrix} 3 & 2 \\ -1 & 0 \end{bmatrix}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีทำ

เพราะว่า $$A-\lambda \mathrm{I}=\begin{bmatrix}3 & 2\\ -1 & 0\end{bmatrix}-\lambda\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}$$

$$=\begin{bmatrix}3-\lambda & 2\\ -1 & -\lambda\end{bmatrix}$$

แล้วพหุนามลักษณะเฉพาะของ A คือ $\det(A-\lambda I)=\begin{vmatrix}3-\lambda & 2\\ -1 & -\lambda\end{vmatrix}=\lambda^2-3\lambda+2$

และสมการลักษณะเฉพาะของ A คือ $\lambda^2-3\lambda+2=0$

ผลเฉลยของสมการนี้คือ $\lambda=1$ และ $\lambda=2$ ซึ่งเป็นค่าเจาะจงของ A #

<u>ตัวอย่าง 2.1.3</u> จงหาค่าเจาะจงและเวกเตอร์เจาะจงของ $A=\begin{bmatrix}1 & 2\\ 2 & 1\end{bmatrix}$

วิธีทำ

สมการลักษณะเฉพาะของ A คือ

$$|A-\lambda I|=0$$

$$\begin{vmatrix}1-\lambda & 2\\ 2 & 1-\lambda\end{vmatrix}=0$$

$$\lambda^2-2\lambda-3=0$$

$$(\lambda+1)(\lambda-3)=0$$

ดังนั้นค่าเจาะจงของ A คือ $\lambda=-1$ และ $\lambda=3$

หาเวกเตอร์เจาะจงได้จาก $(A-\lambda I)X=0$

สำหรับ $\lambda=-1$ จะได้

$$\begin{bmatrix}2 & 2\\ 2 & 2\end{bmatrix}\begin{bmatrix}X_1\\ X_2\end{bmatrix}=\begin{bmatrix}0\\ 0\end{bmatrix}$$ เมื่อ $\begin{bmatrix}X_1\\ X_2\end{bmatrix}$ เป็นเวกเตอร์เจาะจง

$$2X_1+2X_2=0$$

$$X_1=-X_2$$

เพราะฉะนั้นเวกเตอร์เจาะจงที่สมนัยกับ $\lambda=-1$ คือ $k\begin{bmatrix}-1\\ 1\end{bmatrix}$ เมื่อ k เป็นค่าคงที่ใดๆ

สำหรับ $\lambda=3$ จะได้

$$\begin{bmatrix}-2 & 2\\ 2 & -2\end{bmatrix}\begin{bmatrix}X_1\\ X_2\end{bmatrix}=\begin{bmatrix}0\\ 0\end{bmatrix}$$

$$-2X_1+2X_2=0$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$2X_1 - 2X_2 = 0$$

$$X_1 = X_2$$

ดังนั้นเวกเตอร์เจาะจงที่สมนัยกับ $\lambda = 3$ คือ $m\begin{bmatrix}1\\1\end{bmatrix}$ เมื่อ m เป็นค่าคงที่ใดๆ ที่ไม่เป็นศูนย์ #

ทฤษฎีบท ถ้า A เป็นเมทริกซ์ขนาด n x n และ λ เป็นค่าจริงแล้วข้อความต่อไปนี้จะสมมูล

1) λ เป็นค่าเจาะจงของ A
2) ระบบสมการ $(A - \lambda I)X = 0$ มีผลเฉลยที่มีคุณค่า
3) มีเวกเตอร์ที่ไม่เป็นเวกเตอร์ศูนย์ใน R^n ที่ทำให้ $AX = \lambda X$
4) λ เป็นผลเฉลยค่าจริงของสมการลักษณะเฉพาะ $|A - \lambda I| = 0$

เวกเตอร์เจาะจงทั้งหลายที่สมนัยกับ λ เป็นเวกเตอร์ที่ไม่เป็นศูนย์ในปริภูมิผลเฉลยของ $(A - \lambda I)X = 0$ เราเรียกปริภูมิผลเฉลยนี้ว่า **ปริภูมิเจาะจง** ของ A ที่สมนัยกับ λ

<u>ตัวอย่าง 2.1.4</u> จงหามูลฐานสำหรับปริภูมิเจาะจงของ $A = \begin{bmatrix}3 & -2 & 0\\ -2 & 3 & 0\\ 0 & 0 & 5\end{bmatrix}$

วิธีทำ

$$|A - \lambda I| = 0$$

$$\begin{vmatrix}3-\lambda & -2 & 0\\ -2 & 3-\lambda & 0\\ 0 & 0 & 5-\lambda\end{vmatrix} = 0$$

$$(\lambda - 1)(\lambda - 5)^2 = 0$$

เพราะฉะนั้น $\lambda = 1, 5, 5$

สำหรับ $\lambda = 1$ หาเวกเตอร์เจาะจงได้ดังนี้

$$\begin{bmatrix}2 & -2 & 0\\ -2 & 2 & 0\\ 0 & 0 & 4\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix} = \begin{bmatrix}0\\0\\0\end{bmatrix}$$

แก้ระบบสมการจะได้ $x_1 = t, x_2 = t, x_3 = 0$

ฉะนั้นเวกเตอร์เจาะจงที่สมนัยกับ $\lambda = 1$ เป็นเวกเตอร์ที่อยู่ในรูปแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$X = \begin{bmatrix} t \\ t \\ 0 \end{bmatrix} = t \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$$

ดังนั้นมูลฐานสำหรับปริภูมิเจาะจงที่สมนัยกับ $\lambda = 1$ คือ $\begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$

สำหรับ $\lambda = 5$ หาเวกเตอร์เจาะจงได้ดังนี้

$$\begin{bmatrix} -2 & -2 & 0 \\ -2 & -2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

แก้ระบบสมการจะได้ $x_1 = -k$, $x_2 = k$, $x_3 = t$

เวกเตอร์เจาะจงที่สมนัยกับ $\lambda = 5$ เป็นเวกเตอร์เจาะจงที่อยู่ในรูปแบบ

$$X = \begin{bmatrix} -k \\ k \\ 0 \end{bmatrix} = \begin{bmatrix} -k \\ k \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ t \end{bmatrix}$$

$$= k \begin{bmatrix} -1 \\ 1 \\ 0 \end{bmatrix} + t \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

เนื่องจาก $\begin{bmatrix} -1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$ เป็นอิสระเชิงเส้น ดังนั้น มูลฐานสำหรับปริภูมิเจาะจงที่สมนัยกับ $\lambda = 5$ คือ $\left\{ \begin{bmatrix} -1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right\}$ #

2.2 การหาสมการลักษณะเฉพาะ และค่ากำหนดของเมทริกซ์ โดยวิธีของแมกซิมโปเซอร์

ขอนิยามคำว่า **ผลบวกเฉียง** ของเมทริกซ์

นิยาม ถ้า $A = (a_{ij})$ เป็น n x n เมทริกซ์ เราจะนิยามผลบวกเฉียงของ A คือ

$$Tr(A) = \sum_{i=1}^{n} a_{ii}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่าง 2.2.1 กำหนด

$$A = \begin{bmatrix} 2 & 1 & 1 \\ 0 & -1 & 0 \\ 1 & 2 & 3 \end{bmatrix}$$

$$\therefore \quad \text{Tr(A)} = 2 + (-1) + 3 = 4$$

โดยปกติ ถ้า A เป็นเมทริกซ์จตุรัสขนาด n x n ใดๆ เราสามารถหาสมการลักษณะเฉพาะของ A ได้จาก

$$|A - mI| = 0$$

หรือ

$$|mI - A| = 0 \tag{2.21}$$

ในที่นี้ m เป็นค่าเจาะจงของ A

ซึ่งเมื่อกระจายดีเทอร์มิแนนท์ทางซ้ายมือ (2.21) แล้วจะสมการดังนี้

$$m^n + c_1 m^{n-1} + c_2 m^{n-2} + \ldots + c_{n-1} m + c_n = 0 \tag{2.22}$$

สมการ (2.22) เราเรียกว่าสมการลักษณะเฉพาะของ A จากนั้นเราก็ทำการแยกแฟคเตอร์ หรือใช้วิธีทางการวิเคราะห์ตัวเลข หารากของสมการ (2.22) ออกมาได้ n ค่า สมมุติว่ารากทั้ง n ค่านั้นคือ $m_1, m_2, \ldots, m_n$

ค่า m เหล่านี้ เราเรียกว่าค่าเจาะจงของ A

คุณสมบัติของค่ารากเจาะจง คือ

$$m_1 + m_2 + \ldots + m_n = \text{Tr(A)} \tag{2.23}$$

$$m_1 \text{ x } m_2 \text{ x } \ldots \text{x } m_n = \text{Tr(A)} \tag{2.24}$$

แมกซิมโปเชอร์ได้ให้ความสัมพันธ์ระหว่างผลบวกเฉียงของเมทริกซ์สัมประสิทธิ์ของสมการลักษณะเฉพาะ (2.22) ไว้ดังนี้

ถ้าเราให้

$$S_1 = Tr(A), S_2 = Tr(A^2), S_3 = Tr(A^3), \ldots, S_n = Tr(A^n)$$

แล้วจะได้ว่าสัมประสิทธิ์ $C_1, C_2, \ldots, C_n$ ของสมการลักษณะเฉพาะ (2) จะเป็นดังนี้

$$C_1 = -S_1$$

$$C_2 = -\frac{1}{2}(C_1 S_1 + S_2)$$

$$C_3 = -\frac{1}{3}(C_2 S_1 + C_1 S_2 + S_3) \tag{2.25}$$

.

.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$C_n = -\frac{1}{n}(C_{n-1}S_1 + C_{n-2}S_2 + \ldots + C_1S_{n-1} + S_n)$$

และได้อีกว่า

$$\det(A) = (-1)^n C_n \tag{2.26}$$

2.3 ค่าเจาะจงซ้ำ

ไม่จำเป็นที่ค่าเจาะจง n ค่า $\lambda_1, \lambda_2, \ldots, \lambda_n$ ของเมทริกซ์ขนาด nxn ต้องมีค่าจำเพาะ อาจมีบางค่าของค่าเจาะจงอาจจะซ้ำกัน

<u>ตัวอย่าง 2.3.1</u> $$X' = \begin{pmatrix} 3 & -18 \\ 2 & -9 \end{pmatrix} X \tag{2.31}$$

จะได้สมการลักษณะเฉพาะ $(\lambda + 3)^2 = 0$ เพราะ $\lambda_1 = \lambda_2 = -3$ ซึ่งเป็นค่าเจาะจงคงที่ซ้ำกัน 2 ค่า จะได้เวกเตอร์เจาะจงคือ

$$k_1 = \begin{pmatrix} 3 \\ 1 \end{pmatrix}, X_1 = \begin{pmatrix} 3 \\ 1 \end{pmatrix} e^{-3t} \tag{2.32}$$

ซึ่งเป็นผลเฉลยหนึ่งของ (2.41)

ถ้า m เป็นค่าคงที่บวกและ $(\lambda - \lambda_1)^m$ เป็นสมการลักษณะเฉพาะ ในขณะที่ $(\lambda - \lambda_1)^{m+1}$ ไม่ใช่ ดังนั้น λ_1 กล่าวได้เป็นค่าเจาะจงซ้ำกัน m ค่า

1) สำหรับ nxn เมทริกซ์ A อาจจะเป็นได้ที่จะหาเวกเตอร์เจาะจง m เวกเตอร์ $k_1, k_2, \ldots, k_m$ ซึ่งสัมพันธ์กับ λ_1 โดยที่ $m \le n$ โดยผลเฉลยของระบบสมการทั่วไปจะอยู่ในรูปแบบของ

$$c_1k_1e^{\lambda_1 t} + c_2k_2e^{\lambda_2 t} + \ldots + c_mk_me^{\lambda_1 t}$$

2) ถ้ามีเพียงเวกเตอร์เจาะจงสัมพันธ์กับค่าเจาะจง ดังนั้น รูปแบบของผลเฉลยคือ

$$x_1 = k_{11}e^{\lambda_1 t}$$
$$x_2 = k_{21}e^{\lambda_1 t} + k_{22}e^{\lambda_1 t}$$
$$\vdots$$
$$x_m = k_{m1}\frac{t^{m-1}}{(m-1)!}e^{\lambda_1 t} + k_{m2}\frac{t^{m-2}}{(m-2)!}e^{\lambda_1 t} + \ldots + k_{mm}e^{\lambda_1 t}$$

2.3.1 ค่าเจาะจงซ้ำ 2 ค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราจะเริ่มพิจารณาจากค่าเจาะจงที่ซ้ำ 2 ค่า ในตัวอย่างแรกนี้เราสามารถหา 2 เวกเตอร์เจาะจงซึ่งสัมพันธ์กับคู่ค่าเจาะจง

ตัวอย่าง 2.3.2

จงแก้ $$x' = \begin{bmatrix} 1 & -2 & 2 \\ -2 & 1 & -2 \\ 2 & -2 & 1 \end{bmatrix} x$$

วิธีทำ $$\det(A-\lambda I) = \begin{bmatrix} 1-\lambda & -2 & 2 \\ -2 & 1-\lambda & -2 \\ 2 & -2 & 1-\lambda \end{bmatrix} = 0$$

$-(\lambda+1)^2(\lambda-5)=0$ เราจะได้ $\lambda_1 = \lambda_2 = -1$ และ $\lambda_3 = 5$

สำหรับ $\lambda_1 = -1$ ใช้วิธี Gauss-Jordan

$$(A+I|0) = \left[\begin{array}{ccc|c} 2 & -2 & 2 & 0 \\ -2 & 2 & -2 & 0 \\ 2 & -2 & 2 & 0 \end{array}\right]$$

แถวโอเปอเรชั่น

$$\left[\begin{array}{ccc|c} 1 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right]$$

แถวแรกของเมทริกซ์สุดท้าย หมายถึง $k_1 - k_2 + k_3 = 0$ หรือ $k_1 = k_2 - k_3$ เลือก $k_2 = 1 + k_3 = 0$ และ $k_2 = 1, k_3 = 1$ จะได้ $k_1 = 1$ และ $k_1 = 0$ ดังนั้น 2 เวกเตอร์เจาะจงที่สัมพันธ์กับ $\lambda_1 = 1$ คือ

$$k_1 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \quad \text{และ} \quad k_2 = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$$

สุดท้ายสำหรับ $\lambda_3 = 5$

$$(A+5I|0) = \left[\begin{array}{ccc|c} -4 & -2 & 2 & 0 \\ -2 & -4 & -2 & 0 \\ -2 & -2 & -4 & 0 \end{array}\right]$$

แถวโอเปอเรชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\left[\begin{array}{ccc|c} 1 & 0 & -1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right]$$

จะได้ $k_1 = k_3$ และ $k_2 = k_3$ เลือก $k_3 = 1$ ให้ $k_1 = 1, k_2 = -1$ และ เวกเตอร์ที่ 3 คือ

$$k_3 = \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$$

เรารวมผลเฉลยของระบบเป็น

$$x = c_1 \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} e^{-t} + c_2 \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} e^{-t} + c_3 \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} e^{-t} \qquad \#$$

2.3.2 การหาผลเฉลยแบบที่ 2

กำหนดให้ λ_1 เป็นค่าเจาะจงซ้ำ 2 และมีเพียงเวกเตอร์เจาะจง 1 ค่า ที่สัมพันธ์กับค่าเจาะจงผลเฉลยแบบที่ 2 อยู่ในรูปแบบ

$$x_2 = kte^{\lambda_1 t} + pe^{\lambda_1 t} \qquad (2.33)$$

ซึ่ง $k = \begin{bmatrix} k_1 \\ k_2 \\ \vdots \\ k_n \end{bmatrix}$ และ $p = \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{bmatrix}$

แทน (2.43) ลงในระบบ $x' = Ax$ และทำการปรับรูป

$$(Ak - \lambda_1 k)te^{\lambda_1 t} + (Ap - \lambda_1 p - k)e^{\lambda_1 t} = 0$$

$$(A - \lambda_1 I)k = 0 \qquad (2.34)$$

และ $$(A - \lambda_1 I)p = 0 \qquad (2.35)$$

โดยที่ K ต้องเป็นเวกเตอร์เจาะจงของ A ซึ่งสัมพันธ์กับ λ_1 จากการแก้สมการ (2.34) จะได้ $x_1 = ke^{\lambda_1 t}$

<u>ตัวอย่าง 2.3.3</u>

หาผลเฉลยของระบบสมการที่ให้ใน (2.41)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีทำ จาก (2.32) เรารู้ $\lambda_1 = -3$ และ ผลเฉลยหนึ่งเป็น $x_1 = \begin{pmatrix}3\\1\end{pmatrix}e^{-3t}$ ระบุ $k = \begin{pmatrix}3\\1\end{pmatrix}$ และ $p = \begin{pmatrix}p_1\\p_2\end{pmatrix}$ เราสามารถหาจาก (2.35) ซึ่งเราต้องแก้ปัญหา

$(A+3I)p = k$ หรือ $\begin{aligned}6p_1 - 18p_2 &= 3\\ 2p_1 - 6p_2 &= 1\end{aligned}$

ถ้าเลือก $p_1 = 1$ เราจะได้ $p_2 = 1/6$

อย่างไรก็ตาม เราจะเลือก $p_1 = 1/2$ ดังนั้น $p_2 = 0$ จะได้ $p = \begin{pmatrix}1/2\\0\end{pmatrix}$ จาก (2.33) เรา จะได้ $x_2 = \begin{pmatrix}3\\1\end{pmatrix}te^{-3t} + \begin{pmatrix}1/2\\0\end{pmatrix}e^{-3t}$

ผลเฉลยทั่วไปของ (2.31) คือ

$$x = c_1\begin{pmatrix}3\\1\end{pmatrix}e^{-3t} + c_2\left[\begin{pmatrix}3\\1\end{pmatrix}te^{-3t} + \begin{pmatrix}1/2\\0\end{pmatrix}e^{-3t}\right] \qquad \#$$

2.3.3 ค่าเจาะจงซ้ำ 3

เมื่อเมทริกซ์ A มีเพียงเวกเตอร์เจาะจง 1 เวกเตอร์ สัมพันธ์กับค่าเจาะจงทั้ง 3 ค่าเราจะหาผลเฉลยของรูปแบบ (2.33)และผลเฉลยที่ 3บางรูปแบบ

$$x_3 = k\frac{t^2}{2}e^{\lambda_1 t} + pte^{\lambda_1 t} + qe^{\lambda_1 t} \qquad (2.46)$$

ซึ่ง $k = \begin{bmatrix}k_1\\k_2\\\vdots\\k_n\end{bmatrix}$, $p = \begin{bmatrix}p_1\\p_2\\\vdots\\p_n\end{bmatrix}$ และ $Q = \begin{bmatrix}q_1\\q_2\\\vdots\\q_n\end{bmatrix}$

แทน (2.36) ลงในระบบ $x' = Ax$ เราจะได้

$$(A-\lambda_1 I)k = 0 \qquad (2.37)$$

$$(A-\lambda_1 I)p = k \qquad (2.38)$$

$$(A-\lambda_1 I)Q = p \qquad (2.39)$$

ผลเฉลยของ (2.37)และ(2.38) เราใช้รูปแบบผลเฉลย x_1 และ x_2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่าง 2.3.4

แก้ปัญหา $$x' = \begin{bmatrix} 2 & 1 & 6 \\ 0 & 2 & 5 \\ 0 & 0 & 2 \end{bmatrix} x$$

วิธีทำ สมการลักษณะเฉพาะ $(\lambda - 2)^3 = 0$ จะได้ $\lambda_1 = 2$ ซึ่งซ้ำกัน 3 ค่า โดยการแก้ $(A - 2I)k = 0$ เราจะได้

$$k = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

ถัดไปเราจะแก้ระบบ $(A - 2I)p = k$ และ $(A - 2I)Q = P$ เราจะได้

$$p = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \quad \text{และ} \quad Q = \begin{bmatrix} 0 \\ 6/5 \\ 1/5 \end{bmatrix}$$

ใช้ (2.33) และ (2.36) ผลเฉลยทั่วไปของระบบ คือ

$$x = c_1 \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} e^{2t} + c_2 \left(\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} te^{2t} + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} e^{2t} \right) + c_3 \left(\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \frac{t^2}{2} e^{2t} + \begin{bmatrix} 0 \\ 6/5 \\ 1/5 \end{bmatrix} e^{2t} \right) \qquad \#$$

2.4 การประยุกต์ในระบบสมการเชิงอนุพันธ์เชิงเส้น

ในหัวข้อนี้จะกล่าวถึงการนำเอาพีชคณิตเชิงเส้นไปประยุกต์ใช้ในการหาผลเฉลยของระบบสมการเชิงอนุพันธ์เชิงเส้น

พิจารณาระบบสมการเชิงอนุพันธ์เชิงเส้นใน n สมการ

$$\begin{aligned} \frac{dx_1}{dt} &= a_{11}(t)x_1 + a_{12}(t)x_2 + a_{13}(t)x_3 + \ldots + a_{1n}(t)x_n + f_1(t) \\ \frac{dx_2}{dt} &= a_{21}(t)x_1 + a_{22}(t)x_2 + a_{23}(t)x_3 + \ldots + a_{2n}(t)x_n + f_2(t) \\ &\vdots \\ \frac{dx_n}{dt} &= a_{n1}(t)x_1 + a_{n2}(t)x_2 + a_{n3}(t)x_3 + \ldots + a_{nn}(t)x_n + f_n(t) \end{aligned} \qquad (2.41)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อสัมประสิทธิ์ a_{ij} และ f_i เป็นฟังก์ชั่นที่มีความต่อเนื่องในช่วง I

ถ้า $f_i(t) = 0$, $i = 1, 2, \ldots, n$ ในระบบสมการ (2.41) เรียกระบบสมการนี้ว่าเป็นเอกพันธ์และถ้า $f_i(t) \neq 0$ เรียกว่าไม่เอกพันธ์

$$\text{ถ้าให้ } X = \begin{bmatrix} x_1(t) \\ x_2(t) \\ . \\ . \\ . \\ x_n(t) \end{bmatrix}, A(t) = \begin{bmatrix} a_{11}(t) & a_{12}(t) & . & . & . a_{1n}(t) \\ a_{21}(t) & a_{22}(t) & . & . & . a_{2n}(t) \\ . & . & & & . \\ . & . & & & . \\ . & . & & & . \\ a_{n1}(t) & a_{n2}(t) & . & . & . a_{nn}(t) \end{bmatrix}$$

$$\text{และ} \quad F(t) = \begin{bmatrix} f_1(t) \\ f_2(t) \\ . \\ . \\ . \\ f_n(t) \end{bmatrix}$$

แล้วระบบสมการเชิงอนุพันธ์เชิงเส้นใน (2.41) จะเขียนในรูปแบบ

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ . \\ . \\ . \\ x_n \end{bmatrix} = \begin{bmatrix} a_{11}(t) & a_{12}(t) & . & . & . a_{1n}(t) \\ a_{21}(t) & a_{22}(t) & . & . & . a_{2n}(t) \\ . & . & & & . \\ . & . & & & . \\ . & . & & & . \\ a_{n1}(t) & a_{n2}(t) & . & . & . a_{nn}(t) \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ . \\ . \\ . \\ x_n \end{bmatrix} + \begin{bmatrix} f_1(t) \\ f_2(t) \\ . \\ . \\ . \\ f_n(t) \end{bmatrix}$$

หรือ เขียนย่อๆ ได้เป็น

$$\frac{dx}{dt} = A(t)X + F(t) \tag{2.42}$$

และระบบสมการเชิงอนุพันธ์เชิงเส้นเอกพันธ์จะเขียนได้เป็น

$$\frac{dx}{dt} = A(t)X \tag{2.43}$$

โดยทั่วไปมักจะเขียนสมการ (2.42) และ (2.43) แทนด้วย $X' = AX + F$ และ $X' = AX$ ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<u>ตัวอย่าง 2.4.1</u> ระบบสมการเอกพันธ์

$$\frac{dx}{dt} = 2x - 3y$$
$$\frac{dx}{dt} = 6x + 5y$$

มีรูปแบบเมทริกซ์ คือ $\frac{dx}{dt} = \begin{bmatrix} 2 & -3 \\ 6 & 5 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix}$ #

<u>ตัวอย่าง 2.4.2</u> จงเขียนระบบสมการที่กำหนดให้รูปแบบเมทริกซ์

$$\frac{dx_1}{dt} = 3x_1$$
$$\frac{dx_2}{dt} = -2x_2$$
$$\frac{dx_3}{dt} = 5x_3$$

วิธีทำ รูปแบบเมทริกซ์ของระบบสมการ คือ

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 3 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & 5 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$ #

นิยาม เวกเตอร์ผลเฉลย บนช่วง I เป็นเมทริกซ์สดมภ์ $X = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \cdot \\ \cdot \\ \cdot \\ x_n(t) \end{bmatrix}$

ซึ่งสมาชิกเป็นฟังก์ชั่นที่หาอนุพันธ์ได้ซึ่งสอดคล้องกับระบบสมการ $X' = AX + F(t)$ บนช่วงดังกล่าว

<u>ตัวอย่าง 2.4.3</u> จงแสดงว่า $X_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}e^{-2t} = \begin{bmatrix} e^{-2t} \\ -e^{-2t} \end{bmatrix}$ และ $X_2 = \begin{bmatrix} 3 \\ 5 \end{bmatrix}e^{-2t} = \begin{bmatrix} 3e^{-2t} \\ 5e^{-2t} \end{bmatrix}$

เป็นผลเฉลย $X' = \begin{bmatrix} 1 & 3 \\ 5 & 3 \end{bmatrix}X$ บนช่วง $(-\infty, \infty)$

วิธีทำ $\because\ X_1' = \begin{bmatrix} -2e^{-2t} \\ 2e^{-2t} \end{bmatrix}$ และ $AX_1 = \begin{bmatrix} 1 & 3 \\ 5 & 3 \end{bmatrix}\begin{bmatrix} e^{-2t} \\ -e^{-2t} \end{bmatrix} = \begin{bmatrix} -2e^{-2t} \\ 2e^{-2t} \end{bmatrix} = X_1'$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\because\ X_2' = \begin{bmatrix} 18e^{6t} \\ 30e^{6t} \end{bmatrix}$$

และ $$AX_2 = \begin{bmatrix} 1 & 3 \\ 5 & 3 \end{bmatrix}\begin{bmatrix} 3e^{6t} \\ 5e^{6t} \end{bmatrix} = \begin{bmatrix} 18e^{6t} \\ 30e^{6t} \end{bmatrix} = X_2'$$ #

ปัญหาค่าเริ่มต้น

ให้ t_0 แทนจุดบนช่วง I และ $X(t_0) = \begin{bmatrix} x_1(t_0) \\ x_2(t_0) \\ \cdot \\ \cdot \\ \cdot \\ x_n(t_0) \end{bmatrix}$ และ $X_0 = \begin{bmatrix} c_1 \\ c_2 \\ \cdot \\ \cdot \\ \cdot \\ c_n \end{bmatrix}$

เมื่อ c_i , $i = 1, 2, \ldots, n$ เป็นค่าคงที่ที่กำหนดให้แล้วปัญหาจากการแก้ระบบสมการ $\frac{dx}{dt} = A(t)X + F(t)$ และได้ $X(t_0) = X_0$ เป็นปัญหาค่าเริ่มต้นบนช่วงที่กำหนด

ทฤษฎีบท กำหนดให้สมาชิกของเมทริกซ์ A(t) และ F(t) เป็นฟังก์ชั่นที่มีความต่อเนื่องบนช่วง I ซึ่งประกอบด้วยจุด t_0 แล้วจะมีผลเฉลยเพียงอย่างหนึ่งอย่างเดียวเกิดขึ้นในปัญหาค่าเริ่มต้น $\frac{dx}{dt} = A(t)X + F(t)$ ซึ่งได้ $X(t_0) = X_0$ บนช่วงที่กล่าวถึง

ทฤษฎีบท กำหนดให้ $X_1, X_2, \ldots, X_n$ เป็นเซตของเวกเตอร์ผลเฉลยของระบบสมการเชิงอนุพันธ์เชิงเส้นเอกพันธ์ $X' = AX$ บนช่วง I แล้วผลบวกเชิงเส้น $X = c_1X_1 + c_2X_2 + \ldots + c_nX_n$ เป็นผลเฉลยทั่วไปของระบบสมการดังกล่าวนี้ด้วย เมื่อ C_i เป็นค่าคงที่ไม่เจาะจง ซึ่ง I =1 , 2 , ... , n

<u>ตัวอย่าง 2.4.4</u> ผลเฉลยหนึ่งของระบบสมการ

$$X' = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ -2 & 0 & -1 \end{bmatrix}X \quad \text{คือ} \quad X_1 = \begin{bmatrix} \cos t \\ (-1/2)\cos t + (1/2)\sin t \\ -\cos t - \sin t \end{bmatrix}$$

สำหรับค่าคงที่ใดๆ C_1 เวกเตอร์ $X = C_1X_1$ เป็นผลเฉลยด้วย เมื่อ

$$\frac{dX}{dt} = \begin{bmatrix} -c_1\cos t \\ (-1/2)c_1\cos t + (1/2)c_1\sin t \\ -c_1\cos t - c_1\sin t \end{bmatrix}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$AX = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ -2 & 0 & -1 \end{bmatrix} \begin{bmatrix} c_1 \cos t \\ (-1/2)c_1 \cos t + (1/2)c_1 \sin t \\ -c_1 \cos t - c_1 \sin t \end{bmatrix}$$

$$= \begin{bmatrix} -c_1 \sin t \\ (1/2)c_1 \cos t + (1/2)c_1 \sin t \\ -c_1 \cos t + c_1 \sin t \end{bmatrix}$$

ซึ่งแสดงว่า $X' = AX$ #

<u>ตัวอย่าง 2.4.5</u> พิจารณาระบบสมการจากตัวอย่าง 2.3.4

ถ้า $X_2 = \begin{bmatrix} 0 \\ e^t \\ 0 \end{bmatrix}$ แล้ว $X_2' = \begin{bmatrix} 0 \\ e^t \\ 0 \end{bmatrix}$

และ $AX_2 = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ -2 & 0 & -1 \end{bmatrix} \begin{bmatrix} 0 \\ e^t \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ e^t \\ 0 \end{bmatrix} = X_2'$

ดังนั้นจะเห็นได้ว่า X_2 เป็นผลเฉลยของระบบสมการในตัวอย่าง 2.4.4 ด้วย

จากทฤษฎีบท ผลเฉลย $X = C_1X_1 + C_2X_2$

$$= c_1 \begin{bmatrix} \cos t \\ (-1/2)\cos t + (1/2)\sin t \\ -\cos t - \sin t \end{bmatrix} + c_2 \begin{bmatrix} 0 \\ e^t \\ 0 \end{bmatrix}$$

เป็นผลเฉลยอีกอันหนึ่งของระบบสมการดังกล่าว #

นิยาม อิสระเชิงเส้นและไม่อิสระเชิงเส้น

กำหนดให้ $X_1, X_2, \ldots, X_n$ เป็นเซตของเวกเตอร์ผลเฉลย ของระบบสมการเอกพันธ์ $X' = AX$ บนช่วง I เรากล่าวว่าเซตดังกล่าวจะไม่เป็นอิสระเชิงเส้นบนช่วง ถ้ามีค่าคงที่ $C_1, C_2, \ldots, C_n$ ที่ไม่เป็นศูนย์ทั้งหมด ซึ่ง $C_1X_1 + C_2X_2 + \ldots + C_kX_k = 0$ สำหรับทุกค่า t ในช่วงและเซตของเวกเตอร์ดังกล่าว เป็นอิสระเชิงเส้น เมื่อมีค่าคงที่ $C_1, C_2, \ldots, C_n$ เป็นศูนย์ทุกตัว

ทฤษฎีบท เงื่อนไขการเป็นอิสระเชิงเส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

กำหนดให้ $X_1 = \begin{bmatrix} X_{11} \\ X_{21} \\ . \\ . \\ . \\ X_{n1} \end{bmatrix}, X_2 = \begin{bmatrix} X_{12} \\ X_{22} \\ . \\ . \\ . \\ X_{n2} \end{bmatrix}, \ldots, X_n = \begin{bmatrix} X_{1n} \\ X_{2n} \\ . \\ . \\ . \\ X_{nn} \end{bmatrix}$

เป็นเวกเตอร์ผลเฉลย n เวกเตอร์ของระบบสมการเอกพันธ์ $X' = AX$ บนช่วง I เงื่อนไขที่จำเป็นและเพียงพอที่เซตของผลเฉลยจะเป็นอิสระเชิงเส้น คือ วรอนสกี

$$(X_1, X_2, \ldots, X_n) = \begin{vmatrix} X_{11} & X_{12} & . & . & . & X_{1n} \\ X_{21} & X_{22} & . & . & . & X_{2n} \\ . & . & & & & . \\ . & . & & & & . \\ . & . & & & & . \\ X_{n1} & X_{n2} & . & . & . & X_{nn} \end{vmatrix} \neq 0$$

สำหรับทุกๆ ค่า t ใน I

นิยาม เซตหลักมูล

เวกเตอร์ผลเฉลยที่เป็นอิสระเชิงเส้น n เวกเตอร์ใดๆ $X_1, X_2, \ldots, X_n$ ของระบบสมการเอกพันธ์ $X' = AX$ บนช่วง I จะเรียกว่าเป็นเซตหลักมูลของผลเฉลยบนช่วงดังกล่าว

ทฤษฎีบท การมีอยู่ของเซตหลักมูล

เซตหลักมูลของผลคูณของผลเฉลยสำหรับระบบสมการเอกพันธ์ $X' = AX$ บนช่วง I จะมีอยู่เสมอ

นิยาม ผลเฉลยทั่วไปของระบบสมการเอกพันธ์

กำหนดให้ $X_1, X_2, \ldots, X_n$ เป็นเซตหลักมูลของผลเฉลยของระบบสมการเอกพันธ์ $X' = AX$ บนช่วง I แล้วผลเฉลยทั่วไปของระบบสมการในช่วง กำหนดด้วย $X = C_1X_1 + C_2X_2 + \ldots + C_nX_n$ เมื่อ C_i เป็นค่าคงที่ไม่เจาะจง $i = 1, 2, \ldots, n$

ตัวอย่าง 2.4.5 จากตัวอย่าง 2.4.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
43020

$X_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix} e^{-2t}$ และ $X_1 = \begin{bmatrix} 3 \\ 5 \end{bmatrix} e^{6t}$ เป็นผลเฉลยของระบบสมการ $X' = \begin{bmatrix} 1 & 3 \\ 5 & 3 \end{bmatrix} X$ และเป็นผลเฉลยที่เป็นอิสระเชิงเส้นบนช่วง $(-\infty, \infty)$ ดังนั้น สร้างเซตหลักมูลของผลเฉลยบนช่วงดังกล่าว X_1, X_2 แล้วผลเฉลยทั่วไปของระบบสมการนี้ คือ

$$X = C_1X_1 + C_2X_2 = C_1\begin{bmatrix} 1 \\ -1 \end{bmatrix} e^{-2t} + C_2\begin{bmatrix} 3 \\ 5 \end{bmatrix} e^{6t} \qquad \#$$

ทฤษฎีบท กำหนดให้ $X_1, X_2, \ldots, X_k$ เป็นเซตของเวกเตอร์ของผลเฉลยของระบบสมการเอกพันธ์ $X' = AX$ บนช่วง I และให้ X_p เป็นผลเฉลยใดๆ ของระสมการไม่เอกพันธ์ $X' = AX + F(t)$ บนช่วงเดียวกันแล้ว $X = C_1X_1 + C_2X_2 + \ldots + C_kX_k + X_p$ จะเป็นผลเฉลยของระบบสมการไม่เอกพันธ์บนช่วงดังกล่าว เมื่อ $X = C_1, C_2, \ldots, C_k$ เป็นค่าคงที่ใดๆ

นิยาม ผลเฉลยทั่วไปของระบบสมการไม่เอกพันธ์

กำหนดให้ X_p เป็นผลเฉลยที่กำหนดให้ของระบบสมการไม่เอกพันธ์ $X' = AX + F(t)$ บนช่วง I และให้ $X = C_1X_1 + C_2X_2 + \ldots + C_nX_n$ เป็นผลเฉลยทั่วไปบนช่วงเดียวกันที่สมนัยกับของระบบสมการเอกพันธ์ $X' = AX$ แล้วผลเฉลยทั่วไปของระบบสมการไม่เอกพันธ์บนช่วงนี้กำหนดด้วย $X = X_c + X_p$

<u>ตัวอย่าง 2.4.6</u> เมทริกซ์หลักมูล

กำหนดให้ $X_1 = \begin{bmatrix} X_{11} \\ X_{21} \\ \vdots \\ X_{n1} \end{bmatrix}, X_2 = \begin{bmatrix} X_{12} \\ X_{22} \\ \vdots \\ X_{n2} \end{bmatrix}, \ldots, X_n = \begin{bmatrix} X_{1n} \\ X_{2n} \\ \vdots \\ X_{nn} \end{bmatrix}$

เป็นเซตหลักมูลของเวกเตอร์ผลเฉลยของระบบสมการเอกพันธ์ $X' = AX$ บนช่วง I แล้วจะเรียกเมทริกซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\phi(t)=\begin{bmatrix} X_{11} & X_{12} & . & . & . & X_{1n} \\ X_{21} & X_{22} & . & . & . & X_{1n} \\ . & . & & & & . \\ . & . & & & & . \\ . & . & & & & . \\ X_{n1} & X_{n2} & . & . & . & X_{nn} \end{bmatrix}$$ ว่าเป็นเมทริกซ์หลักมูลของระบบสมการบนช่วงดังกล่าว

ตัวอย่าง 2.4.7 เวกเตอร์ $X_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix} e^{-2t} = \begin{bmatrix} e^{-2t} \\ -e^{-2t} \end{bmatrix}$ และ $X_2 = \begin{bmatrix} 3 \\ 5 \end{bmatrix} e^{6t} = \begin{bmatrix} 3e^{6t} \\ 5e^{6t} \end{bmatrix}$ เป็นเซตหลักมูลของผลเฉลยของระบบสมการ $X' = \begin{bmatrix} 1 & 3 \\ 5 & 3 \end{bmatrix} X$ บนช่วง ($-\infty, \infty$) แล้ว เมทริกซ์หลักมูลของระบบสมการดังกล่าวนี้ คือ

$$\phi(t)=\begin{bmatrix} e^{-2t} & 3e^{6t} \\ -e^{-2t} & 5e^{6t} \end{bmatrix}$$ #

การหาผลเฉลยของระบบสมการเชิงอนุพันธ์เชิงเส้นอันดับหนึ่ง

จากระบบสมการ $X' = AX$ เมื่อ A เป็นเมทริกซ์ขนาด n*n ถ้าให้ $X = Ke^{\lambda t}$ แทนใน $X' = AX$ จะได้ $K\lambda e^{\lambda t} = AKe^{\lambda t}$ เพราะว่า $e^{\lambda t} \neq 0$ จึงต้องหารตลอดสมการดังกล่าว จะได้ $AK = \lambda K$ หรือ $(A - \lambda I)K = 0$ ซึ่ง λ เป็นค่าเจาะจงของ A และ K เป็นเวกเตอร์เจาะจงที่สมนัยกับ λ เมื่อ A เป็นเมทริกซ์ขนาด n*n ซึ่งมีค่าเจาะจงต่าง ๆ กัน $\lambda_1, \lambda_2, \ldots, \lambda_n$ แล้วเซตของเวกเตอร์เจาะจง $K_1, K_2, \ldots, K_n$ ที่เป็นอิสระเชิงเส้น จะแสดงถึงผลเฉลยของเซตหลักมูลของระบบสมการ $X' = AX$ บนช่วง $(-\infty, \infty)$ คือ

$$X_1 = K_1 e^{\lambda_1 t}, X_2 = K_2 e^{\lambda_2 t}, \ldots, X_n = K_n e^{\lambda_n t}$$

ทฤษฎีบท ผลเฉลยทั่วไป

กำหนดให้ $\lambda_1, \lambda_2, \ldots, \lambda_n$ เป็นค่าเจาะจงค่าจริงต่างๆ กัน ของ เมทริกซ์สัมประสิทธิ์ A ของระบบสมการเอกพันธ์ $X' = AX$ และให้ $K_1, K_2, \ldots, K_n$ เป็นเวกเตอร์เจาะจงที่สมนัยกับค่าเจาะจงดังกล่าว แล้วผลเฉลยทั่วไปของระบบสมการ $X' = AX$ บนช่วง $(-\infty, \infty)$ จะกำหนดด้วย

$$X = c_1 K_1 e^{\lambda_1 t} + c_2 X_2 + c_n K_n e^{\lambda_n t}$$

ตัวอย่าง 2.4.8 จงแก้ระบบสมการ $\frac{dx}{dt} = 2x + 3y$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\frac{dy}{dt} = 2x + y$$

วิธีทำ เมทริกซ์สัมประสิทธิ์ $A = \begin{bmatrix} 2 & 3 \\ 2 & 1 \end{bmatrix}$

หาค่าเจาะจง และเวกเตอร์เจาะจงของ A จาก

$$|A - \lambda I| = \begin{vmatrix} 2-\lambda & 3 \\ 1 & 1-\lambda \end{vmatrix} = \lambda^2 - 3\lambda - 4 = 0$$

จะได้ $(\lambda + 1)(\lambda - 4) = 0$ นั่นคือ $\lambda = -1$ และ $\lambda = 4$

สำหรับ $\lambda = -1$ หาเวกเตอร์เจาะจงได้ $K_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$

และสำหรับ $\lambda = 4$ หาเวกเตอร์เจาะจงได้ $K_2 = \begin{bmatrix} 3 \\ 2 \end{bmatrix}$

ดังนั้น ผลเฉลยที่เป็นอิสระเชิงเส้นของระบบสมการนี้ คือ

$X_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix} e^{-t}$ และ $X_2 = \begin{bmatrix} 3 \\ 2 \end{bmatrix} e^{4t}$

และผลเฉลยทั่วไป คือ

$$x = c_1 \begin{bmatrix} 1 \\ -1 \end{bmatrix} e^{-t} + c_2 \begin{bmatrix} 3 \\ 2 \end{bmatrix} e^{4t}$$ เมื่อ c_1, c_2 เป็นค่าคงที่ไม่เจาะจง

ตัวอย่าง 2.4.9 จงแก้ระบบสมการ $\frac{dx}{dt} = -4x + y + z$

$$\frac{dy}{dt} = x + 5y - z$$

$$\frac{dz}{dt} = y - 3z$$

วิธีทำ จัดระบบสมการให้อยู่ในรูปแบบเมทริกซ์ จะได้

$$X' = AX \text{ เมื่อ } x' = \begin{bmatrix} x \\ y \\ z \end{bmatrix}', A = \begin{bmatrix} -4 & 1 & 1 \\ 1 & 5 & -1 \\ 0 & 1 & -3 \end{bmatrix}, X = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

$$|A - \lambda I| = \begin{vmatrix} -4-\lambda & 1 & 1 \\ 1 & 5 & -1 \\ 0 & 1 & -3-\lambda \end{vmatrix} = -(\lambda + 3)(\lambda + 4)(\lambda - 5) = 0$$

จะได้ $\lambda_1 = -3$, $\lambda_2 = -4$ และ $\lambda_3 = 5$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับ $\lambda_1 = -3$ จะได้ $K_1 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$

สำหรับ $\lambda_2 = -4$ จะได้ $K_2 = \begin{bmatrix} 10 \\ -1 \\ 1 \end{bmatrix}$

สำหรับ $\lambda_3 = 5$ จะได้ $K_3 = \begin{bmatrix} 1 \\ 8 \\ 1 \end{bmatrix}$

ดังนั้น ผลเฉลยทั่วไปของระบบสมการคือ

$$x = c_1 \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} e^{-3t} + c_2 \begin{bmatrix} 10 \\ -1 \\ 1 \end{bmatrix} e^{-4t} + c_3 \begin{bmatrix} 1 \\ 8 \\ 1 \end{bmatrix} e^{5t}$$

เมื่อ c_1, c_2 และ c_3 เป็นค่าที่ไม่เจาะจง

<u>ตัวอย่าง 2.4.10</u> จงแก้ระบบสมการ $\frac{dx_1}{dt} = 3x_1$

$$\frac{dx_2}{dt} = -2x_2$$

$$\frac{dx_3}{dt} = 5x_3$$

วิธีทำ จัดระบบสมการให้อยู่ในรูปแบบเมทริกซ์ จะได้

$$X' = AX \text{ เมื่อ } x' = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}', A = \begin{bmatrix} 3 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & 5 \end{bmatrix}, X = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

$$|A - \lambda \mathrm{I}| = \begin{vmatrix} 3-\lambda & 0 & 0 \\ 0 & -2-\lambda & 0 \\ 0 & 0 & 5-\lambda \end{vmatrix}$$

$$(3-\lambda)(-2-\lambda)(5-\lambda) = 0$$

$$\lambda = 3, \lambda = -2, \lambda = 5$$

สำหรับ $\lambda = 3$ หาเวกเตอร์เจาะจงได้ $K_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับ $\lambda = -2$ หาเวกเตอร์เจาะจงได้ $K_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$

สำหรับ $\lambda = 5$ หาเวกเตอร์เจาะจงได้ $K_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$

ดังนั้นผลเฉลยทั่วไปของระบบสมการ คือ

$$X = c_1 \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} e^{3t} + c_2 \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} e^{-2t} + c_3 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} e^{5t}$$

เมื่อ c_1, c_2 และ c_3 เป็นค่าที่ไม่เจาะจง

หรือ $x_1 = c_1 e^{3t}$

$x_2 = c_2 e^{-2t}$

$x_3 = c_3 e^{5t}$

จากระบบสมการในตัวอย่าง **2.4.10** เราสามารถหาผลเฉลยได้ง่ายเพราะเมทริกซ์สัมประสิทธิ์ A ในตัวอย่าง **2.4.10** เป็นเมทริกซ์เฉียง ดังนั้น ในตอนนี้เราจะเอาแนวความคิดนี้มาจัดเมทริกซ์สัมประสิทธิ์ในระบบสมการที่ไม่เป็นเมทริกซ์เฉียงให้เป็นเมทริกซ์เฉียง เพื่อง่ายแก่การหาผลเฉลยของระบบสมการเอกพันธ์

$X' = AX$

2.5 ระบบสมการแบบไม่เอกพันธ์

ในหัวข้อนี้พิจารณา $X' = AX + B$ (2.51)

เมื่อ A เป็นเมทริกซ์ค่าคงตัวขนาด n x n และ B เป็นฟังก์ชั่นเวกเตอร์ของ t ดังนั้นจากทฤษฎีบทที่ 5 จะต้องหาผลเฉลยเฉพาะราย X_p ของ (2.61) และรวมกับผลเฉลยของระบบสมการแบบเอกพันธ์ ซึ่งการหาผลเฉลยเฉพาะราย X_p นั้นเป็นวิธีแปรผันของพารามิเตอร์

ตัวอย่างที่ 2.5.1 จงหาผลเฉลยของระบบ

$$\begin{bmatrix} X \\ Y \end{bmatrix}' = \begin{bmatrix} 0 & 1 \\ -2 & 3 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} f(t) \\ g(t) \end{bmatrix} \quad (2.52)$$

ผลเฉลย จากตัวอย่างที่ 1 ในหัวข้อที่ 2 หาผลเฉลยทั่วไปของระบบสมการแบบเอกพันธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\begin{bmatrix} X \\ Y \end{bmatrix}' = \begin{bmatrix} 0 & 1 \\ -2 & 3 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix}$$

จะได้

$$\begin{bmatrix} X \\ Y \end{bmatrix} = a_1\begin{bmatrix} 1 \\ 1 \end{bmatrix}e^{t} + a_2\begin{bmatrix} 1 \\ 2 \end{bmatrix}e^{2t}$$

เมื่อ a_1 และ a_2 เป็นค่าคงตัวไม่เจาะจง

เราจะได้ผลเฉลยของระบบ (2.62) อยู่ในรูป

$$\begin{bmatrix} X \\ Y \end{bmatrix}_p = a_1(t)\begin{bmatrix} 1 \\ 1 \end{bmatrix}e^{t} + a_2(t)\begin{bmatrix} 1 \\ 2 \end{bmatrix}e^{2t}$$

แทนค่าใน (2.52)

$$a_1(t)\begin{bmatrix} 1 \\ 1 \end{bmatrix}e^{t} + 2a_2(t)\begin{bmatrix} 1 \\ 2 \end{bmatrix}e^{2t} + a_1'(t)\begin{bmatrix} 1 \\ 1 \end{bmatrix}e^{t} + a_2'(t)\begin{bmatrix} 1 \\ 2 \end{bmatrix}e^{2t}$$

$$= \begin{bmatrix} 0 & 1 \\ -2 & 3 \end{bmatrix}\begin{bmatrix} 1 \\ 1 \end{bmatrix}a_1(t)e^{t} + \begin{bmatrix} 0 & 1 \\ -2 & 3 \end{bmatrix}\begin{bmatrix} 1 \\ 2 \end{bmatrix}a_2(t)e^{2t} + \begin{bmatrix} f(t) \\ g(t) \end{bmatrix}$$

หรือ

$$a_1'(t)\begin{bmatrix} 1 \\ 1 \end{bmatrix}e^{t} + a_2'(t)\begin{bmatrix} 1 \\ 2 \end{bmatrix}e^{2t} = \begin{bmatrix} f(t) \\ g(t) \end{bmatrix}$$

ดังนั้น

$$a_1'(t)e^{t} + a_2'(t)e^{2t} = f(t)$$
$$a_1'(t)e^{t} + 2a_2'(t)e^{2t} = g(t)$$

ใช้กฎของคราเมอร์หา

$$a_1'(t) = \frac{\begin{vmatrix} f(t) & 1 \\ g(t) & 2 \end{vmatrix}}{\begin{vmatrix} 1 & 1 \\ 1 & 2 \end{vmatrix}} = 2f(t) - g(t)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้น

$$a_1'(t) = [2f(t) - g(t)]e^{-t}$$
$$a_2'(t) = [g(t) - f(t)]e^{-2t}$$
$$a_2'(t) = \frac{\begin{vmatrix} 1 & f(t) \\ 1 & g(t) \end{vmatrix}}{\begin{vmatrix} 1 & 1 \\ 1 & 2 \end{vmatrix}} = g(t) - f(t)$$

ตัวอย่างเช่น ถ้า $f(t) = e^t$ และ $g(t) = 1$ แล้ว

$$a_1'(t) = 2t + e^{-t}$$
$$a_2'(t) = \frac{1}{2}e^{-2t} + e^{-t}$$

ผลเฉลยเฉพาะของ (5) คือ

หรือ

$$\begin{bmatrix} X \\ Y \end{bmatrix}_p = \left(2t + e^t\right)\begin{bmatrix} 1 \\ 1 \end{bmatrix}e^t + \left(-\frac{1}{2}e^{-2t} + e^{-t}\right)\begin{bmatrix} 1 \\ 2 \end{bmatrix}e^{2t}$$

$$X_p = \begin{bmatrix} x \\ y \end{bmatrix} = \left(2te^t + 1\right)\begin{bmatrix} 1 \\ 1 \end{bmatrix} + \left(e^t - \frac{1}{2}\right)\begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

2.6 โปรแกรม Mathematica V. 3

2.6.1 การคำนวณแมททิเมติกา

2.6.1.1 หน้าจออินเตอร์เฟซ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การรัน Mathematica ด้วย Notebook Interfaces

ดับเบิ้ลคลิกที่ไอคอน Mathematica	การแสดงรูปแบบกราฟฟิกสำหรับการเริ่มต้น Mathematica
Mathematica	คำสั่งระบบปฏิบัติการ เพื่อสั่งเริ่มต้น Mathematica
การจบข้อความด้วย Shift-Return	การป้อน input สำหรับ Mathematica
เลือก quit จากเมนู	การออกจาก Mathematica

ถ้าเราใช้คอมพิวเตอร์โดยผ่านทาง graphical interface เราต้องดับเบิ้ลคลิกที่ไอคอน Mathematicaเพื่อการเริ่มต้นของ Mathenamtica แต่ถ้าเราใช้คอมพิวเตอร์โดยผ่านทางระบบปฏิบัติการโดยพื้นฐานเป็นข้อความ (texually based O.S.) เราต้องพิมพ์คำสั่ง Mathematica เพื่อเริ่มต้นของ Mathematica

เมื่อ Mathematica เริ่มทำงานมันจะแสดงหน้าจอว่าเพื่อให้เราป้อน input ลงไปแล้วกดปุ่ม Shift และ Return (หมายถึงปุ่ม enter นั่นเอง) โดยจะกดปุ่ม shift ค้างไว้และกด enter จากนั้น Mathematica จะประมวลผล input และแสดงผลลัพธ์มาให้

2+2

เพียงแค่เราพิมพ์ 2+2 แล้วกดปุ่ม shift – return จากนั้น Mathematica ทำการประมวลผล input แล้วเพิ่มข้อความ In[1]:= ให้ และแสดงผลลัพธ์หลังข้อความ Out[1]=

In[1] := 2+2

Out[1] = 4

ถ้าต้องการออกจาก Mathematica ทำได้โดยการเลือก Quit จากเมนูใน notebook interface

Text-Based Interface

การรัน Mathematica ด้วย trext-base interface

Math	คำสั่งระบบปฏิบัติการเพื่อสั่งเริ่มต้น Mathematica
การจบข้อความด้วย Shift - Return	การป้อน input สำหรับ Mathematica ที่ใช่ในทุกระบบ
การจบข้อความด้วย return	รูปแบบอย่างง่ายของการป้อน input ที่ใช้ได้ในบางระบบ
Control - D หรือ Quit[ ]	การออกจาก Mathematica

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อการเริ่มต้น Mathematica เราจะต้องพิมพ์คำสั่ง math ที่ O.S. prompt ในบางระบบอาจจะเริ่มต้น Mathematica ด้วยการดับเบิ้ลคลิกที่ไอคอน Mathematica Kernel

เมื่อ Mathematica เริ่มต้นการทำงานมันจะพิมพ์ In[1] := ขึ้นมาเพื่อเป็นการบอกว่าพร้อมสำหรับการป้อน input แล้ว และเมื่อทำการป้อน input เสร็จในบางระบบจะกดปุ่ม shift – return บางระบบจะกดปุ่ม return (หรือ enter) เพียงปุ่มเดียวหลังจากนั้น Mathematica จะทำการประมวลผล input และแสดงผลลัพธ์หลังข้อความ Out[1] =

สำหรับการออกจาก Mathematica นั้นทำได้โดยการกด Control – D หรือ พิมพ์ Quit[ ] ที่ input prompt

2.6.2 คำสั่งเกี่ยวกับพีชคณิตเชิงเส้น

2.6.2.1 การสร้างเมทริกซ์

Table[f , {i,m} , {j,n}]	สร้างเมทริกซ์ขนาด m x n ที่ f คือฟังก์ชั่นของ i และ j ซึ่งให้ค่าของ i ณ ตำแหน่ง j
IdentityMatrix[n]	สร้างเมทริกซ์เอกลักษณ์ขนาด n x n
Table[0 , {m} , {n}]	เป็นเมทริกซ์ศูนย์
Table[Random[] , {m} , {n}]	สร้างเมทริกซ์ขนาด m x n โดยทำการสุ่มค่า

ตัวอย่าง

สร้างเมทริกซ์ขนาด 2 x 2

```
In[1]:= Table[ a[i,j] , {i,2} , {j,2} ]
Out[1]:= {{a[1,1],a[1,2]},{a[2,1],a[2,2]}}
```

สร้างเมทริกซ์เอกลักษณ์ขนาด n x n

```
In[2]:= IdentityMatrix[3]
Out[2]:= {{1,0,0} , {0,1,0} , {0,0,1}}
```

2.6.2.2 คำสั่งที่เกี่ยวกับเมทริกซ์

Transpose[m]	ทำการ Transpose เมทริกซ์ m
Inverse[m]	ทำการ Inverse เมทริกซ์ m
Det[m]	ทำการ Det เมทริกซ์ m

ตัวอย่าง

ทำการ Transpose เมทริกซิขนาด 2 x 3 จะได้เมทริกซ์ขนาด 3 x 2

```
In[1]:= Transpose[ {a,b,c} ,{ap,bp,cp} ]
Out[1]:= { {a,ap} , {b,bp} , {c,cp} }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นี้คือค่า determinant ของเมทริกซ์ ตัวอย่างขนาด 2x2

```
In[2]:= Det[{ {a,b} , {c,d} }]
Out[2]:= -bc+ad
```

2.6.2.3 คำสั่งหาค่าเจาะจงและเวกเตอร์เจาะจง

Eigenvalues[m]	รายการค่า Eigenvalues ของเมทริกซ์ m
Eigenvectors[m]	รายการค่า Eigenvectors ของเมทริกซ์ m
Eigensystem[m]	รายการของรูปแบบ {eigenvalues , eigenvectors}

ตัวอย่าง

เมทริกซ์ขนาด 2 x 2

```
In[1]:= m = {{2.3,4.5},{6.7,-1.2}}
Out[1]:= {{2.3,4.5},{6.7,-1.2}}
```

เมทริกซ์นี้มีค่า Eigenvalues 2 ค่า ซึ่งดปีนจำนวนจริงทั้ง 2 ค่า

```
In[2]:= Eigenvalues[m]
Out[1]:= {6.3103,-5.21303}
```

มี Eigenvectors 2 ค่า

```
In[3]:= Eigenvectors[m]
Out[1]:= {{0.746335,0.66557},{-0.513839,0.857886}}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
วิธีดำเนินการ

3.1 ระบบงาน

3.1.1 ส่วนนำเข้าข้อมูล

เป็นระบบที่นำเข้าข้อมูลอย่างง่าย ข้อมูลที่น้ำเข้า คือ ระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่ง

3.1.2 ส่วนวิเคราะห์และประมวลผล

จากข้อมูลส่วนนำเข้า นำข้อมูลที่ได้มาวิเคราะห์และนำไปคำนวณหาสมการลักษณะเฉพาะ , ค่าเจาะจง , เวกเตอร์เจาะจง และผลเฉลย

3.1.3 ส่วนแสดงผล

นำข้อมูลที่ได้จากการประมวลผลมาแสดงทางจอภาพ จะประกอบด้วย สมการลักษณเฉพาะ , ค่าเจาะจง , เวกเตอร์เจาะจง และผลเฉลย

3.2 ขั้นตอนการดำเนินงาน

1) นำข้อมูลเข้าดยที่ข้อมูลที่จะนำเข้าจะเป็นระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่ง และอยู่ในขอบเขตของตัวแปร X , Y และ Z

2) จัดรูปแบบของข้อมูลนำเข้าให้อยู่ในรูปของเมทริกซ์ A พร้อมกับทำการเลือกรูปแบบของระบบสมการ คือ แบบไม่เอกพันธ์หรือเอกพันธ์แล้วจึงเลือกขนาดของเมทริกซ์ 2 x 2 หรือ 3 x 3

3) เลือกวิธีทีจะใช้ในการคำนวณ ประกอบด้วย วิธีแมกซ์ซิมโปเซอร์ หรือ วิธี Det(A - mI)

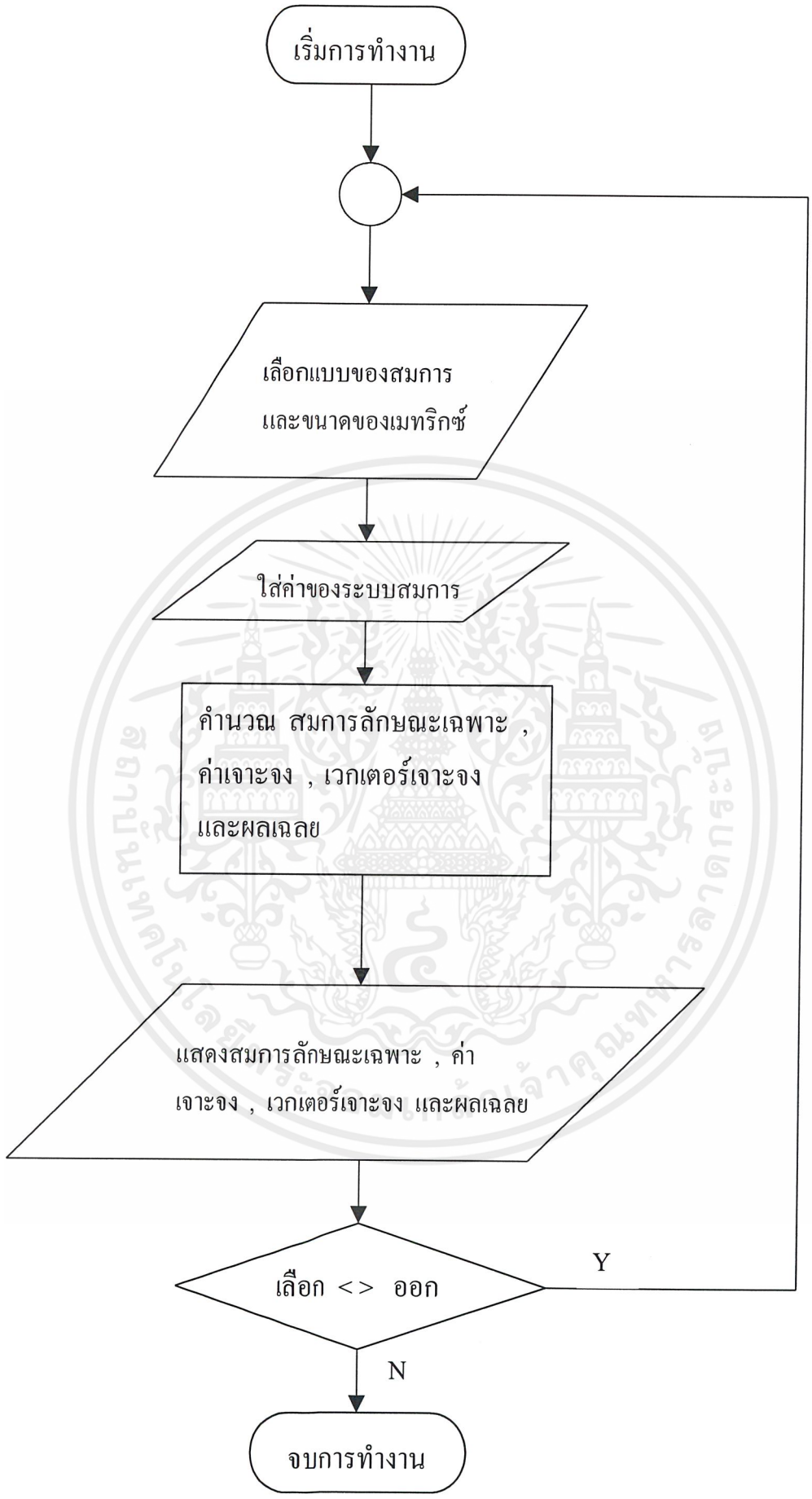
4) ทำการคำนวณหาสมการลักษณะ , ค่าเจาะจง , เวกเตอร์เจาะจง และผลเฉลย

5) แสดงข้อมูลที่ได้หลังจากการคำนวณ ประกอบด้วย สมการลักษณะ , ค่าเจาะจง , เวกเตอร์เจาะจง และผลเฉลยทางหน้าจอแสดงผล

6) จบการทำงาน

รูป 3.1 แสดง System Flow Chart

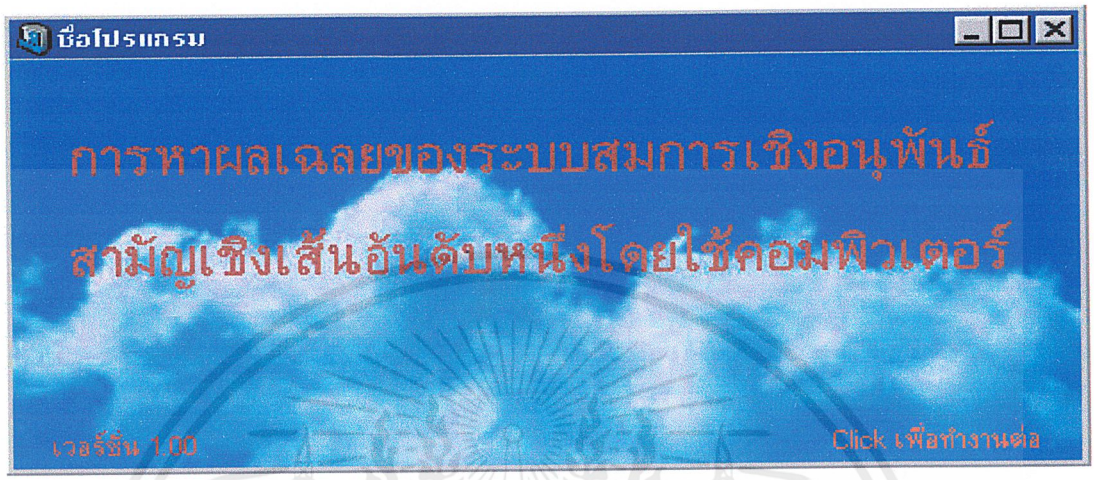
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ขั้นตอนต่างๆในการใช้งานโปรแกรม

1. เมื่อทำการรันโปรแกรม จะปรากฏดังนี้



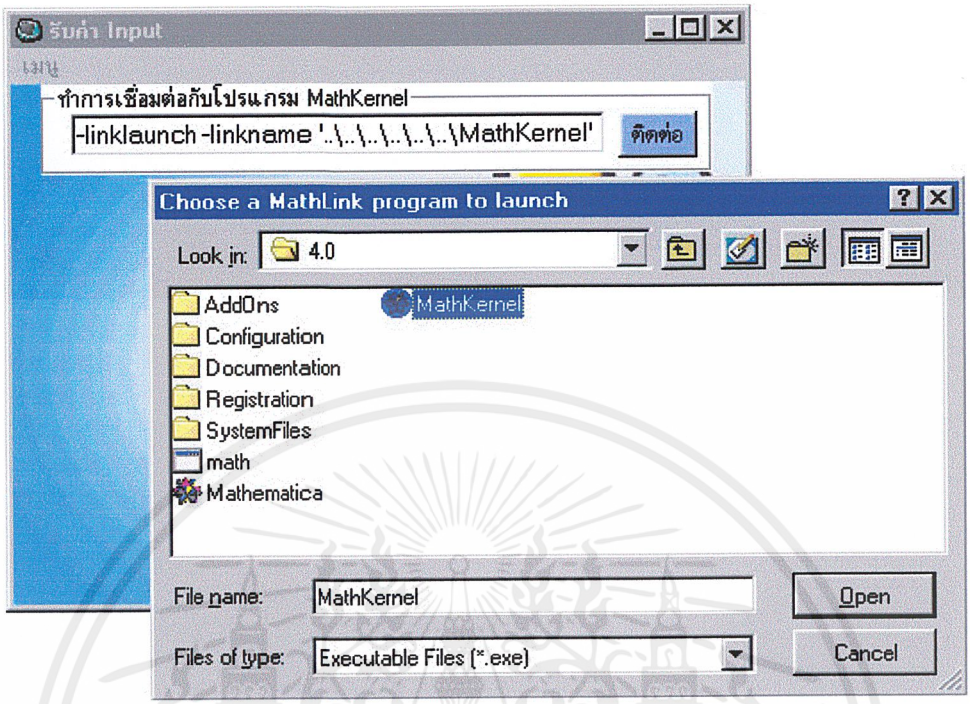
รูป 3.2 แสดงชื่อโปรแกรม

2. แสดงไดเรกเทอรี่ของโปรแกรมแมทเคอร์เนล

รูป 3.3 ระบุไดเรกเทอรี่โดยใช้คีย์บอร์ด

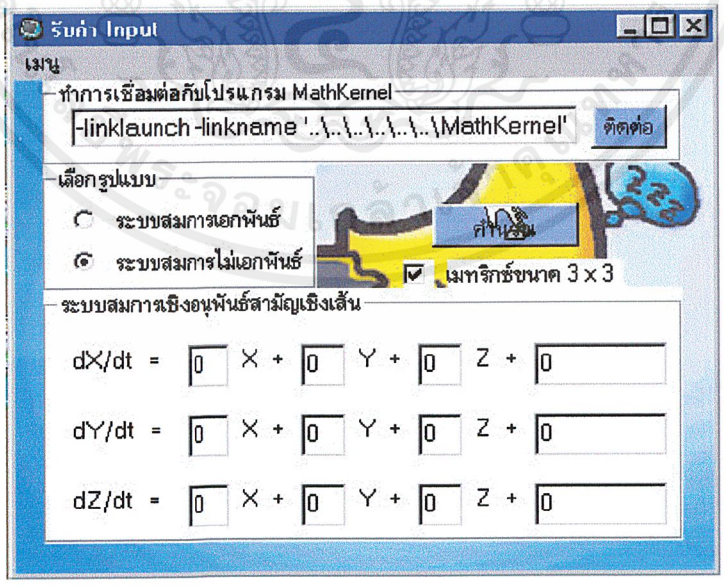
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากคลิกปุ่ม “ติดต่อ”



รูป 3.4 ระบุไดเรกเทอรี่โดยใช้ฟอร์มไดเรกเทอรี่

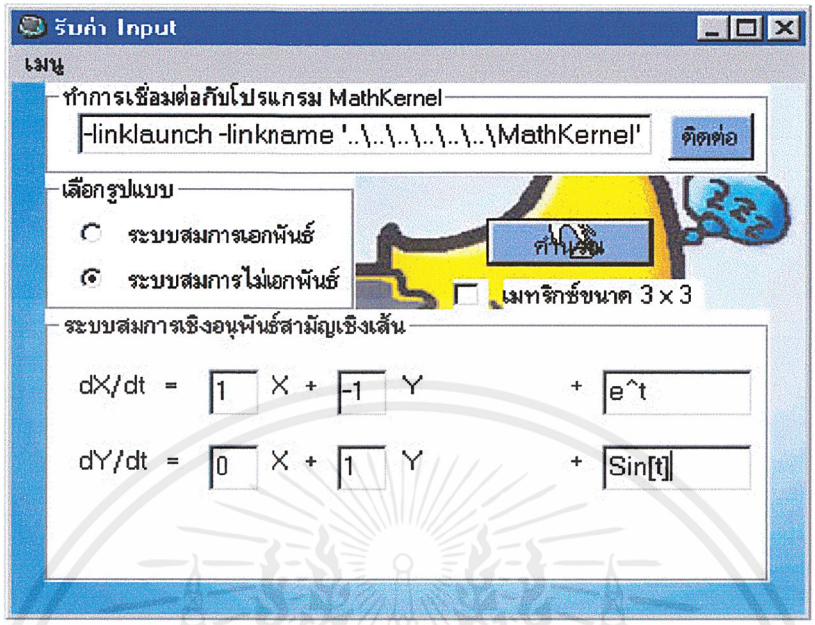
หลังจากทำการเชื่อมต่อกับโปรแกรมแมทเคอร์เนล



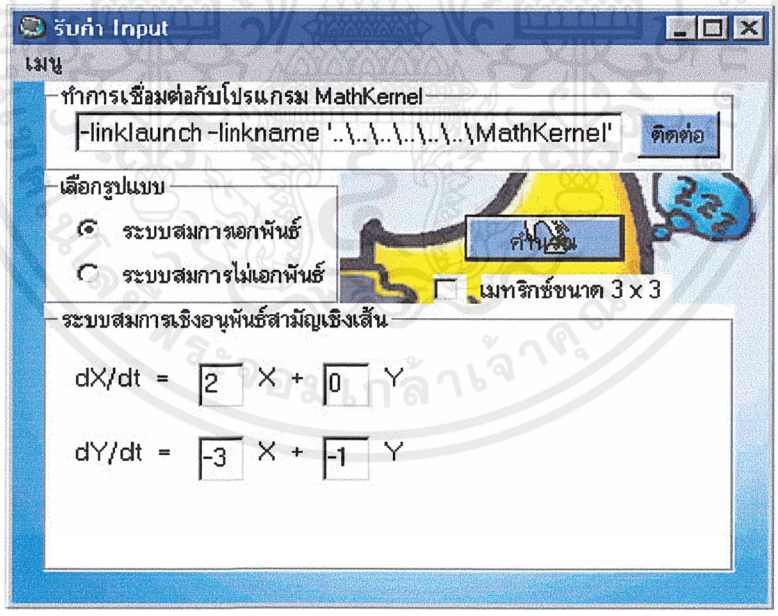
รูป 3.5 หน้าจอรับค่าอินพุตแบบ 3x3 แบบไม่เอกพันธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. รูปแบบต่างๆ ของหน้าจออินพุต

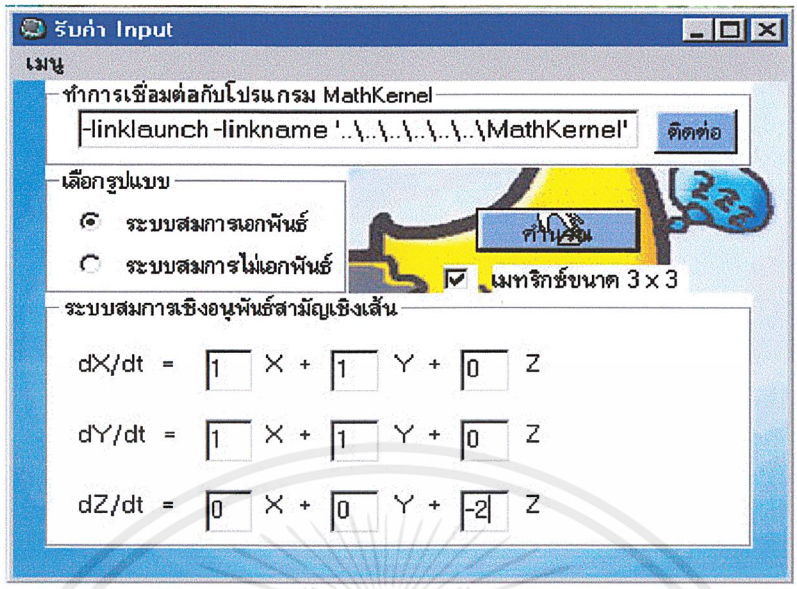


รูป 3.6 หน้าจอรับค่าอินพุตแบบ 2x2 แบบไม่เอกพันธ์



รูป 3.7 หน้าจอรับค่าอินพุตแบบ 2x2 แบบเอกพันธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.8 หน้าจอรับค่าอินพุตแบบ 3x3 แบบเอกพันธ์

รูป 3.9 แสดงค่าเอาท์พุตด้วยวิธี det(A-mI) ของระบบสมการแบบเอกพันธ์ ขนาด 3x3

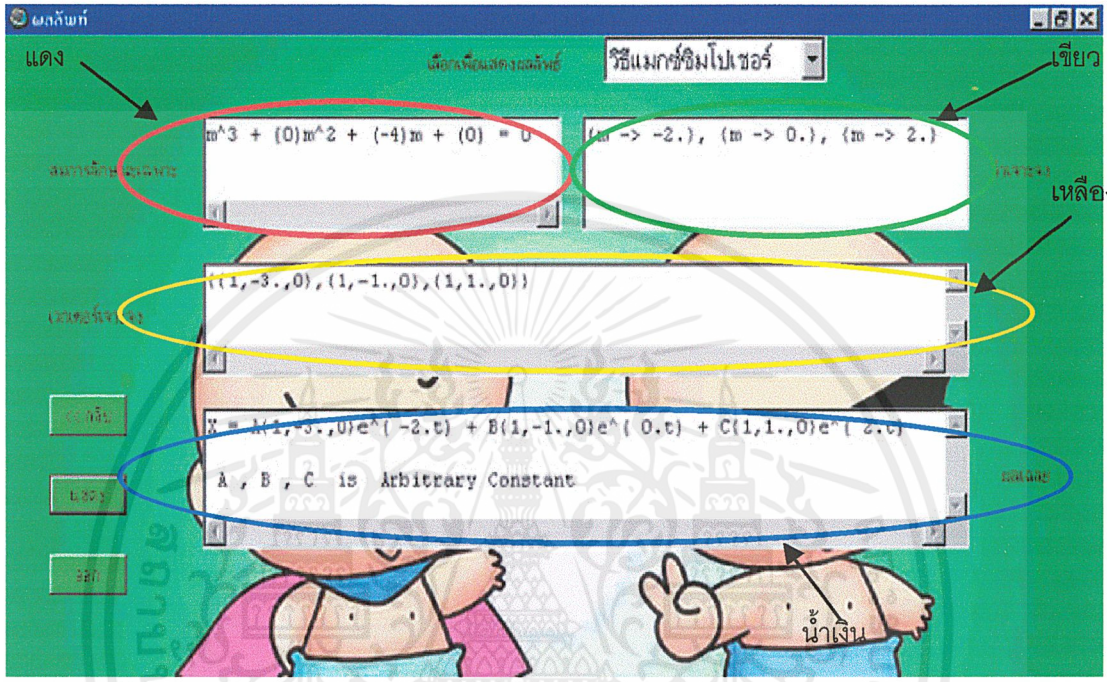
<u>หมายเหตุ</u> รูปแบบของฟังก์ชั่นที่มีตัวแปร t ของระบบสมการแบบไม่เอกพันธ์ ตัวอย่างดังนี้

- มุม เช่น Sin[t] , Cos[2t] , Tan[3t] , Cot[t] , Sec[3t] , Cosec[4t] , Sin[t]Cos[t] และ Tan[t]Sec[2t] เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เอกซ์โปแนนเชียล เช่น e^t , e^(2t) , e^(4t) , 2te^(t) , Sin[t]e^(2t) และ tCos[t]e^(2t) เป็นต้น

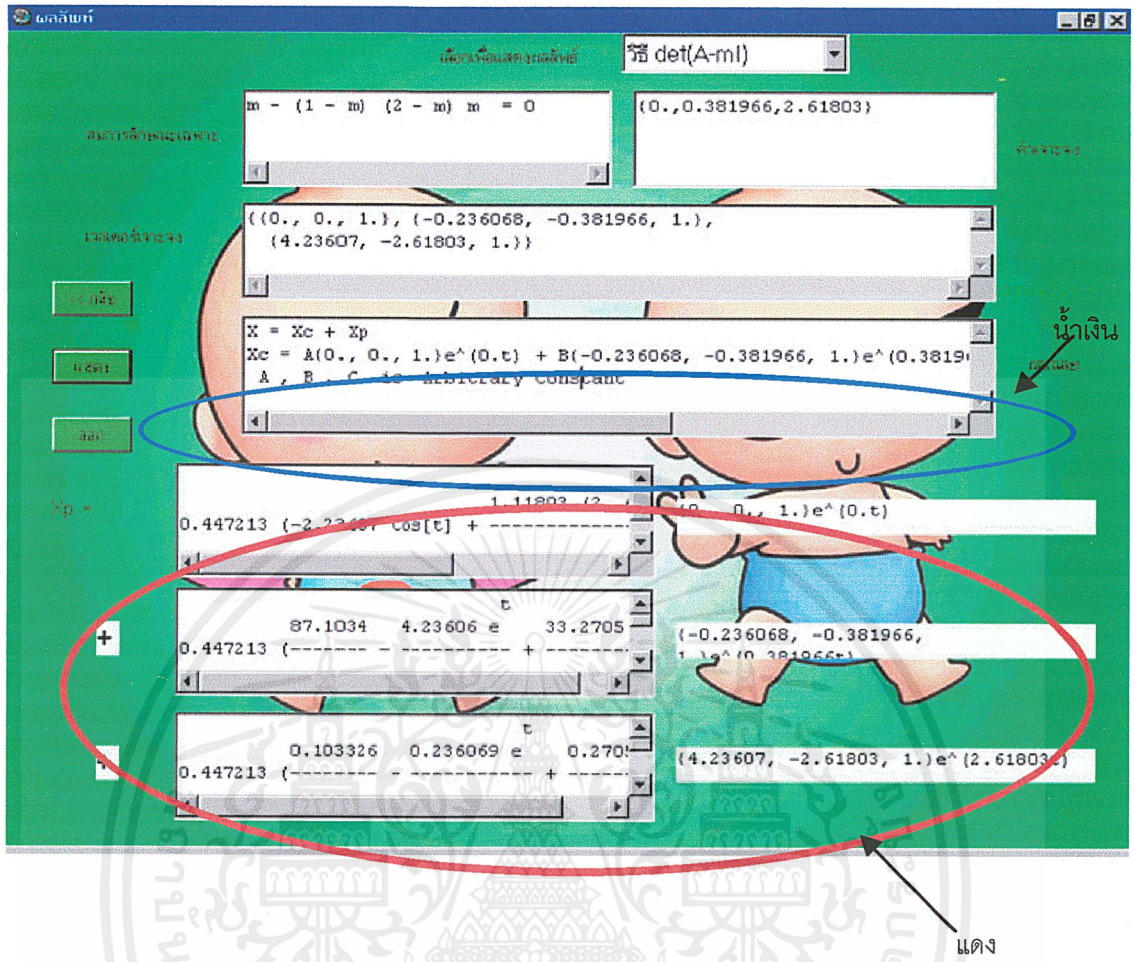
4. ส่วนแสดงผล



รูป 3.10 แสดงค่าเอาท์พุตด้วยวิธีแมกซ์ซิมโปเซอร์ของระบบสมการแบบเอกพันธ์ ขนาด 3x3

- ในวงแดงเป็นสมการลักษณะเฉพาะ
- ในวงเขียวเป็นค่าเจาะจง
- ในวงเหลืองเป็นเวกเตอร์เจาะจง
- ในวงสีน้ำเงินเป็นผลเฉลยของระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่งแบบเอกพันธ์ ด้วยวิธี det(A-mI)

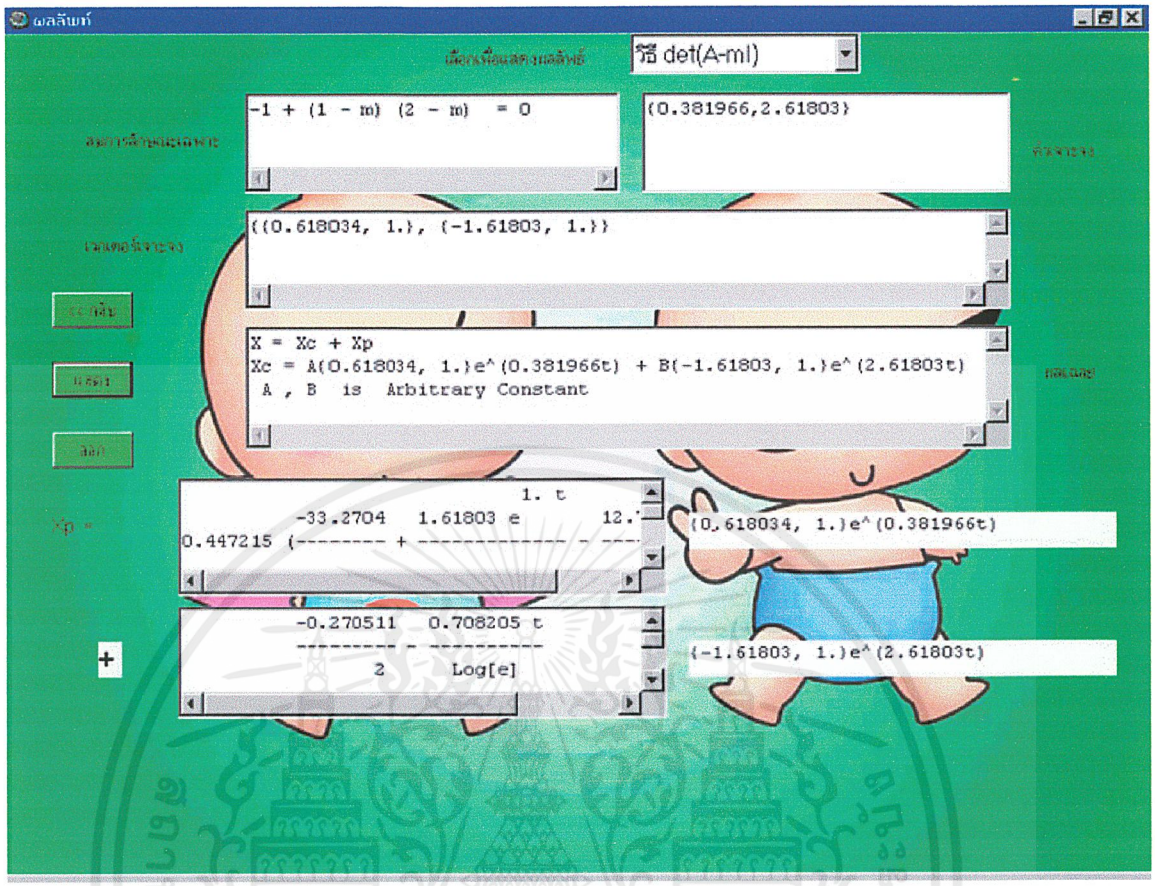
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.11 แสดงค่าเอาท์พุตด้วยวิธี det(A-mI) ของระบบสมการแบบไม่เอกพันธ์ ขนาด 3x3

- ในวงน้ำเงินเป็นค่าของ X_c
- ในวงแดงเป็นค่าของ X_p

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.12 แสดงค่าเอาท์พุตด้วยวิธี det(A-mI) ของระบบสมการแบบไม่เอกพันธ์ ขนาด 2x2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 วิธีการทำงานของโปรแกรม

1. การเชื่อมต่อกับโปรแกรมแมททิเมติกา หลังจากที่ได้ทำการระบุไดเรกเทอรี่ของโปรแกรมแมททิเมติกาแล้ว จะทำการส่งคำสั่ง MLInitialize(0) ไปยังโปรแกรมแมททิเมติกาเพื่อทำการตรวจสอบว่าโปรแกรมพร้อมทำงานหรือไม่ ถ้าค่าที่ส่งกลับมามีค่าไม่เท่ากับก็จะทำการเชื่อมต่อโดยใช้คำสั่ง MLOpenString(ep,"ไดเรกเทอรี่ของแมททิเมติกา" , err) โดยจะเป็นการส่งในรูปแบบของ String

2. ตรวจสอบค่าอินพุตหลังจากกดปุ่ม "คำนวณ"

   2.1 กรณีของระบบสมการแบบเอกพันธ์

   - ตรวจสอบว่าค่าอินพุตที่ใส่จะต้องไม่มีค่าว่าง ถ้าไม่มีค่าก็ใส่ให้ค่าเป็นศูนย์

   2.2 กรณีของระบบสมการแบบไม่เอกพันธ์

   - ตรวจสอบว่าอินพุตที่ใส่จะต้องไม่เป็นค่าว่าง
   - ในช่องที่ใส่ค่าฟังก์ชั่น f(t) , g(t) และ h(t) จะต้องไม่เป็นศูนย์พร้อมกันทั้ง 3 ค่า เพราะจะกลายเป็นระบบเอกพันธ์แทนซึ่งไม่ตรงกับที่เลือกไว้

3. สร้างเมตริกซ์ A จากอินพุตจะมีขนาด 2x2 หรือ 3x3

$$A = \begin{pmatrix} a(1,1) & a(1,2) & a(1,3) \\ a(2,1) & a(2,2) & a(2,3) \\ a(3,1) & a(3,2) & a(3,3) \end{pmatrix}$$

4. คำนวณค่าหลังจากสร้างเมตริกซ์ A เสร็จทำการเลือกวิธี $\det(A - \lambda I)$ หรือวิธีแมกซ์ซิมโปเซอร์จากคอมโบบ็อกซ์ในหน้าแสดงค่าเอาท์พุต

   4.1) เลือกวิธีแมกซ์ซิมโปเซอร์

   4.1.1) สร้างเมตริกซ์ $B = A^2$ และ $C = A^3$ โดยใช้

```
For i = 1 to 2 or 3
    For j = 1 to 2  or 3
        For h = 1 to 2  or 3
            b(i,j) = b(i,j) + (a(i,h) x a(h,j))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าต้องการหา C จะเปลี่ยนสเตทเมนท์เป็น

$$C(i,j) = C(i,j) + (b(i,h) + a(h,j))$$

4.1.2) หาค่า Trace(A), $Tr(A^2)$, $Tr(A^3)$ ก็คือเป็นการหาผลบวกเฉียงของเมตริกซ์โดยที่

$$Tr(A) = a(1,1) + a(2,2) + a(3,3)$$
$$Tr(A^2) = b(1,1) + b(2,2) + b(3,3)$$
$$Tr(A^3) = c(1,1) + c(2,2) + c(3,3)$$

4.1.2) คำนวณหาค่าสัมประสิทธิ์ของสมการลักษณะเฉพาะโดยที่

$$s_1 = Tr(A)\ ,\ s_2 = Tr(A^2)\ ,\ s_3 = Tr(A^3)$$
$$c_1 = -s_1\ ,\ c_2 = -1/2(c_1 c_2 + s_2),\ c_3 = -1/3(c_2 c_1 + c_1 c_2 + s_1)$$

จะได้สมการลักษณะเฉพาะ

$m^2 + c_1 m + c_2 = 0$ หรือ

$m^3 + c_1 m^2 + c_2 m + c_3 = 0$

4.1.3) หาค่าเจาะจงจากสมการลักษณะเฉพาะในข้อ 3.1.3 โดยใช้

Solve[$m^2 + c_1 m + c_2 == 0$] หรือ

Solve[$m^3 + c_1 m^2 + c_2 m + c_3 == 0$]

ซึ่งเป็นคำสั่งในโปรแกรมแมททิเมติกา

4.1.5). เมื่อรับค่าจากคำสั่งที่ส่งไปในข้อ 3.1.4 เราก็จะนำค่านั้นมาแยกเอาส่วนเอาค่าเจาะจงออกมาใส่ตัวแปรก็จะได้เป็นค่าเจาะจงออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่าง {{m -> 1}, {m -> -1}, {m ->2}} เป็นค่าที่ได้รับจากการใช้คำสั่ง Solveในโปรแกรมแมททิเมติกา

4.1.6). ตรวจสอบว่าค่าเจาะจงถ้าเป็นจำนวนเชิงซ้อนให้จบการทำงาน

4.1.7). ตรวจสอบว่าค่าเจาะจงว่ามีค่าซ้ำหรือไม่จะมี 2 แบบคือซ้ำ 2 หรือซ้ำ 3

4.1.7.1) หาค่าของเมตริกซ์ (A – mI) โดย m คือ ค่าเจาะจงของตัวที่ซ้ำ

4.1.7.2) หาค่าเวกเตอร์เจาะจงของตัวที่ซ้ำตัวที่ 2 โดยใช้วิธี

$$(A - mI)P = k$$

ตังอย่าง

$$P = (A - mI)^{-1}K$$

$$\begin{pmatrix} p_1 \\ p_2 \\ p_3 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 2 \\ 2 & -1 & 3 \end{pmatrix} \begin{pmatrix} k_1 \\ k_2 \\ k_3 \end{pmatrix}$$

ถ้า $|A - mI| <> 0$ ก็จะใช้คำสั่ง

Inverse[{a(1,1),a(2,1),a(3,1)},{a(1,2),a(2,2),a(3,2)},{a(1,3),a(2,3),a(3,3)}}]

หาค่า $(A - mI)^{-1}$ แล้วนำ $(A - mI)^{-1}$ มาคูณกับ k แล้วจะได้

$$p_1 = a_1(1,1)k_1 + a_1(1,2)k_2 + a_1(1,2)k_3$$
$$p_1 = a_1(2,1)k_1 + a_1(2,2)k_2 + a_1(3,2)k_3$$
$$p_1 = a_1(3,1)k_1 + a_1(3,2)k_2 + a_1(3,2)k_3$$

ถ้า $|A - mI| = 0$ เราจะใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่าง

$$(A - mI) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 2 \\ 2 & -1 & 3 \end{pmatrix}, \ (A - mI),P = k$$

รอบแรกจะดูว่าในแต่ละแถวมีจำนวนของค่าที่ไม่เท่ากับ 0 มีจำนวน 1 ค่าหรือไม่ ถ้ามีก็จะให้ตำแหน่งของ X ที่ตรงกับตำแหน่งของสัมประสิทธิ์ตัวนั้นมีค่าเป็น k_1

$$1.X_1 + 0 + 0 = k_1 \ \text{->} \ X_1 = k_1$$

รอบที่ 2 หาจำนวนของค่าที่ไม่เท่ากับ 0 มีจำนวน 2 ค่าจะมี 2 กรณี (แต่ถ้าในกรณีที่ค่า 2 ค่านั้นได้ถูกหามาก่อนแล้วในรอบนี้ก็จะไม่ทำการคำนวณอีก)

- กรณีที่ 1 ยังไม่มีค่า X ที่หามาแล้วในรอบแรกก็จะทำการกำหนดให้ค่า X ตัวใดตัวหนึ่งใน 2 ค่านั้นเท่ากับ 0 แล้วค่าอีกตัวจะเท่ากับ

$$0 + 2X_2 = k \ \text{->} \ X_2 = k$$

- กรณีที่ 2 มีค่าของ X บางค่าใน 2 ค่านั้นได้หามาแล้วในรอบแรก ให้นำค่าตัวนั้นมาใช้ในการคำนวณด้วย

รอบที่ 3 หาจำนวนค่าที่ไม่เท่ากับ 0 มีจำนวน 3 ค่า จะมี 3 กรณี

- กรณีที่ 1 มีค่าของ X ที่ยังไม่ได้หาทั้ง 3 ค่า จำทำการกำหนดค่าของ X 2 ค่าจาก 3 ค่ามีค่าเท่ากับ 0 และ 1 ส่วนค่าที่เหลือก็จะได้จากการคำนวณโดยการแทนค่าของ X ที่กำหนดไว้ลงในสมการ
- กรณีที่ 2 มีค่าที่ยังไม่ได้คำนวณ 1 ค่าทำการคำนวณได้เลย
- กรณีที่ 3 มีค่าที่ยังไม่ได้คำนวณ 2 ค่า จะทำการกำหนดค่าใดค่าหนึ่งเป็น 1

4.1.8) ได้ผลเฉลยแบบเอกพันธ์หรือ x_c ในแบบไม่เอกพันธ์

$$x = Avector(1)e^{value(1)t} + Bvector(2)e^{value(2)t} \quad \text{หรือ}$$

$$x = Avector(1)e^{value(1)t} + Bvector(2)e^{value(2)t} + Cvector(3)e^{value(3)t}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.9) ถ้าเป็นระบบสมการแบบไม่เอกพันธ์จะทำตั้งแต่ข้อ 3.1.1 ถึง3.1.8 ก็จะได้ค่าสมการลักษณะเฉพาะ , ค่าเจาะจง , เวกเตอร์เจาะจง และ X_C

4.1.10) หาค่า $a_1'(t)$, $a_2'(t)$ และ $a_3'(t)$ โดยการใช้กฎของคราเมอร์หาเพื่อหาสมการ x_p

$$a_1'(t) = \frac{\begin{vmatrix} f(t) & m(1,2) \\ g(t) & m(2,2) \end{vmatrix}}{\begin{vmatrix} m(1,1) & m(1,2) \\ m(2,1) & m(2,2) \end{vmatrix}} \quad \text{และ} \quad a_2'(t) = \frac{\begin{vmatrix} m(1,1) & f(t) \\ m(2,1) & g(t) \end{vmatrix}}{\begin{vmatrix} m(1,1) & m(1,2) \\ m(2,1) & m(2,2) \end{vmatrix}}$$

โดยจะใช้คำสั่ง A=Det[{{ f_t , m(1, 2) },{ g_t , m(2, 2) }}] , B=Det[{{m(1, 1) , f_t },{ m(2, 1) , g_t }}] , L=Det[{{ m(1, 1) , m(2, 1) },{ m(1, 2) & ", m(2, 2) }}] , a_1'(t) = A = (A*e^(- (values(1) t))/L) และ a_2'(t) = B = (B*e^(- (values(2) t))/L)

- ถ้า det ของตัวหารมีค่าเท่ากับ 0 ให้จบการทำงานเพราะตัวหารเป็น 0 ไม่ได้ในสูตรของคราเมอร์
- ถ้า det < > 0 นำค่า $a_1'(t)$, $a_2'(t)$, $a_3'(t)$ ที่ได้จากวิธีของคราเมอร์มาใส่ในคำสั่ง

Integrate[$a_1'(t)$,t] และ Expand[%] (ทำการแจกแจงค่าในคำตอบหลังจากทำการ Integrate) โดยค่าที่ได้จะเป็นค่าของ $a_1'(t)$, $a_2'(t)$ และ $a_3'(t)$

4.1.11) จัดรูปแบบของค่า x_p โดย

$$x_ย = a_1(t)\,\text{vector}(1)e^{\text{value}(1)t} + a_2(t)\text{vector}(2)e^{\text{value}(2)t} \quad \text{หรือ}$$

$$x_ย = a_1(t)\,\text{vector}(1)e^{\text{value}(1)t} + a_2(t)\text{vector}(2)e^{\text{value}(2)t} + a_3(t)\,\text{vector}(3)e^{\text{value}(3)t}$$

4.2) เลือกวิธี det(A – mI)

4.2.1) กรณีระบบสมการแบบเอกพันธ์

4.2.1.1) หาสมการลักษณะเฉพาะโดยใช้ | A – mI| = 0 แล้วส่งค่าที่ได้ไปจัดรูปที่โปรแกรมแมททิเมติกา

4.2.1.2) หาค่าเจาะจงโดยใช้คำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Eigenvalues[{{a(1,1),a(2,1),a(3,1)},{a(2,1),a(2,2),a(2,3)},{a(3,1),a(3,2),a(3,3)}}]

โดยจะส่งไปคำนวณที่โปรแกรมแมททิเมติกาซึ่งจะอยู่ในรูป

{{m = -1},{m = 2},{m=1}}

แล้วทำการตัดผลที่ได้เอาเฉพาะค่าเจาะจงใส่ในตัวแปร

4.2.1.3) ตรวจสอบค่าเจาะจงว่าเป็นจำนวนเชิงซ้อนทำโดยการหาค่า I จากผลที่ได้จากข้อ 3.2.1.2 ถ้าพบก็จบการทำงาน

4.2.1.4) ตรวจสอบค่าเจาะจงซ้ำว่าทำงานเหมือนหัวข้อ 3.1.7

4.2.1.5) จะได้ผลเฉลยของระบบสมการแบบเอกพันธ์หรือค่า x_c ในระบบสมการแบบไม่เอกพันธ์เหมือนข้อ 3.1.8

4.2.2) กรณีระบบสมการแบบไม่เอกพันธ์

จะทำตามหัวข้อ 3.2.1.1 ถึง 3.2.1.5 จะได้ สมการลักษณะเฉพาะ , ค่าเจาะจง , เวกเตอร์เจาะจง และ x_c

ส่วน x_p จะทำเหมือนหัวข้อ 3.1.10

5. แสดงค่าเอาท์พุตที่ได้จากการคำนวณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การอภิปรายผล

ผลลัพธ์ที่ได้จากการศึกษาและทำการสร้างโปรแกรมสำเร็จรูปแสดงความสัมพันธ์ของผลเฉลยของระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่ง โดยวิธีทางเมทริกซ์สามารถประเมินผลได้ดังนี้

4.1 ส่งเสริมให้นักศึกษาเกิดความเข้าใจในเรื่องการหาผลเฉลยระบบสมการเชิงอนุพันธ์สามัญเชิงเส้น

การเรียนการสอนในวิชาพีชคณิตเชิงเส้นเรื่องการประยุกต์ในระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่งนั้น นักศึกษาจะต้องทำความเข้าใจใจเรื่องการสมการลักษณะเฉพาะค่าเจาะจงและเวกเตอร์เจาะจงก่อน จึงนำมาประยุกต์ใช้ในเรื่องการหาผลเฉลยระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่ง เนื่องจากนักศึกษาต้องทำความเข้าใจในหลายเรื่องและการคำนวณค่อนข้างจะยาก ด้วยสาเหตุนี้จึงได้นำคอมพิวเตอร์มาใช้ในการเรียนการสอนในวิชาพีชคณิตเชิงเส้นเรื่องของการประยุกต์ในระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่ง ซึ่งจะทำให้นักศึกษาสามารถศึกษาและเข้าใจใน การหาผลเฉลยของระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่งได้ดีมากขึ้น

4.2 ด้านการใช้งานและความเข้าใจ

เนื่องจากโปรแกรมที่ได้จัดทำขึ้นนี้เป็น โปรแกรมที่ใช้งานบนระบบปฏิบัติการวินโดว์ผู้ใช้สามารถเลือกคำสั่งการทำงานต่างๆ ของโปรแกรมได้โดยการใช้เม้าส์และยังสามารถเข้าใจในด้านการใช้งานได้ง่าย

4.3 ข้อเสนอแนะที่ควรแก้ไข

1) เพิ่มเติมวิธีการหาผลเฉลย ของระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่งด้วยวิธีอื่นๆ เช่น วิธีเมทริกซ์เฉียง เป็นต้น
2) สามารถทำการพิมพ์ค่าเอาต์พุตออกทางเครื่องพิมพ์ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) สามารถรับค่าสัมประสิทธิ์ที่มีค่าเป็นจำนวนอตรรกยะและเศษส่วนได้

4) สามารถรับค่าอินพุตของระบบสมการที่มีขนาดของเมทริกซ์มากกว่า 3 x 3 ได้

5) ไม่จำเป็นต้องมีโปรแกรมแมททิเมติกาในเครื่องคอมพิวเตอร์ก็สามารถหาผลเฉลยได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการวิจัยและเสนอแนะ

5.1 สรุปผลการวิจัย

ปัญหาพิเศษที่จัดทำขึ้นนี้ได้จัดทำ โปรแกรมหาผลเฉลยของระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่งของวิชาพีชคณิตเชิงเส้น ซึ่งโปรแกรมนี้จะแสดงผลเฉลยของระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่ง ทำการหาสมการลักษณะเฉพาะ ค่าเจาะจง เวกเตอร์เจาะจง และจะนำมาซึ่งผลเฉลยของระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นอันดับหนึ่ง โดยที่โปรแกรมนี้ได้สร้างขึ้นมาจากโปรแกรมวิชวลเบซิก 6 ทั้งในส่วนของการรับค่าอินพุตและเอาท์พุต เมื่อทำการรับค่าอินพุตก็จะนำค่าอินพุตนั้นมาทำการคำนวณโดยทำการส่งค่าไปยังโปรแกรมแมททิเมติกา ซึ่งได้ทำการลิงค์ระหว่างโปรแกรมวิชวลเบซิก 6 และโปรแกรมแมททิเมติกาไว้แล้ว จากนั้นจึงทำการแสดงผลเฉลยลงบนหน้าจอเอาท์พุต แต่โปรแกรมนี้มีข้อจำกัดหลายอย่างเช่น จำเป็นต้องมีโปรแกรมแมททิเมติกาอยู่บนเครื่องคอมพิวเตอร์นั้น ไม่สามารถรับค่าอินพุตเป็นจำนวนอตรรกยะและเศษส่วนได้

5.2 ข้อเสนอแนะ

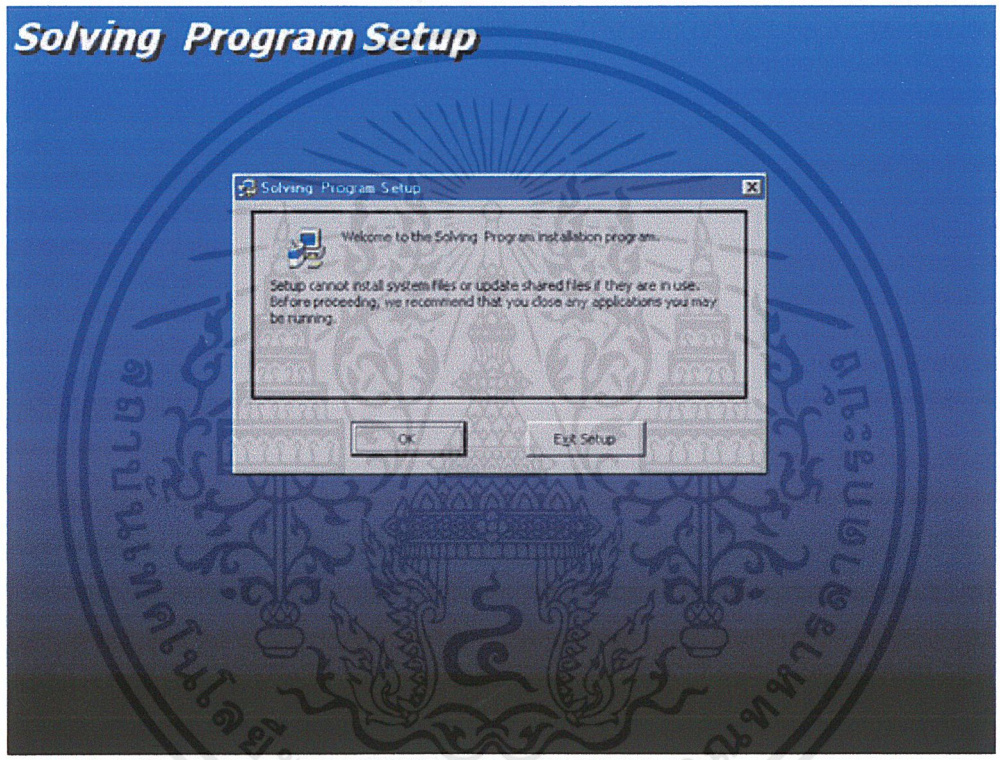
เนื่องจากปัญหาพิเศษในหัวข้อการประยุกต์ในระบบสมการเชิงอนุพันธ์สามัญเชิงเส้นซึ่งเป็นส่วนหนึ่งของวิชาพีชคณิต ดังนั้นโปรแกรมนี้จึงเหมาะที่จะนำไปใช้ในการเรียนการสอนในวิชาพีชคณิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

วิธีการติดตั้งโปรแกรม Matrix.exe

1) กดปุ่ม setup จะแสดงหน้าจอ

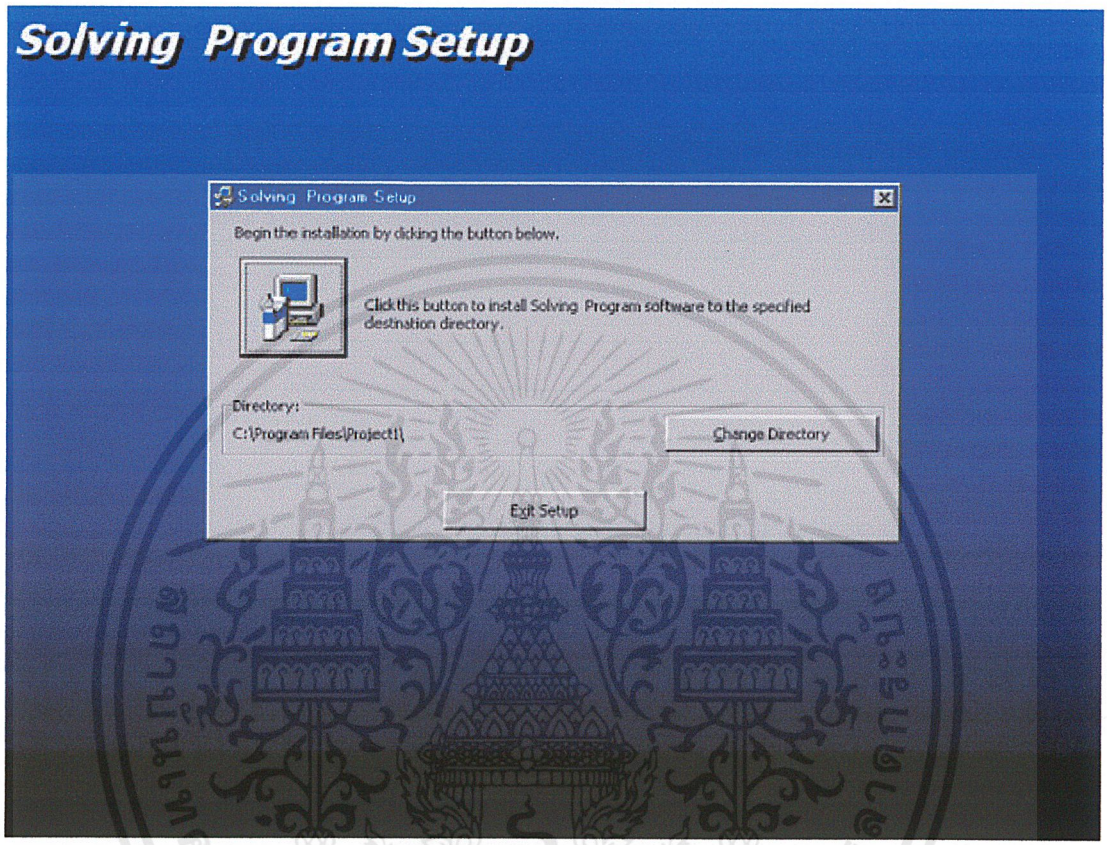


รูปที่ ผ.1 แสดงหน้าจอแรกของการติดตั้ง

แล้วกดปุ่ม “OK”

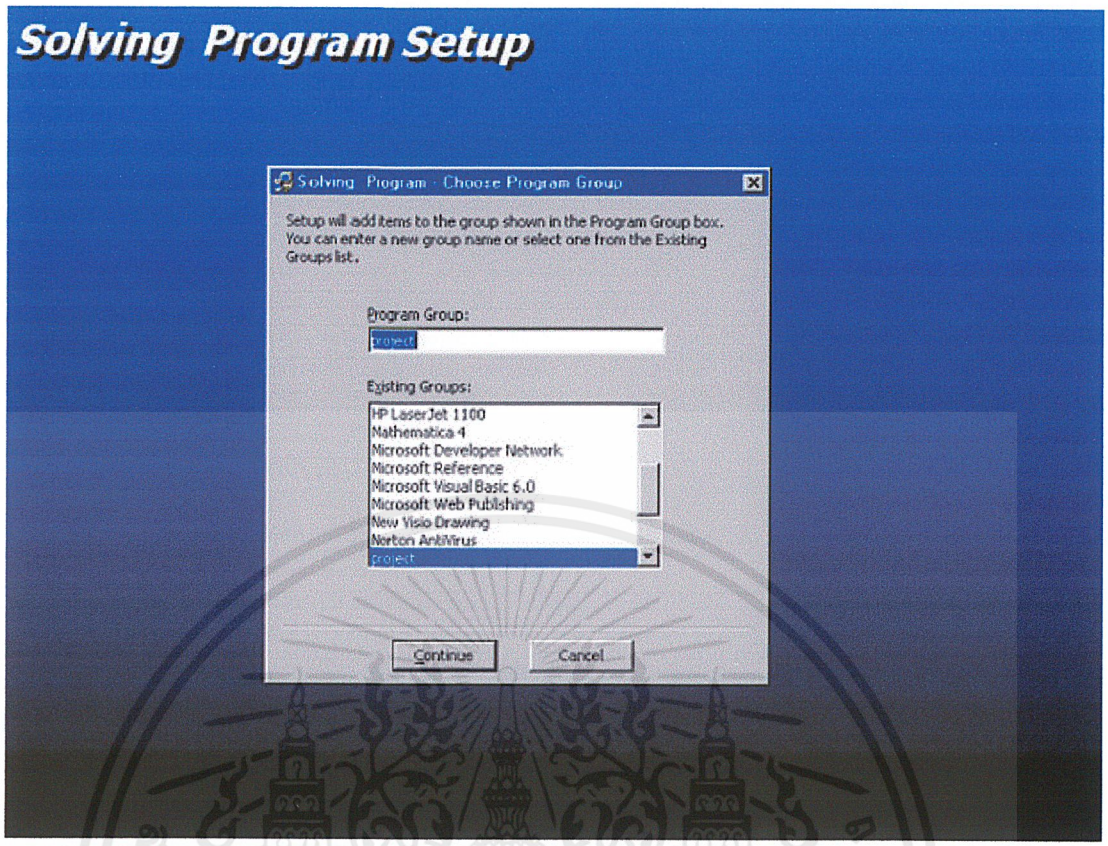
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) แสดงหน้าจอที่ให้ทำการระบุไดเรกเทอรี่ ถ้าต้องการเปลี่ยนไดเรกเทอรี่ให้กดปุ่ม “CHANGE DIRECTORY” เมื่อระบุเสร็จแล้วให้กดปุ่ม จะปรากฏหน้าจอ



รูปที่ ผ.2 หน้าจอแสดงการระบุไดเรกเทอรี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

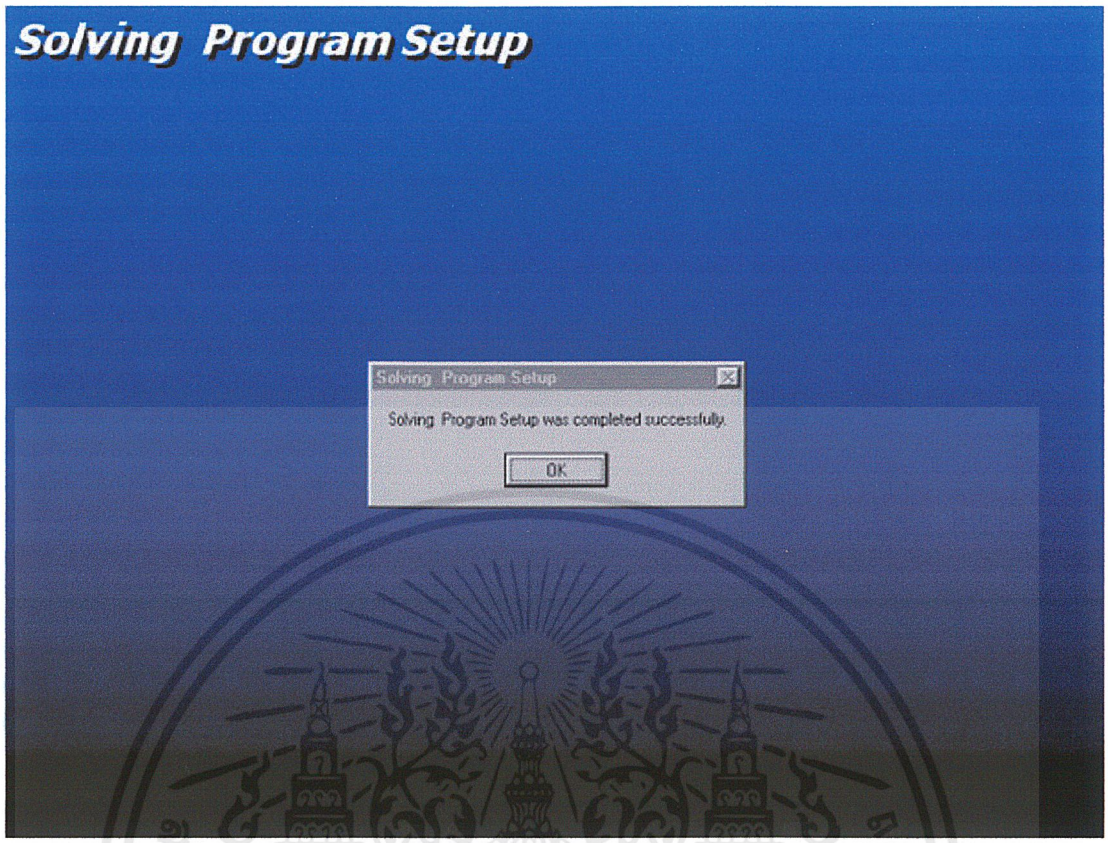


รูปที่ ผ.3 แสดงขั้นตอนการติดตั้ง

ให้กดปุ่ม "Continue"

3) เมื่อเสร็จการติดตั้งจะแสดงหน้าจอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ผ.4 เสร็จสิ้นการติดตั้ง

ให้กดปุ่ม "OK" เพื่อเสร็จการติดตั้งโปรแกรม Matrix.exe

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

แสดงชุดคำสั่งของโปรแกรม

1. หน้าแสดงชื่อโปรแกรม
   - เมื่อกดที่ Label1 จะแสดงหน้าต่างที่จะทำการเชื่อมต่อ Mathkernal

```
Private Sub Label1_Click()
    Load Form4
    Form4.Show
    Unload Form3
End Sub
```

- เมื่อกดที่ Label2 จะแสดงหน้าต่างที่จะทำการเชื่อมต่อ Mathkernal

```
Private Sub Label2_Click()
    Load Form4
    Form4.Show
    Unload Form3
End Sub
```

- เมื่อกดที่ Label3 จะแสดงหน้าต่างที่จะทำการเชื่อมต่อ Mathkernal

```
Private Sub Label3_Click()
    Load Form4
    Form4.Show
    Unload Form3
End Sub
```

2. หน้าจอทำการเชื่อมต่อและรับค่าอินพุต
   - เมื่อกดปุ่ม “ติดต่อ”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Command1_Click()
    Dim i As Integer
    If connected = False Then
        DoConnect
    End If
If connected = True Then
    Frame1.Visible = True
    Frame3.Visible = True
    Label1.Visible = True
    Label3.Visible = True
    Label4.Visible = True
    Label8.Visible = True
    Label9.Visible = True
    Label10.Visible = True
    Label11.Visible = True
    Label12.Visible = True
    Label13.Visible = True
    Label14.Visible = True
    Label15.Visible = True
    Label16.Visible = True
    Label17.Visible = True
    Label18.Visible = True
    Label19.Visible = True
    Label20.Visible = True
    Label21.Visible = True
    Label22.Visible = True
    Label23.Visible = True
    Label24.Visible = True
    Label25.Visible = True
    Label7(0).Visible = True
    Label7(1).Visible = True
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Label7(2).Visible = True
        For i = 0 To 11
          Text1(i).Visible = True
        Next i
        Option1.Visible = True
        Option2.Visible = True
        Check1.Visible = True
        Command3.Visible = True
    End If
End Sub
```

- โปรแกรมย่อยที่ทำการเชื่อมต่อกับโปรแกรมแมทเคอร์แนล

```
Sub DoConnect()
    Dim initstring As String
    inistring = ""
    If connected = False Then
      ep = MLInitialize(0)
      If (ep = 0) Then
        MsgBox ("MLInitialize failed")
      Else
        link = MLOpenString(ep, Form4.Text1(12).Text, err)
        If (err <> 0) Then
          MsgBox ("Error: " & err)
        Else
          res = MLConnect(link)
        End If
      End If
      If (res <> 0) Then
        connected = True
        res = MLDoCompute("")
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            res = MLPutFunction(link, "EvaluatePacket", 1)
            res = MLPutFunction(link, "ToExpression", 1)
            initstring = "$Display := OpenTemporary[];" & Chr(13) & Chr(10)
            initstring = initstring & "$DisplayFunction = (Module[{tmpstream = $Display,
tmpfile}, tmpfile = tmpstream[[1]]; Close[tmpstream]; Display[tmpfile, #1, " & Chr(34) &
"Metafile" & Chr(34) & "]; LinkWrite[$ParentLink, DisplayEndPacket[tmpfile]]]; #1)&;"
& Chr(13) & Chr(10)
            initstring = initstring & "SetOptions[{ContourGraphics, ContourPlot,
DensityGraphics, DensityPlot, Graphics, Graphics3D, GraphicsArray, ListContourPlot,
ListDensityPlot, ListPlot, ListPlot3D, ParametricPlot, ParametricPlot3D, Plot, Plot3D,
SurfaceGraphics}, AspectRatio -> 2];" & Chr(13) & Chr(10)
            initstring = initstring & "SetOptions[ " & Chr(34) & "stdout" & Chr(34) & ",
PageWidth->65];" & Chr(13) & Chr(10)
            initstring = initstring & "Format[LineBreak[_]]:= " & Chr(34) & Chr(34) & ";" &
Chr(13) & Chr(10)
            initstring = initstring & "Format[Continuation[_]]:= " & Chr(34) & " " & Chr(34)
& ";"
            res = MLPutString(link, initstring)
        End If
    End If
End Sub
```

- ฟังก์ชั่นของโปรแกรมย่อย “DoConnect”

```
Function MLDoCompute(ByVal buff As String) As Integer
    Static inputMode
    Dim pkt As Integer
    If link = 0 Then
        MLDoCompute = 0
    End If
    If buff <> "" Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    If inputMode = False Then
      res = MLPutFunction(link, "EnterTextPacket", 1)
    Else
      res = MLPutFunction(link, "TextPacket", 1)
      inputMode = False
    End If
    res = MLPutString(link, buff)
    If MLError(link) <> 0 Then
      MLDoCompute = 0
    End If
  End If
  Do While done = False
    pkt = MLNextPacket(link)
    Select Case pkt
      Case ILLEGALPKT
        done = True
      Case INPUTNAMEPKT, INPUTPKT
        done = True
      Case RETURNTEXTPKT
        out = get_text
        out = filterme(out)
      Case TEXTPKT
        done = True
      Case INPUTSTRPKT
        done = True
      Case SUSPENDPKT
        res = MLNewPacket(link)
      Case RESUMEPKT
        res = MLNewPacket(link)
    End Select
    res = MLNewPacket(link)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        If MLError(link) <> 0 Then
          done = True
          MLDoCompute = 0
        End If
      Loop
      MLDoCompute = pkt
    End Function
```

- แสดงหน้าการใช้งานของโปรแกรมโดยใช้ “Internet Explore”

```
Public Explorer As SHDocVw.InternetExplorer
Private Sub use_Click()
  On Error GoTo error1
  Set Explorer = New SHDocVw.InternetExplorer
  Explorer.Visible = True
  Explorer.Navigate "C:\WINDOWS\Desktop\การใช้งานโปรแกรม.htm"
  Exit Sub
error1:
  MsgBox "ไม่สามารถเปิดเอกสารได้", vbOKOnly, "ผิดพลาด"
End Sub
```

- แสดงรายชื่อคณะผู้จัดทำ

```
Private Sub about_Click()
  Form1.Show
  Form4.Hide
End Sub
```

- ออกจากโปรแกรม

```
Private Sub exit_Click()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    If MsgBox("ต้องการออกจากโปรแกรมใช่หรือไม่", vbYesNo, "ยืนยัน") = 6 Then
      If connected = True Then
        DoDisconnect
      End If
      End
    End If
End Sub
```

- โปรแกรมย่อยทำการยกเลิกการเชื่อมต่อกับโปรแกรมแมทเคอร์แนล

```
Sub DoDisconnect()
   If link <> 0 Then
      MLClose (link)
   End If
   If ep <> 0 Then
      MLDeinitialize (ep)
   End If
   connected = False
End Sub
```

- เมื่อกดปุ่ม “คำนวณ”

```
Private Sub Command3_Click()
   If (Option1.Value = True) Then
      If Check1.Value = Checked Then
         If ((Text1(3).Text <> "") And (Text1(7).Text <> "") And (Text1(11).Text <> "")
And (Text1(0).Text <> "") And (Text1(1).Text <> "") And (Text1(2).Text <> "") And
(Text1(4).Text <> "") And (Text1(5).Text <> "") And (Text1(6).Text <> "") And (Text1
(8).Text <> "") And (Text1(9).Text <> "") And (Text1(10).Text <> "")) Then
            GoTo con
         Else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            MsgBox "ใส่ค่าไม่ครบ", vbOKOnly, "ผิดพลาด"
            GoTo out
          End If
        ElseIf Check1.Value = Unchecked Then
          If ((Text1(0).Text <> "") And (Text1(1).Text <> "") And (Text1(4).Text <> "")
And (Text1(5).Text <> "")) Then
            GoTo con
          Else
            MsgBox "ใส่ค่าไม่ครบ", vbOKOnly, "ผิดพลาด"
            GoTo out
          End If
        End If
      ElseIf (Option2.Value = True) Then
        If Check1.Value = Checked Then
          If ((Text1(3).Text <> "") And (Text1(7).Text <> "") And (Text1(11).Text <> "")
And (Text1(0).Text <> "") And (Text1(1).Text <> "") And (Text1(2).Text <> "") And
(Text1(4).Text <> "") And (Text1(5).Text <> "") And (Text1(6).Text <> "") And (Text1
(8).Text <> "") And (Text1(9).Text <> "") And (Text1(10).Text <> "")) Then
            If ((Text1(3).Text = "0") And (Text1(7).Text = "0") And (Text1(11).Text = "0"))
Then
              MsgBox "ใส่ค่าไม่ถูกต้อง", vbOKOnly, "ผิดพลาด"
              GoTo out
            Else
              GoTo con
            End If
          Else
            MsgBox "ใส่ค่าไม่ครบ", vbOKOnly, "ผิดพลาด"
            GoTo out
          End If
        ElseIf Check1.Value = Unchecked Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        If ((Text1(0).Text <> "") And (Text1(1).Text <> "") And (Text1(4).Text <> "")
    And (Text1(5).Text <> "")) Then
            If ((Text1(3).Text = "0") And (Text1(7).Text = "0") And (Text1(11).Text = "0"))
    Then
              MsgBox "ใส่ค่าไม่ถูกต้อง", vbOKOnly, "ผิดพลาด"
              GoTo out
            Else
              GoTo con
            End If
          Else
            MsgBox "ใส่ค่าไม่ครบ", vbOKOnly, "ผิดพลาด"
            GoTo out
          End If
        End If
      End If
    con:
      If Option2.Value = True Then
        f_t = Text1(3).Text
        g_t = Text1(7).Text
        If Check1.Value = Checked Then
          h_t = Text1(11).Text
        End If
      End If
        Form5.Show
        Form4.Hide
    out:
    End Sub
```

3. หน้าต่างแสดงค่าเอาท์พุต
   - กลับไปยังหน้าต่างรับค่าอินพุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Command1_Click()
   Form4.Show
   Unload Me
End Sub
```

- ออกจากโปรแกรม

```
 Private Sub Command2_Click()
   If MsgBox("ต้องการออกจากโปรแกรมใช่หรือไม่", vbYesNo, "ยืนยัน") = 6 Then
      If connected = True Then
         DoDisconnect
      End If
      End
   End If
End Sub
```

- เมื่อกดปุ่ม “แสดง”

```
Private Sub Command3_Click()
   Dim cc1, cc2, cc3, Inverse, m2(1 To 3), pack, m3(1 To 3), allvector(1 To 3), chrac1,
chrac2, values_S As String
   Dim i, j, vector_R, vector_R1, vector_R2, vector_R3, values_R As Integer
   For i = 1 To 3
      vector(i) = ""
      values(i) = ""
      output(i) = ""
      pack = ""
   Next i
   Clearoutput
   If Combo1.Text = "" Then
      MsgBox "ยังไม่ได้เลือกวิธี", vbOKOnly, "ผิดพลาด"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    GoTo out
  Else
    If sizeA3x3 = True Then
      k = 3
    Else
      k = 2
    End If
    Clearmatrix
    CreateA
    res = MLDoCompute(chrac)
    normal1 = out
    res = MLDoCompute(eiA)
    normal2 = out
    Findnumber (normal2)
    k = 3
    R = 0
    If sizeA3x3 = False Then
      k = 2
      values(3) = "a"
    End If
    For i = 1 To k
      values(i) = output(i)
      values(i) = "N[" & values(i) & "]"
      res = MLDoCompute(values(i))
      values(i) = out
      vector(i) = ""
    Next i
    If sizeA3x3 = True Then
      normal2 = "{" & values(1) & "," & values(2) & "," & values(3) & "}"
    Else
      normal2 = "{" & values(1) & "," & values(2) & "}"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End If
If (values(1) = values(2)) And (values(2) = values(3)) Then
   R = 3
ElseIf ((values(1) = values(2)) And (values(2) <> values(3))) Then
   R = 2
ElseIf ((values(3) = values(2)) And (values(1) <> values(3))) Then
   R = 2
End If
res = MLDoCompute(eivA)
normal3 = out
Findvectors (normal3)
For i = 1 To k
   vector(i) = vector1(i)
   vector(i) = "N[" & vector(i) & "]"
   res = MLDoCompute(vector(i))
   vector(i) = out
Next i
If (Combo1.Text = "วิธีแมกซ์ซิมโปเซอร์") Then
   Maxime
   cc1 = c1
   cc2 = c2
   cc3 = c3
   If sizeA3x3 = False Then
      Text1.Text = "m^2 + (" & cc1 & ")m + (" & cc2 & ") = 0"
   Else
      Text1.Text = "m^3 + (" & cc1 & ")m^2 + (" & cc2 & ")m + (" & cc3 & ") = 0"
   End If
   Solve
   Maxim_values
   For i = 1 To 3
      check_v(i) = values_m(i)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            Next i
            If Check_I = True Then
              MsgBox "ค่าเจาะจงเป็นจำนวนเชิงซ้อน", vbOKOnly, "รูปแบบไม่ถูกต้อง"
              GoTo out
            End If
            Maxim_vectors1
            Show_vector
            If sizeA3x3 = False Then
              allvector(1) = "{" & vector_m(1, 1) & "," & vector_m(1, 2) & "}"
              allvector(2) = "{" & vector_m(2, 1) & "," & vector_m(2, 2) & "}"
           Else
allvector(1) = "{" & vector_m(1, 1) & "," & vector_m(1, 2) & "," & vector_m(1, 3) & "}"
allvector(2) = "{" & vector_m(2, 1) & "," & vector_m(2, 2) & "," & vector_m(2, 3) & "}"
allvector(3) = "{" & vector_m(3, 1) & "," & vector_m(3, 2) & "," & vector_m(3, 3) & "}"
            End If
            If (R = 2) And (sizeA3x3 = False) Then
              vector_m(1, 3) = "0"
              Repeated vector_m(1, 1), vector_m(1, 2), vector_m(1, 3), values_m(1)
              For i = 1 To 2
                vector_m(2, i) = vector_m(1, i) & "t+(" & Ans(i) & ")"
              Next i
              allvector(1) = "{" & vector_m(1, 1) & "," & vector_m(1, 2) & "}"
              allvector(2) = "{" & vector_m(1, 1) & "t+(" & Ans(1) & ")," & vector_m(1, 2) &
"t+(" & Ans(2) & ")}"
              Text3.Text = "{" & allvector(1) & "," & allvector(2) & "}"
           End If
           If (R = 2) And (sizeA3x3 = True) Then
             If values_m(1) = values_m(2) Then
               values_R = 1
               For i = 1 To 3
                 output(i) = vector_m(1, i)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            Next i
            vector_R = 2
            vector_R1 = 1
            vector_R2 = 3
        ElseIf values_m(3) = values_m(2) Then
            values_R = 2
            For i = 1 To 3
                output(i) = vector_m(2, i)
            Next i
            vector_R = 3
            vector_R1 = 2
            vector_R2 = 1
        ElseIf values_m(1) = values_m(3) Then
            values_R = 1
            For i = 1 To 3
                output(i) = vector_m(3, i)
            Next i
            vector_R = 3
            vector_R1 = 1
            vector_R2 = 2
        End If
        values_S = values_m(values_R)
        Repeated output(1), output(2), output(3), values_S
        For i = 1 To 3
            vector_m(vector_R, i) = output(i) & "t+(" & Ans(i) & ")"
        Next i
        allvector(vector_R2) = "{" & vector_m(vector_R2, 1) & "," & vector_m
(vector_R2, 2) & "," & vector_m(vector_R2, 3) & "}"
        allvector(vector_R1) = "{" & vector_m(vector_R1, 1) & "," & vector_m
(vector_R1, 2) & "," & vector_m(vector_R1, 3) & "}"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            allvector(vector_R) = "{" & output(1) & "t+(" & Ans(1) & ")," & output(2) &
"t+(" & Ans(2) & ")," & output(3) & "t+(" & Ans(3) & ")}"
            Text3.Text = "{" & allvector(vector_R2) & "," & allvector(vector_R1) & "," &
allvector(vector_R) & "}"
        End If
        If (R = 3) Then
            Repeated vector_m(1, 1), vector_m(1, 2), vector_m(1, 3), values_m(1)
            For i = 1 To 3
               vector_m(2, i) = vector_m(1, i) & "t+(" & Ans(i) & ")"
             vector_m(3, i) = vector_m(1, i) & "(t^2)/2+(" & Ans(i) & ")t+(" & Ans1(i) & ")"
            Next i
 allvector(1) = "{" & vector_m(1, 1) & "," & vector_m(1, 2) & "," & vector_m(1, 3) & "}"
            allvector(2) = "{" & vector_m(1, 1) & "t+(" & Ans(1) & ")," & vector_m(1, 2) &
"t+(" & Ans(2) & ")," & vector_m(1, 3) & "t+(" & Ans(3) & ")}"
            allvector(3) = "{" & vector_m(1, 1) & "(t^2)/2+(" & Ans(1) & ")t+(" & Ans1(1)
& ")," & vector_m(1, 1) & "(t^2)/2+(" & Ans(2) & ")t+(" & Ans1(2) & ")," & vector_m(1,
1) & "(t^2)/2+(" & Ans(3) & ")t+(" & Ans1(3) & ")}}"
            Text3.Text = "{" & allvector(1) & "," & allvector(2) & "," & allvector(3) & "}"
        End If
        If sizeA3x3 = False Then
            Text4.Text = "X = A" & allvector(1) & "e^(" & values_m(1) & "t) + B" &
allvector(2) & "e^(" & values_m(2) & "t)" & CRLF & CRLF & " A , B  is  Arbitrary
Constant "
        Else
            Text4.Text = "X = A" & allvector(1) & "e^(" & values_m(1) & "t) + B" &
allvector(2) & "e^(" & values_m(2) & "t) + C" & allvector(3) & "e^(" & values_m(3) &
"t)" & CRLF & CRLF & " A , B , C  is  Arbitrary Constant "
        End If
        If Form4.Option2.Value = True Then
            openXp
            For i = 1 To k
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        For j = 1 To k
            m(j, i) = vector_m(i, j)
        Next j
        values_1(i) = values_m(i)
    Next i
    FindInverse allvector(1), allvector(2), allvector(3)
        pack = "Integrate[A,t]"
        res = MLDoCompute(pack)
        pack = "Expand[%*e^(" & values_m(1) & "t)]"
        res = MLDoCompute(pack)
        pack = "A=Expand[(%*" & allvector(1) & ")]"
        res = MLDoCompute(pack)
        pack = "Integrate[B,t]"
        res = MLDoCompute(pack)
        pack = "Expand[%*e^(" & values_m(2) & "t)]"
        res = MLDoCompute(pack)
        pack = "B=Expand[(%*" & allvector(2) & ")]"
        res = MLDoCompute(pack)
        If sizeA3x3 = True Then
            pack = "Integrate[C,t]"
            res = MLDoCompute(pack)
            pack = "Expand[%*e^(" & values_m(3) & "t)]"
            res = MLDoCompute(pack)
            pack = "C=Expand[(%*" & allvector(3) & ")]"
            res = MLDoCompute(pack)
        End If
        If sizeA3x3 = False Then
            Text4.Text = "X = Xc + Xp" & CRLF & "Xc = A" + allvector(1) + "e^(" +
values_m(1) + "t) + B" + allvector(2) + "e^(" + values_m(2) + "t)" & CRLF & " A , B  is
Arbitrary Constant "
            pack = "Expand[A+B]"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                res = MLDoCompute(pack)
                Text5.Text = out
            Else
                Text4.Text = "X = Xc + Xp" & CRLF & "Xc = A" + allvector(1) + "e^(" +
values_m(1) + "t) + B" + allvector(2) + "e^(" + values_m(2) + "t) + C" + allvector(3) + "e^
(" + values_m(3) + "t)" & CRLF & " A , B , C  is  Arbitrary Constant "

                pack = "Expand[A+B+C]"
                res = MLDoCompute(pack)
                Text5.Text = out
            End If
        End If
    ElseIf (Combo1.Text = "วิธี det(A-mI)") Then
        Text1.Text = normal1 + "  = 0"
        Text2.Text = normal2
        For i = 1 To k
          check_v(i) = values(i)
        Next i
        If Check_I = True Then
          MsgBox "ค่าเจาะจงเป็นจำนวนเชิงซ้อน", vbOKOnly, "รูปแบบไม่ถูกต้อง"
          GoTo out
        End If
        If (R = 2) And (sizeA3x3 = False) Then
          Findnumber (vector(1))
          output(3) = "0"
          Repeated output(1), output(2), output(3), values(1)
          Findnumber (vector(1))
          vector(2) = "{" & output(1) & "t+(" & Ans(1) & ")," & output(2) & "t+(" & Ans
(2) & ")}"
          normal3 = "{" & vector(1) & "," & vector(2) & "}"
        End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If (R = 2) And (sizeA3x3 = True) Then
  If values(1) = values(2) Then
    values_R = 1
    Findnumber (vector(2))
    If output(1) = 0 And output(2) = 0 And output(3) = 0 Then
      Findnumber (vector(1))
      vector_R = 2
    Else
      GoTo out2
    End If
  ElseIf values(3) = values(2) Then
    values_R = 2
    Findnumber (vector(3))
    If output(1) = 0 And output(2) = 0 And output(3) = 0 Then
      Findnumber (vector(2))
      vector_R = 3
    Else
      GoTo out2
    End If
  ElseIf values(1) = values(3) Then
    values_R = 1
    Findnumber (vector(3))
    If output(1) = 0 And output(2) = 0 And output(3) = 0 Then
      Findnumber (vector(1))
      vector_R = 3
    Else
      GoTo out2
    End If
  End If
  values_S = values(values_R)
  Repeated output(1), output(2), output(3), values_S
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        vector(vector_R) = "{" & output(1) & "t+(" & Ans(1) & ")," & output(2) & "t+("
& Ans(2) & ")," & output(3) & "t+(" & Ans(3) & ")}"
        normal3 = "{" & vector(1) & "," & vector(2) & "," & CRLF & vector(3) & "}"
      End If
      If (R = 3) Then
        Findnumber (vector(1))
        Repeated output(1), output(2), output(3), values(1)
        Findnumber (vector(1))
        vector(2) = "{" & output(1) & "t+(" & Ans(1) & ")," & output(2) & "t+(" & Ans
(2) & ")," & output(3) & "t+(" & Ans(3) & ")}"
        vector(3) = "{" & output(1) & "(t^2)/2+(" & Ans(1) & ")t+(" & Ans1(1) & ")," &
output(2) & "(t^2)/2+(" & Ans(2) & ")t+(" & Ans1(2) & ")," & output(3) & "(t^2)/2+(" &
Ans(3) & ")t+(" & Ans1(3) & ")}"
        normal3 = "{" & vector(1) & "," & vector(2) & "," & CRLF & vector(3) & "}"
      End If
out2:
      Text3.Text = normal3
      If (Form4.Option2.Value = True) Then
        openXp
          For i = 1 To k
            Findnumber (vector(i))
            For j = 1 To k
              m(j, i) = output(j)
            Next j
            values_1(i) = values(i)
          Next i
          FindInverse vector(1), vector(2), vector(3) 'หาค่าของ a1'(t) , a2'(t) , a3'(t)
          pack = "Integrate[A,t]"
          res = MLDoCompute(pack)
          pack = "Expand[%*e^(" & values(1) & "t)]"
          res = MLDoCompute(pack)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            pack = "A=Expand[(%*" & vector(1) & ")]"
            res = MLDoCompute(pack)
            pack = "Integrate[B,t]"
            res = MLDoCompute(pack)
            pack = "Expand[%*e^(" & values(2) & "t)]"
            res = MLDoCompute(pack)
            pack = "B=Expand[(%*" & vector(2) & ")]"
            res = MLDoCompute(pack)
        If sizeA3x3 = True Then
            pack = "Integrate[C,t]"
            res = MLDoCompute(pack)
            pack = "Expand[%*e^(" & values(3) & "t)]"
            res = MLDoCompute(pack)
            pack = "C=Expand[(%*" & vector(3) & ")]"
            res = MLDoCompute(pack)
        End If
        If sizeA3x3 = False Then
            Text4.Text = "X = Xc + Xp" & CRLF & "Xc = A" + vector(1) + "e^(" +
values(1) + "t) + B" + vector(2) + "e^(" + values(2) + "t)" & CRLF & " A , B  is  Arbitrary
Constant "
            pack = "Expand[A+B]"
            res = MLDoCompute(pack)
            Text5.Text = out
        Else
            Text4.Text = "X = Xc + Xp" & CRLF & "Xc = A" + vector(1) + "e^(" +
values(1) + "t) + B" + vector(2) + "e^(" + values(2) + "t) + C" + vector(3) + "e^(" + values
(3) + "t)" & CRLF & " A , B , C  is  Arbitrary Constant "
            pack = "Expand[A+B+C]"
            res = MLDoCompute(pack)
            Text5.Text = out
        End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Else
            If sizeA3x3 = False Then
                Text4.Text = "X = A" + vector(1) + "e^(" + values(1) + "t) + B" + vector(2) +
"e^(" + values(2) + "t)" & CRLF & " A , B  is  Arbitrary Constant "
            Else
                Text4.Text = "X = A" + vector(1) + "e^(" + values(1) + "t) + B" + vector(2) +
"e^(" + values(2) + "t) + C" + vector(3) + "e^(" + values(3) + "t)" & CRLF & " A , B , C  is
Arbitrary Constant "
            End If
        End If
    End If
  End If
out:
End Sub
```

- โปรแกรมย่อย

```
Sub Clearoutput()
  Form5.Text1.Text = ""
  Form5.Text2.Text = ""
  Form5.Text3.Text = ""
  Form5.Text4.Text = ""
End Sub

Sub Clearmatrix()
  Dim i, j As Integer
  For i = 1 To 3
    For j = 1 To 3
      a(i, j) = 0
      b(i, j) = 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            c(i, j) = 0
        Next j
    Next i
End Sub

Sub CreateA()
    Dim i, j As Integer
    If Form4.Check1.Value = Unchecked Then
        sizeA3x3 = False
    ElseIf Form4.Check1.Value = Checked Then
        sizeA3x3 = True
    End If
    For i = 1 To 3
    For j = 1 To 3
        a(i, j) = 0
        b(i, j) = 0
        c(i, j) = 0
        chrac = ""
        eivA = ""
        eiA = ""
    Next j
    Next i
    If sizeA3x3 = True Then
        a(1, 1) = Form4.Text1(0).Text
        a(1, 2) = Form4.Text1(1).Text
        a(1, 3) = Form4.Text1(2).Text
        a(2, 1) = Form4.Text1(4).Text
        a(2, 2) = Form4.Text1(5).Text
        a(2, 3) = Form4.Text1(6).Text
        a(3, 1) = Form4.Text1(8).Text
        a(3, 2) = Form4.Text1(9).Text
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        a(3, 3) = Form4.Text1(10).Text
    Else
        a(1, 1) = Form4.Text1(0).Text
        a(1, 2) = Form4.Text1(1).Text
        a(2, 1) = Form4.Text1(4).Text
        a(2, 2) = Form4.Text1(5).Text
    End If
    CRLF = Chr(13) & Chr(10)
    If sizeA3x3 = True Then
        chrac = "Expand[((" + Form4.Text1(0).Text + "-m)(" + Form4.Text1(5).Text + "-m)("
+ Form4.Text1(10).Text + "-m))+(" + Form4.Text1(1).Text  + "*" + Form4.Text1(6).Text
+ "*" + Form4.Text1(8).Text + ")+(" + Form4.Text1(2).Text + "*" + Form4.Text1(4).Text
+ "*" + Form4.Text1(9).Text + ")-(" + Form4.Text1(8).Text + "*(" + Form4.Text1(5).Text
+ "-m)*" + Form4.Text1(2).Text + ")-(" + Form4.Text1(9).Text  + "*" + Form4.Text1
(6).Text + "*(" + Form4.Text1(0).Text + "-m))-((" + Form4.Text1(10).Text + "-m)*" +
Form4.Text1(4).Text + "*" + Form4.Text1(1).Text + "  )]"
        eiA = "N[Eigenvalues[{{" & Form4.Text1(0).Text &  "," & Form4.Text1(1).Text &
"," & Form4.Text1(2).Text & "},{" & Form4.Text1(4).Text & "," & Form4.Text1(5).Text
& "," & Form4.Text1(6).Text & "},{" & Form4.Text1(8).Text & "," & Form4.Text1
(9).Text & "," & Form4.Text1(10).Text + "}}]]"
        eivA = "N[Eigenvectors[{{" & Form4.Text1(0).Text & "," & Form4.Text1(1).Text &
"," & Form4.Text1(2).Text & "},{" & Form4.Text1(4).Text & "," &  Form4.Text1(5).Text
& "," & Form4.Text1(6).Text & "},{" & Form4.Text1(8).Text & "," & Form4.Text1
(9).Text & "," & Form4.Text1(10).Text + "}}]]"
    Else
        chrac = "Expand[((" + Form4.Text1(0).Text + "-m)(" + Form4.Text1(5).Text + "-m))-
(" + Form4.Text1(4).Text + "*" + Form4.Text1(1).Text + ")]"
        eiA = "N[Eigenvalues[{{" & Form4.Text1(0).Text & "," & Form4.Text1(1).Text & "
},{" & Form4.Text1(4).Text & "," & Form4.Text1(5).Text & "}}]]"
        eivA = "N[Eigenvectors[{{" & Form4.Text1(0).Text & "," & Form4.Text1(1).Text &
"},{" & Form4.Text1(4).Text & "," & Form4.Text1(5).Text & "}}]]"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    End If
End Sub

Sub Findnumber(ByVal instring As String)
    Dim n, i, z As Integer
    Dim done As Boolean
    done = True
    i = 1
    z = 0
    instring = LTrim(instring)
    instring = Mid$(instring, 2)
    n = Len(instring)
    instring = Mid$(instring, 1, n - 1)
    Do While done = True
      z = InStr(1, instring, ",")
      If z <> 0 Then
        output(i) = Mid$(instring, 1, z - 1)
        output(i) = LTrim(output(i))
        i = i + 1
        instring = Mid$(instring, z + 1)
      Else
        instring = LTrim(instring)
        output(i) = instring
        done = False
      End If
    Loop
End Sub

Sub Findvectors(ByVal instring As String)
    Dim n, i, z As Integer
    Dim done As Boolean
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    done = True
    i = 1
    z = 0
    instring = Mid$(instring, 2)
    n = Len(instring)
    instring = Mid$(instring, 1, n - 1)
    Do While done = True
      z = InStr(1, instring, "}")
      If z <> 0 Then
        vector1(i) = Mid$(instring, 1, z)
        vector1(i) = LTrim(vector1(i))
        i = i + 1
        instring = Mid$(instring, z + 2)
      Else
        done = False
      End If
    Loop
End Sub

Sub Maxime()
    Dim Tr, s2, s3, i As Integer
    Tr = 0
    s2 = 0
    s3 = 0
    c2 = 0
    c3 = 0
    aa
    aaa
    For i = 1 To 3
      Tr = Tr + a(i, i)
      s2 = s2 + b(i, i)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        If sizeA3x3 = True Then
          s3 = s3 + c(i, i)
        End If
      Next i
      s1 = Tr
      c1 = -s1
      c2 = -((c1 * s1) + s2) / 2
      If sizeA3x3 = True Then
        c3 = -((c2 * s1) + (c1 * s2) + s3) / 3
      End If
    End Sub

    Sub aa()
      Dim i, j, h, k As Integer
      k = 2
      If sizeA3x3 = True Then
        k = 3
      End If
      For i = 1 To k
        For j = 1 To k
          For h = 1 To k
            b(i, j) = b(i, j) + (a(i, h) * a(h, j))
          Next h
        Next j
      Next i
    End Sub

    Sub aaa()
      Dim i, j, h, k As Integer
      k = 2
      If sizeA3x3 = True Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        k = 3
    End If
    For i = 1 To k
        For j = 1 To k
            For h = 1 To k
                c(i, j) = c(i, j) + (b(i, h) * a(h, j))
            Next h
        Next j
    Next i
End Sub

Sub Solve()
    Dim answer1, answer2 As String
    Dim n, z As Integer
    Dim done As Boolean
    done = False
    equation = Form5.Text1.Text
    equation = LTrim(equation)
    n = Len(equation)
    equation = Mid(equation, 1, n - 3)
    equation = "N[Solve[" & equation & "==0,m]]"
    MLDoCompute (equation)
    answer = out
    Do While done = False
        z = InStr(1, answer, "|")
        If z <> 0 Then
            answer1 = Mid(answer, 1, z - 1)
            answer2 = Mid(answer, z + 2)
            answer = answer1 & "," & answer2
        Else
            done = True
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        End If
    Loop
End Sub

Sub Maxim_values()
    Dim i, z, Y As Integer
    Dim got As String
    Dim done As Boolean
    i = 1
    done = False
    answer = Mid(answer, 2)
    z = Len(answer)
    answer = Mid(answer, 1, z - 1)
    Form5.Text2.Text = answer
    Do While done = False
      z = InStr(1, answer, "}")
      Y = InStr(1, answer, ">")
      If z <> 0 Then
        got = Mid(answer, Y + 1, z - Y - 1)
        values_m(i) = got
        i = i + 1
        answer = Mid(answer, z + 3)
      Else
        done = True
      End If
    Loop
End Sub

Sub Maxim_vectors1()
    Dim i, j, l, k, k2(1 To 3), kk As Integer
    Dim Check1(1 To 3) As Boolean
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim ans3(1 To 3) As String
k = 2
If sizeA3x3 = True Then
   k = 3
End If
For i = 1 To k
   For j = 1 To k
      Check1(j) = False
      k2(j) = 0
   Next j
   Calculate_vector (values_m(i))
   For j = 1 To k
   kk = 0
      For l = 1 To k
         If d(j, l) <> "0" Then
            kk = kk + 1
            k2(kk) = l
         End If
      Next l
      If kk = 1 Then
         ans3(k2(1)) = "0"
         Check1(k2(1)) = True
      End If
   Next j
   For j = 1 To k
      k2(j) = 0
   Next j
   For j = 1 To k
   kk = 0
      For l = 1 To k
         If d(j, l) <> "0" Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            kk = kk + 1
            k2(kk) = l
          End If
        Next l
        If kk = 2 Then
          If (Check1(k2(1)) = False) And (Check1(k2(2)) = True) Then
    ans3(k2(1)) = "N[(0-(" & d(j, k2(2)) & "*" & ans3(k2(2)) & "))/(" & d(j, k2(1)) & ")]"
            res = MLDoCompute(ans3(k2(1)))
            ans3(k2(1)) = out
            Check1(k2(1)) = True
          ElseIf (Check1(k2(1)) = True) And (Check1(k2(2)) = False) Then
    ans3(k2(2)) = "N[(0-(" & d(j, k2(1)) & "*" & ans3(k2(1)) & "))/(" &   d(j, k2(2)) & ")]"
            res = MLDoCompute(ans3(k2(2)))
            ans3(k2(2)) = out
            Check1(k2(2)) = True
          ElseIf (Check1(k2(1)) = False) And (Check1(k2(2)) = False) Then
            ans3(k2(1)) = "1"
    ans3(k2(2)) = "N[(0-(" & d(j, k2(1)) & "*" & ans3(k2(1)) & "))/(" & d(j, k2(2)) & ")]"
            res = MLDoCompute(ans3(k2(2)))
            ans3(k2(2)) = out
            Check1(k2(1)) = True
            Check1(k2(2)) = True
          End If
        End If
      Next j
      For j = 1 To k
        k2(j) = 0
      Next j
      For j = 1 To k
      kk = 0
        For l = 1 To k
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            If d(j, l) <> "0" Then
               kk = kk + 1
               k2(kk) = l
            End If
         Next l
         If kk = 3 Then
            If (Check1(1) = False) And (Check1(2) = False) And (Check1(3) = False) Then
               ans3(k2(1)) = "1"
               ans3(k2(2)) = "0"
               ans3(k2(3)) = "N[(0-(" & d(j, k2(1)) & "*" & ans3(k2(1)) & ")-(" & d(j, k2(2))
& "*" & ans3(k2(2)) & "))/(" & d(j, k2(3)) & ")]"
               res = MLDoCompute(ans3(k2(3)))
               ans3(k2(3)) = out
               Check1(k2(1)) = True
               Check1(k2(2)) = True
               Check1(k2(3)) = True
ElseIf (Check1(k2(1)) = True) And (Check1(k2(2)) = False) And (Check1(k2(3))  = False) Then
               ans3(k2(2)) = "1"
               ans3(k2(3)) = "N[(0-(" & d(j, k2(1)) & "*" & ans3(k2(1)) & ")-(" & d(j, k2(2))
& "*" & ans3(k2(2)) & "))/(" & d(j, k2(3)) & ")]"
               res = MLDoCompute(ans3(k2(3)))
               ans3(k2(3)) = out
               Check1(k2(2)) = True
               Check1(k2(3)) = True
ElseIf (Check1(k2(1)) = False) And (Check1(k2(2)) = True) And  (Check1(k2(3)) = False) Then
               ans3(k2(1)) = "1"
               ans3(k2(3)) = "N[(0-(" & d(j, k2(1)) & "*" & ans3(k2(1)) & ")-(" & d(j, k2(2))
& "*" & ans3(k2(2)) & "))/(" & d(j, k2(3)) & ")]"
               res = MLDoCompute(ans3(k2(3)))
               ans3(k2(3)) = out
               Check1(k2(1)) = True
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                Check1(k2(3)) = True
ElseIf (Check1(k2(1)) = False) And (Check1(k2(2)) = False) And    (Check1(k2(3)) = True) Then
                ans3(k2(1)) = "1"
                ans3(k2(2)) = "N[(0-(" & d(j, k2(1)) & "*" & ans3(k2(1)) & ")-(" & d(j, k2(3))
    & "*" & ans3(k2(3)) & "))/(" & d(j, k2(2)) & ")]"
                res = MLDoCompute(ans3(k2(2)))
                ans3(k2(2)) = out
                Check1(k2(1)) = True
                Check1(k2(2)) = True
ElseIf (Check1(k2(1)) = True) And (Check1(k2(2)) = True) And    (Check1(k2(3)) = False) Then
                ans3(k2(3)) = "N[(0-(" & d(j, k2(1)) & "*" & ans3(k2(1)) & ")-(" & d(j, k2(2))
    & "*" & ans3(k2(2)) & "))/(" & d(j, k2(3)) & ")]"
                res = MLDoCompute(ans3(k2(3)))
                ans3(k2(3)) = out
                Check1(k2(3)) = True
ElseIf (Check1(k2(1)) = True) And (Check1(k2(2)) = False) And    (Check1(k2(3)) = True) Then
                ans3(k2(2)) = "N[(0-(" & d(j, k2(1)) & "*" & ans3(k2(1)) & ")-(" & d(j, k2(3))
    & "*" & ans3(k2(3)) & "))/(" & d(j, k2(2)) & ")]"
                res = MLDoCompute(ans3(k2(2)))
                ans3(k2(2)) = out
                Check1(k2(2)) = True
ElseIf (Check1(k2(1)) = False) And (Check1(k2(2)) = True) And    (Check1(k2(3)) = True) Then
                ans3(k2(1)) = "N[(0-(" & d(j, k2(2)) & "*" & ans3(k2(2)) & ")-(" & d(j, k2(3))
    & "*" & ans3(k2(3)) & "))/(" & d(j, k2(1)) & ")]"
                res = MLDoCompute(ans3(k2(1)))
                ans3(k2(1)) = out
                Check1(k2(1)) = True
            End If
        End If
    Next j
    If Check1(1) = False Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ans3(1) = "1"
      End If
      If Check1(2) = False Then
        ans3(2) = "1"
      End If
      If Check1(3) = False And sizeA3x3 = True Then
        ans3(3) = "1"
      End If
      For j = 1 To k
        vector_m(i, j) = ans3(j)
      Next j
    Next i
End Sub

Sub Calculate_vector(ByVal eigenvalue As String)
    Dim i, j, k As Integer
    k = 2
    If sizeA3x3 = True Then
      k = 3
    End If
    For i = 1 To k
      For j = 1 To k
        If i = j Then
          d(i, j) = a(i, j) & "-(" & eigenvalue & ")"
          res = MLDoCompute(d(i, j))
          d(i, j) = out
        Else
          d(i, j) = a(i, j)
        End If
      Next j
    Next i
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Sub

Sub Repeated(ByVal r4 As String, ByVal r2 As String, ByVal r3 As String, ByVal v As
String)
   Dim i, j  As Integer
   Dim RI(1 To 3, 1 To 3), command, d  As String
   For i = 1 To k
   For j = 1 To k
      RI(i, j) = ""
   Next j
   Next i
   For i = 1 To k
   For j = 1 To k
      If i = j Then
         Row(i, j) = "N[" & a(i, j) & "-(" & v & ")]"
         res = MLDoCompute(Row(i, j))
         Row(i, j) = out
      Else
         Row(i, j) = "N[" & a(i, j) & "]"
         res = MLDoCompute(Row(i, j))
         Row(i, j) = out
      End If
   Next j
   Next i
   RB(1) = r4
   RB(2) = r2
   RB(3) = r3
   If sizeA3x3 = False Then
      d = "Det[{{" & Row(1, 1) & "," & Row(1, 2) & "},{" & Row(2, 1) & "," & Row(2, 2)
& "}}]"
      res = MLDoCompute(d)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    If (out = "0.") Then
      Condition
      For i = 1 To k
        Ans(i) = Ans2(i)
        Ans(i) = "N[" & Ans(i) & "]"
        res = MLDoCompute(Ans(i))
        Ans(i) = out
      Next i
    ElseIf (out <> "0.") Then
      GoTo out2
    End If
    GoTo out1
  Else
    d = "Det[{{" & Row(1, 1) & "," & Row(1, 2) & "," & Row(1, 3) & "},{" & Row(2, 1)
& "," & Row(2, 2) & "," & Row(2, 3) & "},{" & Row(3, 1) & "," & Row(3, 2) & "," &
Row(3, 3) & "}}]"
    res = MLDoCompute(d)
    If (out = "0.") Then
      Condition
      For i = 1 To 3
        Ans(i) = Ans2(i)
        Ans(i) = "N[" & Ans(i) & "]"
        res = MLDoCompute(Ans(i))
        Ans(i) = out
      Next i
      If R = 3 Then
        For i = 1 To 3
          RB(i) = Ans(i)
        Next i
        Condition
        For i = 1 To 3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            Ans1(i) = Ans2(i)
            Ans1(i) = "N[" & Ans1(i) & "]"
            res = MLDoCompute(Ans1(i))
            Ans1(i) = out
          Next i
        End If
      ElseIf (out <> "0.") Then
        GoTo out2
      End If
      GoTo out1
    End If
out2:
If sizeA3x3 = False Then
      command = "Inverse[{{" & Row(1, 1) & "," & Row(1, 2) & "},{" & Row(2, 1) & ","
& Row(2, 2) & "}}]"
   Else
      command = "Inverse[{{" & Row(1, 1) & "," & Row(1, 2) & "," & Row(1, 3) & "},{"
& Row(2, 1) & "," & Row(2, 2) & "," & Row(2, 3) & "},{" & Row(3, 1) & "," & Row(3,
2) & "," & Row(3, 3) & "}}]"
   End If
   res = MLDoCompute(command)
   Findvectors (out)
   For i = 1 To k
     Findnumber (vector1(i))
     For j = 1 To k
       RI(i, j) = output(j)
     Next j
   Next i
   If sizeA3x3 = False Then
     For i = 1 To k
       Ans(i) = "N[(" & RI(i, 1) & ")" & r4 & "+(" & RI(i, 2) & ")" & r2 & "]"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      Next i
    Else
      For i = 1 To k
        Ans(i) = "N[(" & RI(i, 1) & ")" & r4 & "+(" & RI(i, 2) & ")" & r2 & "+(" & RI(i, 3)
& ")" & r3 & "]"
      Next i
    End If
    For i = 1 To k
      command = Ans(i)
      res = MLDoCompute(command)
      Ans(i) = out
    Next i
    If R = 3 Then
      For i = 1 To 3
        Ans1(i) = "N[(" & RI(i, 1) & ")" & Ans(1) & "+(" & RI(i, 2) & ")" & Ans(2) & "+
(" & RI(i, 3) & ")" & Ans(3) & "]"
      Next i
      For i = 1 To k
        command = Ans1(i)
        res = MLDoCompute(command)
        Ans1(i) = out
      Next i
    End If
  out1:
  End Sub

  Sub Condition()
    Dim check(1 To 3) As Boolean
    Dim kk, i, j, k2(1 To 3) As Integer
    For i = 1 To k
      check(i) = False
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        k2(i) = 0
    Next i
    For i = 1 To k
        kk = 0
        For j = 1 To k
            If Row(i, j) <> "0." Then
                kk = kk + 1
                k2(kk) = j
            End If
        Next j
        If kk = 1 Then
            Ans2(k2(1)) = RB(i) & "/" & Row(i, k2(1))
            check(k2(1)) = True
        End If
    Next i
For i = 1 To k
        k2(i) = 0
    Next i
    For i = 1 To k
        kk = 0
        For j = 1 To k
            If Row(i, j) <> "0." Then
                kk = kk + 1
                k2(kk) = j
            End If
        Next j
        If kk = 2 Then
            If (check(k2(1)) = False) And (check(k2(2)) = True) Then
                Ans2(k2(1)) = "N[(" & RB(i) & "-(" & Row(i, k2(2)) & "*" & Ans2(k2(2)) &
"))/(" & Row(i, k2(1)) & ")]"
                res = MLDoCompute(Ans2(k2(1)))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            Ans2(k2(1)) = out
            check(k2(1)) = True
        ElseIf (check(k2(1)) = True) And (check(k2(2)) = False) Then
            Ans2(k2(2)) = "N[(" & RB(i) & "-(" & Row(i, k2(1)) & "*" & Ans2(k2(1)) &
"))/(" & Row(i, k2(2)) & ")]"
            res = MLDoCompute(Ans2(k2(2)))
            Ans2(k2(2)) = out
            check(k2(2)) = True
        ElseIf (check(k2(1)) = False) And (check(k2(2)) = False) Then
            Ans2(k2(1)) = "0"
            Ans2(k2(2)) = "N[(" & RB(i) & "-(" & Row(i, k2(1)) & "*" & Ans2(k2(1)) &
"))/(" & Row(i, k2(2)) & ")]"
            res = MLDoCompute(Ans2(k2(2)))
            Ans2(k2(2)) = out
            check(k2(2)) = True
            check(k2(1)) = True
        End If
      End If
    Next i
For i = 1 To k
    k2(i) = 0
  Next i
  For i = 1 To k
    kk = 0
    For j = 1 To k
      If Row(i, j) <> "0." Then
        kk = kk + 1
        k2(kk) = j
      End If
    Next j
    If kk = 3 And sizeA3x3 = True Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If (check(1) = False) And (check(2) = False) And (check(3) = False) Then
    Ans2(k2(1)) = "0"
    Ans2(k2(2)) = "1"
    Ans2(k2(3)) = "(" & RB(i) & "-(" & Row(i, k2(1)) & "*" & Ans2(k2(1)) & ")-("
& Row(i, k2(2)) & "*" & Ans2(k2(2)) & "))/(" & Row(i, k2(3)) & ")"
    check(k2(1)) = True
    check(k2(2)) = True
    check(k2(3)) = True
ElseIf (check(k2(1)) = T rue) And (check(k2(2)) = False) And (check(k2(3)) =
False) Then
    Ans2(k2(2)) = "1"
    Ans2(k2(3)) = "(" & RB(i) & "-(" & Row(i, k2(1)) & "*" & Ans2(k2(1)) & ")-("
& Row(i, k2(2)) & "*" & Ans2(k2(2)) & "))/(" & Row(i, k2(3)) & ")"
    check(k2(2)) = True
    check(k2(3)) = True
ElseIf (check(k2(1)) = False) And (check(k2(2)) = True) And (check(k2(3)) =
False) Then
    Ans2(k2(1)) = "0"
    Ans2(k2(3)) = "(" & RB(i) & "-(" & Row(i, k2(1)) & "*" & Ans2(k2(1)) & ")-("
& Row(i, k2(2)) & "*" & Ans2(k2(2)) & "))/(" & Row(i, k2(3)) & ")"
    check(k2(1)) = True
    check(k2(3)) = True
ElseIf (check(k2(1)) = False) And (check(k2(2)) = False) And (check(k2(3)) =
True) Then
    Ans2(k2(2)) = "1"
    Ans2(k2(1)) = "(" & RB(i) & "-(" & Row(i, k2(3)) & "*" & Ans2(k2(3)) & ")-
(" & Row(i, k2(2)) & "*" & Ans2(k2(2)) & "))/(" & Row(i, k2(1)) & ")"
    check(k2(1)) = True
    check(k2(2)) = True
ElseIf (check(k2(1)) = True) And (check(k2(2)) = True) And (check(k2(3)) = False) Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            Ans2(k2(3)) = "(" & RB(i) & "-(" & Row(i, k2(1)) & "*" & Ans2(k2(1)) & ")-("
& Row(i, k2(2)) & "*" & Ans2(k2(2)) & "))/(" & Row(i, k2(3)) & ")"
            check(k2(3)) = True
  ElseIf (check(k2(1)) = True) And (check(k2(2)) = False) And (check(k2(3)) = True) Then
            Ans2(k2(2)) = "(" & RB(i) & "-(" & Row(i, k2(1))  & "*" & Ans2(k2(1)) & ")-("
& Row(i, k2(3)) & "*" & Ans2(k2(3)) & "))/(" & Row(i, k2(2)) & ")"
            check(k2(2)) = True
  ElseIf (check(k2(1)) = False) And (check(k2(2)) = True) And (check(k2(3)) = True) Then
            Ans2(k2(1)) = "(" & RB(i) & "-(" & Row(i, k2(3))  & "*" & Ans2(k2(3)) & ")-("
& Row(i, k2(2)) & "*" & Ans2(k2(2)) & "))/(" & Row(i, k2(1)) & ")"
            check(k2(1)) = True
          End If
       End If
    Next i
If check(1) = False Then
       Ans2(1) = "1"
    End If
    If check(2) = False Then
       Ans2(2) = "1"
    End If
    If check(3) = False And sizeA3x3 = True Then
       Ans2(3) = "1"
    End If
End Sub
Sub openXp()
    Form5.Text5.Text = ""
    Form5.Label6.Visible = True
    Form5.Text5.Visible = True
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Sub FindInverse(ByVal number1 As String, ByVal number2 As String, ByVal number3 As String)
   Dim i As Integer
   Dim det, det1, det2, det3 As String
   If sizeA3x3 = False Then
      det = "L=Det[{{" & vector_m(1, 1) & "," & vector_m(2, 1) & "},{" & vector_m(1, 2) & "," & vector_m(2, 2) & "}}]"
      res = MLDoCompute(det)
      If out <> "0." Then
         det1 = "A=Det[{{" & f_t & "," & m(1, 2) & "},{" & g_t & "," & m(2, 2) & "}}]"
         res = MLDoCompute(det1)
         det2 = "B=Det[{{" & m(1, 1) & "," & f_t & "},{" & m(2, 1) & "," & g_t & "}}]"
         res = MLDoCompute(det2)
         m1(1) = "A=(A*e^(- (" & values_m(1) & "t))/L)"
         m1(2) = "B=(B*e^(- (" & values_m(2) & "t))/L)"
         For i = 1 To 2
           res = MLDoCompute(m1(i))
         Next i
      Else
 MsgBox "ตัวหารเป็นศูนย์จึงหาค่า Xp ด้วยวิธีของคราเมอร์ไม่ได้" , vbOKOnly, "ผิดพลาด"
      End If
   Else
      det = "L=Det[{{" & vector_m(1, 1) & "," & vector_m(2, 1) & "," & vector_m(3, 1) & "},{" & vector_m(1, 2) & "," & vector_m(2, 2) & "," & vector_m(3, 2) & "},{" & vector_m(1, 3) & "," & vector_m(2, 3) & "," & vector_m(3, 3) & "}}]"
      res = MLDoCompute(det)
      If out <> "0." Then 'ถ้า det เป็น 0 จะใช้กฎของคราเมอร์หาไม่ได้
         det1 = "A=Det[{{" & f_t & "," & m(1, 2) & m(1, 3) & "},{" & g_t & "," & m(2, 2) & m(2, 3) & "},{" & h_t & "," & m(3, 2) & m(3, 3) & "}}]"
         res = MLDoCompute(det1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        det2 = "B=Det[{{" & m(1, 1) & "," & f_t & m(1, 3) & "},{" & m(2, 1) & "," & g_t
& m(2, 3) & "},{" & m(3, 1) & "," & h_t & m(3, 3) & "}}]"
        res = MLDoCompute(det2)
        det3 = "C=Det[{{" & m(1, 1) & "," & m(1, 2) & f_t & "},{" & m(2, 1) & "," & m
(2, 2) & g_t & "},{" & m(3, 1) & "," & m(3, 2) & h_t & "}}]"
        res = MLDoCompute(det3)
        m1(1) = "A*e^(- (" & values_m(1) & "t))/L"
        m1(2) = "B*e^(- (" & values_m(2) & "t))/L"
        m1(3) = "C*e^(- (" & values_m(3) & "t))/L"
        For i = 1 To 3
          res = MLDoCompute(m1(i))
        Next i
      Else
        MsgBox "ตัวหารเป็นศูนย์จึงหาค่า Xp ไม่ได้", vbOKOnly, "ผิดพลาด"
      End If
    End If
End Sub

Function Check_I() As Boolean
    Dim check(1 To 3) As String
    Dim i, z, Y As Integer
    Check_I = False
    For i = 1 To 3
      check(i) = check_v(i)
      z = InStr(1, check(i), "I")
      If z <> 0 Then
        Check_I = True
      End If
    Next i
End Function
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

กิตติ ภักดีวัฒนะกุล และ จำลอง ครูอุตสาหะ 2542. **Visual Basic 6.** พิมพ์ครั้งที่ 6 กรุงเทพฯ : บริษัท เคทีพี คอมพ์ แอนด์ คอนซัลท์ จำกัด

ผ่องพรรณ รัตนธนาวันต์. 2535. **พีชคณิตเชิงเส้น.** กรุงเทพฯ : คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

Dennis G.Zill. and Michael R.Cullen. **Advanced Engineering Mathematics.** 2ed. Jones And Bartlett Publishers

Stephen Wolfram. **The Mathematica Book**. 3ed. Addison-Wesley Publishing Company.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้