



โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด

สำหรับโปรแกรมเฉพาะ

Convert file input to keyboard signal for individual Program

นางสาวจุฬารัตน์ นุ่นมีศรี รหัส 48362230
นางสาววิจิตรา มหายศ รหัส 48362445

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ...../...../.....
เลขทะเบียน..... 1A113260
เลขเรียกหนังสือ..... ๙/ร.
มหาวิทยาลัยนเรศวร ๙ ๖๘/๗

2551

ปริญญาบัตรนี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2551



ใบรับรองโครงการ

หัวข้อโครงการ โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับ
โปรแกรมเฉพาะ

ผู้ดำเนินโครงการ นางสาวจุฬารัตน์ นุ่นมีศรี รหัส 48362230
นางสาวจิตรรา มหายศ รหัส 48362445

อาจารย์ที่ปรึกษา ดร.พนมขวัญ ริยะมงคล

สาขาวิชา วิศวกรรมคอมพิวเตอร์

ภาควิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์

ปีการศึกษา 2551

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนครสวรรค์ อนุมัติให้โครงการฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะกรรมการสอบ โครงการวิศวกรรม

พนม

.....ประธานกรรมการ

(ดร.พนมขวัญ ริยะมงคล)

มณฑิลา

.....กรรมการ

(ดร.มณฑิลา สงฆ์จันทร์)

ไพศาล

.....กรรมการ

(ดร.ไพศาล มณีสว่าง)

หัวข้อโครงการ	โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับโปรแกรมเฉพาะ	
ผู้ดำเนินโครงการ	นางสาวจุฬารัตน์ นุ่นมีศรี	รหัส 48362230
	นางสาววิจิตรา มหายศ	รหัส 48362445
อาจารย์ที่ปรึกษา	ดร.พนมขวัญ ริยะมงคล	
สาขาวิชา	วิศวกรรมคอมพิวเตอร์	
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์	
ปีการศึกษา	2551	

บทคัดย่อ

โครงการนี้เราได้พัฒนาโปรแกรมสำหรับแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับโปรแกรมเฉพาะ เพื่อใช้ในการเพิ่มประสิทธิภาพการทำงานของโปรแกรมเฉพาะที่สามารถรับค่าข้อมูลจากคีย์บอร์ดได้เพียงอย่างเดียว โดยใช้หลักการของโปรแกรม AutoIt ที่สามารถพัฒนาคำสั่งที่ใช้สั่งการ โปรแกรมต่างๆ บน Microsoft Window

ทั้งนี้เราได้มีการพัฒนาลักษณะของโปรแกรมเฉพาะในรูปแบบต่างๆ เพื่อใช้ในการทดสอบโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดอีกด้วย ซึ่งจากผลการทดลองพบว่าโปรแกรมสามารถแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดแล้วส่งไปยังโปรแกรมเฉพาะได้อย่างถูกต้อง

Project title Convert file input to keyboard signal for individual Program

Name Miss Chularat Noonmeesri ID 48362230
Miss Wijitra Mahayod ID 48362445

Project advisor Panomkhawn Riyamongkol, Ph.D.

Major Computer Engineering

Department Electrical and Computer Engineering

Academic year 2008

.....

Abstract

This project develops a program for converting a file input to be a keyboard signal for individual programs. It is built up the capability of individual programs that only accept alphabet file input by concept of AutoIt program. The AutoIt program is used to write a code to control program on Microsoft Window.

For experiments, this project develops the individual program in different forms to test the convert file program. Our experiments show that convert file program can send alphabet data to the individual program correctly.

กิตติกรรมประกาศ

โครงการวิศวกรรมคอมพิวเตอร์สำเร็จได้ด้วยดี ก็เนื่องจากความอนุเคราะห์จากท่านอาจารย์ที่ปรึกษา อาจารย์พนมขวัญ ธิยะมงคล ที่กรุณาให้คำปรึกษา แนะนำวิธีการในการทำงาน ตลอดจนการตรวจสอบการทำงานพร้อมทั้งเสนอแนะทางแก้ไขตลอดระยะเวลาการทำโครงการ สุดท้ายต้องขอขอบพระคุณอาจารย์ทุกท่านและเพื่อนๆ ทุกคนที่ยังไม่ได้เอ่ยนามที่คอยสนับสนุนในการทำโครงการครั้งนี้



สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ก
บทคัดย่อภาษาอังกฤษ.....	ข
กิตติกรรมประกาศ.....	ค
สารบัญ.....	ง
สารบัญตาราง.....	ช
สารบัญรูป.....	ฅ
บทที่ 1 บทนำ.....	1
1.1 ที่มาและความสำคัญ.....	1
1.2 วัตถุประสงค์ของโครงการ.....	2
1.3 ขอบข่ายของโครงการ.....	2
1.4 ขั้นตอนการดำเนินงาน.....	2
1.5 แผนการดำเนินงาน.....	3
1.6 ผลที่คาดว่าจะได้รับ.....	3
1.7 งบประมาณของโครงการ.....	4
บทที่ 2 หลักการและทฤษฎีที่เกี่ยวข้อง.....	5
2.1 หลักการทำงานของคีย์บอร์ด.....	5
2.2 การเข้ารหัสและถอดรหัส(Encoder and Decoder).....	7
2.3 รหัสแอสกี (American Standard Code Interchange, ASCII).....	11
2.4 หลักการทำงานของโปรแกรม AutoIt.....	11

สารบัญ(ต่อ)

	หน้า
บทที่ 3 วิธีดำเนินการ.....	19
3.1 ลักษณะการทำงานร่วมกันของ โปรแกรมแปลงไฟล์ตัวอักษร เป็นสัญญาณอินพุทของคีย์บอร์ดกับ โปรแกรมเฉพาะ.....	19
3.2 การทำงานของ โปรแกรมเฉพาะและการออกแบบ โปรแกรม.....	19
3.3 การเข้าถึงส่วนประกอบใน โปรแกรมเฉพาะ.....	24
3.4 การออกแบบการทำงานของ โปรแกรมแปลง ไฟล์ตัวอักษร เป็นสัญญาณอินพุทของคีย์บอร์ด.....	25
3.5 การทำงานของ โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด.....	32
บทที่ 4 ผลการทดลอง.....	35
4.1 ผลการทดลองการใช้ โปรแกรมแปลง ไฟล์ตัวอักษรเป็นสัญญาณอินพุท ของคีย์บอร์ดกับ โปรแกรมเฉพาะและ ไฟล์อินพุทรูปแบบต่างๆ.....	35
4.2 ผลการทดลองการใช้โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุท ของคีย์บอร์ดกับ โปรแกรมเฉพาะที่ใช้ในชีวิตประจำวัน.....	51
บทที่ 5 สรุปผล.....	61
5.1 สรุปผลการทดลอง.....	61
5.2 ปัญหาและอุปสรรค.....	62
5.3 ข้อเสนอแนะ.....	62
5.4 สรุป.....	62

สารบัญ(ต่อ)

	หน้า
บรรณานุกรม.....	63
ภาคผนวก ก.....	64
ภาคผนวก ข.....	82
ประวัติผู้เขียน โครงการงาน.....	88



สารบัญตาราง

ตารางที่	หน้า
2.1 แสดงรูปแบบของกราฟิกที่สามารถใช้ได้ ใน AutoIt.....	14
2.2 แสดงฟังก์ชันการจัดการไฟล์ใน AutoIt.....	15
2.3 แสดงฟังก์ชันกล่องข้อความใน AutoIt.....	16
2.4 แสดงฟังก์ชันควบคุมเมาส์ใน AutoIt.....	17
2.5 แสดงฟังก์ชันการจัดการหน้าต่างโปรแกรมใน AutoIt.....	17
2.6 แสดงฟังก์ชันคำสั่งควบคุม control ต่างๆ ของโปรแกรมอื่นใน AutoIt.....	18
3.1 แสดงรูปแบบการเขียนคำสั่งของฟังก์ชันเปิดไฟล์.....	26
3.2 แสดงรูปแบบการเขียนคำสั่งของฟังก์ชันกล่องข้อความ.....	27
3.3 แสดงรูปแบบการเขียนคำสั่งของฟังก์ชันการอ่านไฟล์ที่ละบรรทัด.....	28
3.4 แสดงรูปแบบการเขียนคำสั่งของฟังก์ชันการแบ่งสตริง.....	29
3.5 แสดงรูปแบบการเขียนคำสั่งของฟังก์ชันการรอโปรแกรมเฉพาะ.....	31
3.6 แสดงรูปแบบการเขียนคำสั่งการส่งข้อความไปยังโปรแกรมเฉพาะ.....	31
3.7 แสดงรูปแบบการเขียนคำสั่งให้โปรแกรมกดปุ่ม.....	32

สารบัญรูป

รูปที่	หน้า
2.1 แสดงเส้น Matrix 8 x 4.....	8
2.2 แสดงวงจรเข้ารหัส.....	8
2.3 แสดงการใช้วงจร Logic เขียนFunction	9
2.4 แสดง Diode Matrix สำหรับวงจรถอดรหัส BCD -8421 เป็นเลขฐานสิบ.....	10
2.5 แสดงวงจร Gate combination สำหรับวงจรถอดรหัส BCD -8421 เป็นเลขฐานสิบ.....	10
2.6 แสดงลักษณะเครื่องมือ SciTE Script Editors ของโปรแกรม AutoIt.....	12
2.7 แสดงลักษณะเครื่องมือ AutoIt Window Info ของโปรแกรม AutoIt.....	13
2.8 แสดงลักษณะเครื่องมือ Aut2Exe ของโปรแกรม AutoIt.....	14
3.1 แสดงผังการทำงาน โดยรวม.....	19
3.2 แสดงโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ซึ่งแสดงข้อมูลในฟอร์มสุดท้าย.....	20
3.3 แสดงโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ซึ่งแสดงข้อมูลในตารางฐานข้อมูล.....	21
3.4 แสดงโปรแกรมเฉพาะที่พัฒนาด้วยภาษา C#.....	22
3.5 แสดงโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java.....	23
3.6 แสดงเครื่องมือ AutoIt Window Info	24
3.7 แสดงตัวอย่างคำสั่ง โปรแกรมแปลงข้อมูลไฟล์ตัวอักษรเป็นสัญญาณอินพุทจากคีย์บอร์ด....	25
3.8 แสดงหน้าต่างกล่องเปิดไฟล์.....	27
3.9 แสดงขั้นตอนการทำงานของ โปรแกรมแปลงข้อมูลไฟล์ตัวอักษรเป็นสัญญาณอินพุท จากคีย์บอร์ด.....	33
4.1 โปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในฟอร์มสุดท้าย ขณะที่ยังไม่ได้มีการกรอกข้อมูล.....	36
4.2 แสดงไฟล์ข้อมูลชนิด Text file ซึ่งนำมาทดสอบกับโปรแกรมเฉพาะ ที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในฟอร์มสุดท้าย.....	37

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.3 แสดงไฟล์ข้อมูลชนิด Excel file ซึ่งนำมาทดสอบกับโปรแกรมเฉพาะ ที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในฟอร์มสุดท้าย.....	37
4.4 แสดงโค้ด โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับโปรแกรม เฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในฟอร์มสุดท้าย.....	38
4.5 แสดงผลการทดสอบกับ โปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในฟอร์มสุดท้าย.....	40
4.6 โปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในตารางฐานข้อมูล ขณะที่ยังไม่ได้มีการกรอกข้อมูล.....	41
4.7 แสดงไฟล์ข้อมูลชนิด Text file ซึ่งนำมาทดสอบกับ โปรแกรมเฉพาะ ที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในตารางฐานข้อมูล.....	42
4.8 แสดงไฟล์ข้อมูลชนิด Excel file ซึ่งนำมาทดสอบกับ โปรแกรมเฉพาะ ที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในตารางฐานข้อมูล.....	42
4.9 แสดงโค้ด โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับ โปรแกรม เฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในตารางฐานข้อมูล.....	43
4.10 แสดงผลการทดสอบกับโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลในตารางฐานข้อมูล	44
4.11 โปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูล ในตารางฐานข้อมูล ขณะที่ยังไม่ได้มีการกรอกข้อมูล.....	45
4.12 แสดงไฟล์ข้อมูลชนิด Text file ซึ่งนำมาทดสอบกับ โปรแกรมเฉพาะ ที่พัฒนาด้วยด้วย ภาษา C#.....	46
4.13 แสดงไฟล์ข้อมูลชนิด Excel file ซึ่งนำมาทดสอบกับ โปรแกรมเฉพาะ ที่พัฒนาด้วยด้วย ภาษา C#.....	46
4.14 แสดงโค้ด โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด สำหรับ โปรแกรมเฉพาะที่พัฒนาด้วยด้วยภาษา C#.....	47

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.15 แสดงผลการทดสอบกับ โปรแกรมเฉพาะที่พัฒนาด้วยภาษา C#.....	48
4.16 แสดงผลการทดสอบ โปรแกรมเฉพาะที่พัฒนาด้วยภาษา C# กับ ไฟล์ข้อมูลที่เป็นภาษาไทย.....	49
4.17 แสดงผลการทดสอบการระบุส่วนประกอบ โปรแกรมเฉพาะ ที่พัฒนาด้วยภาษา Java ด้วยเครื่องมือ AutoIt Window Info.....	50
4.18 หน้าต่างการกรอกข้อมูลสมาชิกของ โปรแกรมร้านเช่าหนังสือ ขณะที่ยังไม่ได้มีการกรอกข้อมูล.....	51
4.19 แสดงไฟล์ข้อมูลชนิด Text file ซึ่งนำมาทดสอบกับ ข้อมูลสมาชิกของ โปรแกรมร้านเช่าหนังสือ.....	52
4.20 แสดงไฟล์ข้อมูลชนิด Excel file ซึ่งนำมาทดสอบกับข้อมูลสมาชิก ของโปรแกรมร้านเช่าหนังสือ.....	52
4.21 แสดงโค้ด โปรแกรมแปลง ไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด สำหรับใช้กับข้อมูลสมาชิกโปรแกรมร้านเช่าหนังสือ.....	53
4.22 แสดงผลการทดสอบกับการกรอกข้อมูลสมาชิก โปรแกรมร้านเช่าหนังสือ.....	54
4.23 แสดงผลฐานข้อมูลสมาชิก โปรแกรมร้านเช่าหนังสือหลังจากการทดสอบ กับ โปรแกรมแปลง ไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด.....	55
4.24 หน้าต่างการกรอกข้อมูลพนักงานของ โปรแกรม MyRestuarant ขณะที่ยังไม่ได้มีการกรอกข้อมูล.....	56
4.25 แสดงไฟล์ข้อมูลชนิด Text file ซึ่งนำมาทดสอบกับ โปรแกรม MyRestuarant	57
4.26 แสดงไฟล์ข้อมูลชนิด Excel file ซึ่งนำมาทดสอบกับ โปรแกรม MyRestuarant	57
4.27 แสดงโค้ด โปรแกรมแปลง ไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด สำหรับใช้กับข้อมูลพนักงาน โปรแกรม MyRestuarant.....	58
4.28 แสดงผลการทดสอบกรอกข้อมูลพนักงานของ โปรแกรม MyRestuarant.....	59

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.29 แสดงผลฐานข้อมูลพนักงานโปรแกรม MyRestuarant หลังจากการทดสอบ กับโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของกีบอร์ด.	60
ก.1 แสดงโค้ดฟอร์ม 1 ของโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ซึ่งแสดงผลข้อมูลในฟอร์มสุดท้าย.....	64
ก.2 แสดงโค้ดฟอร์ม 2 ของโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ซึ่งแสดงผลข้อมูลในฟอร์มสุดท้าย.....	65
ก.3 แสดงโค้ดฟอร์ม 3 ของโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ซึ่งแสดงผลข้อมูลในฟอร์มสุดท้าย.....	66
ก.4 แสดงโค้ดส่วนที่ 1 ของโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ซึ่งแสดงผลข้อมูลในตารางฐานข้อมูล.....	67
ก.5 แสดงโค้ดส่วนที่ 2 ของโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ซึ่งแสดงผลข้อมูลในตารางฐานข้อมูล.....	68
ก.6 แสดงโค้ดฟอร์ม1 ของโปรแกรมเฉพาะที่พัฒนาด้วยภาษา C#.....	69
ก.7 แสดงโค้ดฟอร์ม1 ของโปรแกรมเฉพาะที่พัฒนาด้วยภาษา C#.....	70
ก.8 แสดงโค้ดฟอร์ม1 ของโปรแกรมเฉพาะที่พัฒนาด้วยภาษา C#.....	71
ก.9 แสดงโค้ดส่วนที่ 1 ของฟอร์ม 1 ในโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java.....	72
ก.10 แสดงโค้ดส่วนที่ 2 ของฟอร์ม 1 ในโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java.....	73
ก.11 แสดงโค้ดส่วนที่ 3 ของฟอร์ม 1 ในโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java.....	74
ก.12 แสดงโค้ดส่วนที่ 4 ของฟอร์ม 1 ในโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java.....	75
ก.13 แสดงโค้ดส่วนที่ 5 ของฟอร์ม 1 ในโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java.....	76
ก.14 แสดงโค้ดส่วนที่ 1 ของฟอร์ม 2 ในโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java.....	77
ก.15 แสดงโค้ดส่วนที่ 2 ของฟอร์ม 2 ในโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java.....	78
ก.16 แสดงโค้ดส่วนที่ 3 ของฟอร์ม 2 ในโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java.....	79
ก.17 แสดงโค้ดส่วนที่ 4 ของฟอร์ม 2 ในโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java.....	80

สารบัญรูป(ต่อ)

รูปที่	หน้า
ก.18 แสดงโค้ดส่วนที่ 5 ของฟอร์ม 2 ในโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java.....	81
ข.1 แสดงโค้ดส่วนที่ 1 ของโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด สำหรับโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในฟอร์มสุดท้าย..	82
ข.2 แสดงโค้ดส่วนที่ 2 ของโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด สำหรับโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในฟอร์มสุดท้าย..	83
ข.3 แสดงโค้ดโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับโปรแกรม เฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในตารางฐานข้อมูล.....	84
ข.4 แสดงโค้ดโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด สำหรับโปรแกรมเฉพาะที่พัฒนาด้วยภาษา C#.....	85
ข.5 แสดงโค้ดโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด สำหรับใช้กับข้อมูลสมาชิก โปรแกรมร้านเช่าหนังสือ.....	86
ข.5 แสดงโค้ดโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด สำหรับใช้กับข้อมูลสมาชิก โปรแกรม MyRestuarant.....	87

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

ปัจจุบันมีการนำคอมพิวเตอร์เข้ามาใช้ในชีวิตประจำวันอย่างแพร่หลาย รวมไปถึงในหน่วยงานต่างๆ และสิ่งที่ตามมาคือ ได้มีการพัฒนาโปรแกรมเพื่อใช้งานกับคอมพิวเตอร์มากขึ้น ไม่ว่าจะเป็นโปรแกรมจัดการเอกสารที่ใช้ตามสำนักงาน โปรแกรมจัดการรูปภาพที่ใช้ในงานถ่ายภาพ หรือโปรแกรมเล่นเพลงที่ใช้อยู่ตามบ้าน ฯลฯ ก็ล้วนถูกพัฒนาขึ้นอย่างต่อเนื่อง เพื่อตอบสนองความต้องการของผู้ใช้

โปรแกรมเฉพาะ คือโปรแกรมที่เขียนขึ้นมาเพื่อใช้ทำงานเฉพาะอย่าง ซึ่งในบางโอกาสอาจพบข้อจำกัด ซึ่งเป็นอุปสรรคในการนำมาใช้งานจึงต้องมีการพัฒนา โปรแกรมเฉพาะเหล่านี้ให้สามารถทำงานได้อย่างเต็มประสิทธิภาพ และครอบคลุมการให้งานที่สุด การพัฒนาโปรแกรมขึ้นมา เพื่อใช้งานกับ โปรแกรมเฉพาะนี้ก็จะเป็นอีกแนวทางหนึ่งในการพัฒนาด้านการใช้งานเช่นกัน

การใช้งานโปรแกรมสำเร็จรูปที่สามารถรับค่าอินพุตจากคีย์บอร์ดเท่านั้น สิ่งที่เป็นอุปสรรคคือ เราไม่สามารถนำข้อมูลผ่านทางช่องทางอื่นที่สะดวกกว่าได้ เช่น หากเรามีไฟล์ข้อมูลเป็นไฟล์อักษรอยู่แล้ว หากเราสามารถนำไฟล์อักษรนี้เข้าไปเป็นอินพุตได้โดยไม่ต้องพิมพ์จากคีย์บอร์ด การใช้งานก็จะสะดวกรวดเร็วขึ้น จึงได้มีแนวคิดในการที่จะพัฒนาโปรแกรมขึ้นมาเพื่อแปลงค่าจากไฟล์อักษร ให้เป็นอินพุตส่งต่อไปยังโปรแกรมเฉพาะนี้

แนวคิดในการพัฒนาโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุตของคีย์บอร์ดเพื่อใช้กับ โปรแกรมเฉพาะนี้ มีจุดมุ่งหมายสำคัญที่จะทำให้ผู้ใช้งานสามารถใช้โปรแกรมได้อย่างมีประสิทธิภาพ สะดวก และรวดเร็ว ประหยัดเวลาในการพิมพ์ข้อมูลเข้าสู่โปรแกรมในกรณีที่มีไฟล์ข้อมูลตัวอักษรในส่วนของงานนั้นอยู่แล้ว จึงคาดหวังเป็นอย่างยิ่งว่าจะได้รับประโยชน์สูงสุดในการศึกษาปฏิบัติโครงการครั้งนี้

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อศึกษาวิธีการในการแปลงไฟล์อักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับโปรแกรมเฉพาะ
2. เพื่อออกแบบโปรแกรมแปลงไฟล์อักษรเป็นสัญญาณอินพุทของคีย์บอร์ดได้
3. เพื่อให้ได้โปรแกรมที่สามารถแปลงไฟล์อักษรเป็นสัญญาณอินพุทของคีย์บอร์ด

1.3 ขอบข่ายของโครงการ

1. ศึกษาโปรแกรมเฉพาะที่สามารถรับค่าอินพุทจากคีย์บอร์ดได้เท่านั้น
2. ศึกษาวิธีการทำงานของโปรแกรมและภาษาที่ใช้ในการพัฒนาโปรแกรมนั้น
3. นำผลที่ศึกษาได้มาทำการวิเคราะห์และออกแบบโปรแกรมที่จะแปลงไฟล์อักษรเป็นสัญญาณอินพุทของคีย์บอร์ดเพื่อให้โปรแกรมเฉพาะใช้งานอย่างมีประสิทธิภาพมากขึ้น

1.4 ขั้นตอนของการดำเนินงาน

1. ศึกษาและค้นคว้าข้อมูลเกี่ยวกับโปรแกรมเฉพาะที่สามารถรับค่าอินพุทจากคีย์บอร์ดได้เท่านั้น ว่าทำงานอย่างไร และใช้ภาษาอะไรในการพัฒนาโปรแกรมนั้น
2. ออกแบบโปรแกรมที่จะมาช่วยในการทำงานของโปรแกรมเฉพาะ ให้มีประสิทธิภาพมากขึ้น
3. เขียนโปรแกรม เพื่อสามารถใช้งานร่วมกับโปรแกรมเฉพาะตัวนั้นได้
4. สรุปผลการทำงานและจัดทำรูปเล่มโครงการ

1.5 แผนการดำเนินงาน

กิจกรรม	ปี 2551 - 2552										
	มี.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	
1. ศึกษาค้นคว้าเกี่ยวกับหัวข้อที่ใช้ในการทำโครงการ ว่ามีความรู้เรื่องใดเข้ามาเกี่ยวข้องกับ	←→										
2. ทำความเข้าใจเกี่ยวกับการใช้งาน โปรแกรมเฉพาะที่สามารถรับค่าอินพุตจากคีย์บอร์ดได้เพียงอย่างเดียว		←→									
3. พัฒนาโปรแกรมเพื่อนำมาใช้งานกับ โปรแกรมเฉพาะ				←→							
4. ทดสอบ โปรแกรม และ ค้นหาข้อผิดพลาดในการใช้งานจริง							←→				
5. สรุปผล โปรแกรมที่สมบูรณ์ และจัดทำรูปเล่มโครงการ								←→			

1.6 ผลที่คาดว่าจะได้รับ

1. เพื่อให้ได้โปรแกรมแปลงไฟล์อักษรเป็นสัญญาณอินพุตของคีย์บอร์ด
2. สามารถพัฒนา โปรแกรมเพื่อให้ใช้งานกับ โปรแกรมบางชนิดได้
3. สามารถนำไปใช้จริงในชีวิตประจำวันได้อย่างมีประสิทธิภาพ

1.7 งบประมาณของโครงการ

1. ค่าถ่ายเอกสารและค่าเช่าเล่มโครงการฉบับสมบูรณ์	เป็นเงิน	1,000 บาท
2. ค่าวัสดุสำนักงาน	เป็นเงิน	400 บาท
3. ค่าวัสดุคอมพิวเตอร์	เป็นเงิน	<u>600 บาท</u>
	รวมเป็นเงินทั้งสิ้น	<u>2,000 บาท</u>

(สองพันบาทถ้วน)

หมายเหตุ ถัวเฉลี่ยทุกรายการ



บทที่ 2

หลักการและทฤษฎีที่เกี่ยวข้อง

2.1 หลักการการทำงานของคีย์บอร์ด

คีย์บอร์ดเป็นอุปกรณ์รับเข้าพื้นฐานที่ต้องมีในคอมพิวเตอร์ทุกเครื่อง ซึ่งจะรับข้อมูลจากการกดแป้นพิมพ์ แล้วเปลี่ยนเป็นรหัสเพื่อส่งต่อไปให้กับคอมพิวเตอร์ แป้นพิมพ์ที่ใช้ในการป้อนข้อมูลจะมีจำนวนตั้งแต่ 50 แป้นขึ้นไป แผงแป้นอักขระส่วนใหญ่จะมีแป้นตัวเลขแยกไว้ต่างหาก เพื่อที่จะทำการป้อนข้อมูลที่เป็นตัวเลขได้ง่ายและสะดวกขึ้น

การวางตำแหน่งของแป้นพิมพ์จะเป็นไปตามมาตรฐานของระบบพิมพ์สัมผัสเครื่องพิมพ์ดีด ที่จะมีการใช้แป้นยกแตร (Shift) เพื่อให้สามารถพิมพ์อักษรอังกฤษได้ทั้งตัวพิมพ์เล็กและตัวพิมพ์ใหญ่ ซึ่งระบบจะรับรหัสตัวอักษรอังกฤษส่งเป็นรหัส 7 หรือ 8 บิต นั่นคือ เมื่อมีการกดแป้นพิมพ์แผงอักขระจะส่งรหัส 7 หรือ 8 บิต นี้เข้าไปในคอมพิวเตอร์เพื่อนำค่าไปแสดงผลลัพท์ทางหน้าจอคอมพิวเตอร์ ส่วนกลุ่มแป้นพิมพ์ที่ใช้พิมพ์ภาษาไทยก็จะเป็นแป้นพิมพ์เดียวกับภาษาอังกฤษ แต่ก็จะใช้แป้นพิมพ์ลักษณะพิเศษ เป็นหนึ่งทำหน้าที่สลับเปลี่ยนการพิมพ์ภาษาไทยหรือภาษาอังกฤษภายใต้การควบคุมของซอฟต์แวร์อีกชั้นหนึ่ง ปัจจุบันได้แบ่งคีย์บอร์ดของคอมพิวเตอร์ออกเป็น 2 ประเภท คือ

2.1.1 แบบสัมผัส

เป็นคีย์บอร์ดซึ่งการกดแป้นคีย์จะทำให้กระแสไฟฟ้าไหลครบวงจร ซีปที่อยู่ในคีย์บอร์ดจะเป็นตัวเก็บค่าในรูปแบบสแกนโค้ดไว้ในหน่วยความจำชั่วคราว และส่งสัญญาณขัดจังหวะไปยังซีพียูเพื่อทำการประมวลผลต่อไป โดยจะมีไบออสทำหน้าที่แบบสแกนโค้ดเป็นรหัสแอสกี

2.1.2 แบบประจุ

วิธีนี้จะเป็นการทำให้คีย์บอร์ดเกิดการถ่ายเทประจุระหว่างเพลตโลหะใต้แป้นพิมพ์ เมื่อปริมาณประจุไฟฟ้าในตัวเก็บประจุเปลี่ยนจะทำให้กระแสไฟฟ้าขึ้น จะทำให้สามารถรู้ว่าเป็นใดถูกกด

คีย์บอร์ด (Keyboard) นั้น เมื่อพิจารณารายละเอียดแยกลงไป ก็จะได้เห็นว่าแต่ละปุ่มนั้นก็คือ สวิตช์ (Switch) ไฟฟ้าอันหนึ่งเท่านั้นเอง การกดคีย์บอร์ดก็คือ การกดสวิตช์เหล่านั้นให้เชื่อมต่อวงจรไฟฟ้าให้ถึงกันเกิดเป็นสัญญาณไปยังตัวเครื่อง ซึ่งหน่วยประมวลผลจะรับรู้ และตีความสัญญาณเหล่านั้นอีกทีหนึ่งว่าเป็นคำสั่งให้ทำอะไร ปัจจุบันนี้ตัวคีย์บอร์ดก็ยังมีเทคโนโลยีที่แตกต่างกันออกไปหลายแบบ ทำให้มีลักษณะ และการทำงานแตกต่างกันออกไป แบบที่แพร่หลายที่สุดก็คือ การใช้สวิตช์ไฟฟ้าแบบธรรมดาที่มีการแตะกันของหน้าสัมผัสหรือเรียกว่า Contact เพื่อให้กระแสไฟฟ้าไหลผ่าน แต่ปัจจุบันนี้ได้มีการนำเอาสวิตช์แบบหน้าสัมผัสคาร์บอนมาใช้กันมากขึ้น

ตามปกติเมื่อสวิตช์ไฟฟ้าแบบธรรมดา มักจะมีสายเส้นหนึ่งผ่านสวิตช์ตัวหนึ่ง เมื่อกดสวิตช์ก็จะมีกระแสไหลในสายเส้นนั้น ทำให้รับรู้ได้ว่ามีการกดสวิตช์หรือไม่ แต่ในกรณีของ คีย์บอร์ดแล้ว

จำนวนสวิทช์หรือคีย์มีมากถึง 101 ถึง 105 คีย์ จึงมีวิธีการที่พยายามจะลดสายที่ต่อกับคีย์ต่าง ๆ ลง โดยจัดสวิทช์ให้อยู่ในรูปของตารางขนาด 13×8 โดยมีสายในแนวตั้ง 8 สาย และแนวนอน 13 สายพาดผ่านกันเกิดเป็นจุดตัดของสาย 104 จุด ซึ่งจุดเหล่านั้นคือ ตำแหน่งที่จะวางสวิทช์แต่ละตัว สายในแนวตั้งเส้นหนึ่งกับสายในแนวนอนอีกเส้นหนึ่งจะต่อถึงกัน ทำให้มีกระแส

ไฟฟ้าไหลในสายคู่หนึ่งวิธีนี้จะเป็นการช่วยลดจำนวนสายที่ต่อจากสวิทช์หรือคีย์ลงไปได้มาก เช่น ในกรณีของ 165 คีย์ จะเหลือเพียง 8 สาย เท่านั้น หรือ 64 คีย์ก็เหลือเพียง $8 + 8 = 16$ สาย เมื่อจัดคีย์บอร์ดเป็นรูปเมตริกซ์ (Matrix) แล้วการจะบอกให้เครื่องรู้ว่าคีย์ใดถูกกด ก็ทำได้โดย การกวาด ดูกระแสไฟฟ้า ที่อยู่ในสายต่าง ๆ ว่ามีการเปลี่ยนแปลงไปอย่างไร โดยเริ่มจากสายที่ต่อกับเมตริกซ์ ด้านใดด้านหนึ่ง จะเริ่มส่งข้อมูลออกมาในลักษณะสัญญาณดิจิทัล คือมีเพียง 2 สถานะ ที่เป็น 0 คือไม่มีสัญญาณ และ 1 มีไฟฟ้า

สัญญาณที่ส่งออกมาจึงอาจเขียนแทนเป็นไฟฟ้ารหัสได้ เช่น ถ้า เมตริกซ์มีด้านละ 4 สาย ก็เขียนได้เป็น 1000, 0100, 0100, 0000 โดยรหัสที่ออกมานั้น จะมีเพียงสายเดียวเท่านั้นที่เป็น 1 และ 1 ตัวนี้จะถูกกวาดไปตามสายที่ 4 อย่างทั่วถึงภายในที่รวดเร็ว ส่วนอีกด้านหนึ่งของเมตริกซ์นั้น สายอีก 4 สายที่เหลืออยู่ก็จะทำหน้าที่เป็นตัวรับสัญญาณตามปกติ ถ้าไม่มีการกดคีย์ใด ๆ สายทั้ง 2 ชุดคือแนวนอนและแนวตั้ง จะไม่ต่อถึงกันเลย ดังนั้นไม่ว่าแนวตั้งจะส่ง รหัสอะไรออกมา สายทางแนวนอนจะรับสัญญาณได้เป็น 0 หมาดเสมอ แต่ถ้ามีการกดคีย์ใด ๆ เมื่อเลข 1 ถูกกวาดมาตรงกับสายที่ผ่านคีย์นั้น สัญญาณเลข 1 คือ มีกระแสไฟฟ้าก็จะถ่ายทอดผ่านคีย์มาถึงสายทางแนวนอนด้วยทำให้ทางแนวนอนรับรู้ได้ว่ามีการกดคีย์ใด โดยดูจากสัญญาณที่ส่งออกมา ขณะนั้นว่าสายใดเป็นสาย 1 และสายใดที่รับ 1 ดังกล่าวได้ ก็แสดงว่าสวิทช์หรือคีย์ที่ถูกกด คือ ตัวที่อยู่ในตำแหน่งของสายทั้งสองนั้นตัดกัน วงจรภายในของคีย์บอร์ดก็จะนำสัญญาณทั้งสองนี้ซึ่งเป็นตัวบอกว่าคีย์ใดถูกกด ไปเข้ารหัสแทนตัวอักษรประจำคีย์นั้นแล้วส่งให้แก่หน่วยประมวลผลต่อไป การติดต่อกับหน่วยประมวลผล

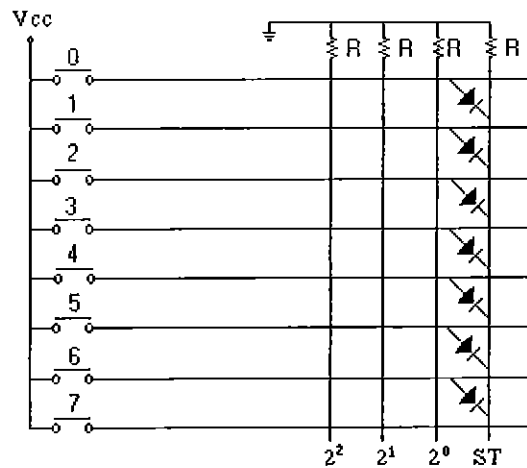
โดยทั่วไปคีย์บอร์ดนั้นจะมีวงจรควบคุมโดยเฉพาะที่เรียกว่า คีย์บอร์ดคอนโทรลเลอร์ (Keyboard Controller) ซึ่งอาจจะเป็นวงจรแบบวอยไปจนถึงไมโครโปรเซสเซอร์ขนาดเล็ก สำหรับงานควบคุมโดยเฉพาะ ทั้งนี้ขึ้นอยู่กับความซับซ้อนของการทำงานของคีย์บอร์ดแต่ละเครื่อง ตัวคอนโทรลเลอร์นี้เองจะเป็นตัวที่ทำหน้าที่สแกนเมตริกซ์ของคีย์เพื่อดูว่ามีการกดคีย์ใดจัดการเข้ารหัสข้อมูลที่ตรงกับสัญญาณประจำคีย์นั้น ๆ เพื่อส่งข้อมูลให้แก่หน่วยประมวลผล และยังทำหน้าที่เกี่ยวกับการแก้ปัญหาเรื่อง Bounce และ Rollover ด้วย ในการเข้ารหัสจากการกดคีย์เพื่อส่งให้หน่วยประมวลผลนั้น บางแบบอาจจะเข้ารหัสเป็นรหัสมาตรฐาน แทนข้อมูลที่จะนำไปประมวลผลจริง ๆ ไปเลย เช่น รหัส ASCII ซึ่งย่อมาจาก American Standard Code For Information Interchange ในกรณีนี้ข้อมูลที่รับไปจะสามารถนำไปประมวลผลได้ทันที กับอีกวิธีหนึ่งคือการเข้ารหัสพิเศษที่เรียกว่า SCAN CODE ซึ่งเป็นรหัสเฉพาะที่ใช้ภายในเครื่อง อาจจะเหมือนหรือต่างกับรหัสที่แทนข้อมูลจริง เช่น รหัส ASCII เป็นบางส่วนก็ได้ เมื่อหน่วยประมวลผลรับ Scan code นี้ไปแล้ว จะไปแปลงเป็นข้อมูลจริงอีกที

ในการส่งข้อมูลจาก Keyboard Controller ไปยังหน่วยประมวลผลนั้น ในบางกรณี อาจจะทำให้ทำการส่งข้อมูลแบบอนุกรมหรือ Serial คือส่งทีละบิตต่อเนื่องกันไปตามลำดับแทนที่จะส่งบิตพร้อม ๆ กันไปแบบขนาน (Parallel) ทั้งนี้เพราะบางครั้งสายที่ต่อ ระหว่างคีย์บอร์ด กับตัวเครื่อง อาจเป็นสายเส้นเดียวเพื่อความสะดวกในการใช้งานให้คีย์บอร์ดแยกออกมาต่างจากตัวเครื่องได้ หรือบางทีก็อาจจะไม่มีสาย แต่ใช้การส่งข้อมูลด้วยแสงอินฟราเรดแทน ซึ่งจะส่งข้อมูลได้เพียงทีละบิตเท่านั้น ดังนั้น ในกรณีเหล่านี้จึงต้องมีอุปกรณ์พิเศษที่ทำหน้าที่ส่งสัญญาณข้อมูลทั้ง 8 บิต หรือ 16 บิต จากคีย์บอร์ดให้เรียงต่อกันออกไปทีละบิต และทางตัวรับที่เครื่องนั้นก็จะต้องมีอุปกรณ์ประเภทเดียวกันที่ทำหน้าที่นำสัญญาณข้อมูลที่มาทีละบิตกลับมาเรียงต่อกันเป็น 1 ไบต์ 8 บิต หรือ 2 ไบต์ 16 บิต เพื่อส่งให้แก่หน่วยประมวลผลอีกทีหนึ่งในแบบขนานผ่านทางบัสข้อมูลในกรณีการติดต่อ กับหน่วยประมวลผล ไม่ว่าจะเป็นแบบ Polling หรือ Interrupt ก็จะทำให้ตัวอุปกรณ์แปลง ข้อมูลนี้แทนที่จะเป็น Controller ที่อยู่แยกออกไป

2.2 การเข้ารหัส และ ถอดรหัส (Encoder and Decoder)

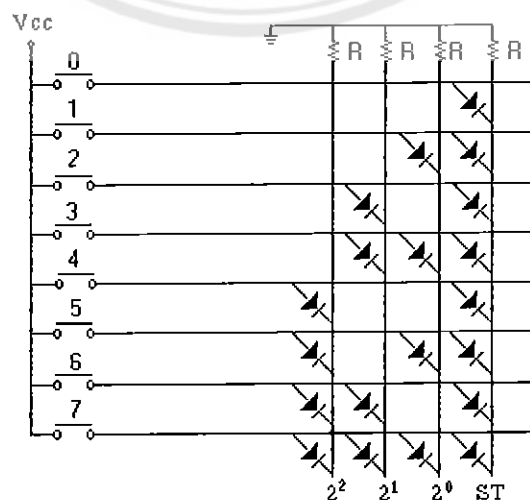
2.2.1 การเข้ารหัส (Encoder)

ในเรื่องของวงจร Logic แล้ว หมายถึง การเปลี่ยนแปลงระดับของ Logic จากสวิตช์มาเป็นสัญญาณ Logic ตามรหัสที่เราต้องการ เช่น เรามีสวิตช์อยู่ 10 ตัว กดสวิตช์เลขหมายใดก็ให้ Output ของวงจรเป็นสัญญาณ Logic ของรหัส Binary เป็นต้น ซึ่งเราจะให้ Output ของวงจรเป็นรหัสอะไรก็ขึ้นอยู่กับความต้องการของผู้ใช้ ในทำนองเดียวกัน การถอดรหัส (Decoder) ก็หมายความว่า วงจรซึ่งเปลี่ยนรหัสทาง Input เพื่อเป็นรหัสอื่นตามต้องการ เช่น เปลี่ยนรหัส BCD เป็นแรงดันของ เลขฐานสิบ เป็นต้น วงจรเข้ารหัส - ถอดรหัส สามารถสร้างขึ้นมาจาก Diode ซึ่งต่อกันแบบ Matrix หรือ จากวงจร Gate Combination ก็ได้ การออกแบบวงจรเข้ารหัส เราอาจสร้างวงจรเข้ารหัสได้จาก Diode Matrix หรือ วงจร Gate การสร้างวงจร จาก Diode Matrix จากความต้องการเปลี่ยนสัญญาณ Logic จากสวิตช์ 8 ตัว เพื่อให้ได้ Output เป็น Octal code ซึ่ง Output ต้องมี 3 Output คือ Output 2^2 , 2^1 และ 2^0 นอกจากนี้ควรจะมีอีก Output หนึ่งใช้แสดงการกดหมายเลขเพิ่มขึ้นมา เรียกว่า Output Strobe ส่วนทาง Input จะต้องเป็น 8 เส้น เท่ากับจำนวนของสวิตช์ นั่นคือสามารถเขียนเส้น Matrix ได้เป็น 8×4 ดังรูปที่ 2.1



รูปที่ 2.1 แสดงเส้น Matrix 8 × 4

เมื่อได้เส้น Matrix แล้ว ก็ต่อสวิทช์เข้ากับเส้น Matrix ส่วนอีกปลายด้านหนึ่งของสวิทช์ต่อร่วมกับ +Vcc สำหรับด้าน Output 4 เส้น ต่อปลายเข้ากับความต้านทาน ปลายของความต้านทาน ต่อลง Ground จากนั้นก็มาพิจารณา Output Strobe (ST) Output นี้จะเป็น Logic 1 ทุกครั้งที่สวิทช์ Input ใด Input หนึ่งถูกกด นั่นคือ เป็นลักษณะการ OR ของ Diode โดยด้าน Cathode ต่อร่วมกับขา ST ส่วนด้าน Anode ก็ต่อเข้ากับสวิทช์แต่ละตัว จากนั้นลองตรวจดู โดยการสมมุติว่าสวิทช์ตัวใดตัวหนึ่งถูกกด แล้วจะได้ Output เป็น Logic 1 จริงหรือไม่ เช่น สมมุติว่า สวิทช์หมายเลข 4 ถูกกด แรงดันจาก +Vcc ก็ จะผ่าน Diode ไปตกคร่อม R เป็น Output ST โดยที่แรงดันดังกล่าวไม่สามารถผ่านย้อนไปยัง Diode ตัวอื่นได้เลย ต่อไปก็ดูว่า รหัสทางด้าน Output เป็นอย่างไรถ้าสวิทช์ถูกกด เช่น ถ้ากดสวิทช์หมายเลข 7 Output จะต้องเป็น Logic 1 หากเราใส่ Diode ในลักษณะเดิมตามเส้น Output ดังรูปที่ 2.2



รูปที่ 2.2 แสดงวงจรเข้ารหัส

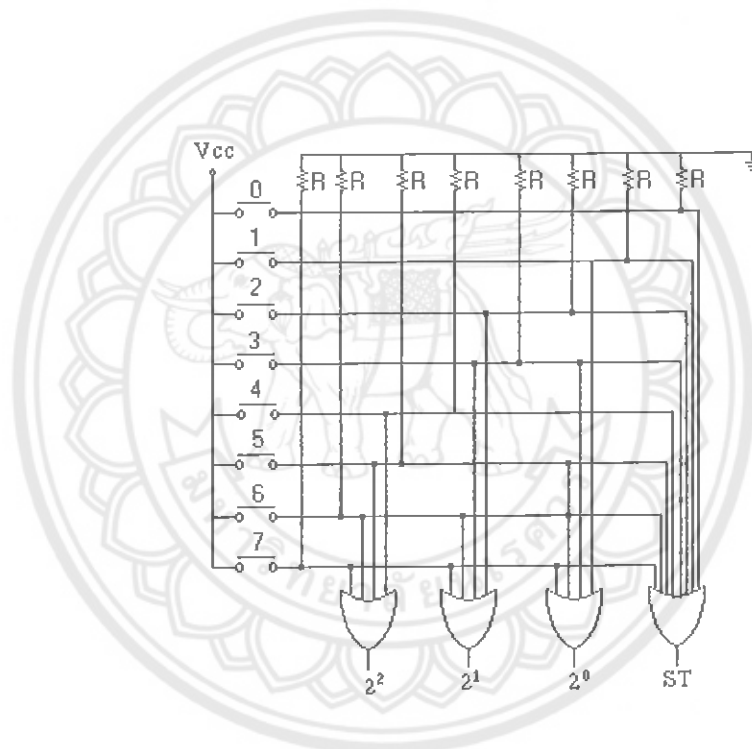
สำหรับสวิตช์ใดที่ให้ Output บาง Output เป็น Logic 0 แลของสวิตช์นั้นใน Output เป็น Logic 0 ก็ปล่อยว่าง เช่น สวิตช์หมายเลข 6 รหัสจะเป็น 110 ดังนั้น Output 20 ที่แถว 6 จะไม่มี Diode ต่ออยู่ ข้อสังเกตคือ สวิตช์ 0 จะไม่มี Diode ต่ออยู่เลย แต่ Output ST ก็ยังเป็น Logic 1 ถ้าสวิตช์นี้ถูกกด นั่นคือข้อแตกต่างระหว่างการกดสวิตช์หมายเลข 0 กับการไม่กดซึ่งจะไม่มี Logic 1 ที่ OUT ST ในกรณีที่เราใช้วงจร Logic เราเขียน Function ได้ดัง

$$ST = 0 + 1 + 2 + 3 + 4 + 5 + 6 + 7$$

$$2^0 = 1 + 3 + 5 + 7$$

$$2^1 = 2 + 3 + 6 + 7$$

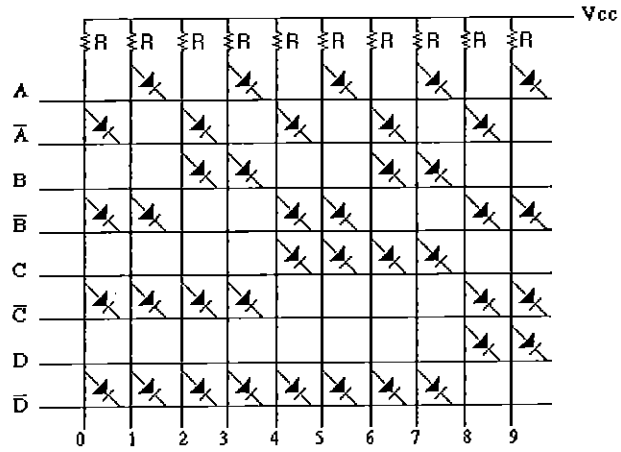
$$2^2 = 4 + 5 + 6 + 7$$



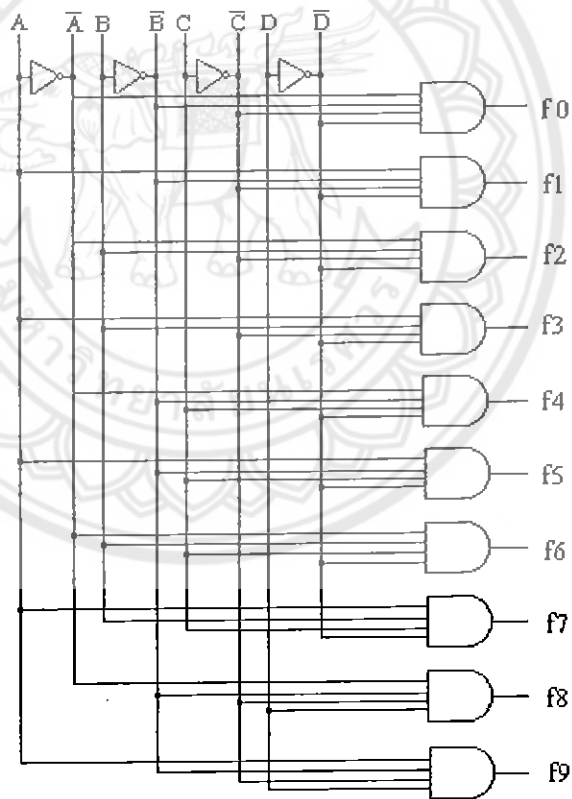
รูปที่ 2.3 แสดงการใช้วงจร Logic เขียน Function

2.2.2 ถอดรหัส (Decoder)

เมื่อมีวงจรเข้ารหัส ก็ต้องมีวงจรถอดรหัส (Decoder) วงจรถอดรหัสนี้จะรับสัญญาณระดับ Logic ในรูปของรหัส เช่น รหัส Binary BCD เมื่อปล่อยออกมาเป็นเลขฐานสิบ ซึ่งก็ทำได้ทั้งจากวงจร Diode และ Gate combination



รูปที่ 2.4 แสดง Diode Matrix สำหรับวงจรถอดรหัส BCD -8421 เป็นเลขฐานสิบ



รูปที่ 2.5 แสดงวงจร Gate combination สำหรับวงจรถอดรหัส BCD -8421 เป็นเลขฐานสิบ

วงจรถอดรหัสที่กล่าวมาเป็นแนวทางในการศึกษาโครงสร้าง และรายละเอียดตลอดจนความคิด และการออกแบบ เพื่อให้เข้าใจการทำงานของวงจร ซึ่งใจทางปฏิบัติแล้วบริษัทผู้ผลิตจะผลิตเป็น IC

Chip เพียง Chip เดียว ไม่ต้องยุ่งยากออกแบบ เช่น ถ้าเป็น BCD to Decimal Decoder ก็จะเลือกใช้ IC เบอร์ 7441, 7442 หรือ 7445 ถ้าเป็น Excess - 3 to Decimal Decoder ก็เลือกใช้ IC เบอร์ 7443 หรือ 7444 ถ้าเป็น BCD to 7 Segment ก็เลือกใช้ IC เบอร์ 7446, 7447, 7448 หรือ 7449 เป็นต้น

2.3 รหัสแอสกี (American Standard Code Interchange, ASCII)

เดิมการแทนรหัสฐานสองด้วยพยัญชนะในภาษาต่างๆเป็นการกำหนดกันเอง ขึ้นอยู่กับว่าใครพัฒนาขึ้นมาทำให้การส่งผ่านข้อมูลระหว่างเครื่องคอมพิวเตอร์ต่างชนิดไม่สามารถส่งผ่านกันได้ เพราะใช้รหัสในเลขฐานสองไม่เหมือนกัน ดังนั้นเพื่อให้สื่อสารกันได้ จึงจำเป็นต้องมีการ กำหนดมาตรฐานของรหัสขึ้น สำหรับภาษาอังกฤษ เรียกว่า รหัสแอสกี (American Standard Code Interchange, ASCII)

รหัสที่เป็นมาตรฐาน คือ รหัส ASCII ย่อมาจาก American Standard Code For Information Interchange (ASCII) อ่านว่า แอส-กี เป็นรหัสที่พัฒนาขึ้นโดยสถาบันมาตรฐานแห่งชาติสหรัฐอเมริกา (American National Standard Institute: ANSI อ่านว่า แอน-ชาย) เรียกว่า ASCII Code ซึ่งเป็นที่นิยมในกลุ่มผู้สร้างเครื่องคอมพิวเตอร์ทั่วไป รหัสนี้ได้มาจากรหัสขององค์การมาตรฐานระหว่างประเทศ (International Standardization Organization: ISO) ขนาด 7 บิต ซึ่งสามารถสร้างรหัสที่แตกต่างกันได้ถึง 128 รหัส (ตั้งแต่ 000 0000 ถึง 111 1111) โดยกำหนดให้ 32 รหัสแรกเป็น 000 0000 ถึง 001 1111 ทำหน้าที่เป็นสั่งควบคุม เช่น รหัส 000 1010 แทนการเลื่อนบรรทัด (Line Feed) ในเครื่องพิมพ์ เป็นต้น และอีก 96 รหัสถัดไป (32-95) ใช้แทนอักษรและสัญลักษณ์พิเศษอื่นๆ รหัส ASCII ใช้วิธีการกำหนดการแทนรหัสเป็นเลขฐานสิบ ทำให้ง่ายต่อการจำและใช้งาน นอกจากนั้นยังสามารถเขียนมโนรูปของเลขฐานสิบหกได้ด้วย ดังนั้น ASCII Code จึงเป็นรหัสที่เขียนได้ 3 แบบ รหัส ASCII สามารถใช้แทนข้อมูลอักขระและคำสั่งได้มากขึ้น และมีการขยายเป็นรหัสแบบ 8 บิต

อักขระที่ไม่แสดงผลและมองไม่เห็นในการพิมพ์หรือการแสดงบนจอภาพอย่าง อักขระว่าง (Null character) ไม่ควรสับสนกับอักขระช่องว่าง (whitespace) โดยอักขระช่องว่างแสดงผลเป็นช่องว่างสำหรับเว้นวรรค เป็นต้น แต่อักขระว่างนั้น ไม่แสดงผล

2.4 หลักการทำงานของโปรแกรม AutoIt

AutoIt เป็นโปรแกรมที่มีภาษาคัดลอกภาษาเบสิกถูกออกแบบมาเพื่อสั่งการอัตโนมัติในรูปแบบของหน้าต่างกราฟิกและการเขียนคำสั่งสคริปต์ทั่วไป โดยเลียนแบบการกดแป้นพิมพ์ การขยับเมาส์ และการจัดการหน้าต่างหรือปุ่มควบคุมต่างๆ ในโปรแกรมอื่นๆ ให้เป็นไปตามลำดับโดยอัตโนมัติ ซึ่งไม่สามารถกระทำได้ในภาษาอื่น ซึ่งโปรแกรมนี้มีขนาดเล็กมาก และสามารถดำเนินการได้ด้วยตัวเอง อีกทั้งยังสามารถทำงานได้บนระบบปฏิบัติการ Window ทุกเวอร์ชัน โดยมีลักษณะเด่นคือ

- โครงสร้างของภาษาที่ใช้สามารถเรียนรู้ได้ง่าย
- สามารถเขียนแบบการกดคีย์บอร์ดและการขยับเมาส์ได้
- สามารถจัดการกับหน้าต่างและตัวปฏิบัติการของ โปรแกรมได้
- ใช้ได้กับตัวควบคุมพื้นฐานทุกชนิดของ Window
- สามารถสร้างส่วนติดต่อกับผู้ใช้ในรูปแบบของกราฟิก (GUI)
- เรียกใช้งาน โดยตรงกับ external DLL และ Window API
- เข้าได้กับระบบ Unicode และ x64
- ใช้ได้กับ Window 95/ 98/ ME/ NT4/ 2000/ XP/ 2003/ Vista/ 2008

2.4.1 เครื่องมือต่างๆ ของโปรแกรม

1. SciTE Script Editors เป็นเครื่องมือที่ใช้ในการเขียนสคริปต์ของโปรแกรม ซึ่งลักษณะพิเศษของเครื่องมือชนิดนี้คือ จะมีการเน้นสีส่วนของ โค้ดสคริปต์ที่เป็นคีย์เวิร์ดหรือฟังก์ชันของ โปรแกรม เพื่อให้สะดวกในการตรวจสอบความผิดพลาดของ โค้ดก่อนจะทำการรันสคริปต์

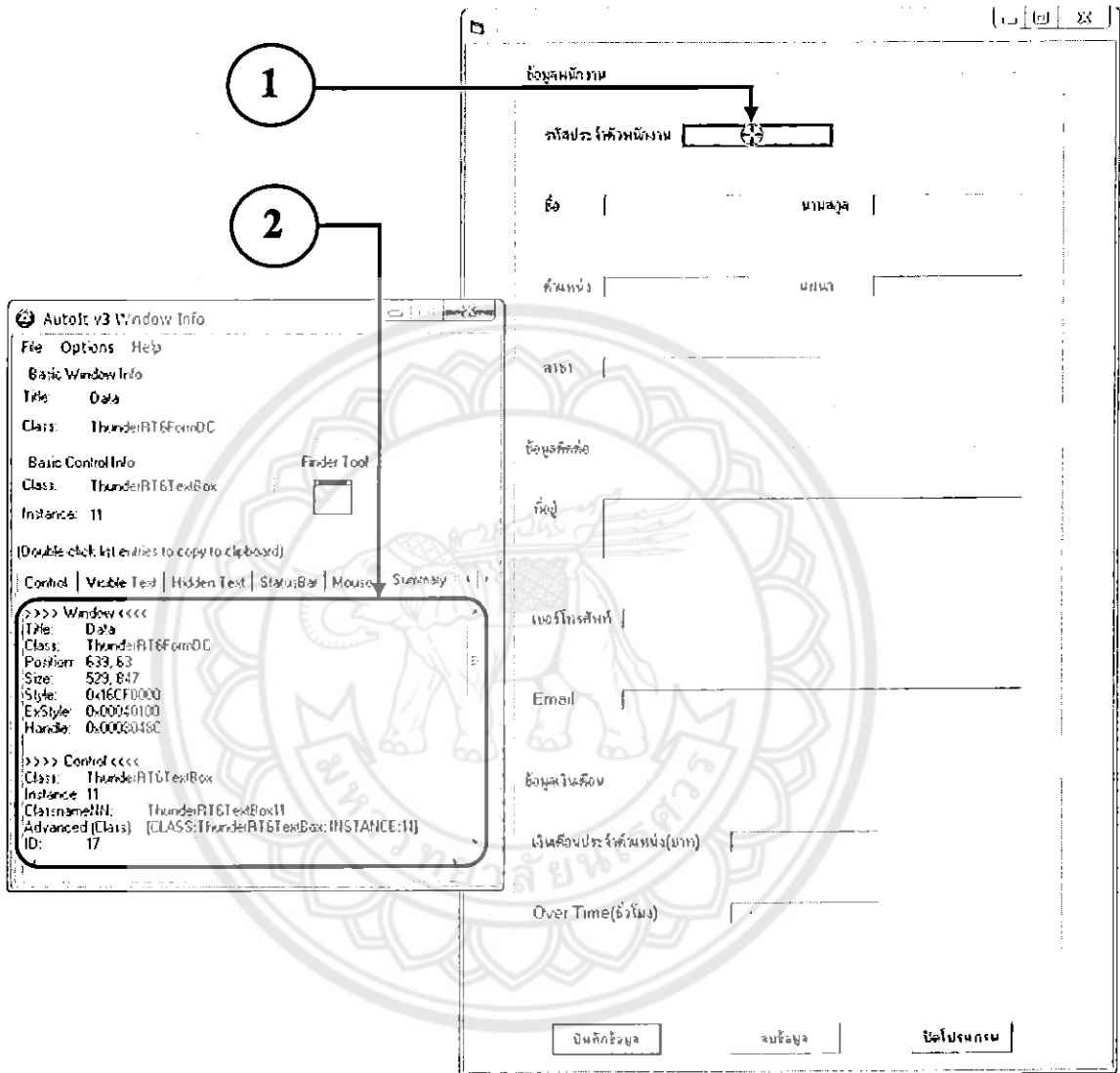
```

D:\Paboo Document\Project\Program\Project\VB1\RunProject\au1 - SciTE4AutoIt3
File Edit Search View Tools Options Language Buffers Help
D  Q Q
1 $file = FileOpenDialog( "Please select a file to open", "C:\Program Files\AutoIt3\Scripts", 1 )
2
3 IF $error Then
4   MsgBox(4096, "Error", "Please select a file to open")
5   Exit
6 EndIf
7
8 $line_no = 1
9
10 While True
11   $sline = FileReadLine($file,$line_no)
12   If $error Then
13     If $error = -1 Then
14       MsgBox(4096, "Error", "Please select a file to open")
15       ExitLoop
16     ElseIf $error = 1 Then
17       MsgBox(4096, "Error", "Please select a file to open")
18       ContinueLoop
19     EndIf
20   EndIf
21
22   $line_no = $line_no + 1
23   $col = StringSplit($sline, ",")
24
25   If $col[0] = 11 Then
26     WinWait( "AutoIt" )
27     ControlSend( "AutoIt", "17,$col[1] )
28     ControlSend( "AutoIt", "16,$col[2] )
29     ControlSend( "AutoIt", "15,$col[3] )
30     ControlSend( "AutoIt", "14,$col[4] )
31     ControlSend( "AutoIt", "13,$col[5] )
32     ControlSend( "AutoIt", "12,$col[6] )
33     ControlSend( "AutoIt", "7,$col[7] )
34     ControlSend( "AutoIt", "8,$col[8] )
35     ControlSend( "AutoIt", "6,$col[9] )
36     ControlSend( "AutoIt", "5,$col[10] )
37     ControlSend( "AutoIt", "4,$col[11] )
38
39     Sleep(1000)
40     ControlClick( "AutoIt", "10" )
41
42     WinWait( "msgbox" )
43     Sleep(1000)
44     ControlClick( "msgbox", "17" )
45   Else
46     MsgBox(4096, "Error", "Please select a file to open")
47   EndIf
48 WEnd
49 FileClose($file)
50
51 Exit

```

รูปที่ 2.6 แสดงลักษณะเครื่องมือ SciTE Script Editors ของโปรแกรม AutoIt

2. AutoIt Window Info เป็นเครื่องมือที่ใช้ในการระบุรายละเอียดของส่วนประกอบของโปรแกรมที่ต้องการเขียนสคริปต์ โดยมีอุปกรณ์ชื่อว่า Finder Tool สำหรับนำไปชี้ยังส่วนประกอบของโปรแกรมที่ต้องการสั่งการ แล้วจะมีการแสดงรายละเอียดต่างๆ บนหน้าต่างของ AutoIt Window Info

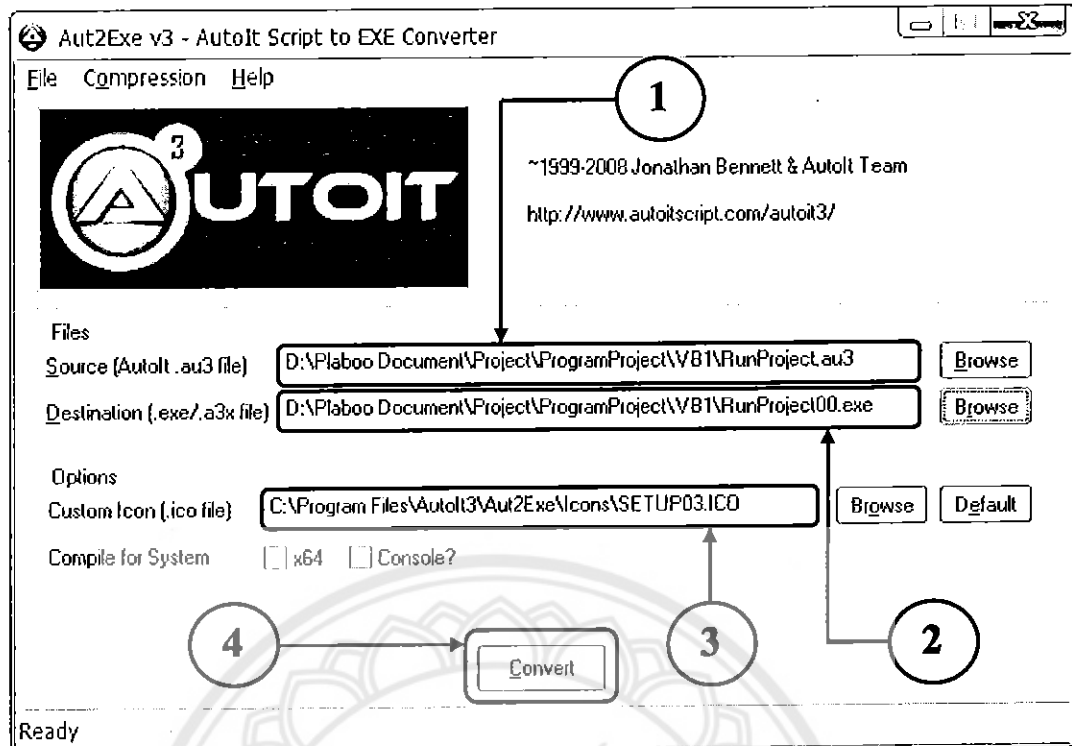


รูปที่ 2.7 แสดงลักษณะเครื่องมือ AutoIt Window Info ของโปรแกรม AutoIt

ส่วนที่ 1 แสดง Finder Tool ซึ่งจะสังเกตเห็นเมื่อนำไปชี้ที่ส่วนประกอบของโปรแกรมที่ต้องการทราบรายละเอียดแล้ว จะปรากฏกรอบสี่ล้อมรอบเครื่องมือขึ้น

ส่วนที่ 2 แสดง Information ซึ่งเป็นรายละเอียดของส่วนประกอบที่นำ Finder Tool ไปชี้ โดยรายละเอียดตรงส่วนนี้เองที่นำไปใช้ในการเขียนสคริปต์เพื่อสั่งการให้กับโปรแกรม

3. Aut2Exe เป็นเครื่องมือที่ใช้ในการเปลี่ยนคำสั่งในสคริปต์ให้เป็นตัวสั่งการของโปรแกรม (เปลี่ยนเป็นไฟล์นามสกุล .exe)



รูปที่ 2.8 แสดงลักษณะเครื่องมือ Aut2Exe ของ โปรแกรม AutoIt

ส่วนที่ 1 แสดง Source ซึ่งใช้รับไฟล์สคริปต์ที่เขียนด้วยโปรแกรม AutoIt (.au3) ที่ต้องการเปลี่ยนเป็นไฟล์สั่งการ (.exe)

ส่วนที่ 2 แสดง Destination ซึ่งใช้กำหนดชื่อไฟล์สั่งการ และที่อยู่ของไฟล์ซึ่งมีนามสกุลเป็น .exe หรือ .a3x

ส่วนที่ 3 แสดง Custom Icon ซึ่งใช้กำหนดรูปร่างของไอคอนไฟล์สั่งการ

ส่วนที่ 4 แสดง Convert ซึ่งเป็นปุ่มสำหรับการเริ่มการแปลงไฟล์สคริปต์ให้เป็นไฟล์สั่งการ

2.4.2 คุณสมบัติด้านกราฟิก (GUI Reference)

คือส่วนติดต่อกับผู้ใช้ที่โปรแกรม AutoIt สามารถสร้างขึ้นมาเพื่อใช้งานได้ ซึ่งประกอบด้วยส่วนต่างๆดังนี้

ตารางที่ 2.1 แสดงรูปแบบของกราฟิกที่สามารถใช้ได้ ใน AutoIt

Label	กล่องสำหรับใส่ข้อความ
Button	ปุ่มกด
Input	เส้นควบคุมที่สามารถใส่ข้อความได้
Edit	เส้นควบคุม(หลายเส้น)ที่สามารถใส่ข้อความได้
Checkbox	กล่องที่กำหนดการเลือกหรือไม่เลือก

ตารางที่ 2.1 (ต่อ) แสดงรูปแบบของกราฟิกที่สามารถใช้ได้ ใน AutoIt

Radio	ชุดของปุ่มวงกลม
Combo	กล่องรายการ
List	รายการข้อความ
Date	ตัวบอกเวลา
Pic	ตัวใส่รูปภาพ
Icon	ตัวใส่ไอคอน
Progress	แถบ progress
Tab	แถบของกลุ่มตัวควบคุมต่างๆ
UpDown	ตัวควบคุมที่สามารถเพิ่ม input control ได้
Avi	แสดงวิดีโอในรูปแบบของ AVI
Menu	กล่องเมนูซึ่งอยู่บริเวณด้านบนของหน้าต่างโปรแกรม
ContextMenu	กล่องเมนูที่ปรากฏเมื่อทำการคลิกขวา
TreeView	ตัวควบคุมที่แสดงหน้าต่าง file explorer.
Slider	ตัวควบคุมเลื่อน
Listview	ตัวควบคุมสำหรับแสดงข้อมูลในคอลัมน์
ListviewItem	ตัวควบคุมสำหรับแสดงรายการใน listview control
Graphic	ตัวควบคุมสำหรับแสดงกราฟิกของ GUICtrlSetGraphic
Dummy	ตัวจำลอง user control

2.4.3 ฟังก์ชันต่างๆใน AutoIt

1. การจัดการไฟล์ โดยเป็นฟังก์ชันสำหรับจัดการกับไฟล์ เช่น การเปิดปิดไฟล์ การอ่านไฟล์ การเขียนไฟล์ โดยมีตัวอย่างฟังก์ชันดังต่อไปนี้

ตารางที่ 2.2 แสดงฟังก์ชันการจัดการไฟล์ใน AutoIt

ConsoleRead	การอ่านจาก STDIN stream ของ AutoIt script process
ConsoleWrite	การเขียนข้อมูลลงบน STDOUT stream
ConsoleWriteError	การเขียนข้อมูลลงบน STDERR stream
DirCopy	คัดลอก directory และ sub directories ของไฟล์
DirCreate	สร้าง directory และ folder
DirGetSize	แสดงค่าขนาดของ byte ที่ต้องการใน directory

ตารางที่ 2.2 (ต่อ) แสดงฟังก์ชันการจัดการไฟล์ใน AutoIt

DirMove	ย้าย directory และ sub directories ของไฟล์
DirRemove	ลบ directory และ folder
FileChangeDir	เปลี่ยน directory ที่กำลังใช้งาน
FileClose	ปิด Text file ที่เปิดอยู่ก่อนหน้านี้
FileCopy	คัดลอกไฟล์
FileCreateShortcut	สร้างไฟล์ Shortcut (.lnk) จากไฟล์
FileDelete	ลบไฟล์
FileExists	ตรวจสอบว่าไฟล์หรือ directory มีอยู่จริง
FileGetShortcut	เรียกข้อมูลรายละเอียดของ Shortcut (.lnk)
FileGetSize	แสดงผลขนาดของไฟล์เป็น Byte
FileGetTime	แสดงผลเวลาและรายละเอียดของไฟล์
FileGetVersion	แสดงผลข้อมูลเวอร์ชันของไฟล์
FileInstall	Include และ Install ไฟล์
FileMove	ย้ายไฟล์
FileOpen	เปิด Text file สำหรับการอ่านและเขียน
FileOpenDialog	แสดงหน้าต่างเปิดไฟล์
FileRead	อ่านจำนวนอักขระของไฟล์ที่เปิดอยู่
FileReadLine	อ่านบรรทัดข้อความของไฟล์ที่เปิดอยู่
FileRecycle	นำไฟล์หรือ directory ไปยัง Recycle bin
FileRecycleEmpty	ลบข้อมูลใน Recycle bin
FileSaveDialog	แสดงหน้าต่างบันทึกไฟล์
FileSelectFolder	แสดงหน้าต่างค้นหาโฟลเดอร์

2. กล่องข้อความ เป็นฟังก์ชันที่ใช้แสดงกล่องข้อความเพื่อเตือนหรือถามคำตอบจากผู้ใช้งาน โดยมีหลายรูปแบบดังนี้

ตารางที่ 2.3 แสดงฟังก์ชันกล่องข้อความใน AutoIt

InputBox	แสดงกล่องข้อความถามค่าอินพุต
MsgBox	แสดงกล่องข้อความเพื่อเตือนหรือถามคำตอบ
ProgressOn	แสดงกล่องความคืบหน้าการทำงาน

ตารางที่ 2.3 (ต่อ) แสดงฟังก์ชันกล่องข้อความใน AutoIt

ProgressOff	ปิดกล่องความคืบหน้าการทำงาน
ProgressSet	กำหนดค่าต่างๆ ของกล่องความคืบหน้าการทำงาน
SplashImageOn	แสดงกล่อง pop up ที่เป็นรูปภาพ
SplashTextOn	แสดงกล่อง pop up ที่เป็นตัวอักษร
SplashOff	ปิดกล่อง pop up
ToolTip	สร้าง tooltip บนหน้าต่าง โปรแกรม

3. ควบคุมเมาส์ เป็นฟังก์ชันที่ใช้ควบคุมการทำงานของเมาส์ให้เป็นไปตามที่ผู้ใช้งานต้องการ โดยมีรูปแบบฟังก์ชันดังนี้

ตารางที่ 2.4 แสดงฟังก์ชันควบคุมเมาส์ใน AutoIt

MouseClick	สั่งการให้เมาส์คลิกตัวดำเนินการ
MouseClickDrag	สั่งการให้เมาส์คลิกและลากเมาส์
MouseDown	สั่งการให้เมาส์เลื่อนลงจากตำแหน่งปัจจุบัน
MouseGetCursor	แสดงเมาส์ในรูปแบบ Cursor ต่างๆ
MouseGetPos	แสดงเมาส์ในตำแหน่งที่กำหนด
MouseMove	เคลื่อนย้ายเมาส์
MouseUp	สั่งการให้เมาส์เลื่อนขึ้นจากตำแหน่งปัจจุบัน
MouseWheel	สั่งการให้หมุนล้อของเมาส์

4. การจัดการหน้าต่างโปรแกรม เป็นฟังก์ชันที่ใช้กับหน้าต่างโปรแกรมไม่ว่าจะเป็นการอรันสคริปต์เมื่อหน้าต่างโปรแกรมมีการเปิด หรือการปรับเปลี่ยนขนาดของหน้าต่างโปรแกรม โดยมีตัวอย่างฟังก์ชันดังนี้

ตารางที่ 2.5 แสดงฟังก์ชันการจัดการหน้าต่างโปรแกรมใน AutoIt

WinActivate	สั่งการให้เปิดหน้าต่างโปรแกรม
WinActive	ตรวจสอบการเปิดหน้าต่างโปรแกรม
WinClose	ปิดหน้าต่าง โปรแกรม
WinFlash	ย่อหน้าต่าง โปรแกรมลง taskbar

ตารางที่ 2.5 (ต่อ) แสดงฟังก์ชันการจัดการหน้าต่าง โปรแกรมใน AutoIt

WinGetText	ส่งค่าข้อความของหน้าต่าง โปรแกรม
WinGetTitle	ส่งค่าชื่อเต็มของหน้าต่าง โปรแกรม
WinList	ส่งค่ารายการของหน้าต่าง โปรแกรม
WinMenuItem	เรียก menu item ของหน้าต่าง โปรแกรม
WinMinimizeAll	ย่อขนาดทุกหน้าต่าง โปรแกรม
WinMinimizeAllUndo	ยกเลิกการย่อหน้าต่าง โปรแกรมก่อนหน้า
WinMove	ย้าย หรือปรับขนาดหน้าต่าง โปรแกรม
WinSetOnTop	ปรับหน้าต่าง โปรแกรมให้อยู่ข้างบนเสมอ
WinSetState	ปรับรูปแบบหน้าต่าง โปรแกรมให้ซ่อน, แสดง, ย่อ หรือขยายขนาด
WinSetTitle	เปลี่ยนชื่อหน้าต่าง โปรแกรม
WinSetTrans	ปรับลักษณะความโปร่งใสของหน้าต่าง โปรแกรม
WinWait	รอการเปิดหน้าต่าง โปรแกรมเพื่อรันสคริปต์
WinWaitActive	รอการเปิดใช้หน้าต่าง โปรแกรมเพื่อรันสคริปต์
WinWaitClose	รอการปิดหน้าต่าง โปรแกรม
WinWaitNotActive	รอการหยุดใช้งานหน้าต่าง โปรแกรม

5. คำสั่งควบคุม control ต่างๆ ของ โปรแกรม เป็นฟังก์ชันที่สั่งให้มีการกระทำต่อส่วนประกอบต่างๆ ใน โปรแกรม เช่น สั่งให้ทำการคลิกเมาส์ที่ปุ่มในโปรแกรม สั่งให้ทำการพิมพ์ข้อความลงกล่องใส่ข้อความในโปรแกรม เป็นต้น ซึ่งมีรูปแบบฟังก์ชันดังต่อไปนี้

ตารางที่ 2.6 แสดงฟังก์ชันคำสั่งควบคุม control ต่างๆ ของ โปรแกรมอื่นใน AutoIt

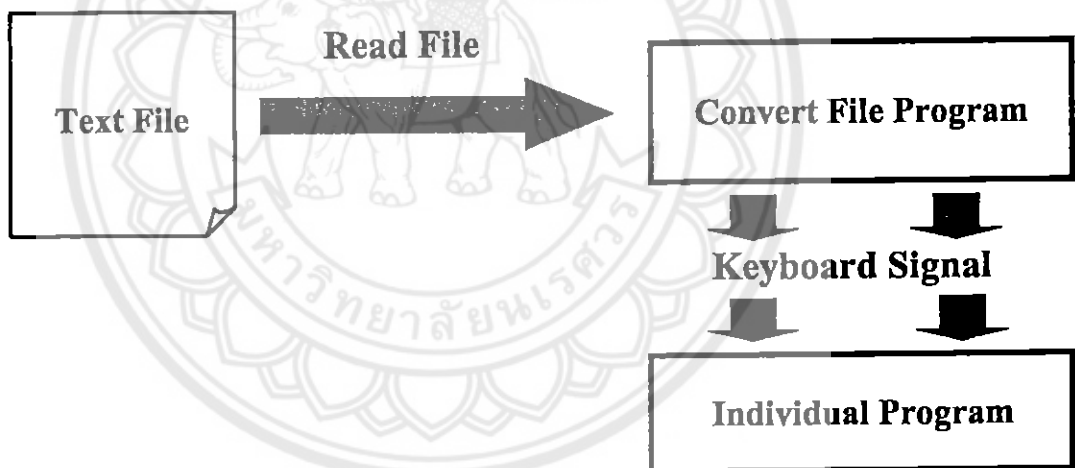
ControlClick	คลิกเมาส์ที่ปุ่มที่ต้องการ
ControlCommand	ส่งคำสั่งการ control ในโปรแกรม
ControlGetPos	ส่งค่าตำแหน่งและขนาดของหน้าต่าง โปรแกรม
ControlGetText	ส่งค่าข้อความ
ControlHide	ซ่อน control ในโปรแกรม
ControlMove	ย้าย control ในโปรแกรม
ControlSetText	ตั้งค่าข้อความของ control ในโปรแกรม
ControlShow	แสดง control เมื่อถูกซ่อนไว้

บทที่ 3

วิธีการดำเนินการ

3.1 ลักษณะการทำงานร่วมกันของโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดกับโปรแกรมเฉพาะ

ลักษณะการทำงานจะมีส่วนที่เข้ามาเกี่ยวข้องด้วยกัน 3 ส่วน คือ ส่วนของไฟล์อักษรที่จะนำข้อมูลมาใช้งาน ส่วนของโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด และส่วนของโปรแกรมเฉพาะที่สามารถรับข้อมูลเข้าได้โดยการพิมพ์จากคีย์บอร์ดเท่านั้น ซึ่งการทำงานโดยรวมคือ มีการอ่านไฟล์ตัวอักษรเข้ามายังโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด จากนั้นโปรแกรมแปลงไฟล์ดังกล่าวจะทำการส่งสัญญาณอินพุทไปยังโปรแกรมเฉพาะ เพื่อทำการสั่งการทำงานภายในโปรแกรมเฉพาะ



รูปที่ 3.1 แสดงผังการทำงาน โดยรวม

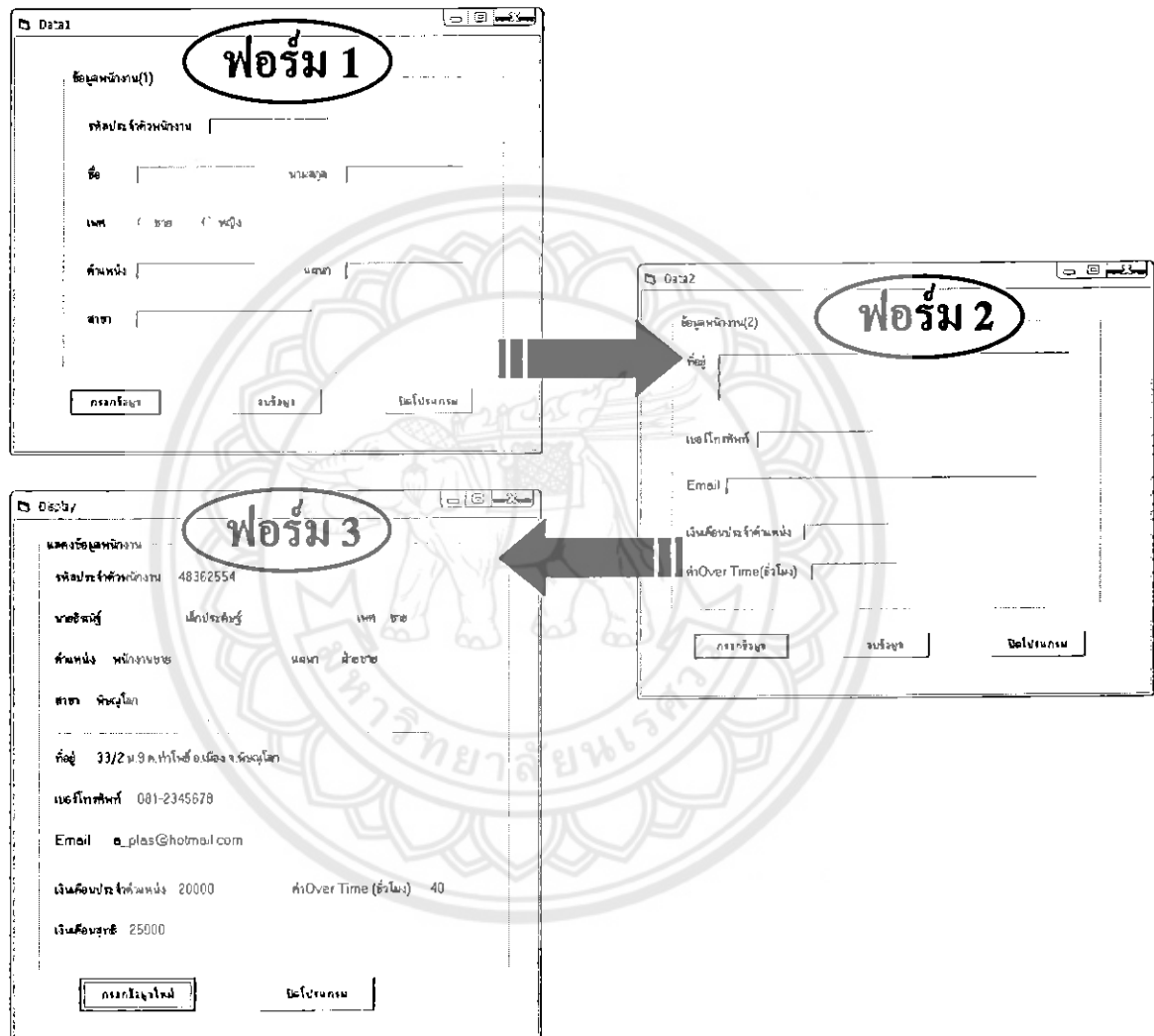
3.2 การทำงานของโปรแกรมเฉพาะและการออกแบบโปรแกรม

ลักษณะการทำงานของโปรแกรมเฉพาะที่นำมาศึกษานั้น ถูกกำหนดว่าจะต้องเป็นโปรแกรมที่สามารถรับค่าอินพุทจากคีย์บอร์ดได้เพียงอย่างเดียวเท่านั้น โดยอาจมีลักษณะเป็นช่องสำหรับกรอกข้อมูล หรือเป็นเช็คบ็อกซ์สำหรับเลือกข้อมูลที่ต้องการ ซึ่งจากลักษณะดังกล่าว จึงได้มีการออกแบบโปรแกรมเฉพาะที่สามารถรับค่าอินพุทจากคีย์บอร์ด และแสดงผลข้อมูลในหลายรูปแบบด้วยกัน ดังนี้

3.2.1 โปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic

แบ่งออกได้ 2 รูปแบบ ตามลักษณะการแสดงผลข้อมูล

1. โปรแกรมที่แสดงผลข้อมูลในฟอร์มสุดท้าย จะประกอบไปด้วยฟอร์มสำหรับกรอกข้อมูล ซึ่งสามารถมีได้หลายฟอร์ม และฟอร์มสำหรับสรุปผลข้อมูลเป็นฟอร์มสุดท้าย โดยมีรูปแบบหน้าต่าง โปรแกรมและ โค้ดคำสั่งของ Form ต่างๆ ดังนี้



รูปที่ 3.2 แสดง โปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ซึ่งแสดงผลข้อมูลในฟอร์มสุดท้าย

จากรูปที่ 3.2 แสดงถึงลักษณะของโปรแกรมที่พัฒนาด้วยภาษา Visual Basic โดยมีฟอร์มทั้งหมด 3 ฟอร์ม ซึ่งฟอร์มที่ 1 และ 2 ใช้สำหรับกรอกข้อมูลที่ต้องการลงในช่องกรอกข้อมูล และสำหรับฟอร์มที่ 3 เป็นฟอร์มที่ใช้แสดงผลข้อมูลที่ทำการกรอกไปแล้ว

2. โปรแกรมที่แสดงผลข้อมูลในตารางฐานข้อมูล ประกอบด้วยฟอร์มสำหรับกรอกข้อมูลซึ่งเมื่อกรอกเรียบร้อยแล้ว จะแสดงผลข้อมูลในตาราง และเก็บไปยังฐานข้อมูลของโปรแกรม โดยมีรูปแบบหน้าต่างโปรแกรมและ ไลค์ค้ำสั่งดังนี้

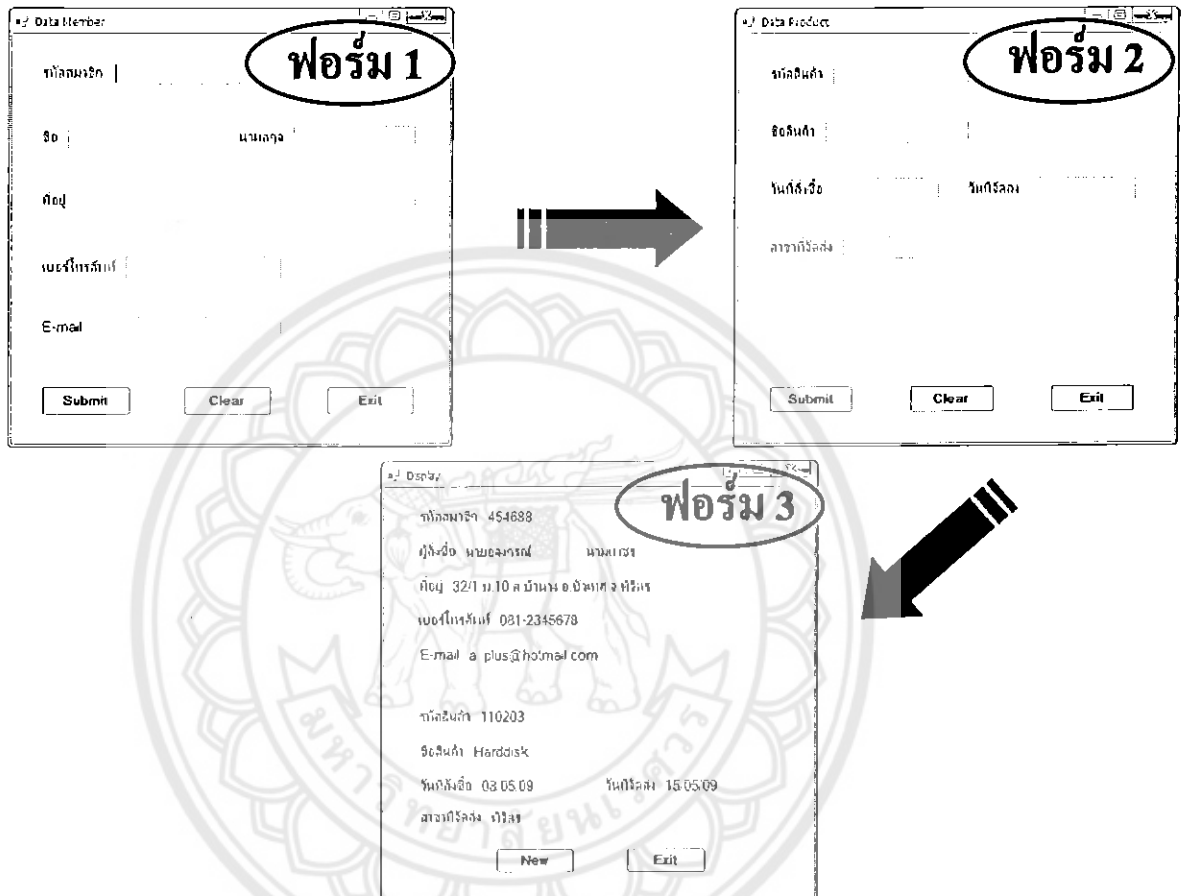
ตารางสอบข้อมูล			
เลขที่	เลขประจำตัว	ค่าน้ำหนัก	ชื่อ
1	0001	นางสาว	สุบใจ
2	0002	นางสาว	พรพทมน

รูปที่ 3.3 แสดงโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ซึ่งแสดงผลข้อมูลในตารางฐานข้อมูล

จากรูปที่ 3.3 จะเห็นได้ว่ารูปแบบของโปรแกรมเฉพาะเป็นแบบที่มีช่องให้กรอกข้อมูล และมี ส่วนของตารางฐานข้อมูลสำหรับแสดงรายละเอียดข้อมูลที่ได้ทำการกรอกไปแล้ว โดยโปรแกรม ส่วนประกอบอื่นๆ เป็นปุ่มสำหรับเพิ่มข้อมูล บันทึกข้อมูลที่กรอก ลบข้อมูล แก้ไขข้อมูล เลื่อนการดูข้อมูล และปุ่มการออกจากโปรแกรม

3.2.2 โปรแกรมเฉพาะที่พัฒนาด้วยภาษา C#

เป็นโปรแกรมที่ถูกพัฒนาขึ้นเพื่อใช้ศึกษาดัชนีโปรแกรมเฉพาะในรูปแบบอินเทอร์เฟซที่ต่างๆ กัน ประกอบด้วยฟอร์มสำหรับการกรอกข้อมูลและฟอร์มสำหรับการแสดงผล โดยมีรูปแบบหน้าต่าง โปรแกรมและ โค้ดคำสั่งดังนี้



รูปที่ 3.4 แสดง โปรแกรมเฉพาะที่พัฒนาด้วยภาษา C#

จากรูปที่ 3.4 จะเห็นได้ว่า โปรแกรมประกอบไปด้วยฟอร์มจำนวน 3 ฟอร์ม โดยฟอร์ม 1 และ 2 เป็นฟอร์มสำหรับการกรอกข้อมูลที่ต้องการลงไป และจะแสดงผลข้อมูลทั้งหมดที่กรอกของสองฟอร์มข้างต้น ในฟอร์ม 3

3.2.3 โปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java

เป็นโปรแกรมที่ถูกพัฒนาขึ้นเพื่อใช้ศึกษาลักษณะโปรแกรมเฉพาะในรูปแบบอินเทอร์เฟซที่ต่างๆ กัน โดยมีรูปแบบของโปรแกรมดังนี้

The image displays two screenshots of a Java-based web form for employee information. The first screenshot, labeled 'ฟอร์ม 1', shows the form with empty input fields. The second screenshot, labeled 'ฟอร์ม 2', shows the form with pre-filled data for an employee named 'นางสาวเน็กส์ อังสน์'.

ฟอร์ม 1 (Empty Form):

- Employee Information:
 - Employee ID:
 - First Name:
 - Sur Name:
 - Careers:
 - Department:
 - Branch:
- Contact Information:
 - Address:
 - Telephone Number:
 - E-mail:
- Month Conditional Information:
 - Month Conditional Rank:
 - Over Time:
- Buttons: Enter new information

ฟอร์ม 2 (Filled Form):

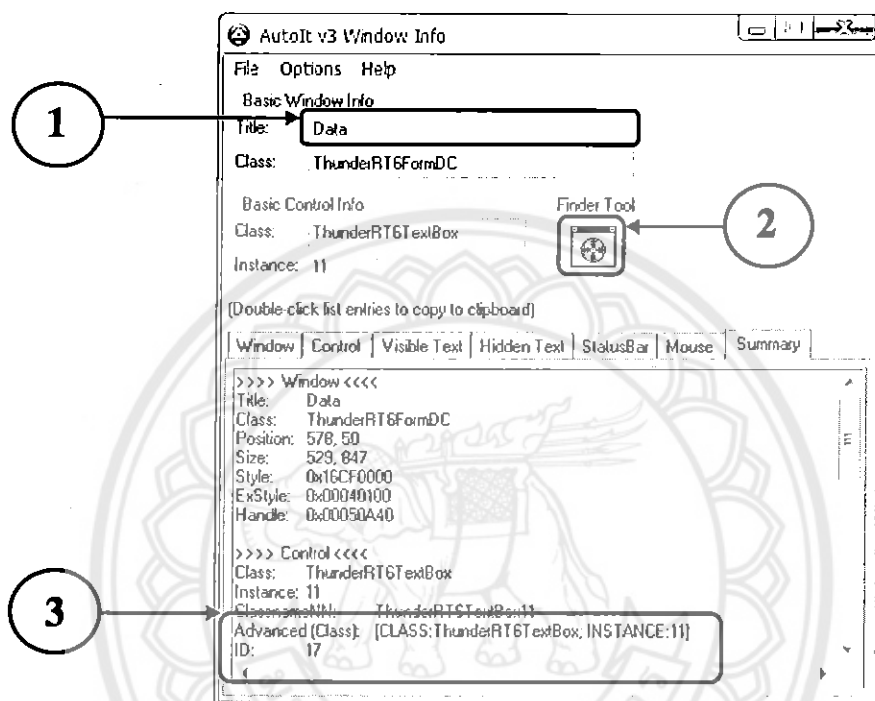
- Employee Information:
 - Employee ID: 4200009
 - First Name: นางสาวเน็กส์
 - Sur Name: อังสน์
 - Careers: พนักงานบัญชี
 - Department: บัญชี
 - Branch: วิทยุโสต
- Contact Information:
 - Address: 123 หมู่ 8 ตำบลใหม่ อำเภอเมืองภูเก็ต
 - Telephone Number: 084-1234567
 - E-mail: garh1rat@hotmail.com
- Month Conditional Information:
 - Month Conditional Rank: 15000
 - Over Time: 40
- Buttons: Enter new Information, EXIT

รูปที่ 3.5 แสดงโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java

จากรูปที่ 3.5 แสดงรูปแบบของโปรแกรมเฉพาะที่มีลักษณะเป็นช่องสำหรับกรอกข้อมูลในฟอร์ม 1 และจะแสดงผลข้อมูลในฟอร์ม 2

3.3 การเข้าถึงส่วนประกอบในโปรแกรมเฉพาะ

การเข้าถึงส่วนประกอบจะมีเครื่องมือชื่อว่า AutoIt Window Info ที่ใช้ในการระบุชื่อของส่วนประกอบภายในโปรแกรมเฉพาะที่ต้องการส่งคำสั่งในส่วนประกอบเหล่านั้นปฏิบัติตามคำสั่งของโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับโปรแกรม



รูปที่ 3.6 แสดงเครื่องมือ AutoIt Window Info

จากรูปที่ 3.6 แสดงส่วนประกอบต่างๆ ของ AutoIt Window Info ดังนี้

ส่วนที่ 1 Title เป็นชื่อของหน้าต่างโปรแกรมเฉพาะ

ส่วนที่ 2 Finder Tool เป็นเครื่องมือที่ใช้ชี้ไปยังส่วนประกอบของโปรแกรมเฉพาะ

ส่วนที่ 3 controlID เป็นชื่อของส่วนประกอบภายในโปรแกรมเฉพาะ ซึ่งอาจนำมาใช้ได้ทั้ง

ส่วนของ Advanced(class) และ ID

จากรูปที่ 3.7 แสดงการทำงานของโค้ดคำสั่งได้ดังต่อไปนี้
บรรทัดที่ 1 เป็นส่วนที่ใช้ในการเปิดไฟล์ที่ต้องการส่งไปยัง โปรแกรมเฉพาะ โดยมีรูปแบบของ
คำสั่งดังนี้

```
$file = OpenFileDialog("Select file to process...",
    "::{450D8FBA-AD25-11D0-98A8-0800361B1103}",
    "Text file (*.txt)|Excel file (*.csv)")
```

\$file คือตัวแปรที่ใช้รับค่าจากไฟล์อักษร

FileOpenDialog คือฟังก์ชันสำหรับเปิดไฟล์ โดยมีรูปแบบฟังก์ชัน เป็น

ตารางที่ 3.1 แสดงรูปแบบการเขียนคำสั่งของฟังก์ชันเปิดไฟล์

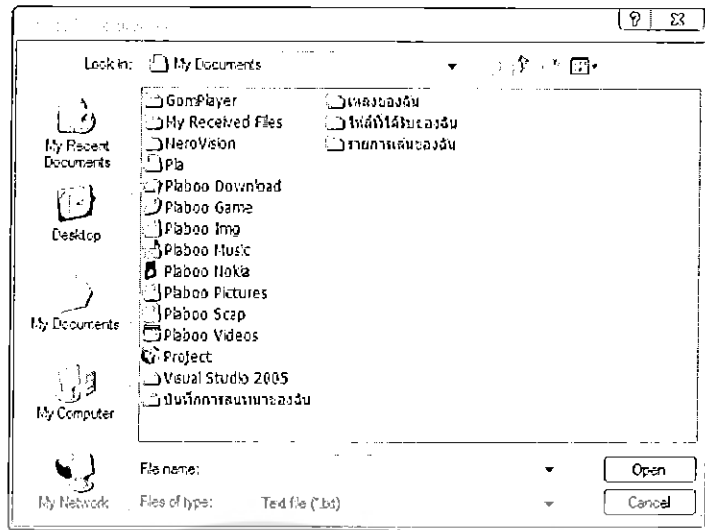
FileOpenDialog ("title", "init dir", "filter" [, options [, "default name"]])	
title	ข้อความที่ใช้เป็นชื่อของกล่องเปิดไฟล์
init dir	ที่อยู่แรกที่แสดงในการเลือกไฟล์
filter	ชนิดไฟล์ เช่น "All (*.*)" หรือ "Text files (*.txt)"
option	ทางเลือกเพิ่มเติม 1 = มีไฟล์แล้ว (ถ้าผู้ใช้พิมพ์ชื่อไฟล์) 2 = มีที่อยู่แล้ว (ถ้าผู้ใช้พิมพ์ที่อยู่ แล้วปิดท้ายด้วยเครื่องหมาย /) 3 = เลือกได้หลายไฟล์ 4 = สร้างไฟล์ใหม่ (ถ้าไม่มีไฟล์อยู่)
default name	ชื่อไฟล์เริ่มต้น ซึ่งปกติมีค่าเป็นช่องว่าง ("")

ชื่อกล่องเปิดไฟล์ (title) คือ "Select file to process..."

ที่อยู่แรกในการเปิดไฟล์ (init dir) คือ "::{450D8FBA-AD25-11D0-98A8-0800361B1103}"

ซึ่งจะแสดงที่อยู่เป็น My document

ชนิดไฟล์ (filter) คือ "Text file (*.txt)|Excel file (*.csv)" ซึ่งจะแสดงไฟล์ชนิดตัวอักษรที่ถูก
บันทึกด้วยนามสกุล txt และ csv โดยจะปรากฏหน้าต่างของกล่องเปิดไฟล์ดังรูป



รูปที่ 3.8 แสดงหน้าต่างกล่องเปิดไฟล์

บรรทัดที่ 3 – 6 เป็นการใช้นิพจน์ If ซึ่งเป็นประโยคเงื่อนไขให้แสดงข้อความเมื่อเกิดข้อผิดพลาดในการเปิดไฟล์ มีรูปแบบคำสั่งดังนี้

```

If @error Then
    MsgBox(4096, "", "No file(s) chosen or operation cancel!!")
Exit
EndIf
    
```

MsgBox คือนิพจน์ในการแสดงกล่องข้อความ โดยมีรูปแบบคือ

ตารางที่ 3.2 แสดงรูปแบบการเขียนคำสั่งของฟังก์ชันกล่องข้อความ

MsgBox (flag, "title", "text" [, timeout])	
flag	แสดงชนิดของกล่องข้อความและปุ่มกด
title	ข้อความที่ใช้เป็นชื่อของกล่องข้อความ
text	ข้อความภายในกล่องข้อความ
timeout	การตั้งเวลาให้กล่องข้อความปิดอัตโนมัติภายในเวลาที่กำหนด(วินาที)

ชนิดของกล่องข้อความและปุ่มกด (flag) คือ 4096 (System modal)

ข้อความภายในกล่องข้อความ (text) คือ “No file(s) chosen or operation cancel!!” ซึ่งไม่มีชื่อของกล่องข้อความ (title) และการตั้งเวลาปิด (timeout)

บรรทัดที่ 8 เป็นการประกาศตัวแปรเก็บค่าหมายเลขบรรทัดที่ทำการอ่านจากไฟล์ โดยกำหนดให้เริ่มที่บรรทัดที่ 1 มีคำสั่งเป็น

```
$line_no = 1
```

บรรทัดที่ 10 – 48 เป็นคำสั่งในการวนลูป While แล้วจบด้วย WEnd ซึ่งกระทำตามคำสั่งภายในจนกว่าเงื่อนไขเป็นเท็จ

บรรทัดที่ 11 เป็นการอ่านไฟล์เข้ามาทีละบรรทัด โดยนำมาเก็บไว้ที่ตัวแปร \$line ซึ่งมีคำสั่งคือ

```
$line = FileReadLine($file,$line_no)
```

FileReadLine คือฟังก์ชันที่ใช้ในการอ่านไฟล์เข้าสู่โปรแกรมทีละบรรทัด มีรูปแบบคือ

ตารางที่ 3.3 แสดงรูปแบบการเขียนคำสั่งของฟังก์ชันการอ่านไฟล์ทีละบรรทัด

FileReadLine (filchandle or "filename" [, line])	
filehandle	การเข้าสู่ไฟล์ ซึ่งเรียกผ่านฟังก์ชัน FileOpen
line	บรรทัดที่ทำการอ่านจากไฟล์ บรรทัดแรกคือ 1 (ไม่ใช่ 0) บรรทัดสุดท้ายคือ - 1

การเข้าสู่ไฟล์ (handle) จะเรียกผ่านตัวแปร \$file ส่วนบรรทัดที่ทำการอ่าน (line) จะทำการอ่านผ่านตัวแปร \$line_no

บรรทัดที่ 12 – 20 เป็นคำสั่งเงื่อนไข If ... Else ... EndIf ใช้ในการทดสอบว่าอ่านไฟล์มีข้อผิดพลาดเกิดขึ้นหรือไม่ โดยถ้าผิดพลาดที่บรรทัดสุดท้ายจะแสดงข้อความ "End of file, operation completed." แล้วออกจากโปรแกรม แต่ถ้าผิดพลาดที่ที่บรรทัดอื่นจะแสดงข้อความ "Error while reading, Skip this line..." แล้วทำตามคำสั่งบรรทัดต่อไป คั่งโค้ดคำสั่งด้านล่าง

```

If @error Then
    If @error = -1 Then
        MsgBox(4096, "", "End of file, operation completed.", 5)
        ExitLoop
    ElseIf @error = 1 Then
        MsgBox(4096, "", "Error while reading, Skip this line...", 5)
        ContinueLoop
    EndIf
EndIf

```

บรรทัดที่ 22 – 23 เป็นการเลื่อนบรรทัดที่นำเข้ามาจากไฟล์ และทำการแตกตัวอักษรออกจากตัวแบ่งซึ่งในโปรแกรมนี้ได้ใช้เครื่องหมายจุลภาค (,) ในการแบ่ง ดังโค้ดคำสั่ง

```

$line_no = $line_no + 1
$col     = StringSplit($line, ",")

```

\$line_no คือตัวแปรที่ใช้เก็บค่าหมายเลขบรรทัด ที่ถูกกำหนดให้เพิ่มขึ้นทีละ 1

\$col คือตัวแปรที่ใช้เก็บค่าคอลัมน์ของข้อมูลแต่ละบรรทัด โดยถูกกำหนดให้รับข้อมูลที่ละตัวที่ถูกคั่นด้วยเครื่องหมายจุลภาค (,) ตามฟังก์ชัน

ตารางที่ 3.4 แสดงรูปแบบการเขียนคำสั่งของฟังก์ชันการแบ่งสตริง

StringSplit ("string", "delimiters" [, flag])	
string	ค่าสตริงที่ใช้ในการประเมินค่า
delimiters	อักขระหนึ่งหรือหลายตัว ที่ใช้จำกัดสตริง
flag	ทางเลือกเพิ่มเติมในการแบ่งสตริง

บรรทัดที่ 25 เป็นการใช้ฟังก์ชันเงื่อนไข (If... Else... EndIf) ในการตรวจสอบว่า รูปแบบของไฟล์ที่รับเข้ามาเป็นไปตามที่ต้องการหรือไม่ นั่นคือ มีจำนวนคอลัมน์ครบตามที่ต้องการหรือไม่

If \$col[0] = 11 Then

จากโค้ดจะเป็นจำนวนคอลัมน์เท่ากับ 11 ซึ่งกำหนดให้เป็นตัวแปรอาร์เรย์เก็บจำนวนคอลัมน์ของข้อมูลเมื่อทดสอบเงื่อนไขแล้วว่าเป็นจริงจึงทำตามคำสั่งในบรรทัดที่ 26 – 44 ซึ่งเป็นการระบุการนำข้อมูลไปใส่ในช่องข้อมูล และการกดปุ่มของโปรแกรมเฉพาะ คั่ง โค้ดด้านล่าง

```
WinWait("Data")
ControlSend("Data", "", 17, $col[1])
ControlSend("Data", "", 16, $col[2])
ControlSend("Data", "", 15, $col[3])
ControlSend("Data", "", 14, $col[4])
ControlSend("Data", "", 13, $col[5])
ControlSend("Data", "", 12, $col[6])
ControlSend("Data", "", 7, $col[7])
ControlSend("Data", "", 8, $col[8])
ControlSend("Data", "", 6, $col[9])
ControlSend("Data", "", 3, $col[10])
ControlSend("Data", "", 4, $col[11])
```

Sleep(1000)

ControlClick("Data", "", 10)

WinWait("Display")

Sleep(3000)

ControlClick("Display", "", 17)

WinWait เป็นฟังก์ชันที่ใช้ในการหยุดการดำเนินการของโปรแกรมก่อนที่จะมีการเปิดโปรแกรมเฉพาะ ซึ่งในโค้ดคำสั่งจะเป็นการรอหน้าต่างโปรแกรมชื่อว่า Data และ Display

ตารางที่ 3.5 แสดงรูปแบบการเขียนคำสั่งของฟังก์ชันการรอโปรแกรมเฉพาะ

WinWait ("title" [, "text" [, timeout]])	
title	ชื่อของหน้าต่างโปรแกรมที่ตรวจสอบ
text	ข้อความของหน้าต่างโปรแกรมที่ตรวจสอบ
timeout	ตั้งเวลารอการเปิดโปรแกรม

ControlSend เป็นฟังก์ชันที่ใช้ในการส่งข้อความอัตโนมัติไปยังโปรแกรมเฉพาะ โดยมีรูปแบบของฟังก์ชันดังนี้

ตารางที่ 3.6 แสดงรูปแบบการเขียนคำสั่งการส่งข้อความไปยังโปรแกรมเฉพาะ

ControlSend ("title", "text", controlID, "string" [, flag])	
title	ชื่อหน้าต่างโปรแกรมที่ต้องการส่งข้อความเข้า
text	ข้อความหน้าต่างโปรแกรมที่ต้องการส่งข้อความเข้า
controlID	ชื่อส่วนประกอบหน้าต่างโปรแกรมที่ต้องการส่งข้อความเข้า
string	ค่าสตริงที่ต้องการส่งไปยังโปรแกรม
flag	ตัวเลือกเพิ่มเติม

ในโค้ดมีชื่อหน้าต่างโปรแกรม (title) เป็น Data แต่ไม่มีข้อความหน้าต่างโปรแกรม (text) มีชื่อส่วนประกอบหน้าต่างโปรแกรม (controlID) เป็นตัวเลข ID ที่สามารถดูได้จากการใช้เครื่องมือ AutoIt Window Info ในการชี้ไปยังโปรแกรมเฉพาะ เพื่อตรวจสอบชื่อหรือตัวเลข ID ของโปรแกรมเฉพาะ และมีค่าสตริงที่ต้องการส่ง (string) เป็นค่าที่รับจากตัวแปรอาร์เรย์ \$col[]

Sleep เป็นฟังก์ชันที่ใช้ในการหยุดการดำเนินโปรแกรมชั่วคราว โดยมีการกำหนดการหน่วงเวลาเป็นมิลลิวินาที ซึ่งในโค้ดจะมีการหน่วงเวลาเป็น 1000 และ 3000 มิลลิวินาที

ControlClick เป็นฟังก์ชันที่ใช้ในการสั่งการให้โปรแกรมกดปุ่มที่ต้องการ โดยมีรูปแบบดังนี้

ตารางที่ 3.7 แสดงรูปแบบการเขียนคำสั่งให้โปรแกรมกดปุ่ม

ControlClick ("title", "text", controlID [, button [, clicks [, x [, y]]]])	
title	ชื่อหน้าต่างโปรแกรมที่ต้องการส่งคำสั่งกดปุ่ม
text	ข้อความหน้าต่างโปรแกรมที่ต้องการส่งคำสั่งกดปุ่ม
controlID	ชื่อส่วนประกอบหน้าต่างโปรแกรมที่ต้องการส่งคำสั่งกดปุ่ม
button	ปุ่มที่ต้องการกดมี "left", "right", "middle", "main", "menu", "primary", "secondary" ค่าเริ่มต้นคือ ปุ่มซ้าย
clicks	จำนวนการกดปุ่มของเมาส์ ค่าเริ่มต้นคือ 1
x	ตำแหน่ง x ที่กดบนปุ่ม ค่าเริ่มต้นคือ ตรงกลาง
y	ตำแหน่ง y ที่กดบนปุ่ม ค่าเริ่มต้นคือ ตรงกลาง

ในโค้ดคำสั่งมีการสั่งให้กดปุ่มในหน้าต่างโปรแกรม (title) ชื่อว่า Data แต่ไม่มีข้อความหน้าต่างโปรแกรม (text) มีชื่อส่วนประกอบหน้าต่างโปรแกรม (controlID) เป็นตัวเลข ID ที่สามารถดูได้จากการใช้เครื่องมือ AutoIt Window Info ในการชี้ไปยังโปรแกรมเฉพาะ เพื่อตรวจสอบชื่อหรือตัวเลข ID ของโปรแกรมเฉพาะ

หากเงื่อนไขไม่เป็นจริง ก็มีจำนวนลอคัลนี้ไม่ครบตามที่ต้องการ จะมีการแจ้งเตือนด้วยกล่องข้อความว่าข้อความไม่เป็นไปตามรูปแบบ ให้ข้ามบรรทัดข้อมูลนี้ไป ("Data not match, this line skipped...")

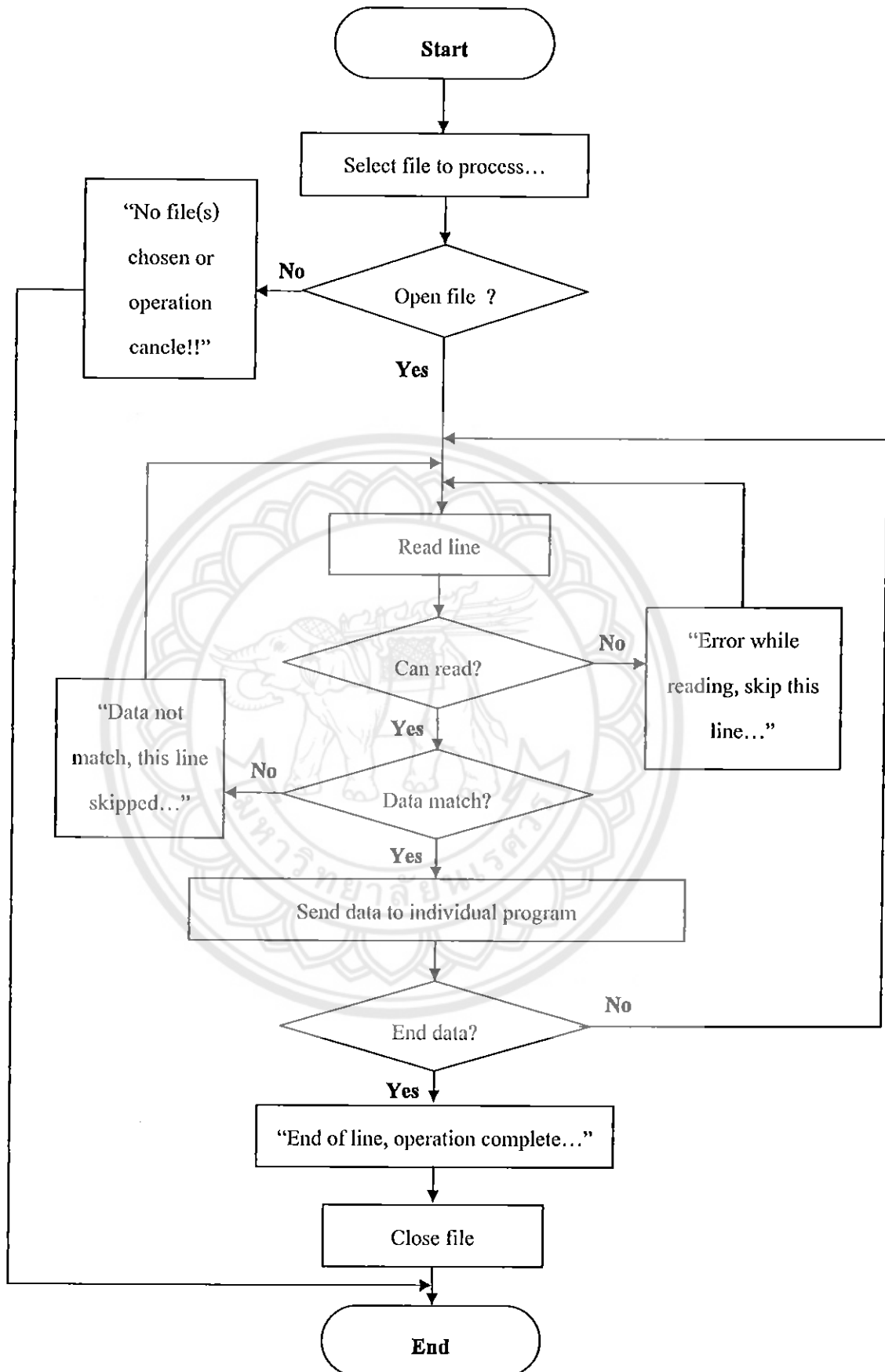
FileClose(\$file) เป็นคำสั่งในการปิดไฟล์ที่เราทำการอ่านเรียบร้อยแล้ว

Exit เป็นการปิดคำสั่งโปรแกรม

ซึ่งคำสั่งส่วนที่เป็นการระบุการนำข้อมูลไปใส่และการกดปุ่มในโปรแกรมเฉพาะสามารถเปลี่ยนไปตามชื่อและรูปแบบของ โปรแกรมเฉพาะ ซึ่งแต่ละ โปรแกรมจะมีรูปแบบการนำข้อมูลไปใส่ และการกดปุ่มที่แตกต่างกัน

3.5 การทำงานของโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด

โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทจากคีย์บอร์ด มีขั้นตอนการทำงานของโปรแกรมดังนี้



รูปที่ 3.9 แสดงขั้นตอนการทำงานของโปรแกรมแปลงข้อมูล
ไฟล์ตัวอักษรเป็นสัญญาณอินพุตจากคีย์บอร์ด

จากรูปที่ 3.9 แสดงขั้นตอนการทำงานของโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับโปรแกรมเฉพาะ ซึ่งมีการทำงานเริ่มจากโปรแกรมเมื่อทำการเปิดโปรแกรมขึ้นมาจะปรากฏกล่องเปิดไฟล์เพื่อทำการเลือกไฟล์ที่จะใช้เป็นอินพุท หากไม่มีการเลือกไฟล์แล้วทำการปิดกล่องเปิดไฟล์ โปรแกรมจะแสดงกล่องข้อความว่า “No file(s) chosen or operation cancel!!” แล้วจบโปรแกรม

ถ้ามีการเปิดไฟล์ โปรแกรมจะทำการตรวจสอบว่าทำการอ่านข้อมูลเข้ามาได้หรือไม่ หากไม่สามารถอ่านบรรทัดข้อมูลใดได้ โปรแกรมจะแสดงกล่องข้อความ “Error while reading, skip this line...” แล้วทำการข้ามบรรทัดนั้นไป เริ่มอ่านข้อมูลในบรรทัดต่อไปเข้ามาในโปรแกรม

ถ้ามีการอ่านบรรทัดข้อมูลเข้ามาได้ โปรแกรมจะทำการตรวจสอบว่าข้อมูลมีจำนวนและรูปแบบเป็นไปตามที่กำหนดหรือไม่ หากมีรูปแบบไม่ตรงตามที่กำหนด โปรแกรมจะแสดงกล่องข้อความ “Data not match, this line skipped...” แล้วทำการข้ามบรรทัดข้อมูลนั้นไป กลับไปอ่านข้อมูลบรรทัดต่อไปเข้ามาในโปรแกรม

ถ้าข้อมูลเป็นไปตามรูปแบบที่ต้องการ โปรแกรมจะส่งข้อมูลบรรทัดนั้นไปยังโปรแกรมเฉพาะเพื่อทำการใส่ข้อมูลลงในโปรแกรมเฉพาะ จากนั้นโปรแกรมจะทำการตรวจสอบว่าหมดบรรทัดข้อมูลแล้วหรือไม่ ถ้ายังไม่หมดข้อมูลก็จะทำการอ่านข้อมูลบรรทัดต่อไปเข้ามา

ถ้าข้อมูลในไฟล์หมดแล้ว โปรแกรมจะแสดงกล่องข้อความ “End of line, operation complete...” แล้วทำการปิดไฟล์ข้อมูลที่ใช้ จากนั้นจึงปิดตัวเองลง

บทที่ 4

ผลการทดลอง

4.1 ผลการทดลองการใช้โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดกับโปรแกรมเฉพาะและไฟล์อินพุทรูปแบบต่างๆ

ในการทดลองนั้น เราต้องให้อินพุทเป็นไฟล์อักษรที่มีจำนวนข้อมูลเท่ากับจำนวนช่องของข้อมูลที่จะกรอกใน โปรแกรมเฉพาะ โดยคั่นข้อมูลแต่ละช่องด้วยเครื่องหมายจุดภาค (.) และแบ่งข้อมูลของแต่ละชุดด้วยการเว้นบรรทัด ซึ่งไฟล์อินพุทที่นำมาทดลองจะเป็นไฟล์อักษรสองชนิดด้วยกันคือ Text file (.txt) และ Excel file (.csv)

นอกจากนี้เรายังให้อินพุทเป็น โปรแกรมเฉพาะที่มีรูปแบบต่างๆ กันในการทดสอบ โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด เพื่อให้ทราบว่าโปรแกรมนี้สามารถใช้กับโปรแกรมเฉพาะที่มีรูปแบบต่างๆ กันได้หรือไม่



4.1.1 การทดสอบโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในฟอร์มสุดท้าย

1. หน้าต่าง โปรแกรมที่ยังไม่ได้มีการกรอกข้อมูล

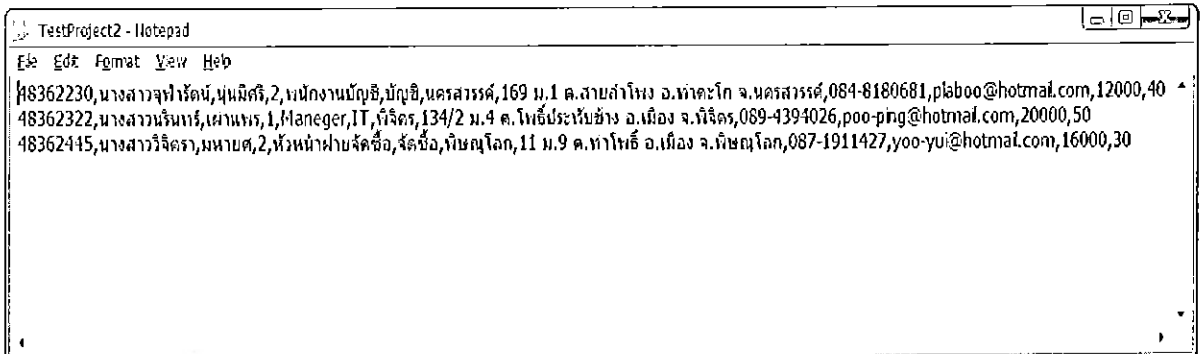
The image shows three overlapping windows from a Visual Basic application:

- Data1 (Employee Information Form 1):** Contains fields for 'รหัสประจำตัวพนักงาน' (Employee ID), 'ชื่อ' (Name), 'นามสกุล' (Surname), 'เพศ' (Gender) with radio buttons for 'ชาย' (Male) and 'หญิง' (Female), 'ตำแหน่ง' (Position), 'แผนก' (Department), and 'สาขา' (Branch). It has buttons for 'กรอกข้อมูล' (Enter Data) and 'ลบข้อมูล' (Delete Data).
- Data2 (Employee Information Form 2):** Contains fields for 'ที่อยู่' (Address), 'เบอร์โทรศัพท์' (Phone Number), 'Email', 'เงินเดือนประจำตำแหน่ง' (Monthly Salary), and 'ค่าOver Time(ชั่วโมง)' (Overtime Hours). It has buttons for 'ลบข้อมูล' (Delete Data) and 'ปิดโปรแกรม' (Close Program).
- Display (Output Form):** Shows the data entered in the previous forms. Fields include 'แสดงข้อมูลพนักงาน' (Display Employee Data), 'รหัสประจำตัวพนักงาน', 'ชื่อ', 'นามสกุล', 'ตำแหน่ง', 'สาขา', 'ที่อยู่', 'เบอร์โทรศัพท์', 'Email', 'เงินเดือนประจำตำแหน่ง' (displayed as 0), 'ค่าOver Time (ชั่วโมง)' (displayed as 0), and 'เงินเดือนสุทธิ' (Net Salary, displayed as 0). It has buttons for 'กรอกข้อมูลใหม่' (Enter New Data) and 'ปิดโปรแกรม' (Close Program).

รูปที่ 4.1 โปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในฟอร์มสุดท้าย ขณะที่ยังไม่ได้มีการกรอกข้อมูล

จากรูปที่ 4.1 จะเห็นได้ว่ามีหน้าต่างสำหรับกรอกข้อมูลจำนวนสองหน้า เมื่อทำการกรอกข้อมูลเรียบร้อยแล้ว โปรแกรมจะทำการแสดงผลการกรอกข้อมูลในหน้าต่างสุดท้าย โดยข้อมูลที่จะต้องกรอกมีทั้งหมด 12 ช่อง

2. ไฟล์ข้อมูลที่นำมาทดสอบมีสองชนิดคือ Text file (.txt) และ Excel file (.csv) โดยมีรายละเอียดของข้อมูลเหมือนกัน



```

TestProject2 - Notepad
File Edit Format View Help
48362230,นางสาวจุฬารัตน์,บุณยศิริ,2,พนักงานบัญชี,บัญชี,นครสวรรค์,169 ม.1 ต.สายลำโพง อ.ท่าตะโก จ.นครสวรรค์,084-8180681,plaboo@hotmail.com,12000,40
48362322,นางสาวนรินทร์,แผนพร,1,Maneger,IT,พิจิตร,134/2 ม.4 ต.โพธิ์ประทับช้าง อ.เมือง จ.พิจิตร,089-4394026,poo-ping@hotmail.com,20000,50
48362445,นางสาววิจิตรา,นนายศ,2,หัวหน้าฝ่ายบัญชี,จัดซื้อ,พิษณุโลก,11 ม.9 ต.ท่าโพธิ์ อ.เมือง จ.พิษณุโลก,087-1911427,yoo-yui@hotmail.com,16000,30
  
```

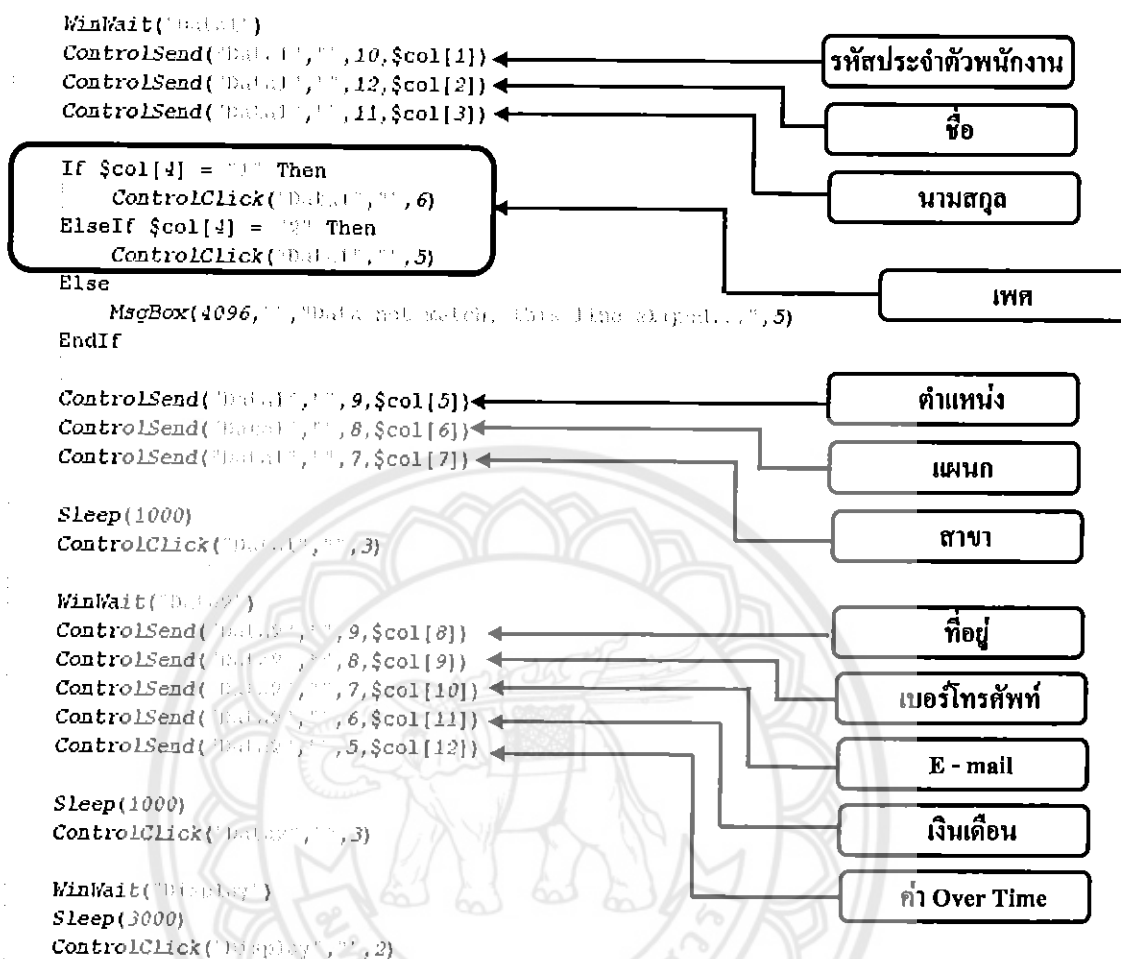
รูปที่ 4.2 แสดงไฟล์ข้อมูลชนิด Text file ซึ่งนำมาทดสอบกับ โปรแกรมเฉพาะ
ที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในฟอร์มสุดท้าย

	A	B	C	D	E	F	G	H	I	J	K	L
1	48362230	นางสาวจุฬารัตน์	บุณยศิริ	2	พนักงานบัญชี	บัญชี	นครสวรรค์	169 ม.1 ต.สายลำโพง อ.ท่าตะโก จ.นครสวรรค์	084-8180681	plaboo@hotmail.com	12000	40
2	48362322	นางสาวนรินทร์	แผนพร	1	Maneger	IT	พิจิตร	134/2 ม.4 ต.โพธิ์ประทับช้าง อ.เมือง จ.พิจิตร	089-4394026	poo-ping@hotmail.com	20000	50
3	48362445	นางสาววิจิตรา	นนายศ	2	หัวหน้าฝ่ายบัญชี	จัดซื้อ	พิษณุโลก	11 ม.9 ต.ท่าโพธิ์ อ.เมือง จ.พิษณุโลก	087-1911427	yoo-yui@hotmail.com	16000	30
4												
5												

รูปที่ 4.3 แสดงไฟล์ข้อมูลชนิด Excel file ซึ่งนำมาทดสอบกับ โปรแกรมเฉพาะ
ที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในฟอร์มสุดท้าย

จากรูปที่ 4.2 และ 4.3 แสดงไฟล์ข้อมูลที่ต้องการนำไปกรอกลงใน โปรแกรมเฉพาะ จะเห็นได้ว่าในไฟล์ข้อมูลจะมีจำนวนข้อมูลตามจำนวนของช่องข้อมูลในโปรแกรมเฉพาะ โดยค้นข้อมูลที่จะกรอกแต่ละช่องด้วยเครื่องหมายจุดภาค ในส่วนของรูปที่ 4.3 จะมีการค้นข้อมูลโดยการเก็บเป็นเซลล์ซึ่งข้อมูลที่มีการค้นประกอบไปด้วย ข้อมูลรหัสประจำตัวพนักงาน, ชื่อ, นามสกุล, เพศ, ตำแหน่ง, แผนก, สาขา, ที่อยู่, เบอร์โทรศัพท์, E-mail, เงินเดือนประจำตำแหน่ง, ค่า Over Time

3. โค้ด โปรแกรมแปลงไฟล์อักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับ โปรแกรมเฉพาะ



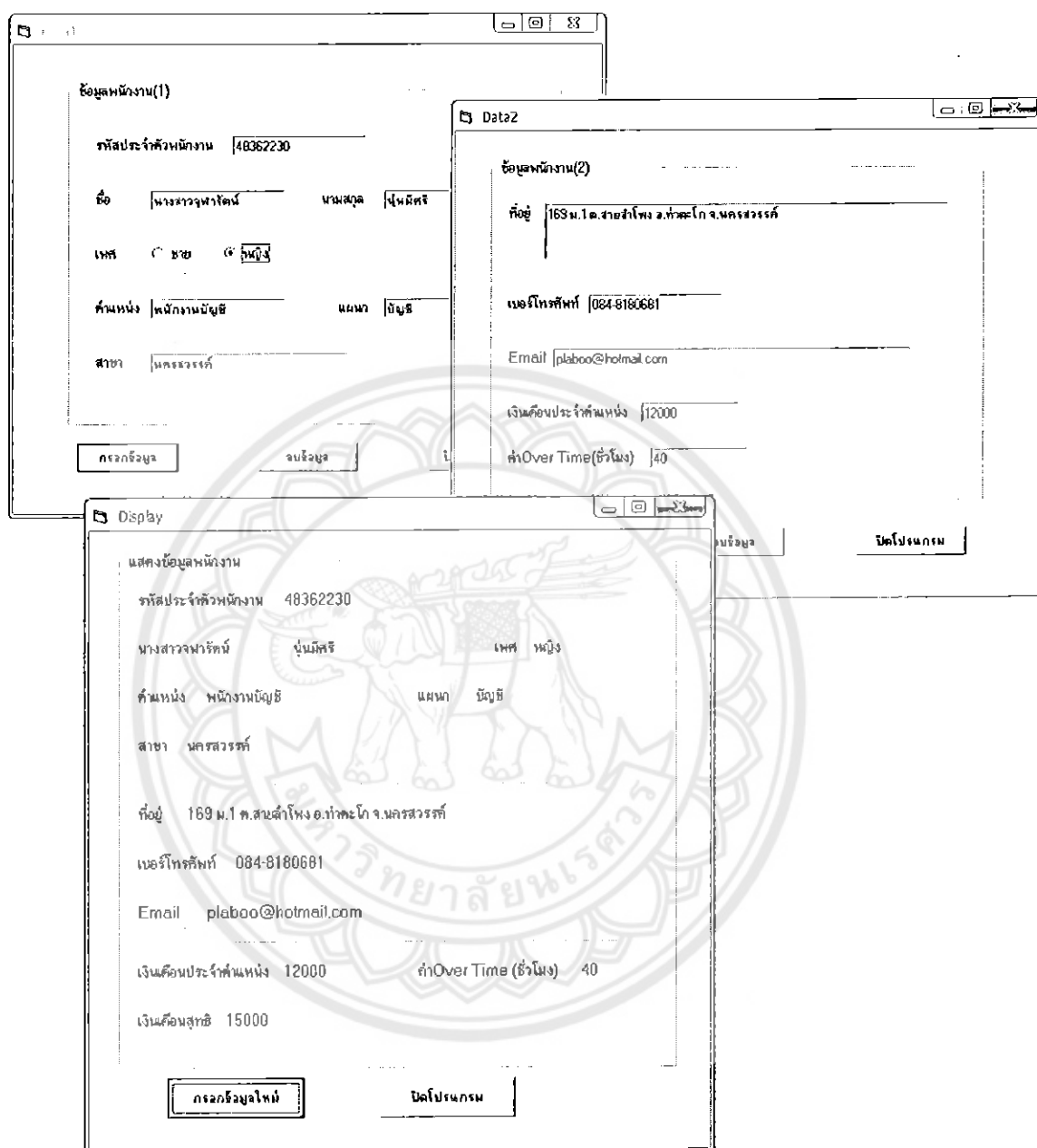
รูปที่ 4.4 แสดง โค้ด โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับ โปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในฟอร์มสุดท้าย

จากรูปที่ 4.4 อธิบายความสัมพันธ์ของโค้ดคำสั่งและโปรแกรมเฉพาะ โดยจะเห็นมีคำสั่งให้นำข้อมูลในแต่ละคอลัมน์ (\$col[]) มาใส่ในแต่ละช่องตามที่ระบุในรูปแบบ ส่วนในการเลือกเพศนั้นจะใช้การตรวจสอบเงื่อนไข ว่ามีข้อมูลเป็นเลขใดระหว่าง 1 (ชาย) และ 2 (หญิง) แล้วจะทำการเลือกในกลุ่มเลือกข้อมูล โดยข้อมูลที่กรอกมีรายละเอียดดังนี้

- \$col[1] จะเป็นการกรอกข้อมูลในช่องรหัสประจำตัวพนักงาน
- \$col[2] จะเป็นการกรอกข้อมูลในช่องชื่อ
- \$col[3] จะเป็นการกรอกข้อมูลในช่องนามสกุล
- \$col[4] จะเป็นการกรอกข้อมูลในช่องเพศ
- \$col[5] จะเป็นการกรอกข้อมูลในช่องตำแหน่ง
- \$col[6] จะเป็นการกรอกข้อมูลในช่องแผนก
- \$col[7] จะเป็นการกรอกข้อมูลในช่องสาขา
- \$col[8] จะเป็นการกรอกข้อมูลในช่องที่อยู่
- \$col[9] จะเป็นการกรอกข้อมูลในช่องเบอร์โทรศัพท์
- \$col[10] จะเป็นการกรอกข้อมูลในช่องE-mail
- \$col[11] จะเป็นการกรอกข้อมูลในช่องเงินเดือน



4. ผลการทดลองกับโปรแกรมเฉพาะจะพบว่าสามารถนำเข้าข้อมูลจากไฟล์ข้อมูลมาใส่ในโปรแกรมเฉพาะได้อย่างถูกต้อง โดยได้ผลที่เหมือนกันทั้ง Text file และ Excel file ดังรูป



รูปที่ 4.5 แสดงผลการทดสอบกับโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในฟอร์มสุดท้าย

จากรูปที่ 4.5 จะแสดงให้เห็นว่า ข้อมูลที่นำเข้ามาจากไฟล์อักษรทั้งสองชนิดคือ Text file และ Excel file โดยผ่านโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของกีบอร์ด สามารถทำการกรอกลงในโปรแกรมเฉพาะได้อย่างถูกต้องครบถ้วน

4.1.2 การทดสอบโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลในตาราง ฐานข้อมูล

1. หน้าต่าง โปรแกรมที่ยังไม่ได้มีการกรอกข้อมูล

Student File

เลขที่

เลขประจำตัว ค้นหา

ค่าน้ำหนัก

ชื่อ

สกุล

ชั้นเรียน

วันเดือนปีเกิด

ที่อยู่

ตรวจสอบข้อมูล			
เลขที่	เลขประจำตัว	ค่าน้ำหนัก	ชื่อ

< >

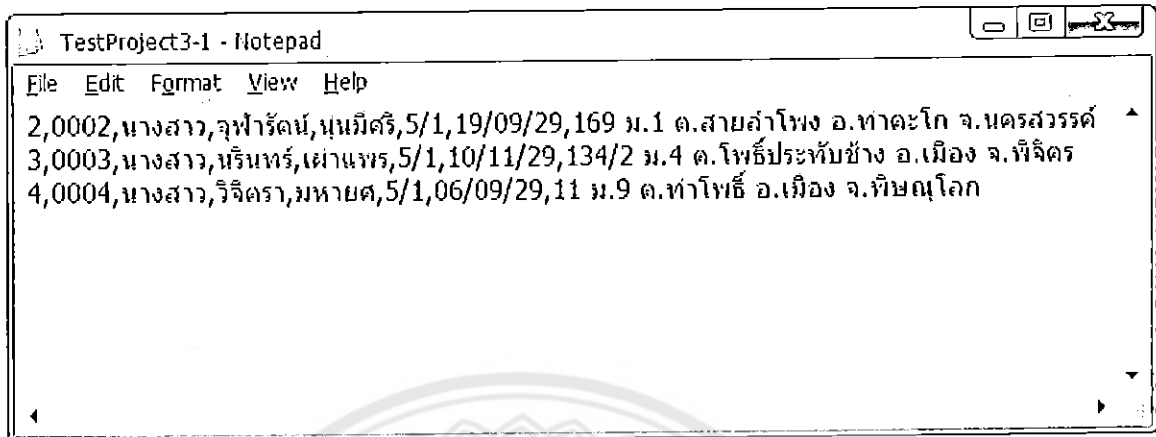
เพิ่ม บันทึก ลบ แก้ไข ออกจากโปรแกรม

Adodc1

รูปที่ 4.6 โปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในตารางฐานข้อมูล
ขณะที่ยังไม่ได้มีการกรอกข้อมูล

จากรูปที่ 4.6 แสดงหน้าต่างของโปรแกรมเฉพาะที่ยังไม่ได้มีการกรอกข้อมูล จะเห็นได้ว่ามีจำนวนช่องสำหรับกรอกข้อมูลจำนวน 8 ช่อง และมีตารางฐานข้อมูลสำหรับแสดงผลข้อมูลอีกหนึ่งตาราง ซึ่งยังไม่มีข้อมูลใดในฐานข้อมูล

2. ไฟล์ข้อมูลที่นำมาทดสอบมีสองชนิดคือ Text file (.txt) และ Excel file (.csv) โดยมีรายละเอียดของข้อมูลเหมือนกัน



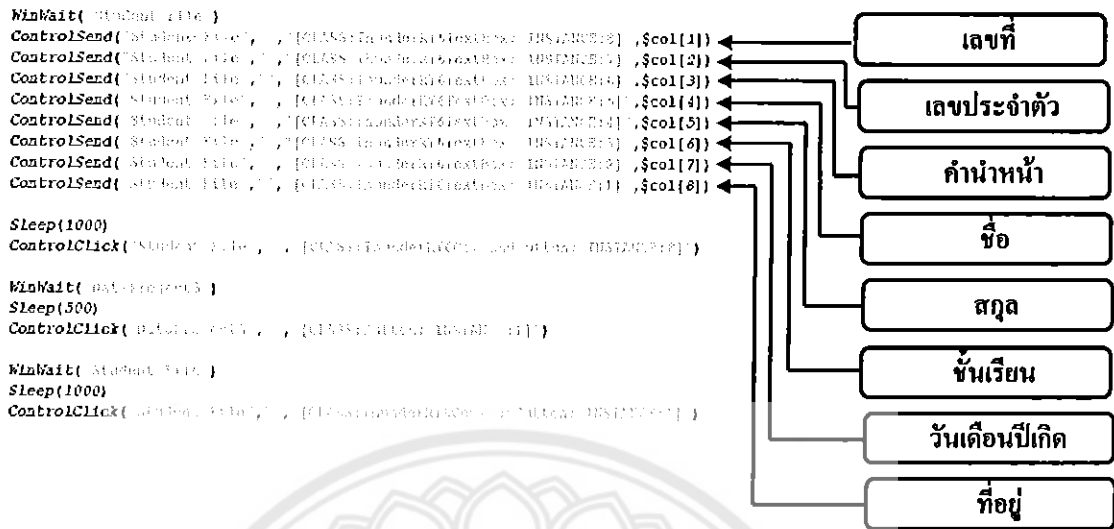
รูปที่ 4.7 แสดงไฟล์ข้อมูลชนิด Text file ซึ่งนำมาทดสอบกับ โปรแกรมเฉพาะที่พัฒนาด้วย
ภาษา Visual Basic ที่แสดงผลข้อมูลในตารางฐานข้อมูล

	A	B	C	D	E	F	G	H
1	2	0002	นางสาว	จุฬารัตน์	นุณมีศรี	5/1	19/09/29	169 ม.1 ต.สายลำโพง อ.ท่าตะโก จ.นครสวรรค์
2	3	0003	นางสาว	นรินทร์	เฝ้าเพชร	5/1	10/11/29	134/2 ม.4 ต.โพธิ์ประทับช้าง อ.เมือง จ.พิจิตร
3	4	0004	นางสาว	วิจิตรา	มหายศ	5/1	06/09/29	11 ม.9 ต.ท่าโพธิ์ อ.เมือง จ.พิษณุโลก
4								
5								

รูปที่ 4.8 แสดงไฟล์ข้อมูลชนิด Excel file ซึ่งนำมาทดสอบกับ โปรแกรมเฉพาะที่พัฒนาด้วย
ภาษา Visual Basic ที่แสดงผลข้อมูลในตารางฐานข้อมูล

จากรูปที่ 4.7 และ 4.8 แสดงไฟล์ข้อมูลที่ต้องการนำไปกรอกลงใน โปรแกรมเฉพาะ จะเห็นได้ว่าในไฟล์ข้อมูลจะมีจำนวนข้อมูลตามจำนวนของช่องข้อมูลใน โปรแกรมเฉพาะ โดยคั่นข้อมูลที่จะกรอกแต่ละช่องด้วยเครื่องหมายจุดภาค ในส่วนของรูปที่ 4.3 จะมีการคั่นข้อมูลโดยการเก็บเป็นเซลล์ ซึ่งข้อมูลที่มีการคั่นประกอบไปด้วย ข้อมูลเลขที่, เลขประจำตัว, คำนำหน้า, ชื่อ, สกุล, ชั้นเรียน, วันเดือนปีเกิด, ที่อยู่

3. โค้ด โปรแกรมแปลงไฟล์อักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับ โปรแกรมเฉพาะ



รูปที่ 4.9 แสดง โค้ด โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับ โปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในตารางฐานข้อมูล

จากรูปที่ 4.9 แสดง โค้ดคำสั่งของ โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับ โปรแกรมเฉพาะ ที่มีรูปแบบการส่งข้อมูลไปยังโปรแกรมเฉพาะ โดยการส่งข้อมูลแต่ละคอลัมน์ (\$col[]) ในไฟล์ข้อมูล ไปยังช่องข้อมูลของ โปรแกรมเฉพาะ

\$col[1] จะเป็นการกรอกข้อมูลในช่องเลขที่

\$col[2] จะเป็นการกรอกข้อมูลในช่องเลขประจำตัว

\$col[3] จะเป็นการกรอกข้อมูลในช่องคำนำหน้า

\$col[4] จะเป็นการกรอกข้อมูลในช่องชื่อ

\$col[5] จะเป็นการกรอกข้อมูลในช่องสกุล

\$col[6] จะเป็นการกรอกข้อมูลในช่องชั้นเรียน

\$col[7] จะเป็นการกรอกข้อมูลในช่องวันเดือนปีเกิด

\$col[8] จะเป็นการกรอกข้อมูลในช่องที่อยู่

3. ผลการทดลองกับโปรแกรมเฉพาะจะพบว่าสามารถนำเข้าข้อมูลจากไฟล์ข้อมูลมาใส่ในโปรแกรมเฉพาะได้อย่างถูกต้อง โดยได้ผลที่เหมือนกันทั้ง Text file และ Excel file ดังรูป

Student File

เลขที่ 4

เลขประจำตัว 0004

คำน้าหน้า นางสาว

ชื่อ วิจิตร

สกุล มหาศ

ชั้นเรียน 5/1

วันเดือนปีเกิด 06/09/29

ที่อยู่ 11 ม.9 ต.ท่าโพธิ์ อ.เมือง จ.พิษณุโลก

ตรวจสอบข้อมูล			
เลขที่	เลขประจำตัว	คำน้าหน้า	ชื่อ
1	0001	นางสาว	สุขใจ
2	0002	นางสาว	งามวิมล
3	0003	นางสาว	นรินทร์

Adodc1

เพิ่ม

บันทึก

ลบ

แก้ไข

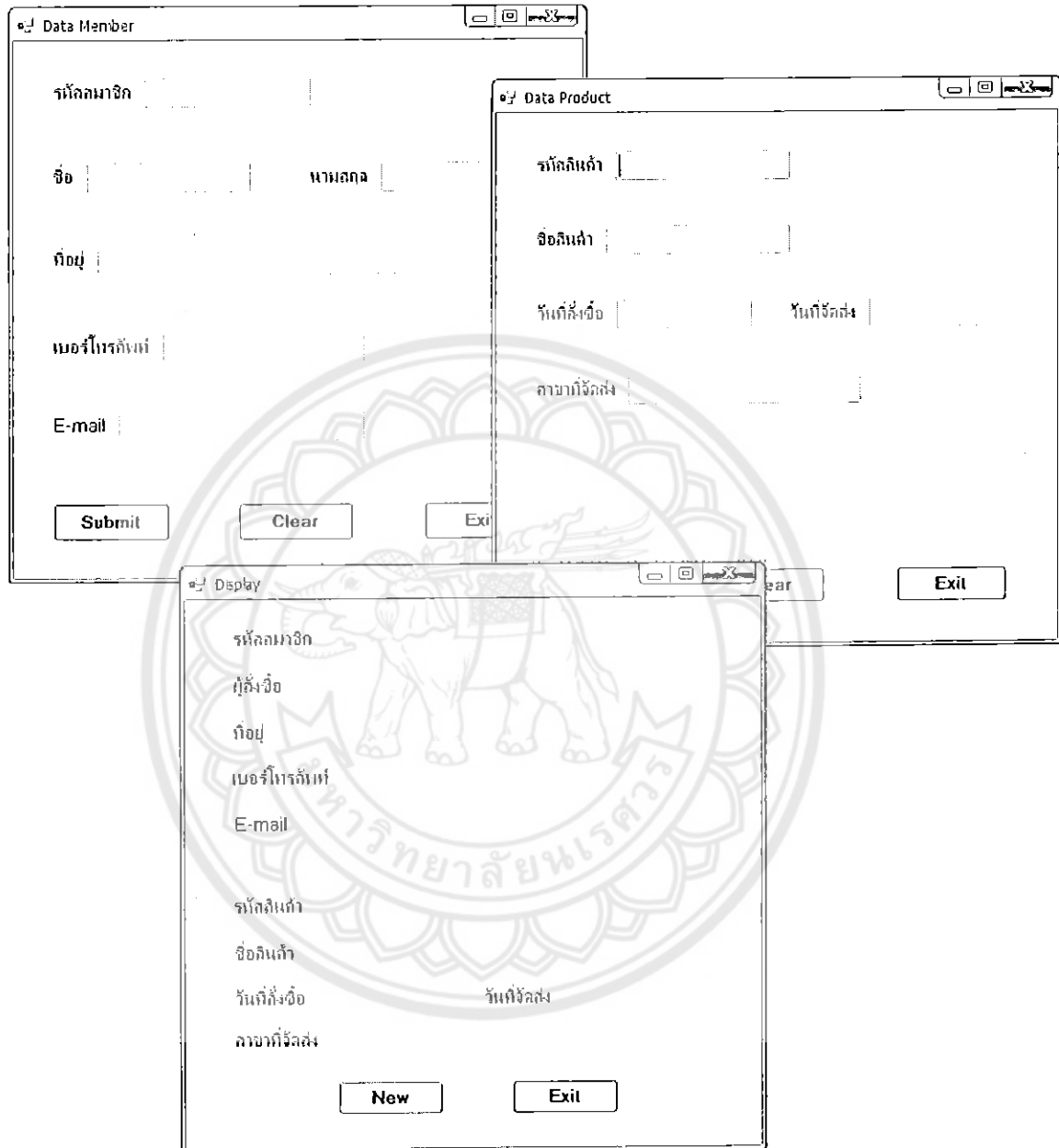
ออกจากระบบ

รูปที่ 4.10 แสดงผลการทดสอบกับโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลในตารางฐานข้อมูล

จากรูปที่ 4.10 เมื่อทดลองรับค่าไฟล์โดยผ่านโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับโปรแกรมเฉพาะแล้ว พบว่าสามารถรอกข้อมูลจากไฟล์ตัวอักษรทั้งชนิด Text file และ Excel file ได้อย่างถูกต้องครบถ้วน และแสดงผลข้อมูลในตารางฐานข้อมูลได้อย่างถูกต้องเช่นกัน

4.1.3 การทดสอบกับโปรแกรมเฉพาะที่พัฒนาด้วยภาษา C#

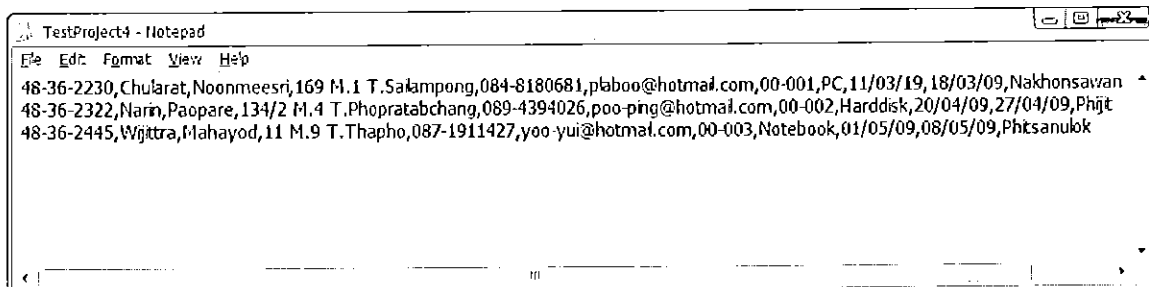
1. หน้าต่างโปรแกรมที่ยังไม่ได้มีการกรอกข้อมูล



รูปที่ 4.11 โปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในตารางฐานข้อมูล ขณะที่ยังไม่ได้มีการกรอกข้อมูล

จากรูปที่ 4.11 จะเห็นได้ว่ามีหน้าต่างสำหรับกรอกข้อมูลจำนวนสองหน้า เมื่อทำการกรอกข้อมูลเรียบร้อยแล้ว โปรแกรมจะทำการแสดงผลการกรอกข้อมูลในหน้าต่างสุดท้าย โดยข้อมูลที่จะต้องกรอกมีทั้งหมด 11 ช่อง

2. ไฟล์ข้อมูลที่นำมาทดสอบมีสองชนิดคือ Text file (.txt) และ Excel file (.csv) โดยมีรายละเอียดของข้อมูลที่เหมือนกัน



```

TestProject4 - Notepad
File Edit Format View Help
48-36-2230,Chularat, Noonmeesri, 169 M.1 T.Salampong, 084-8180681, plaboo@hotmail.com, 00-001, PC, 11/03/19, 18/03/09, Nakhonsawan
48-36-2322, Narin, Paopare, 134/2 M.4 T. Phopratabchang, 089-4394026, poo-ping@hotmail.com, 00-002, Harddisk, 20/04/09, 27/04/09, Phit
48-36-2445, Wjitra, Mahayod, 11 M.9 T. Thapho, 087-1911427, yoo-yui@hotmail.com, 00-003, Notebook, 01/05/09, 08/05/09, Phitsanulok
  
```

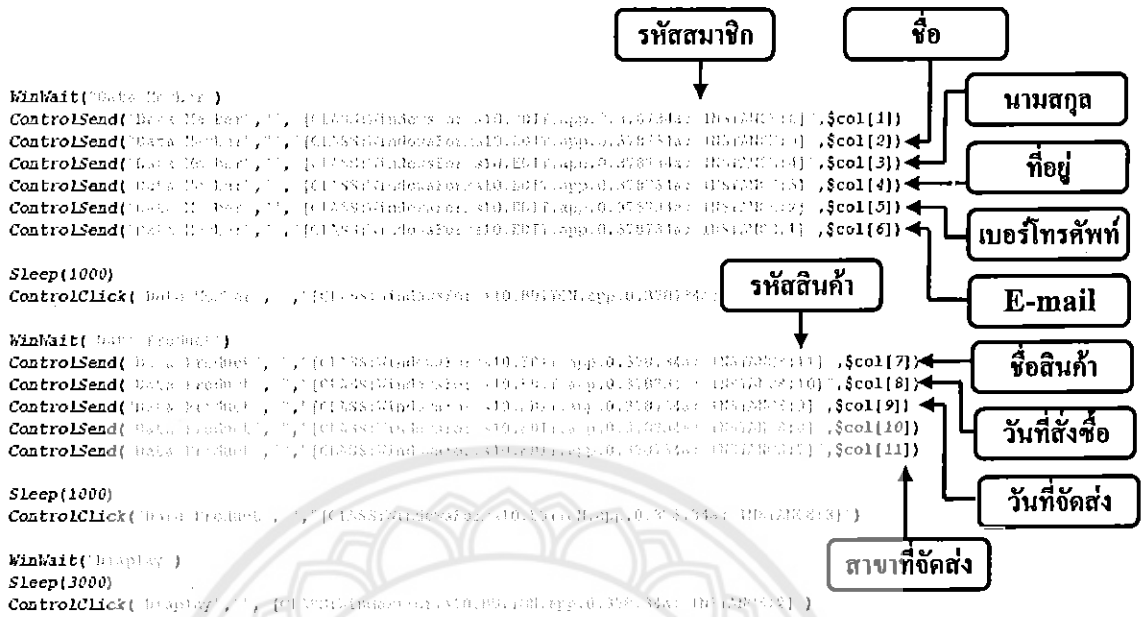
รูปที่ 4.12 แสดงไฟล์ข้อมูลชนิด Text file ซึ่งนำมาทดสอบกับ โปรแกรมเฉพาะ
ที่พัฒนาด้วยด้วยภาษา C#

	A	B	C	D	E	F	G	H	I	J	K
1	48-36-2230	Chularat	Noonmeesri	169 M.1 T.Salampong	084-8180681	plaboo@hotmail.com	0-0001	PC	11/3/2009	18/3/2009	Nakhonsawan
2	48-36-2322	Narin	Paopare	134/2 M.4 T. Phopratabchang	089-4394026	poo-ping@hotmail.com	0-0002	Harddisk	11/3/2009	18/3/2009	Phit
3	48-36-2445	Wjitra	Mahayod	11 M.9 T. Thapho	087-1911427	yoo-yui@hotmail.com	0-0003	Notebook	11/3/2009	18/3/2009	Phitsanulok
4											
5											

รูปที่ 4.13 แสดงไฟล์ข้อมูลชนิด Excel file ซึ่งนำมาทดสอบกับ โปรแกรมเฉพาะ
ที่พัฒนาด้วยด้วยภาษา C#

จากรูปที่ 4.12 และ 4.13 แสดงไฟล์ข้อมูลที่ต้องการนำไปกรอกลงใน โปรแกรมเฉพาะ จะเห็นได้ว่าในไฟล์ข้อมูลจะมีจำนวนข้อมูลตามจำนวนของช่องข้อมูลในโปรแกรมเฉพาะ โดยคั่นข้อมูลที่จะกรอกแต่ละช่องด้วยเครื่องหมายจุดภาค ในส่วนของรูปที่ 4.3 จะมีการคั่นข้อมูลโดยการเก็บเป็นเซลล์ ซึ่งข้อมูลที่มีการคั่นประกอบไปด้วย ข้อมูลรหัสสมาชิก, ชื่อ, นามสกุล, ที่อยู่, เบอร์โทรศัพท์, E-mail, รหัสสินค้า, ชื่อสินค้า, วันที่สั่งซื้อ, วันที่จัดส่ง, สาขาที่จัดส่ง

3. โค้ดโปรแกรมแปลงไฟล์อักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับโปรแกรมเฉพาะ



รูปที่ 4.14 แสดง โค้ด โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด สำหรับ โปรแกรมเฉพาะที่พัฒนาด้วยด้วยภาษา C#

จากรูปที่ 4.14 แสดง โค้ดคำสั่งของ โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของ คีย์บอร์ดสำหรับ โปรแกรมเฉพาะ ที่มีรูปแบบการส่งข้อมูลไปยัง โปรแกรมเฉพาะ โดยการส่งข้อมูลแต่ละ คอลัมน์ (\$col[]) ในไฟล์ข้อมูลไปกรอกช่องข้อมูลของ โปรแกรมเฉพาะ โดยอัตโนมัติ

\$col[1] จะเป็นการกรอกข้อมูลในช่องรหัสสมาชิก

\$col[2] จะเป็นการกรอกข้อมูลในช่องชื่อ

\$col[3] จะเป็นการกรอกข้อมูลในช่องนามสกุล

\$col[4] จะเป็นการกรอกข้อมูลในช่องที่อยู่

\$col[5] จะเป็นการกรอกข้อมูลในช่องเบอร์โทรศัพท์

\$col[6] จะเป็นการกรอกข้อมูลในช่องE-mail

\$col[7] จะเป็นการกรอกข้อมูลในช่องรหัสสินค้า

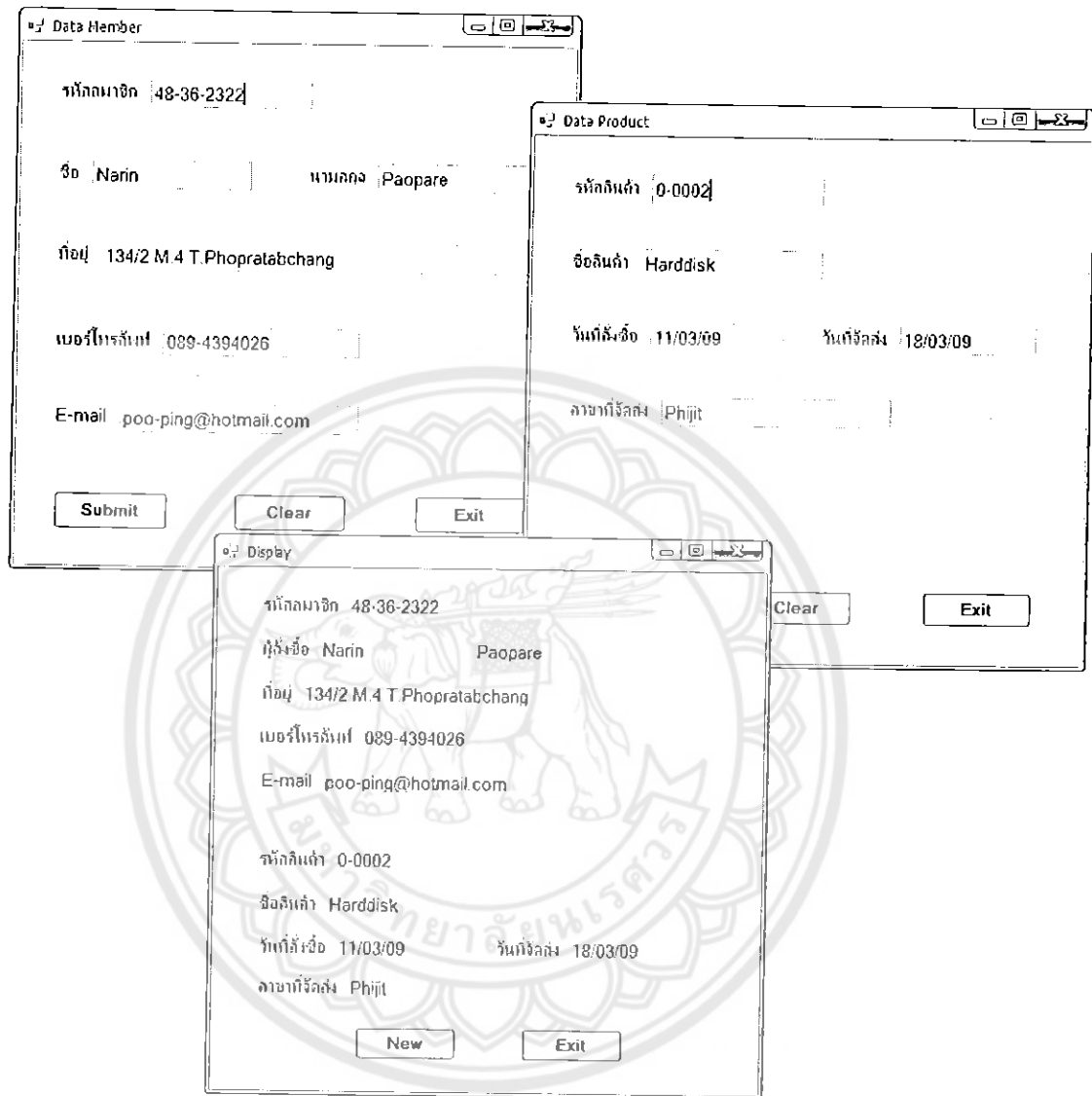
\$col[8] จะเป็นการกรอกข้อมูลในช่องชื่อสินค้า

\$col[9] จะเป็นการกรอกข้อมูลในช่องวันที่สั่งซื้อ

\$col[10] จะเป็นการกรอกข้อมูลในช่องวันที่จัดส่ง

\$col[11] จะเป็นการกรอกข้อมูลในช่องสาขาที่จัดส่ง

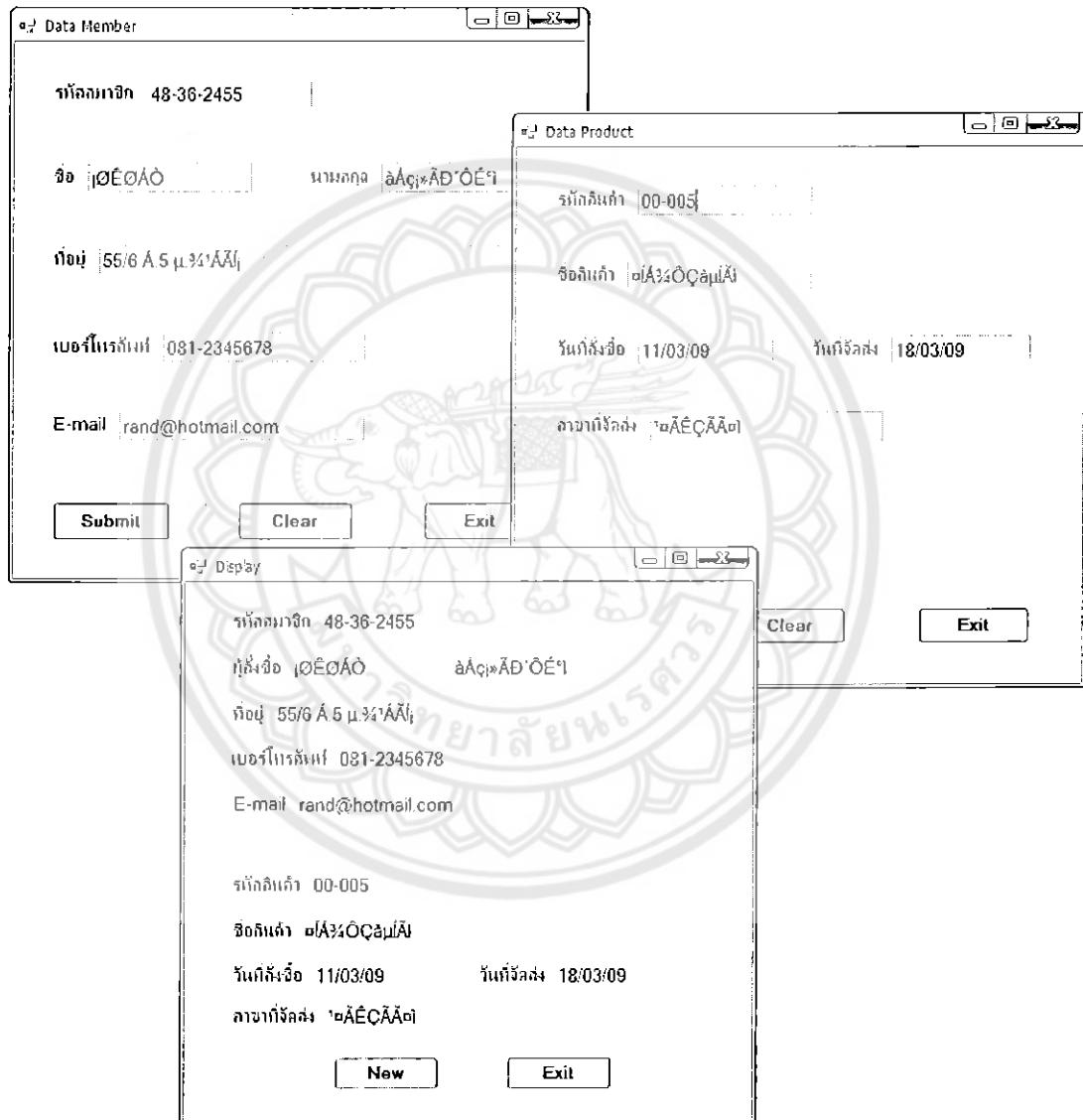
4. ผลการทดลองกับโปรแกรมเฉพาะจะพบว่าสามารถนำเข้าข้อมูลจากไฟล์ข้อมูลมาไว้ในโปรแกรมเฉพาะได้อย่างถูกต้อง โดยได้ผลที่เหมือนกันทั้ง Text file และ Excel file ดังรูป



รูปที่ 4.15 แสดงผลการทดสอบกับ โปรแกรมเฉพาะที่พัฒนาด้วยภาษา C#

จากรูปที่ 4.15 จะแสดงให้เห็นว่า ข้อมูลที่นำเข้ามาจากไฟล์อักษรทั้งสองชนิดคือ Text file และ Excel file โดยผ่านโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด สามารถทำการกรอกลงในโปรแกรมเฉพาะได้อย่างถูกต้องครบถ้วน

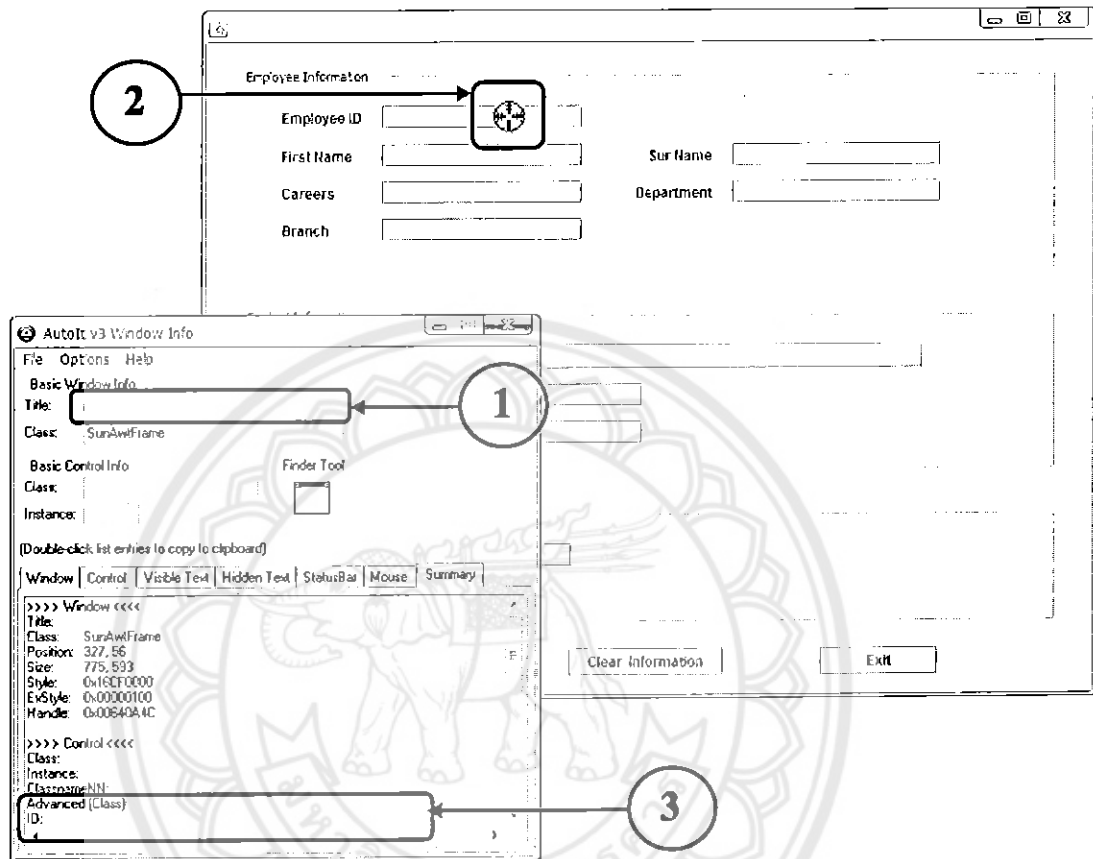
จากการทดลองจะเห็นได้ว่า โปรแกรมเฉพาะที่นำมาตรวจสอบสามารถรับและแสดงค่าที่ได้รับจากไฟล์ข้อมูลทั้งสองชนิด คือ Text file และ Excel file ได้อย่างถูกต้อง โดยรับผ่านโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับโปรแกรมเฉพาะที่มีการปรับปรุงรูปแบบ Source code ให้เป็นไปตามลักษณะการใช้งานของโปรแกรมเฉพาะแต่ละรูปแบบ ซึ่งพบว่า ในส่วนของโปรแกรมเฉพาะที่พัฒนาด้วยภาษา C# ไม่สามารถรับค่าผ่านโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับโปรแกรมเฉพาะจากไฟล์ที่มีข้อมูลเป็นภาษาไทยได้



รูปที่ 4.16 แสดงผลการทดสอบ โปรแกรมเฉพาะที่พัฒนาด้วยภาษา C# กับ ไฟล์ข้อมูลที่เป็นภาษาไทย

จากรูปที่ 4.16 แสดงให้เห็นว่า โปรแกรมเฉพาะที่พัฒนาด้วยภาษา C# ไม่สามารถรับข้อมูลที่เป็นภาษาไทยผ่าน โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดได้ โดยจะสังเกตได้ว่า ผลที่แสดงออกมาไม่ออกมาให้รูปแบบภาษาไทยที่ถูกต้อง

นอกจากนี้ เรายังไม่สามารถใช้โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด สำหรับโปรแกรมเฉพาะกับโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java ได้ เนื่องจากเครื่องมือที่ใช้ในการอ่านค่าส่วนประกอบของโปรแกรม ไม่สามารถใช้กับโปรแกรมที่พัฒนาด้วยภาษา Java ได้



รูปที่ 4.17 แสดงผลการทดสอบการระบุส่วนประกอบ โปรแกรมเฉพาะที่พัฒนาด้วย ภาษา Java ด้วยเครื่องมือ AutoIt Window Info

จากรูปที่ 4.17 จะเห็นได้ว่าเมื่อนำส่วนที่ 2 (Finder Tool) ไปชี้ที่ตัวโปรแกรมที่พัฒนาด้วยภาษา Java แล้วไม่สามารถระบุส่วนที่ 1 (Title) ซึ่งเป็นชื่อของหน้าต่างโปรแกรมเฉพาะ และส่วนที่ 3 (controlID) ซึ่งเป็นชื่อของส่วนประกอบภายในโปรแกรมเฉพาะ ได้ทั้งส่วนที่เป็น Advanced(class) และ ID

4.2 ผลการทดลองการใช้โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดกับโปรแกรมเฉพาะที่ใช้ในชีวิตประจำวัน

ในการทดลองนั้น เรายังไม่สามารถระบุประสิทธิภาพการทำงานของโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดได้อย่างสมบูรณ์ เนื่องจากยังขาดการนำไปใช้งานจริงในชีวิตประจำวัน จึงได้มีการนำตัวอย่างโปรแกรมเฉพาะที่ใช้อยู่ทั่วไปมาทำการทดสอบ

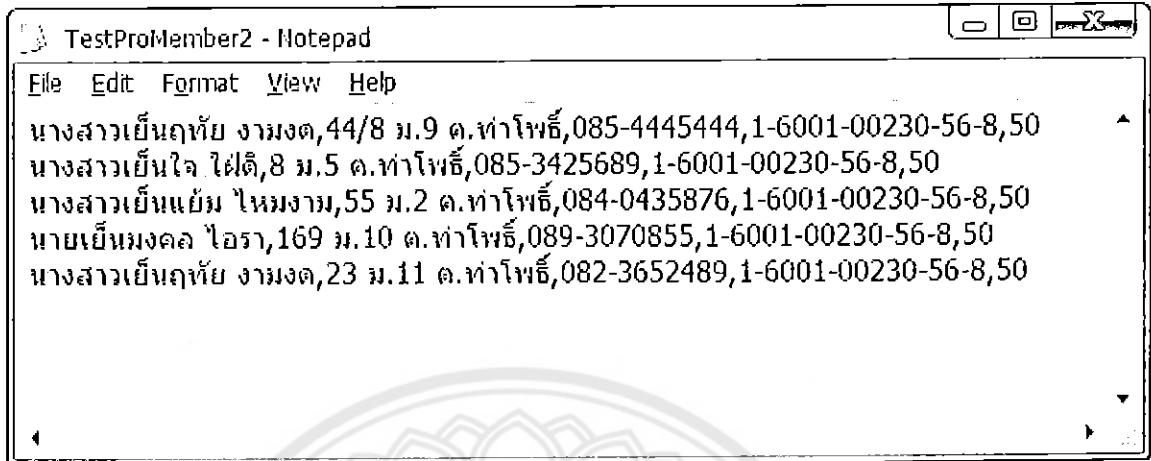
4.2.1 การทดสอบกับข้อมูลสมาชิกของโปรแกรมร้านเช่าหนังสือ

1. หน้าต่างโปรแกรมที่ยังไม่ได้มีการกรอกข้อมูล

รูปที่ 4.18 หน้าต่างกรอกข้อมูลสมาชิกของโปรแกรมร้านเช่าหนังสือ ขณะที่ยังไม่ได้มีการกรอกข้อมูล

จากรูปที่ 4.18 จะเห็นได้ว่ามีหน้าต่างสำหรับกรอกข้อมูลสมาชิกร้านเช่าหนังสือ เมื่อทำการกรอกข้อมูลเรียบร้อยแล้ว เราสามารถเข้าไปตรวจสอบรายละเอียดที่กรอกได้ที่เมนูแก้ไข/ลบสมาชิก โดยข้อมูลที่จะต้องกรอกมีทั้งหมด 5 ช่อง

2. ไฟล์ข้อมูลที่น่ามาทดสอบมีสองชนิดคือ Text file (.txt) และ Excel file (.csv) โดยมีรายละเอียดของข้อมูลที่เหมือนกัน



รูปที่ 4.19 แสดงไฟล์ข้อมูลชนิด Text file ซึ่งนำมาทดสอบกับข้อมูลสมาชิก
ของโปรแกรมร้านเช่าหนังสือ

	A	B	C	D	E
1	นางสาวเย็นฤทัย งามงด	44/8 ม.9 ต.ท่าโพธิ์	085-4445444	1-6001-00230-56-8	50
2	นางสาวเย็นใจ ใฝ่ดี	8 ม.5 ต.ท่าโพธิ์	085-3425689	1-6001-00230-56-8	50
3	นางสาวเย็นแยม ไหมงาม	55 ม.2 ต.ท่าโพธิ์	084-0435876	1-6001-00230-56-8	50
4	นายเย็นมงคล ไชรา	169 ม.10 ต.ท่าโพธิ์	089-3070855	1-6001-00230-56-8	50
5	นางสาวเย็นฤทัย งามงด	23 ม.11 ต.ท่าโพธิ์	082-3652489	1-6001-00230-56-8	50
6					
7					
8					

รูปที่ 4.20 แสดงไฟล์ข้อมูลชนิด Excel file ซึ่งนำมาทดสอบกับข้อมูลสมาชิก
ของโปรแกรมร้านเช่าหนังสือ

จากรูปที่ 4.19 และ 4.20 แสดงไฟล์ข้อมูลที่ต้องการนำไปกรอกลงในโปรแกรมเฉพาะ จะเห็นได้ว่าในไฟล์ข้อมูลจะมีจำนวนข้อมูลตามจำนวนของช่องข้อมูลในโปรแกรมเฉพาะ โดยคั่นข้อมูลที่จะกรอกแต่ละช่องด้วยเครื่องหมายจุดภาค ในส่วนของรูปที่ 4.3 จะมีการคั่นข้อมูลโดยการเก็บเป็นเซลล์ ซึ่งข้อมูลที่มีการคั่นประกอบไปด้วย ข้อมูลชื่อ-นามสกุล, ที่อยู่, เบอร์โทรศัพท์, หมายเลขบัตรประชาชน และค่าสมัคร

4. ผลการทดลองกับโปรแกรมเฉพาะจะพบว่าสามารถนำเข้าข้อมูลจากไฟล์ข้อมูลมาใส่ในโปรแกรมเฉพาะได้อย่างถูกต้อง โดยได้ผลที่เหมือนกันทั้ง Text file และ Excel file ดังรูป

โปรแกรมร้านเช่าหนังสือ (Unregister Version) - [ลงทะเบียนสมาชิกใหม่]

การเช่า/คืนหนังสือ สมาชิก หนังสือ หนังสืองาน ตรวจสอบ สำหรับผู้จัดการ เกี่ยวกับโปรแกรมนี้
ออกจากโปรแกรมนี้

UserID: [00000000]

ชื่อ-นามสกุล: [นายเอ็นเมงคล ไชย]

ที่อยู่: [169 ม.10 ต.ท่าโพธิ์]

เบอร์โทรศัพท์: [089-3070855]

หมายเลขบัตรประชาชน: [1-6001-00230-56-8]

ลงทะเบียนสมาชิก (A) ออกจากการลงทะเบียน

สถานะ: NUM ศุกร์ 15/5/2552 4.53

เมนู: เช่าหนังสือ คืนหนังสือ ลงทะเบียนสมาชิกใหม่ Book ลงทะเบียนหนังสือใหม่ ออกจากโปรแกรม เกี่ยวกับโปรแกรมนี้

รูปที่ 4.22 แสดงผลการทดสอบกับการกรอกข้อมูลสมาชิก โปรแกรมร้านเช่าหนังสือ

จากรูปที่ 4.22 แสดงให้เห็นว่าสามารถกรอกข้อมูลผ่านทางโปรแกรมแปลงตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดโดยรับค่าจากไฟล์ทั้งสองชนิดได้ได้อย่างถูกต้อง

โปรแกรมร้านเช่าหนังสือ (Unregister Version) - [แก้ไข/ลบ สมาชิก]

การเช่า/คืนหนังสือ สมาชิก หนังสือ หนังสืองาน ตรวจสอบ สำหรับผู้จัดการ เกี่ยวกับโปรแกรมนี้
ออกจากโปรแกรมนี้

UserID: [00001] ค้นหา (F)

ชื่อ-นามสกุล: นางสาวเย็นฤทัย งามงด

ที่อยู่: 44/8 ม.9 ต.ท่าโพธิ์

เบอร์โทรศัพท์: 085-4445444 วันที่ออก: 15/5/2552

หมายเลขบัตรประชาชน: 1-6001-00230-56-8 วันหมดอายุ: 14/5/2553







วันที่ลงทะเบียน: 15/5/2552 จำนวนปีหมดอายุ: 1 แก้ไขวันหมดอายุ

แก้ไข สมาชิก (E) ลบ สมาชิก (D) ออก พิมพ์บัตร (P)

Record: 1

รายชื่อสมาชิก	
รหัสสมาชิก	ชื่อ-นามสกุล
00001	นางสาวเย็นฤทัย งามงด
00002	นางสาวเย็นใจ ใฝ่ดี
00003	นางสาวเย็นแก้ว ไหมงาม
00004	นายเย็นมงคล ไชรว
00005	นางสาวเย็นฤทัย งามงด

สถานะ: [] NUM: [] ผู้กร้: 15/5/2552 5:00

 เช่าหนังสือ
 คืนหนังสือ
 ลงทะเบียนสมาชิกใหม่
 ลงทะเบียนหนังสือใหม่
 ออกจากโปรแกรม
 เกี่ยวกับโปรแกรมนี้

รูปที่ 4.23 แสดงผลฐานข้อมูลสมาชิกโปรแกรมร้านเช่าหนังสือหลังจากการทดสอบกับโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด

จากรูปที่ 4.23 เมื่อทำการตรวจสอบความถูกต้องจากฐานข้อมูลสมาชิกของโปรแกรมร้านเช่าหนังสือแล้ว พบว่า ข้อมูลที่กรอกผ่านโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสามารถกรอกข้อมูลลงไปได้ครบถ้วน

4.2.2 การทดสอบกับข้อมูลพนักงานโปรแกรม MyRestaurant

1. หน้าต่างโปรแกรมที่ยังไม่ได้มีการกรอกข้อมูล

The screenshot shows the MyRestaurant software interface. At the top, there is a title bar with 'MyRestaurant 2.0 Cash -> DEMO' and a 'DEMO' label. Below the title bar is a grid of employee data with columns numbered 001 to 138. A form titled 'แถมรายการพนักงาน' (Employee Record) is overlaid on the grid. The form contains the following fields:

- รหัส: (Blank)
- ชื่อ : นามสกุล: (Blank)
- ตำแหน่ง: (Blank)
- ที่อยู่: (Blank)
- บัตร ประ: (Blank)
- เงินเดือน: (Blank)

At the bottom of the form, there are five buttons: F12-บันทึก (Record), F1-แสดง (Display), F1-ลบ (Delete), F9-ยกเลิก (Cancel), and ESC-จบ (End).

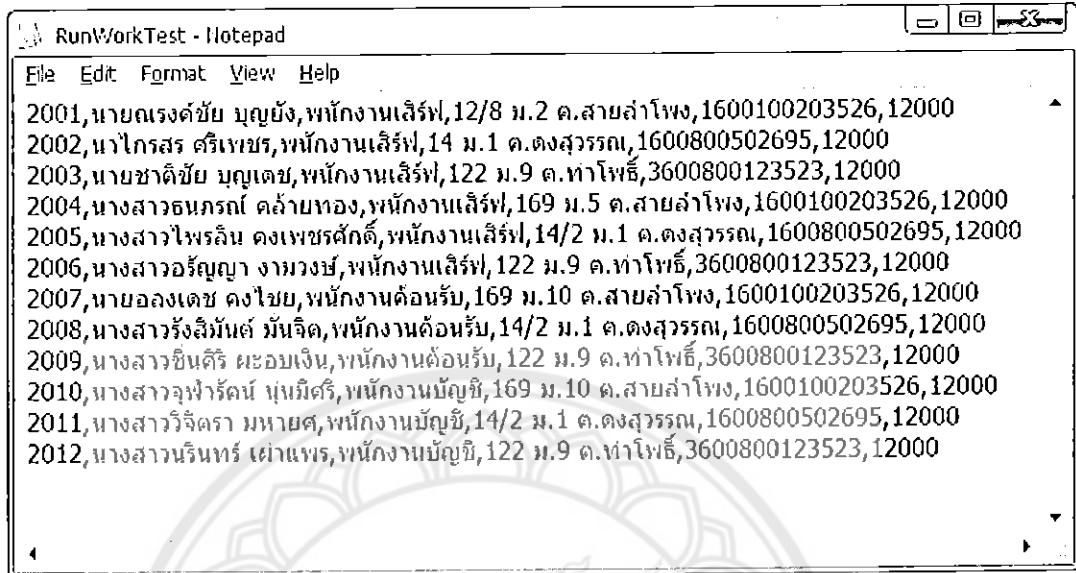
The date and time at the bottom right of the interface are: Date / Time : 15/05/2009 06:17.

รูปที่ 4.24 หน้าต่างการกรอกข้อมูลพนักงานของโปรแกรม MyRestuarant

ขณะที่ยังไม่ได้มีการกรอกข้อมูล

จากรูปที่ 4.24 จะเห็นได้ว่ามีหน้าต่างสำหรับกรอกข้อมูลพนักงานร้านอาหาร เมื่อทำการกรอกข้อมูลเรียบร้อยแล้ว เราสามารถเข้าไปทำการตรวจสอบข้อมูลได้ที่เมนูแสดงข้อมูลพนักงาน (F1-แสดง) โดยข้อมูลที่จะต้องกรอกมีทั้งหมด 6 ช่อง

2. ไฟล์ข้อมูลที่น่ามาทดสอบมีสองชนิดคือ Text file (.txt) และ Excel file (.csv) โดยมีรายละเอียดของข้อมูลที่เหมือนกัน



รูปที่ 4.25 แสดงไฟล์ข้อมูลชนิด Text file ซึ่งนำมาทดสอบกับโปรแกรม MyRestuarant

	A	B	C	D	E	F
1	2001	นายณรงค์ชัย บุญยัง	พนักงานเสิร์ฟ	12/8 ม.2 ค.สายลำโพง	1600100203526	12000
2	2002	นาไกรสร ศรีเพชร	พนักงานเสิร์ฟ	14 ม.1 ค.ดงสุวรรณ	1600800502695	12000
3	2003	นายชาติชัย บุญเดช	พนักงานเสิร์ฟ	122 ม.9 ค.ท่าโพธิ์	3600800123523	12000
4	2004	นางสาวธนภรณ์ คล้ายทอง	พนักงานเสิร์ฟ	169 ม.5 ค.สายลำโพง	1600100203526	12000
5	2005	นางสาวไพรลีน ดงเพชรศักดิ์	พนักงานเสิร์ฟ	14/2 ม.1 ค.ดงสุวรรณ	1600800502695	12000
6	2006	นางสาวอรุณญา งามวงษ์	พนักงานเสิร์ฟ	122 ม.9 ค.ท่าโพธิ์	3600800123523	12000
7	2007	นายอลงเดช ดงไชย	พนักงานต้อนรับ	169 ม.5 ค.สายลำโพง	1600100203526	12000
8	2008	นางสาวรังสิมันต์ มั่นจิต	พนักงานต้อนรับ	14/2 ม.1 ค.ดงสุวรรณ	1600800502695	12000
9	2008	นางสาวชินศิริ ะยอมเงิน	พนักงานต้อนรับ	122 ม.9 ค.ท่าโพธิ์	3600800123523	12000
10	2010	นางสาวจุฬารัตน์ พูนมิตร	พนักงานบัญชี	169 ม.5 ค.สายลำโพง	1600100203526	12000
11	2011	นางสาววิจิตรรา มหายศ	พนักงานบัญชี	14/2 ม.1 ค.ดงสุวรรณ	1600800502695	12000
12	2012	นางสาวรินทร์ร เผ่าแพร	พนักงานบัญชี	122 ม.9 ค.ท่าโพธิ์	3600800123523	12000
13						

รูปที่ 4.26 แสดงไฟล์ข้อมูลชนิด Excel file ซึ่งนำมาทดสอบกับโปรแกรม MyRestuarant

จากรูปที่ 4.25 และ 4.26 แสดงไฟล์ข้อมูลที่ต้องการนำไปกรอกลงในโปรแกรมเฉพาะ จะเห็นได้ว่าในไฟล์ข้อมูลจะมีจำนวนข้อมูลตามจำนวนของช่องข้อมูลในโปรแกรมเฉพาะ โดยคั่นข้อมูลที่จะกรอกแต่ละช่องด้วยเครื่องหมายจุดภาค ในส่วนของรูปที่ 4.3 จะมีการคั่นข้อมูลโดยการเก็บเป็นเซลล์ ซึ่งข้อมูลที่มีการคั่นประกอบไปด้วย ข้อมูลรหัส, ชื่อ-นามสกุล, ตำแหน่ง, ที่อยู่, หมายเลขบัตรประชาชน และเงินเดือน

3. โค้ดโปรแกรมแปลงไฟล์อักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับโปรแกรมเฉพาะ

```

WinWait("ชื่อหน้าต่างโปรแกรม")
ControlSend("ชื่อหน้าต่างโปรแกรม", " ", [CLASS:msctedit][INST:0][TEXT: ] , $col[1])
ControlSend("ชื่อหน้าต่างโปรแกรม", " ", [CLASS:msctedit][INST:0][TEXT: ] , $col[2])
ControlSend("ชื่อหน้าต่างโปรแกรม", " ", [CLASS:msctedit][INST:0][TEXT: ] , $col[3])
ControlSend("ชื่อหน้าต่างโปรแกรม", " ", [CLASS:msctedit][INST:0][TEXT: ] , $col[4])
ControlSend("ชื่อหน้าต่างโปรแกรม", " ", [CLASS:msctedit][INST:0][TEXT: ] , $col[5])
ControlSend("ชื่อหน้าต่างโปรแกรม", " ", [CLASS:msctedit][INST:0][TEXT: ] , $col[6])

Sleep(1000)
ControlClick("ชื่อหน้าต่างโปรแกรม", " ", [CLASS:msctedit][INST:0][TEXT: ] , 1)

```

รหัส
 ชื่อ-นามสกุล
 ตำแหน่ง
 ที่อยู่
 หมายเลขบัตรประชาชน
 เงินเดือน

รูปที่ 4.27 แสดงโค้ด โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด สำหรับใช้กับข้อมูลพนักงาน โปรแกรม MyRestuarant

จากรูปที่ 4.27 แสดงโค้ดคำสั่งของ โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับโปรแกรมเฉพาะ ที่มีรูปแบบการส่งข้อมูลไปยังโปรแกรมเฉพาะ โดยการส่งข้อมูลแต่ละคอลัมน์ (\$col[]) ในไฟล์ข้อมูลไปกรอกช่องข้อมูลของโปรแกรมเฉพาะ โดยอัตโนมัติ

\$col[1] จะเป็นการกรอกข้อมูลในช่องรหัส

\$col[2] จะเป็นการกรอกข้อมูลในช่องชื่อ-นามสกุล

\$col[3] จะเป็นการกรอกข้อมูลในช่องตำแหน่ง

\$col[4] จะเป็นการกรอกข้อมูลในช่องที่อยู่

\$col[5] จะเป็นการกรอกข้อมูลในช่องหมายเลขบัตรประชาชน

\$col[6] จะเป็นการกรอกข้อมูลในช่องเงินเดือน

3. ผลการทดลองกับ โปรแกรมเฉพาะจะพบว่าสามารถนำเข้าข้อมูลจากไฟล์ข้อมูลมาใส่ใน โปรแกรมเฉพาะได้อย่างถูกต้อง โดยได้ผลที่เหมือนกันทั้ง Text file และ Excel file ดังรูป

The screenshot displays a software interface with a grid of data at the top and a form for employee information below. The grid contains 16 columns and 16 rows of numerical data. The form is titled 'แผนกรับกรพนักงาน' and contains the following information:

- รหัส: 2001
- ชื่อ: นายสกุล: นายณรงค์ชัย บุญยัง
- ตำแหน่ง: พนักงานเสิร์ฟ
- ที่อยู่: 12/8 ม.2 คลายสำโพง
- บัตรประชาชน: 1600100203526
- เงินเดือน: 12000

Below the form are five buttons labeled F12-บันทึก, F6-ลบ, F1-ลบ, F9-ยกเลิก, and ESC-จบ. The date and time are shown as 15/05/2009 06:17.

รูปที่ 4.28 แสดงผลการทดสอบกรอกข้อมูลพนักงานของ โปรแกรม MyRestuarant

จากรูปที่ 4.28 แสดงให้เห็นว่ามีการนำข้อมูลจากไฟล์อักษรทั้งชนิดที่เป็น Text file และ Excel file เข้ามาใส่ใน โปรแกรมเฉพาะได้อย่างถูกต้อง

แสดงรายการข้อมูลพนักงานทั้งหมด

รหัส	ชื่อ	ตำแหน่ง
2001	นายณรงค์ชัย บุญยัง	พนักงานเสิร์ฟ
2002	นางไกรสร ศรีเพชร	พนักงานเสิร์ฟ
2003	นายชาติชัย บุญเดช	พนักงานเสิร์ฟ
2004	นางสาวธนกรดี คล้ายทอง	พนักงานเสิร์ฟ
2005	นางสาวไพโรลิน คงเพชรศักดิ์	พนักงานเสิร์ฟ
2006	นางสาวอริญญา งามวงษ์	พนักงานเสิร์ฟ
2007	นายอลงเดช คงไชย	พนักงานต้อนรับ
2008	นางสาวรังสิมันต์ มั่นเจ็ด	พนักงานต้อนรับ
2009	นางสาวชินศิริ ผะอบเงิน	พนักงานต้อนรับ
2010	นางสาวจุฬารัตน์ นุ่นมีศรี	พนักงานบัญชี
2011	นางสาววิจิตรว มหายศ	พนักงานบัญชี
2012	นางสาวนรินทร์ เผ่าแพร่	พนักงานบัญชี

รูปที่ 4.29 แสดงผลฐานข้อมูลพนักงาน โปรแกรม MyRestuarant หลังจากการทดสอบกับโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด

จากรูปที่ 4.29 เป็นการทำการตรวจสอบความถูกต้องจากฐานข้อมูลของโปรแกรม MyRestuarant พบว่า ข้อมูลที่กรอกผ่าน โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสามารถกรอกข้อมูลลงไปได้ครบถ้วน

บทที่ 5

สรุปผล

5.1 สรุปผลการทดลอง

สิ่งที่จำเป็นต้องใช้ในการทดลองนั้น คือ ไฟล์ข้อมูล โปรแกรมเฉพาะ และ โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของกีบบอร์ดสำหรับโปรแกรมเฉพาะ โดยไฟล์ข้อมูลที่สามารถนำมาใช้ในการทดลองได้นั้นมีอยู่สองชนิด คือ Text file (.txt) และ Excel file (.csv) ซึ่งจะทำการรับค่าไฟล์เหล่านี้ผ่านทางโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของกีบบอร์ดสำหรับโปรแกรมเฉพาะ แล้วทำการส่งข้อมูลไปยังโปรแกรมเฉพาะ โดยโปรแกรมเฉพาะที่ใช้ในการทดลองมีทั้งหมด 3 โปรแกรมด้วยกัน คือ โปรแกรมที่พัฒนาด้วยภาษา C# โปรแกรมที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในฟอร์มสุดท้าย และแบบที่มีฐานข้อมูล

โปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic สามารถรับค่าที่ส่งจากโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของกีบบอร์ดสำหรับโปรแกรมเฉพาะได้อย่างถูกต้อง โดยรับได้ทั้งไฟล์ที่เป็น Text file และ Excel file รวมไปถึงสามารถรับได้ทั้งไฟล์ที่เป็นภาษาไทย และภาษาอังกฤษ อีกทั้งยังสามารถรับการสั่งการอื่นๆ เช่น การกดปุ่มต่างๆ ได้อีกด้วย จึงสามารถรับข้อมูลได้ทีละหลายชุดในคราวเดียวกัน

สำหรับโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่มีฐานข้อมูล ก็สามารถบันทึกค่าข้อมูลที่ส่งมาจากโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของกีบบอร์ดสำหรับโปรแกรมเฉพาะลงบนฐานข้อมูลได้อย่างถูกต้องเช่นเดียวกัน

ส่วนโปรแกรมที่พัฒนาด้วยภาษา C# นั้น สามารถรับค่าที่ส่งจากโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของกีบบอร์ดสำหรับโปรแกรมเฉพาะทั้งไฟล์ที่เป็น Text file และ Excel file รวมทั้งคำสั่งในการกดปุ่มต่างๆ และรับข้อมูลที่หลายๆ ชุดได้เช่นเดียวกับโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic แต่ไม่สามารถรับไฟล์ที่มีข้อมูลเป็นภาษาไทยได้

ในส่วนของโปรแกรมที่พัฒนาด้วยภาษา Java นั้น ไม่สามารถระบุส่วนประกอบของโปรแกรมด้วยเครื่องมือ AutoIt Window Info ได้ จึงไม่สามารถรับไฟล์ข้อมูลผ่านทางโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของกีบบอร์ดสำหรับโปรแกรมเฉพาะได้

5.2 ปัญหาและอุปสรรค

1. โปรแกรมไม่สามารถใช้กับโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java ได้
2. โปรแกรมไม่สามารถแปลงไฟล์ข้อมูลที่เป็นภาษาไทย ให้กับโปรแกรมเฉพาะที่พัฒนาด้วยภาษา C# ได้
3. ต้องมีการปรับปรุง โปรแกรมทุกครั้งที่ปรับเปลี่ยน โครงสร้างของ โปรแกรมเฉพาะ

5.3 ข้อเสนอแนะ

ในการปรับปรุง โปรแกรมด้วยเครื่องมือ AutoIt Window Info ควรใช้ Advanced(class) ในการระบุชื่อส่วนประกอบของ โปรแกรมเฉพาะ ซึ่งจะสามารถระบุการส่งข้อมูล ได้ดีกว่า ID

5.4 สรุป

เราได้ทำการศึกษาการแปลงสัญญาณอินพุทของคีย์บอร์ด เพื่อนำไปใช้กับโปรแกรมเฉพาะที่สามารถรับค่าข้อมูลจากคีย์บอร์ดได้เพียงอย่างเดียวเท่านั้น ให้สามารถรับค่าข้อมูลจากไฟล์ตัวอักษรได้ โดยเราได้ทำการพัฒนาโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับ โปรแกรมเฉพาะ ขึ้นมาจากการใช้ลักษณะพิเศษของ โปรแกรม AutoIt ที่สามารถสั่งการโปรแกรมต่างๆ ที่ใช้ใน Microsoft Window ได้ ซึ่งลักษณะดังกล่าวนี้เอง เราจึงนำมาประยุกต์ใช้ในการสั่งการให้กับโปรแกรมเฉพาะให้สามารถรับค่าข้อมูลจากไฟล์ตัวอักษรได้

หลังจากนั้นเราได้ทำการพัฒนาโปรแกรม โดยให้สามารถใช้งานกับ โปรแกรมเฉพาะที่มีรูปแบบของภาษาและรูปแบบการใช้งานที่แตกต่างกัน ได้ในหลายรูปแบบ จากการทดลองโปรแกรมที่พัฒนาขึ้นกับ โปรแกรมเฉพาะเหล่านี้ พบว่าโปรแกรมสามารถที่จะแปลง ไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดส่งผ่านไปยังโปรแกรมเฉพาะได้

บรรณานุกรม

- [1] ชาญชัย หวางศ์. “.” [Online]. Available : http://203.172.183.73/chan/index.php?option=com_docman&task=doc_download&gid=12&Itemid=26. 2549
- [2] ทวีชัย หงส์สุมาลย์. **อินไซต์ ASP.NET ฉบับสมบูรณ์**. กรุงเทพฯ : โปรวิชั่น จำกัด. 2546
- [3] พร้อมเลิศ หล่อวิจิตร. **คู่มือเรียน Visual Basic 2005**. กรุงเทพฯ : โปรวิชั่น จำกัด. 2549
- [4] ไม่ปรากฏชื่อผู้แต่ง. “AutoIt v3 Automate and Script Window Tasks” [Online]. Available : <http://www.autoitscript.com/autoit3/>
- [5] ไม่ปรากฏชื่อผู้แต่ง. “หลักการงานของคีย์บอร์ด.” [Online]. Available : http://hospital.moph.go.th/phetchabun/siam/web_final/vcb1/keyboard.html. 2549



ภาคผนวก ก

โค้ดคำสั่งโปรแกรมเฉพาะ

โค้ดคำสั่งโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ซึ่งแสดงผลข้อมูลในฟอร์มสุดท้าย

```

Private Sub Command1_Click()
Dim name As String
Dim surname As String
Dim id As String
Dim position As String
Dim categories As String
Dim subcate As String

id = Text1.Text
name = Text2.Text
surname = Text3.Text
position = Text4.Text
categories = Text5.Text
subcate = Text6.Text

Form3.Text1.Text = id
Form3.Text2.Text = name
Form3.Text3.Text = surname
Form3.Text4.Text = position
Form3.Text12.Text = categories
Form3.Text5.Text = subcate

If (Option1 = True) Then Form3.Text13.Text = "๐"
If (Option2 = True) Then Form3.Text13.Text = "๑"

Unload Me
Form2.Show

End Sub

Private Sub Command2_Click()
msg = MsgBox("กรุณากดปุ่ม OK เพื่อปิดโปรแกรม", vbOKCancel, "แจ้งเตือน")

If msg = vbOK Then

Text1.Text = ""
Text2.Text = ""
Text3.Text = ""
Text4.Text = ""
Text5.Text = ""
Text6.Text = ""
Else
End If
End Sub

Private Sub Command3_Click()
End
End Sub

```

รูปที่ ก.1 แสดงโค้ดฟอร์ม 1 ของโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic
ซึ่งแสดงผลข้อมูลในฟอร์มสุดท้าย

```

Private Sub Command1_Click()
Dim address As String
Dim telephone As String
Dim email As String
Dim money As String
Dim otpercent As Integer
Dim ot As String

address = Text1.Text
telephone = Text2.Text
email = Text3.Text
money = Text4.Text
money = Val(money)
ot = Text5.Text
ot = Val(ot)

If ot < 5 Then
otpercent = 5
ElseIf ot >= 5 And ot < 11 Then
otpercent = 10
ElseIf ot >= 11 And ot < 16 Then
otpercent = 15
ElseIf ot >= 16 And ot < 26 Then
otpercent = 20
Else
otpercent = 25
End If

Form3.Text6.Text = address
Form3.Text7.Text = telephone
Form3.Text8.Text = email
Form3.Text9.Text = money
Form3.Text10.Text = ot
Form3.Text11.Text = money + (money * otpercent) / 100

Unload Me
Form3.Show
End Sub

Private Sub Command2_Click()
End
End Sub

Private Sub Command3_Click()
msg = MsgBox("กรุณาใส่ข้อมูลให้ครบถ้วน", vbOKCancel, "แจ้งเตือน")
If msg = vbOK Then
Text1.Text = ""
Text2.Text = ""
Text3.Text = ""
Text4.Text = ""
Text5.Text = ""

Form1.Text1.Text = ""
Form1.Text2.Text = ""
Form1.Text3.Text = ""
Form1.Text4.Text = ""
Form1.Text5.Text = ""
Form1.Text6.Text = ""
Unload Me
Form1.Show

Else
End If
End Sub

Private Sub Form_Load()
End Sub

```

รูปที่ ก.2 แสดงโค้ดฟอร์ม 2 ของโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic
ซึ่งแสดงผลข้อมูลในฟอร์มสุดท้าย

```
Private Sub Command1_Click()  
Unload Me  
Form1.Show  
End Sub
```

```
Private Sub Command2_Click()  
End  
End Sub
```

รูปที่ ก.3 แสดง โค้ดของ Form3 ของโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic
ซึ่งแสดงผลข้อมูลในฟอร์มสุดท้าย



โค้ดคำสั่งโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ซึ่งแสดงผลข้อมูลในตารางฐานข้อมูล

```

Private Sub Command1_Click() 'Save
On Error GoTo EL

If Trim(Text1.Text) = "" Then Exit Sub

Adodc1.RecordSource = "select * from student"
Adodc1.Refresh
Adodc1.Recordset.AddNew
Adodc1.Recordset.Fields(0) = Val(Text1.Text)
Adodc1.Recordset.Fields(1) = Text2.Text
Adodc1.Recordset.Fields(2) = Text3.Text
Adodc1.Recordset.Fields(3) = Text4.Text
Adodc1.Recordset.Fields(4) = Text5.Text
Adodc1.Recordset.Fields(5) = Text6.Text
Adodc1.Recordset.Fields(6) = Text7.Text
Adodc1.Recordset.Fields(7) = Text8.Text
Adodc1.Recordset.Update

MsgBox "บันทึกข้อมูลสำเร็จ"

EL:
End Sub

Private Sub Command2_Click() 'Delete
On Error GoTo EL

If Trim(Text2.Text) = "" Then Exit Sub

Adodc1.RecordSource = "select * from student where ชื่อผู้สมัคร = '" & Trim(Text2.Text) & "'"
Adodc1.Refresh

If Not Adodc1.Recordset.EOF Then

Adodc1.Recordset.Delete
Adodc1.Recordset.Update
MsgBox "ลบข้อมูลสำเร็จ"

End If
EL:
End Sub

Private Sub Command3_Click() 'Edit
On Error GoTo EL

If Trim(Text2.Text) = "" Then Exit Sub

Adodc1.RecordSource = "select * from student where ชื่อผู้สมัคร = '" & Trim(Text2.Text) & "'"
Adodc1.Refresh

If Not Adodc1.Recordset.EOF Then

Adodc1.Recordset.Fields(0) = Val(Text1.Text)
Adodc1.Recordset.Fields(1) = Text2.Text
Adodc1.Recordset.Fields(2) = Text3.Text
Adodc1.Recordset.Fields(3) = Text4.Text
Adodc1.Recordset.Fields(4) = Text5.Text
Adodc1.Recordset.Fields(5) = Text6.Text
Adodc1.Recordset.Fields(6) = Text7.Text
Adodc1.Recordset.Fields(7) = Text8.Text
Adodc1.Recordset.Update
MsgBox "แก้ไขข้อมูลสำเร็จ"

End If
EL:
End Sub

Private Sub Command4_Click() 'Exit
End
End Sub

```

รูปที่ ก.4 แสดงโค้ดส่วนที่ 1 ของโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ซึ่งแสดงข้อมูลในตารางฐานข้อมูล

```

Private Sub Command5_Click() 'Find
On Error GoTo EL

If Trim(Text2.Text) = "" Then Exit Sub

Adodc1.RecordSource = "select * from student where ชื่อเล่น" & Trim(Text2.Text) & ""
Adodc1.Refresh

If Not Adodc1.Recordset.EOF Then

Text1.Text = Adodc1.Recordset.Fields(0)
Text2.Text = Adodc1.Recordset.Fields(1)
Text3.Text = Adodc1.Recordset.Fields(2)
Text4.Text = Adodc1.Recordset.Fields(3)
Text5.Text = Adodc1.Recordset.Fields(4)
Text6.Text = Adodc1.Recordset.Fields(5)
Text7.Text = Adodc1.Recordset.Fields(6)
Text8.Text = Adodc1.Recordset.Fields(7)

End If
EL:
End Sub

```

```

Private Sub Command6_Click() 'Left
On Error GoTo EL

Adodc1.Recordset.MovePrevious

If Not Adodc1.Recordset.BOF Then

Text1.Text = Adodc1.Recordset.Fields(0)
Text2.Text = Adodc1.Recordset.Fields(1)
Text3.Text = Adodc1.Recordset.Fields(2)
Text4.Text = Adodc1.Recordset.Fields(3)
Text5.Text = Adodc1.Recordset.Fields(4)
Text6.Text = Adodc1.Recordset.Fields(5)
Text7.Text = Adodc1.Recordset.Fields(6)
Text8.Text = Adodc1.Recordset.Fields(7)

End If
EL:
End Sub

```

```

Private Sub Command7_Click() 'Right
On Error GoTo EL

Adodc1.Recordset.MoveNext

If Not Adodc1.Recordset.EOF Then

Text1.Text = Adodc1.Recordset.Fields(0)
Text2.Text = Adodc1.Recordset.Fields(1)
Text3.Text = Adodc1.Recordset.Fields(2)
Text4.Text = Adodc1.Recordset.Fields(3)
Text5.Text = Adodc1.Recordset.Fields(4)
Text6.Text = Adodc1.Recordset.Fields(5)
Text7.Text = Adodc1.Recordset.Fields(6)
Text8.Text = Adodc1.Recordset.Fields(7)

End If
EL:
End Sub

```

```

Private Sub Command8_Click()
On Error GoTo EL
Text1.Text = ""
Text2.Text = ""
Text3.Text = ""
Text4.Text = ""
Text5.Text = ""
Text6.Text = ""
Text7.Text = ""
Text8.Text = ""

Adodc1.RecordSource = "select * from student"
Adodc1.Refresh

EL:
End Sub

```

รูปที่ ก.5 แสดงโค้ดส่วนที่ 2 ของโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic
ซึ่งแสดงข้อมูลในตารางฐานข้อมูล

โค้ดคำสั่งโปรแกรมเฉพาะที่พัฒนาด้วยภาษา C#

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace WindowsApplication1
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            Form2 f2 = new Form2();
            f2.Show();
            f2.f2textBox6.Text = f1textBox1.Text;
            f2.f2textBox7.Text = f1textBox2.Text;
            f2.f2textBox8.Text = f1textBox3.Text;
            f2.f2textBox9.Text = f1textBox4.Text;
            f2.f2textBox10.Text = f1textBox5.Text;
            f2.f2textBox11.Text = f1textBox6.Text;
            this.Visible = false;
        }

        private void button2_Click(object sender, EventArgs e)
        {
            f1textBox1.Text = "";
            f1textBox2.Text = "";
            f1textBox3.Text = "";
            f1textBox4.Text = "";
            f1textBox5.Text = "";
            f1textBox6.Text = "";
        }

        private void button3_Click(object sender, EventArgs e)
        {
            Close();
        }
    }
}

```

รูปที่ ก.6 แสดงโค้ดฟอร์ม1 ของโปรแกรมเฉพาะที่พัฒนาด้วยภาษา C#

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace WindowsApplication1
{
    public partial class Form2 : Form
    {
        public Form2()
        {
            InitializeComponent();
        }
        private void button1_Click(object sender, EventArgs e)
        {
            Form3 f3 = new Form3();
            f3.Show();
            f3.f3textBox1.Text = f2textBox6.Text;
            f3.f3textBox2.Text = f2textBox7.Text;
            f3.f3textBox3.Text = f2textBox8.Text;
            f3.f3textBox4.Text = f2textBox9.Text;
            f3.f3textBox5.Text = f2textBox10.Text;
            f3.f3textBox6.Text = f2textBox11.Text;
            f3.f3textBox7.Text = f2textBox1.Text;
            f3.f3textBox8.Text = f2textBox2.Text;
            f3.f3textBox9.Text = f2textBox3.Text;
            f3.f3textBox10.Text = f2textBox4.Text;
            f3.f3textBox11.Text = f2textBox5.Text;
            this.Visible = false;
        }
        private void button2_Click(object sender, EventArgs e)
        {
            f2textBox1.Text = "";
            f2textBox2.Text = "";
            f2textBox3.Text = "";
            f2textBox4.Text = "";
            f2textBox5.Text = "";
        }
        private void button3_Click(object sender, EventArgs e)
        {
            Close();
        }
    }
}

```

รูปที่ ก.7 แสดงโค้ดของ Form2 ของโปรแกรมเฉพาะที่พัฒนาด้วยภาษา C#

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace WindowsApplication1
{
    public partial class Form3 : Form
    {
        public Form3()
        {
            InitializeComponent();
        }
        private void button1_Click(object sender, EventArgs e)
        {
            Form1 f1 = new Form1();
            f1.Show();
            this.Visible = false;
        }
        private void button2_Click(object sender, EventArgs e)
        {
            Close();
        }
    }
}
```

รูปที่ ก.8 แสดงโค้ดของ Form3 ของโปรแกรมเฉพาะที่พัฒนาด้วยภาษา C#



โค้ดคำสั่งโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java

```

/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 * FirstApp.java
 * Created on 5 มิ.ย. 2552, 19:49:59
 */
package management;

/**
 *
 * @author chivrunterxhunter
 */
public class FirstApp extends javax.swing.JFrame {
    /** Creates new form FirstApp */
    public FirstApp() {
        initComponents();
        this.setVisible(true) ;
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    @SuppressWarnings({"rawtypes","unchecked"})
    // Editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();
        jLabel1 = new javax.swing.JLabel();
        jTextField1 = new javax.swing.JTextField();
        jLabel2 = new javax.swing.JLabel();
        jTextField2 = new javax.swing.JTextField();
        jLabel3 = new javax.swing.JLabel();
        jTextField3 = new javax.swing.JTextField();
        jLabel4 = new javax.swing.JLabel();
        jTextField4 = new javax.swing.JTextField();
        jLabel5 = new javax.swing.JLabel();
        jTextField5 = new javax.swing.JTextField();
        jLabel6 = new javax.swing.JLabel();
        jTextField6 = new javax.swing.JTextField();
        jPanel2 = new javax.swing.JPanel();
        jLabel7 = new javax.swing.JLabel();
        jTextField7 = new javax.swing.JTextField();
        jLabel8 = new javax.swing.JLabel();
        jTextField8 = new javax.swing.JTextField();
        jLabel9 = new javax.swing.JLabel();
        jTextField9 = new javax.swing.JTextField();
        jPanel3 = new javax.swing.JPanel();
        jLabel10 = new javax.swing.JLabel();
        jTextField10 = new javax.swing.JTextField();
        jLabel11 = new javax.swing.JLabel();
        jTextField11 = new javax.swing.JTextField();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();
        jButton3 = new javax.swing.JButton();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        jPanel1.setBorder(javax.swing.BorderFactory.createTitledBorder(
            javax.swing.BorderFactory.createEtchedBorder(), "Employee Information",
            javax.swing.border.TitledBorder.DEFAULT_JUSTIFICATION,
            javax.swing.border.TitledBorder.DEFAULT_POSITION, new java.awt.Font("Tahoma", 0, 11),
            java.awt.Color.black)); // NOI18N

        jLabel1.setText("Employee ID");
        jLabel2.setText("First Name");
        jLabel3.setText("Sur Name");
        jLabel4.setText("Caveeza");
        jLabel5.setText("Department");
        jLabel6.setText("Branch");
    }
}

```

รูปที่ ก.9 แสดงโค้ดส่วนที่ 1 ของฟอร์ม 1 ในโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java

```

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
jPanel1.setLayout(jPanel1Layout);
jPanel1Layout.setHorizontalGroup(
    jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(33, 33, 33)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel1)
                .addComponent(jLabel2)
                .addComponent(jLabel4)
                .addComponent(jLabel6))
            .addGap(18, 18, 18)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jTextField1, javax.swing.GroupLayout.DEFAULT_SIZE, 174, Short.MAX_VALUE)
                .addComponent(jTextField6, javax.swing.GroupLayout.DEFAULT_SIZE, 174, Short.MAX_VALUE)
                .addComponent(jTextField4, javax.swing.GroupLayout.Alignment.TRAILING,
                    javax.swing.GroupLayout.DEFAULT_SIZE, 174, Short.MAX_VALUE)
                .addComponent(jTextField2, javax.swing.GroupLayout.Alignment.TRAILING,
                    javax.swing.GroupLayout.DEFAULT_SIZE, 174, Short.MAX_VALUE))
            .addGap(46, 46, 46)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel3, javax.swing.GroupLayout.Alignment.TRAILING)
                .addComponent(jLabel5, javax.swing.GroupLayout.Alignment.TRAILING))
            .addGap(15, 15, 15)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                .addComponent(jTextField5)
                .addComponent(jTextField3, javax.swing.GroupLayout.DEFAULT_SIZE, 179, Short.MAX_VALUE))
            .addGap(94, 94, 94))
        );
jPanel1Layout.setVerticalGroup(
    jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(15, 15, 15)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 22,
                    javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE,
                    javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel2)
                .addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE,
                    javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jTextField3, javax.swing.GroupLayout.PREFERRED_SIZE,
                    javax.swing.GroupLayout.PREFERRED_SIZE,
                    javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel3))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel4)
                .addComponent(jTextField5, javax.swing.GroupLayout.PREFERRED_SIZE,
                    javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jTextField4, javax.swing.GroupLayout.PREFERRED_SIZE,
                    javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel6)
                .addComponent(jTextField6, javax.swing.GroupLayout.PREFERRED_SIZE,
                    javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addContainerGap(22, Short.MAX_VALUE))
        );
jPanel2.setBorder(javax.swing.BorderFactory.createTitledBorder(
    javax.swing.BorderFactory.createEtchedBorder(), "Contact Information",
    javax.swing.border.TitledBorder.DEFAULT_JUSTIFICATION,
    javax.swing.border.TitledBorder.DEFAULT_POSITION, new java.awt.Font("Tahoma", 0, 11),
    java.awt.Color.black)); // NOI18N

jLabel7.setText("Address");
jLabel8.setText("Telephone Number");
jLabel9.setText("E - mail");

javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);
jPanel2.setLayout(jPanel2Layout);
jPanel2Layout.setHorizontalGroup(
    jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel2Layout.createSequentialGroup()
            .addGap(33, 33, 33)
            .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(jPanel2Layout.createSequentialGroup()
                    .addGap(18, 18, 18)
                    .addComponent(jTextField7, javax.swing.GroupLayout.PREFERRED_SIZE, 491,
                        javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)

```

รูปที่ ก.10 แสดงโค้ดส่วนที่ 2 ของฟอร์ม 1 ในโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java

```

        .addGroup(javax.swing.GroupLayout.Alignment.LEADING, jPanel2Layout.createSequentialGroup()
            .addComponent(jLabel5)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addComponent(jTextField5))
        .addGroup(javax.swing.GroupLayout.Alignment.LEADING, jPanel2Layout.createSequentialGroup()
            .addComponent(jLabel9)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addComponent(jTextField9, javax.swing.GroupLayout.PREFERRED_SIZE, 256,
                javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addContainerGap(95, Short.MAX_VALUE)
    };
    jPanel2Layout.setVerticalGroup(
        jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel2Layout.createSequentialGroup()
                .addContainerGap()
                .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                    .addComponent(jTextField7, javax.swing.GroupLayout.Alignment.LEADING,
                        javax.swing.GroupLayout.DEFAULT_SIZE, 22, Short.MAX_VALUE)
                    .addComponent(jLabel7, javax.swing.GroupLayout.Alignment.LEADING,
                        javax.swing.GroupLayout.PREFERRED_SIZE, 22, javax.swing.GroupLayout.PREFERRED_SIZE))
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(jLabel6)
                    .addComponent(jTextField6, javax.swing.GroupLayout.PREFERRED_SIZE, 20,
                        javax.swing.GroupLayout.PREFERRED_SIZE))
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(jLabel9)
                    .addComponent(jTextField9, javax.swing.GroupLayout.PREFERRED_SIZE,
                        javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            );

    jPanel3.setBorder(javax.swing.BorderFactory.createTitledBorder(
        javax.swing.BorderFactory.createEtchedBorder(), "Month Conditional Information",
        javax.swing.border.TitledBorder.DEFAULT_JUSTIFICATION,
        javax.swing.border.TitledBorder.DEFAULT_POSITION, new java.awt.Font("Tahoma", 0, 11),
        java.awt.Color.black)); // NOI18N

    jLabel10.setText("Month Conditional Rank");
    jLabel11.setText("Over Time");

    javax.swing.GroupLayout jPanel3Layout = new javax.swing.GroupLayout(jPanel3);
    jPanel3.setLayout(jPanel3Layout);
    jPanel3Layout.setHorizontalGroup(
        jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel3Layout.createSequentialGroup()
                .addGroup(jPanel3Layout.createSequentialGroup().createSequentialGroup()
                    .addGap(32, 32, 32)
                    .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                        .addGroup(jPanel3Layout.createSequentialGroup().createSequentialGroup()
                            .addComponent(jLabel11)
                            .addGap(13, 13, 13)
                            .addComponent(jTextField11, javax.swing.GroupLayout.PREFERRED_SIZE, 105,
                                javax.swing.GroupLayout.PREFERRED_SIZE))
                        .addGroup(jPanel3Layout.createSequentialGroup().createSequentialGroup()
                            .addComponent(jLabel10)
                            .addGap(13, 13, 13)
                            .addComponent(jTextField10, javax.swing.GroupLayout.PREFERRED_SIZE, 105,
                                javax.swing.GroupLayout.PREFERRED_SIZE)))
                    .addContainerGap(413, Short.MAX_VALUE))
            );

    jPanel3Layout.setVerticalGroup(
        jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel3Layout.createSequentialGroup()
                .addGroup(jPanel3Layout.createSequentialGroup().createSequentialGroup()
                    .addContainerGap()
                    .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                        .addComponent(jLabel10)
                        .addComponent(jTextField10, javax.swing.GroupLayout.PREFERRED_SIZE,
                            javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                    .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                        .addComponent(jLabel11)
                        .addComponent(jTextField11, javax.swing.GroupLayout.PREFERRED_SIZE,
                            javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                    .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
            );

```

รูปที่ ก.11 แสดงโค้ดส่วนที่ 3 ของฟอร์ม 1 ในโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java


```

jButton1.setText("OK");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

jButton2.setText("Clear Information");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

jButton3.setText("Exit");
jButton3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton3ActionPerformed(evt);
    }
});

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()
            .addContainerGap(26, true)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                .addComponent(jPanel3, javax.swing.GroupLayout.Alignment.LEADING, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jPanel2, javax.swing.GroupLayout.Alignment.LEADING, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
            .addContainerGap(119, true)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jButton1, javax.swing.GroupLayout.PREFERRED_SIZE, 109, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jButton2)
                .addComponent(jButton3, javax.swing.GroupLayout.PREFERRED_SIZE, 100, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addContainerGap(129, true)
        )
        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addContainerGap(21, true)
                .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jPanel3, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(18, true)
                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(jButton1)
                    .addComponent(jButton2, javax.swing.GroupLayout.PREFERRED_SIZE, 29, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jButton3, javax.swing.GroupLayout.PREFERRED_SIZE, 23, javax.swing.GroupLayout.PREFERRED_SIZE))
                .addContainerGap(15, true)
            )
        )
);
pack();
// </editor-fold> //GEN-END:initComponents

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) //GEN-FIRST:event_jButton3ActionPerformed
    System.exit(0);
//GEN-LAST:event_jButton3ActionPerformed

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) //GEN-FIRST:event_jButton1ActionPerformed
    this.setVisible(false);
    new SecondApp(jTextField1.getText(), jTextField2.getText(), jTextField3.getText(), jTextField4.getText(),
        jTextField5.getText(), jTextField6.getText(), jTextField7.getText(), jTextField8.getText(),
        jTextField9.getText(), jTextField10.getText(), jTextField11.getText());
//GEN-LAST:event_jButton1ActionPerformed

```

รูปที่ ก.12 แสดงโค้ดส่วนที่ 4 ของฟอร์ม 1 ในโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java

```

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton2ActionPerformed
    jTextField1.setText("");
    jTextField2.setText("");
    jTextField3.setText("");
    jTextField4.setText("");
    jTextField5.setText("");
    jTextField6.setText("");
    jTextField7.setText("");
    jTextField8.setText("");
    jTextField9.setText("");
    jTextField10.setText("");
    jTextField11.setText("");
} //GEN-LAST:event_jButton2ActionPerformed

/**
 * Spawns args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new FirstApp();
        }
    });
}

// Variables declaration - do not modify //GEN-BEGIN:variables
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel10;
private javax.swing.JLabel jLabel11;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel6;
private javax.swing.JLabel jLabel7;
private javax.swing.JLabel jLabel8;
private javax.swing.JLabel jLabel9;
private javax.swing.JPanel jPanel1;
private javax.swing.JPanel jPanel2;
private javax.swing.JPanel jPanel3;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField10;
private javax.swing.JTextField jTextField11;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField3;
private javax.swing.JTextField jTextField4;
private javax.swing.JTextField jTextField5;
private javax.swing.JTextField jTextField6;
private javax.swing.JTextField jTextField7;
private javax.swing.JTextField jTextField8;
private javax.swing.JTextField jTextField9;
// End of variables declaration //GEN-END:variables

```

รูปที่ ก.13 แสดงโค้ดส่วนที่ 5 ของฟอร์ม 1 ในโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java

```

/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package management;
import javax.swing.*;

public class SecondApp extends javax.swing.JFrame {
    private String EmID ;
    private String FirstName ;
    private String SurName ;
    private String Career ;
    private String Department ;
    private String Branch ;
    private String Address ;
    private String Telephone ;
    private String EMail ;
    private String Money ;
    private String OverTime ;
    /** Creates new form SecondApp */
    public SecondApp(String ID, String FN, String SN, String CAREER, String DEP, String BR, String ADD, String TEL,
        String MAIL, String Zeny, String OT) {

        initComponents();
        this.setVisible(true);
        this.EmID = ID ;
        this.FirstName = FN ;
        this.SurName = SN ;
        this.Career = CAREER ;
        this.Department = DEP ;
        this.Branch = BR ;
        this.Address = ADD ;
        this.Telephone = TEL ;
        this.EMail = MAIL ;
        this.Money = Zeny ;
        this.OverTime = OT ;

        jTextField1.setText(this.EmID) ;
        jTextField2.setText(this.FirstName) ;
        jTextField3.setText(this.SurName) ;
        jTextField4.setText(this.Career) ;
        jTextField5.setText(this.Department) ;
        jTextField6.setText(this.Branch) ;
        jTextField7.setText(this.Address) ;
        jTextField8.setText(this.Telephone) ;
        jTextField9.setText(this.EMail) ;
        jTextField10.setText(this.Money) ;
        jTextField11.setText(this.OverTime) ;

        /** This method is called from within the constructor to
         * initialize the form.
         * WARNING: Do NOT modify this code. The content of this method is
         * always regenerated by the Form Editor.
         */
        //GEN-BEGIN:initComponents
        private void initComponents() {

            jPanel3 = new javax.swing.JPanel();
            jLabel10 = new javax.swing.JLabel();
            jTextField10 = new javax.swing.JTextField();
            jLabel11 = new javax.swing.JLabel();
            jTextField11 = new javax.swing.JTextField();
            jPanel2 = new javax.swing.JPanel();
            jLabel7 = new javax.swing.JLabel();
            jTextField7 = new javax.swing.JTextField();
            jLabel6 = new javax.swing.JLabel();
            jTextField6 = new javax.swing.JTextField();
            jLabel9 = new javax.swing.JLabel();
            jTextField9 = new javax.swing.JTextField();
            jPanel1 = new javax.swing.JPanel();
            jLabel1 = new javax.swing.JLabel();
            jTextField1 = new javax.swing.JTextField();
            jLabel2 = new javax.swing.JLabel();
            jTextField2 = new javax.swing.JTextField();
            jLabel3 = new javax.swing.JLabel();
            jTextField3 = new javax.swing.JTextField();
            jLabel4 = new javax.swing.JLabel();
            jTextField4 = new javax.swing.JTextField();
            jLabel5 = new javax.swing.JLabel();
            jTextField5 = new javax.swing.JTextField();
            jLabel6 = new javax.swing.JLabel();
            jTextField6 = new javax.swing.JTextField();
            jLabel7 = new javax.swing.JLabel();
            jTextField7 = new javax.swing.JTextField();
            jLabel8 = new javax.swing.JLabel();
            jTextField8 = new javax.swing.JTextField();
            jLabel9 = new javax.swing.JLabel();
            jTextField9 = new javax.swing.JTextField();
            jButton1 = new javax.swing.JButton();
            jButton2 = new javax.swing.JButton();
        }
    }
}

```

รูปที่ ก.14 แสดงโค้ดส่วนที่ 1 ของฟอร์ม 2 ใน โปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java

```

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

jPanel3.setBorder(javax.swing.BorderFactory.createTitledBorder(
    javax.swing.BorderFactory.createEtchedBorder(), "Month Conditional Information",
    javax.swing.border.TitledBorder.DEFAULT_JUSTIFICATION,
    javax.swing.border.TitledBorder.DEFAULT_POSITION, new java.awt.Font("Tahoma", 0, 11),
    java.awt.Color.black)); // NOI18N

jLabel10.setText("Month Conditional Rank");

jLabel11.setText("Over Time");

javax.swing.GroupLayout jPanel3Layout = new javax.swing.GroupLayout(jPanel3);
jPanel3.setLayout(jPanel3Layout);
jPanel3Layout.setHorizontalGroup(
    jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel3Layout.createSequentialGroup()
            .addGap(32, 32, 32)
            .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(jPanel3Layout.createSequentialGroup()
                    .addComponent(jLabel11)
                    .addGap(12, 12, 12)
                    .addComponent(jTextField11, javax.swing.GroupLayout.PREFERRED_SIZE, 105,
                        javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGroup(jPanel3Layout.createSequentialGroup()
                    .addComponent(jLabel10)
                    .addGap(12, 12, 12)
                    .addComponent(jTextField10, javax.swing.GroupLayout.PREFERRED_SIZE, 105,
                        javax.swing.GroupLayout.PREFERRED_SIZE)))
            .addContainerGap(407, Short.MAX_VALUE))
);
jPanel3Layout.setVerticalGroup(
    jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel3Layout.createSequentialGroup()
            .addGap(32, 32, 32)
            .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jTextField10, javax.swing.GroupLayout.PREFERRED_SIZE,
                    javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED))
            .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jTextField11, javax.swing.GroupLayout.PREFERRED_SIZE,
                    javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        );
jPanel2.setBorder(javax.swing.BorderFactory.createTitledBorder(
    javax.swing.BorderFactory.createEtchedBorder(), "Contact Information",
    javax.swing.border.TitledBorder.DEFAULT_JUSTIFICATION,
    javax.swing.border.TitledBorder.DEFAULT_POSITION, new java.awt.Font("Tahoma", 0, 11),
    java.awt.Color.black)); // NOI18N

jLabel7.setText("Address");

jLabel5.setText("Telephone Number");

jLabel9.setText("E - mail");

javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);
jPanel2.setLayout(jPanel2Layout);
jPanel2Layout.setHorizontalGroup(
    jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel2Layout.createSequentialGroup()
            .addGap(33, 33, 33)
            .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(jPanel2Layout.createSequentialGroup()
                    .addComponent(jLabel7)
                    .addGap(12, 12, 12)
                    .addComponent(jTextField7, javax.swing.GroupLayout.PREFERRED_SIZE, 491,
                        javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
                    .addGroup(jPanel2Layout.createSequentialGroup()
                        .addComponent(jLabel5)
                        .addGap(12, 12, 12)
                        .addComponent(jTextField5))
                    .addGroup(jPanel2Layout.createSequentialGroup()
                        .addComponent(jLabel9)
                        .addGap(12, 12, 12)
                        .addComponent(jTextField9, javax.swing.GroupLayout.PREFERRED_SIZE, 255,
                            javax.swing.GroupLayout.PREFERRED_SIZE)))
            .addContainerGap(32, Short.MAX_VALUE))
);
jPanel2Layout.setVerticalGroup(
    jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel2Layout.createSequentialGroup()
            .addGap(32, 32, 32)
            .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                .addComponent(jTextField7, javax.swing.GroupLayout.PREFERRED_SIZE, 22,
                    javax.swing.GroupLayout.PREFERRED_SIZE)

```

รูปที่ ก.15 แสดงโค้ดส่วนที่ 2 ของฟอร์ม 2 ในโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java

```

        .addComponent(jLabel7, javax.swing.GroupLayout.Alignment.LEADING,
            javax.swing.GroupLayout.PREFERRED_SIZE, 22, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jLabel3)
            .addComponent(jTextField5, javax.swing.GroupLayout.PREFERRED_SIZE, 20,
                javax.swing.GroupLayout.PREFERRED_SIZE))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jLabel9)
            .addComponent(jTextField9, javax.swing.GroupLayout.PREFERRED_SIZE,
                javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(19, 19, 19))
    );

jPanel1.setBorder(javax.swing.BorderFactory.createTitledBorder(
    javax.swing.BorderFactory.createEtchedBorder(), "Employee Information",
    javax.swing.border.TitledBorder.DEFAULT_JUSTIFICATION,
    javax.swing.border.TitledBorder.DEFAULT_POSITION, new java.awt.Font("Tahoma", 0, 11),
    java.awt.Color.black)); // NOI18N

jLabel1.setText("Employee ID");
jLabel2.setText("First Name");
jLabel3.setText("Sur Name");
jLabel4.setText("Career");
jLabel5.setText("Department");
jLabel6.setText("Branch");

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
jPanel1.setLayout(jPanel1Layout);
jPanel1Layout.setHorizontalGroup(
    jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(33, 33, 33)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel1)
                .addComponent(jLabel2)
                .addComponent(jLabel4)
                .addComponent(jLabel6))
            .addGap(15, 15, 15)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jTextField1, javax.swing.GroupLayout.DEFAULT_SIZE, 190, Short.MAX_VALUE)
                .addComponent(jTextField6, javax.swing.GroupLayout.DEFAULT_SIZE, 190, Short.MAX_VALUE)
                .addComponent(jTextField5, javax.swing.GroupLayout.Alignment.TRAILING,
                    javax.swing.GroupLayout.DEFAULT_SIZE, 190, Short.MAX_VALUE)
                .addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE, 190,
                    javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(19, 19, 19)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel15)
                .addComponent(jLabel13))
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                .addGroup(jPanel1Layout.createSequentialGroup()
                    .addGap(16, 16, 16)
                    .addComponent(jTextField5, javax.swing.GroupLayout.PREFERRED_SIZE, 179,
                        javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGroup(jPanel1Layout.createSequentialGroup()
                    .addGap(19, 19, 19)
                    .addComponent(jTextField5)))
            .addGap(102, 102, 102))
        );
jPanel1Layout.setVerticalGroup(
    jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(33, 33, 33)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 22,
                    javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE,
                    javax.swing.GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel2)
                .addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE,
                    javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel3)
                .addComponent(jTextField6, javax.swing.GroupLayout.PREFERRED_SIZE,
                    javax.swing.GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel4)
                .addComponent(jTextField4, javax.swing.GroupLayout.PREFERRED_SIZE,
                    javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel5)
            )
        )
    );

```

รูปที่ ก.16 แสดงโค้ดส่วนที่ 3 ของฟอร์ม 2 ในโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java

```

        .addComponent(jTextField5, javax.swing.GroupLayout.PREFERRED_SIZE,
            javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jLabel6)
            .addComponent(jTextField6, javax.swing.GroupLayout.PREFERRED_SIZE,
                javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(22, Short.MAX_VALUE)
    );

    jButton1.setText("Enter new information");
    jButton1.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton1ActionPerformed(evt);
        }
    });

    jButton2.setText("EXIT");
    jButton2.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton2ActionPerformed(evt);
        }
    });

    javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addGap(24, 24, 24)
                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addGroup(layout.createSequentialGroup()
                        .addComponent(jLabel3, javax.swing.GroupLayout.PREFERRED_SIZE,
                            javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addContainerGap())
                    .addGroup(layout.createSequentialGroup()
                        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                            .addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT_SIZE,
                                javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                            .addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT_SIZE,
                                javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
                        .addGap(20, 20, 20)))
                .addGroup(layout.createSequentialGroup()
                    .addGap(160, 160, 160)
                    .addComponent(jButton1)
                    .addGap(155, 155, 155)
                    .addComponent(jButton2, javax.swing.GroupLayout.PREFERRED_SIZE, 135,
                        javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addContainerGap(140, Short.MAX_VALUE))
            );
    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addGap(27, 27, 27)
                .addComponent(jLabel12, javax.swing.GroupLayout.PREFERRED_SIZE,
                    javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(33, 33, 33)
                .addComponent(jLabel13, javax.swing.GroupLayout.PREFERRED_SIZE,
                    javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(18, 18, 18)
                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(jButton2)
                    .addComponent(jButton1))
                .addContainerGap(17, Short.MAX_VALUE)
            );

    pack();
} // </editor-fold> //GEN-END:InitComponents

```

รูปที่ ก.17 แสดงโค้ดส่วนที่ 4 ของฟอร์ม 2 ในโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {GEN-FIRST:event_jButton1ActionPerformed
    this.setVisible(false);
    new FirstApp();
} //GEN-LAST:event_jButton1ActionPerformed

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {GEN-FIRST:event_jButton2ActionPerformed
    System.exit(0);
} //GEN-LAST:event_jButton2ActionPerformed

// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel10;
private javax.swing.JLabel jLabel11;
private javax.swing.JLabel jLabel12;
private javax.swing.JLabel jLabel13;
private javax.swing.JLabel jLabel14;
private javax.swing.JLabel jLabel15;
private javax.swing.JLabel jLabel16;
private javax.swing.JLabel jLabel17;
private javax.swing.JLabel jLabel18;
private javax.swing.JLabel jLabel19;
private javax.swing.JPanel jPanel1;
private javax.swing.JPanel jPanel2;
private javax.swing.JPanel jPanel3;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField10;
private javax.swing.JTextField jTextField11;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField3;
private javax.swing.JTextField jTextField4;
private javax.swing.JTextField jTextField5;
private javax.swing.JTextField jTextField6;
private javax.swing.JTextField jTextField7;
private javax.swing.JTextField jTextField8;
private javax.swing.JTextField jTextField9;
// End of variables declaration//GEN-END:variables

```

รูปที่ ก.18 แสดงโค้ดส่วนที่ 5 ของฟอร์ม 2 ในโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Java


```

46
47      WinWait( "Data2" )
48      ControlSend( "Data2", "{", 9, $col[8] )
49      ControlSend( "Data2", ":", 8, $col[9] )
50      ControlSend( "Data2", ":", 7, $col[10] )
51      ControlSend( "Data2", ":", 6, $col[11] )
52      ControlSend( "Data2", ":", 5, $col[12] )
53
54      Sleep(1000)
55      ControlClick( "Data2", ":", 3 )
56
57      WinWait( "Display" )
58      Sleep(3000)
59      ControlClick( "Display", ":", 2 )
60   Else
61       MsgBox(4096, "", "Data not match, Press like skipd... ", 5)
62   EndIf
63 WEnd
64 FileClose($file)
65
66 Exit

```

รูปที่ ข.2 แสดง โค้ดส่วนที่ 2 ของโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด
สำหรับโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในฟอร์มสุดท้าย



โค้ดคำสั่งโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในตารางฐานข้อมูล

```

1  $file = FileOpenDialog( Select File to process...", "C:\Users\11999-11999-26A8-0900561B11081",
2  "Text File (*.txt)|All files (*.*)")
3
4  If &error Then
5      MsgBox(4096, "No file(s) selected or cancelled.", 5)
6      Exit
7  EndIf
8
9  $line_no = 1
10
11 While True
12     $line = FileReadLine($file,$line_no)
13     If &error Then
14         If &error = -1 Then
15             MsgBox(4096, "End of file, operation completed.", 5)
16             ExitLoop
17         ElseIf &error = 1 Then
18             MsgBox(4096, "Error while reading, skip this line.", 5)
19             ContinueLoop
20         EndIf
21     EndIf
22
23     $line_no = $line_no + 1
24     $col = StringSplit($line, ",")
25
26     If $col(0) = 0 Then
27         WinWait( "Student File", , [CLASSNAME=&HwndFrom: DIALOG] )
28         ControlSend( "Student File", , [CLASSNAME=&HwndFrom: DIALOG] ", $col[1])
29         ControlSend( "Student File", , [CLASSNAME=&HwndFrom: DIALOG] ", $col[2])
30         ControlSend( "Student File", , [CLASSNAME=&HwndFrom: DIALOG] ", $col[3])
31         ControlSend( "Student File", , [CLASSNAME=&HwndFrom: DIALOG] ", $col[4])
32         ControlSend( "Student File", , [CLASSNAME=&HwndFrom: DIALOG] ", $col[5])
33         ControlSend( "Student File", , [CLASSNAME=&HwndFrom: DIALOG] ", $col[6])
34         ControlSend( "Student File", , [CLASSNAME=&HwndFrom: DIALOG] ", $col[7])
35         ControlSend( "Student File", , [CLASSNAME=&HwndFrom: DIALOG] ", $col[8])
36
37         Sleep(1000)
38         ControlClick( "Student File", , [CLASSNAME=&HwndFrom: DIALOG] "Button1")
39
40         WinWait( "Student File", , [CLASSNAME=&HwndFrom: DIALOG] )
41         Sleep(500)
42         ControlClick( "Student File", , [CLASSNAME=&HwndFrom: DIALOG] "Button1")
43
44         WinWait( "Student File", , [CLASSNAME=&HwndFrom: DIALOG] )
45         Sleep(1000)
46         ControlClick( "Student File", , [CLASSNAME=&HwndFrom: DIALOG] "Button1")
47
48     Else
49         MsgBox(4096, "Data not open, this line skipped.", 5)
50     EndIf
51 WEnd
52 FileClose($file)
53
54 Exit

```

รูปที่ ข.3 แสดง โค้ด โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับโปรแกรมเฉพาะที่พัฒนาด้วยภาษา Visual Basic ที่แสดงผลข้อมูลในตารางฐานข้อมูล

โค้ดคำสั่งโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับโปรแกรมเฉพาะที่ใช้กับโปรแกรมร้านหนังสือเช่า

```

1  $file = FileOpenDialog( "Select file to process.", "C:\Program Files\BookRent\BookRent.exe", "All files (*.*)", "Open" )
2  $objFile = File($file)
3
4  If $error Then
5      MsgBox(4096, "Error", "File(s) does not exist or is not a file.")
6      Exit
7  EndIf
8
9  $line_no = 1
10
11 While True
12     $line = FileReadLine($file,$line_no)
13     If $error Then
14         If $error = -1 Then
15             MsgBox(4096, "End of file", "End of file reached.", 5)
16             ExitLoop
17         ElseIf $error = 1 Then
18             MsgBox(4096, "Stream file reading", "Stream file reading failed.", 5)
19             ContinueLoop
20         EndIf
21     EndIf
22
23     $line_no = $line_no + 1
24     $col = StringSplit($line, ",")
25
26     If $col[0] = 5 Then
27
28         WinWait( "BookRent.exe" )
29         ControlSend( "BookRent.exe", "{TAB}" )
30         ControlSend( "BookRent.exe", "{TAB}" )
31         ControlSend( "BookRent.exe", "{TAB}" )
32         ControlSend( "BookRent.exe", "{TAB}" )
33         Sleep(1000)
34         ControlClick( "BookRent.exe", "BookRent.exe", "BookRent.exe", "BookRent.exe" )
35
36         WinWait( "BookRent.exe" )
37         ControlSend( "BookRent.exe", "{TAB}" )
38         ControlClick( "BookRent.exe", "BookRent.exe", "BookRent.exe", "BookRent.exe" )
39         Sleep(500)
40         ControlClick( "BookRent.exe", "BookRent.exe", "BookRent.exe", "BookRent.exe" )
41
42         WinWait( "BookRent.exe" )
43         Sleep(500)
44         ControlClick( "BookRent.exe", "BookRent.exe", "BookRent.exe", "BookRent.exe" )
45
46     Else
47         MsgBox(4096, "Data not match", "Data not match.", 5)
48     EndIf
49 WEnd
50 FileClose($file)
51
52 Exit

```

รูปที่ ข.5 แสดง โค้ด โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด
สำหรับใช้กับข้อมูลสมาชิก โปรแกรมร้านเช่าหนังสือ

โค้ดคำสั่งโปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ดสำหรับโปรแกรมเฉพาะที่ใช้กับโปรแกรม MyRestuarant

```

1  $file = FileOpenDialog( Subject:file to process..., "C:\Program Files\MyRestuarant\MyRestuarant.exe",
2  "Text File (*.txt)|Excel File (*.xls)|",
3
4  If @error Then
5      MsgBox(4096, "No file(s) chosen or operation cancelled!")
6      Exit
7  EndIf
8
9  $line_no = 1
10
11 While True
12     $line = FileReadLine($file,$line_no)
13     If @error Then
14         If @error = -1 Then
15             MsgBox(4096, "End of file, operation completed.", 5)
16             ExitLoop
17         ElseIf @error = 1 Then
18             MsgBox(4096, "Error while reading, skip this line...", 5)
19             ContinueLoop
20         EndIf
21     EndIf
22
23     $line_no = $line_no + 1
24     $col = StringSplit($line, ",")
25
26     If $col[0] = 6 Then
27         WinWait("MyRestuarant")
28         ControlSend("MyRestuarant", "1", [CLASS:msctls.Button.1; INSTANCE:16], $col[1])
29         ControlSend("MyRestuarant", "2", [CLASS:msctls.Button.1; INSTANCE:15], $col[2])
30         ControlSend("MyRestuarant", "3", [CLASS:msctls.Button.1; INSTANCE:4], $col[3])
31         ControlSend("MyRestuarant", "4", [CLASS:msctls.Button.1; INSTANCE:3], $col[4])
32         ControlSend("MyRestuarant", "5", [CLASS:msctls.Button.1; INSTANCE:2], $col[5])
33         ControlSend("MyRestuarant", "6", [CLASS:msctls.Button.1; INSTANCE:1], $col[6])
34
35         Sleep(1000)
36         ControlClick("MyRestuarant", "1", [CLASS:msctls.Button.1; INSTANCE:3])
37     Else
38         MsgBox(4096, "Bad data entered, skip this line...", 5)
39     EndIf
40 WEnd
41 FileClose($file)
42 Exit

```

รูปที่ ข.6 แสดง โค้ด โปรแกรมแปลงไฟล์ตัวอักษรเป็นสัญญาณอินพุทของคีย์บอร์ด
สำหรับใช้กับข้อมูลพนักงาน โปรแกรม MyRestuarant

ประวัติผู้เขียนโครงการ



ชื่อ นางสาวจิวรัตน์ นุ่นมีศรี
 ภูมิลำเนา 169 หมู่ 10 ตำบลสายลำโพง อำเภอท่าตะโก
 จังหวัดนครสวรรค์

ประวัติการศึกษา

- จบมัธยมศึกษาจาก โรงเรียนพนมรอกวิทยา จังหวัดนครสวรรค์
- ปัจจุบันกำลังศึกษาอยู่ระดับปริญญาตรีชั้นปีที่ 4

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
 มหาวิทยาลัยนเรศวร

E-mail : plaboooo@hotmail.com



ชื่อ นางสาววิจิตรา มหาศ
 ภูมิลำเนา 14 หมู่ 4 ตำบลคงสุวรรณ อำเภอดอกคำใต้ จังหวัดพะเยา
 ประวัติการศึกษา

- จบมัธยมศึกษาจาก โรงเรียนจำเมืองวิทยาคม จังหวัดพะเยา
- ปัจจุบันกำลังศึกษาอยู่ระดับปริญญาตรีชั้นปีที่ 4

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
 มหาวิทยาลัยนเรศวร

E-mail : yoo-yui@hotmail.com