

ปริญญานิพนธ์

โปรแกรมช่วยสอนคอส
DOS TRAINING PROGRAM



นายประยงค์ สุภศิริโรจน์
นายประสิทธิ์ แฉ่ตั้ง
นายอำพล ทองระอ่า



เลขหมู่.....	7
เลขทะเบียน.....	1304
วัน เดือน ปี.....	-3 W.F.I. 2538

021072

ปริญญานิพนธ์ฉบับนี้ เป็นส่วนหนึ่งของการศึกษาคามหลักสูตรครุศาสตร์อุตสาหกรรมบัณฑิต
สาขาวิชาอิเล็กทรอนิกส์และคอมพิวเตอร์
ภาควิชาครุศาสตร์วิศวกรรม
คณะครุศาสตร์อุตสาหกรรม
สถาบัน เทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2537

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาควิชาครุศาสตร์ศึกษาศาสตร์

คณะครุศาสตร์อุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญาโท

ชื่อหัวข้อปริญญาโท โปรแกรมช่วยสอนคอมพิวเตอร์

DOS TRAINING PROGRAM

ชื่อนักศึกษา

1. นายประสงค์ ศักดิ์ศิริโรจน์ 36031316
2. นายประสิทธิ์ แซ่ตั้ง 36031317
3. นายอำพล ทองระอา 36031339

อาจารย์ผู้ควบคุมปริญญาโท

1. อาจารย์กิติพงศ์ มะโน
2. อาจารย์วริทธิ์ สมหา
3. อาจารย์สุชิน อาจหาญ

คณะกรรมการสอบปริญญาโท	ลายมือ
1. อาจารย์กิติพงศ์ มะโน	
2. อาจารย์วริทธิ์ อธิพรธรรม	
3. อาจารย์สันติ ตันตระกูล	
4. อาจารย์สมชาย หมั่นสายญาติ	
5. อาจารย์สุชิน อาจหาญ	

วัน/เดือน/ปี ที่สอบ วันที่ 25 ธันวาคม พ.ศ.2537 เวลา 11.00 น. - 12.30 น.

สถานที่สอบ ห้อง ค.301 คณะครุศาสตร์อุตสาหกรรม



ใบรับรองแล้ว

ลงนามในใบรับรองนี้โดย (ในชื่อ) ศาสตราจารย์ ดร. วชิรศักดิ์ วัฒนสิน ๗ อุตสาหกรรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่ควรเผยแพร่หรือใช้เพื่อวัตถุประสงค์อื่นใดโดยไม่ได้รับอนุญาตจากภาควิชาครุศาสตร์ศึกษาศาสตร์

ปฏิญานิตย

เรื่อง โปรแกรมช้สอนคอส

DOS TRAINING PROGRAM

ผู้จัดทำ

1. นายประยงค์ ศกักรติโรจน 36031316
2. นายประสิทธิ์ แซ่ตั้ง 36031317
3. นายอำพล ทองระอา 36031339

อาจารย์ที่ปรึกษา

ลงนาม

(อ. กิตติพงศ์ มะโน)

ลงนาม

(อ. วรวิทย์ สมทา)

ลงนาม

(อ. สชิน อางหาญ)

รับรอง

ลงนาม

(ผศ. ดร. ชีระพล เทพหัสดิน ณ อยุธยา)

หัวหน้าภาควิชาครศาสตร์วิศวกรรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ปีการศึกษา 2537
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญาานิพนธ์

โปรแกรมช่วยสอนคอส

จุดประสงค์

1. เพื่อศึกษาการใช้งานคำสั่งต่าง ๆ ของคอส และเขียนโปรแกรมภาษาซี
2. เพื่อศึกษาการแสดงผล และโปรแกรมทางด้านกราฟฟิกของภาษาซี
3. เพื่อออกแบบโปรแกรมสื่อการเรียนการสอนคอส
4. เพื่อเขียนโปรแกรมสื่อการเรียนการสอนคอส
5. เพื่อนำโปรแกรมไปใช้ในการเรียนการสอนจริงได้อย่างมีประสิทธิภาพ

ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถใช้คำสั่งต่าง ๆ ของคอส ได้อย่างมีประสิทธิภาพ
2. ได้รับความรู้ความเข้าใจในการเขียนโปรแกรมกราฟฟิกของภาษาซี
3. เป็นโปรแกรมสื่อการเรียนการสอนได้
4. นำโปรแกรมที่เขียนขึ้นไปใช้ในการเรียนการสอนจริง ๆ ได้
5. ประหยัดงบประมาณในการทำสื่อการเรียนการสอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมช่วยสอนคอส

นายประสงค์ ศุภเกียรติโรจน์
นายประสิทธิ์ แซ่ตั้ง
นายอำพล ทองระอา

อาจารย์ที่ปรึกษา

อ. กิตติพงศ์ มะโน
อ. วรวิทย์ สมหา
อ. สุชิน อาจหาญ
ปีการศึกษา 2537

บทคัดย่อ

ปฏิญานพนธ์ฉบับนี้เป็นการนำเสนอโปรแกรมช่วยสอนคอส โปรแกรมนี้เหมาะกับกลุ่มบุคคลที่เริ่มต้นศึกษาเกี่ยวกับระบบปฏิบัติการคอส หรือผู้ที่มีความรู้อยู่บ้างแล้วให้มีความเข้าใจลึกซึ้งมากยิ่งขึ้น

การศึกษาเกี่ยวกับระบบปฏิบัติการคอสด้วยตัวเองนั้น อาจจะเกิดปัญหาต่าง ๆ ได้ แต่ปัญหาเหล่านั้นสามารถแก้ไขได้ด้วย โปรแกรมช่วยสอนคอส ซึ่งการเขียนโปรแกรมช่วยสอนคอสนี้จะใช้ภาษา ซี ในโหมดกราฟนิคเขียนทั้งหมดซึ่งจะจำลองหลักการทำงานคำสั่งของคอสและจะแสดงหลักการทำงานของคำสั่งเหล่านั้นเมื่อมีการเรียกใช้หรือสามารถเรียกคู่มือการใช้คำสั่งต่าง ๆ ทั้งหมดของโปรแกรม โดยจะมีการแสดงผลเป็นภาษาไทย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DOS TRAINING PROGRAM

MR. PRAYONG SUPAKESATIROT

MR. PRASIT SEATANG

MR. AMPHON TONGRAHA

ADVISOR

MR. KITIPONG MANO

MR. WORAWIT SOMHA

Mr. SUCHIN HATHAN

1994

ABSTRACT

THIS THESIS IS TO SHOW TO OPERATE DOS. IT COULD BEUTILIZED TO THE DISK OPERATING SYSTEM BEGINER AND ALSO THEN OVICE TO BE ABLE TO GET MORE UNDERSTANDING.

THE DOS'S STUDYING COULD FILD SEVERAL PROBLEMS. BUTDY THE DOS: TRAINING PROGRAM, THOSE KINDS OF PROBLEM CAN BESOLVED. THE PROGRAM WAS WRITEN IN C-LANGUAGE, GRAPHICS MODE. IT WILL SIMULATE THE DOS COMMANDS AND SHOW WHEN YOU USE THAT COMMAND.YOU CAN ALSO LIST OUT ALL COMMAND'S INTRUCTION THOSE WILL SHOW YOU IN THAI

กิติกรรมประกาศ

การจัดทำวิทยานิพนธ์นี้ สามารถสำเร็จลุล่วงไปได้ด้วยดีจากการร่วมมือร่วมใจกันของสมาชิกในกลุ่ม และยังได้รับความกรุณาจากอาจารย์ที่ปรึกษา อาจารย์ภาควิชาครุศาสตร์วิศวกรรมทุกท่านที่ ให้คำแนะนำค่าปรึกษา และให้ความช่วยเหลือในด้านต่าง ๆ เป็นอย่างดี ตลอดจนเพื่อน ๆ ที่ให้ ให้กำลังใจมาตลอด จึงขอขอบคุณไว้ ณ ที่นี้ด้วย



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทที่ 1 กล่าวนำ	1
บทที่ 2 ทฤษฎี และหลักการ	3
2.1 ความหมายของดอส	3
2.2 ความเป็นมาของดอส	4
2.3 ส่วนประกอบของดอส	6
2.3.1 รอม ไบออส	6
2.3.2 บุต แร็กคอร์ด	6
2.3.3 IBMIO.COM และ IBMDOS.COM	6
2.3.4 COMMAND.COM	7
2.4 คำสั่งและการใช้งานดอส	8
2.4.1 ดอส และงานของ ดอส	8
2.4.2 คำสั่งภายในและคำสั่งภายนอก	8
2.4.3 คำสั่งต่าง ๆ ที่ใช้งานของดอส	8
2.4.4 คีย์สำคัญต่าง ๆ ในภาจควบคุมเอ็มเอส ดอส	24
2.5 การแสดงผลภาษาไทยในกราฟิคโหมด ด้วยเทอร์โบซี	26
2.5.1 โครงสร้างของการ์ด VGA	29
2.5.2 หน่วยความจำแสดงผล	31
2.5.3 หน่วยความจำในเท็กซ์โหมด	31
2.5.4 หน่วยความจำในกราฟิคโหมด	31
2.5.5 ชิพ VGA	32
2.5.6 CRT controller	32
2.5.7 Graphics Controller	35
2.5.8 Sequencer	38
2.5.9 Attribute Controller	39
2.5.10 VGA VIDEO DAC	42

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3 การสร้างและการออกแบบ	43
3.1 โปรแกรมคอมพิวเตอร์ภาษาซี	43
3.1.1 การจัดระดับของภาษา	47
3.1.2 การโปรแกรมคอมพิวเตอร์ภาษาซีด้วยเทอร์โบซี	48
3.1.3 ขั้นตอนการทำโปรแกรมโดยใช้เทอร์โบซี	48
3.1.4 การเข้าสู่เทอร์โบซี	50
3.1.5 การเข้าสู่คำสั่งต่าง ๆ ในเทอร์โบซี	52
3.1.6 การสร้างโปรแกรมที่เขียนขึ้น	52
3.1.7 คำสั่ง Edit	52
3.1.8 การใช้คำสั่งในคำสั่ง File	54
3.1.9 การเก็บและเพิ่มข้อมูลลงบนแผ่นบันทึก	55
3.1.10 การใช้คำสั่ง Edit	55
3.1.11 การสั่งให้โปรแกรมปฏิบัติงาน	57
3.2 การใช้งานเทอร์โบซีในกราฟิคโหมด	59
3.2.1 คำสั่งในกราฟิคโหมด	60
3.2.2 คำสั่งในการจัดการเพิ่มข้อมูล	70
3.2.3 การประมวลผลระดับตัวอักษร	73
3.2.3 การประมวลผลระดับข้อความ	74
3.2.5 การประมวลผลระดับระเบียน	76
3.2.6 คำสั่งในการควบคุมทิศทางโปรแกรม	83
3.3 ค้นหาหาข้อมูล	92
3.3.1 โหมดการแสดงผลมาตรฐานของ EGA/VGA	92
3.3.2 การแสดงผลโหมด 10H	95
3.4 การเขียนโปรแกรมการผลภาษาไทยด้วยเทอร์โบซี	96
3.4.1 การจัดระดับตัวอักษร	98
3.5 ชุดคำสั่ง OpenFont()	100
3.6 ชุดคำสั่ง G_Write()	100
3.7 ชุดคำสั่ง DispChar()	103

เรื่อง

หน้า

3.8	เขียนโปรแกรมแสดงส่วนให้ความช่วยเหลือ	108
3.9	การสร้างเท็กซ์ไฟล์	114
3.10	โปรแกรมเช็คคีย์	116
3.11	โปรแกรมแสดงผลคำสั่ง	121
3.11.1	ส่วนตรวจสอบคำสั่ง	122
3.11.2	ส่วนการจัดรูปแบบการแสดงผล	122
3.11.3	ส่วนที่เป็นข้อความ	123
3.12	การเขียนโปรแกรมหลักในการใช้บทเรียน	124
บทที่ 4	การใช้ และการทดสอบการแสดงผลของโปรแกรมช่วยสอนคอมพิวเตอร์	126
4.1	ส่วนของคำอธิบายการใช้งานโปรแกรม	127
4.2	ส่วนของบทเรียน	128
4.3	การกลับสู่คอมพิวเตอร์	134
4.4	การใช้ส่วนความช่วยเหลือ	135
บทที่ 5	บทสรุปและวิจารณ์	138
5.1	สรุป	138
5.2	ปัญหาที่พบในการทำโครงงาน	138
5.3	การแก้ปัญหา	139
5.4	ประโยชน์ที่ได้รับจากการทำโครงงาน	139
5.5	แนวทางการพัฒนา	139
ภาคผนวก		140
	ภาคผนวกที่ ก. การใช้โปรแกรม และอธิบายคีย์ต่าง ๆ	141
	ภาคผนวกที่ ข. ส่วนของโปรแกรม	143

บรรณานุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
รูปที่ 2.1	30
รูปที่ 2.2	32
รูปที่ 2.3	33
รูปที่ 2.4	37
รูปที่ 2.5	37
รูปที่ 2.6	39
รูปที่ 2.7(ก)	41
รูปที่ 2.7(ข)	42
รูปที่ 3.1	46
รูปที่ 3.2	49
รูปที่ 3.3	49
รูปที่ 3.4	51
รูปที่ 3.5	53
รูปที่ 3.6	54
รูปที่ 3.7	58
รูปที่ 3.8	88
รูปที่ 3.9	88
รูปที่ 3.11	96
รูปที่ 3.12	98
รูปที่ 3.13(ก)	99
รูปที่ 3.13(ข)	99
รูปที่ 3.14	107
รูปที่ 3.15	108
รูปที่ 4.1	126
รูปที่ 4.2	127
รูปที่ 4.3	128

รูปที่	หน้า
รูปที่ 4.4 แสดงค่าอธิบายของหัวข้อที่เลือกเขียน	129
รูปที่ 4.5 ผลของการใช้คำสั่ง DIR	129
รูปที่ 4.6 ผลของการใช้คำสั่ง Del *.txt	130
รูปที่ 4.7 ผลของการใช้คำสั่ง tree /f	130
รูปที่ 4.8 เมนูจากการกด F10	131
รูปที่ 4.9 ผลของการคลิกสู่เมนูหลัก	132
รูปที่ 4.10 ผลของการกลับไปยังบทเรียนที่แล้ว	132
รูปที่ 4.11 ผลของการข้ามไปยังบทเรียนบทต่อไป	133
รูปที่ 4.12 ผลของการออกจากโปรแกรม	133
รูปที่ 4.13 ผลของการออกจากเมนูโดยใช้ ESC	134
รูปที่ 4.14 ผลของการคลิกสู่ คอส	134
รูปที่ 4.15 รายละเอียดของส่วนขอความช่วยเหลือ	135
รูปที่ 4.16 รูปแบบของส่วนการใช้งานโปรแกรม	136
รูปที่ 4.17 รูปแบบของส่วนการใช้หมาสเทศ	136
รูปที่ 4.18 รูปแบบของส่วนการใช้ของตัวอย่าง	137

สารบัญตาราง

ตารางที่	หน้า
ตารางที่ 2.1 ตารางอักษร ส.ม.อ.	27
ตารางที่ 2.2 ตารางอักษร เกษตร	28
ตารางที่ 2.3 แสดงรีจิสเตอร์ต่าง ๆ ใน CRT CONTROLLER	34
ตารางที่ 2.4 รีจิสเตอร์ในกราฟฟิกคอนโทรลเลอร์	36
ตารางที่ 2.5 ตารางแสดงฟังก์ชันลอจิก	36
ตารางที่ 2.6 แสดงรีจิสเตอร์ใน SEQUENCE	38
ตารางที่ 2.7 แสดงรีจิสเตอร์ในแอสทรีบิวต์คอนโทรลเลอร์	41
ตารางที่ 2.8 แสดงการแปลงข้อมูลของ DAC	43
ตารางที่ 3.1 การจัดระดับของภาษา	48
ตารางที่ 3.2 ความละเอียดของจอภาพคอมพิวเตอร์	59
ตารางที่ 3.3 ชื่อแมคโคร และค่าคงที่ในการเติม	65
ตารางที่ 3.4 ขั้นตอนการเปิดและปิดแฟ้มข้อมูล	70
ตารางที่ 3.5 โหมดต่าง ๆ ของการจัดการแฟ้มข้อมูล	72
ตารางที่ 3.6 ค่าคงที่ที่ใช้งานกับอุปกรณ์ Input / Output	78
ตารางที่ 3.7 ค่าแมคโคร ค่าต่าง ๆ	82
ตารางที่ 3.8 โหมดแสดงผลมาตรฐานของระบบวีดีโอไอพีเอ็ม	94
ตารางที่ 3.9 สี่มาตรฐานในกราฟิค 16 สี	96

บทที่ 1

บทนำ

ในปัจจุบันความก้าวหน้าทางเทคโนโลยีได้ก้าวไกลไปมาก คอมพิวเตอร์ก็เป็นส่วนหนึ่งของความก้าวหน้าทางเทคโนโลยี เราจะเห็นว่าไม่ว่าจะทำอะไรหรืองานอะไรก็จะมีคอมพิวเตอร์เข้าไปเกี่ยวข้องเกือบทุกอย่าง สำหรับคอมพิวเตอร์ถ้าจะว่าไปแล้วก็เป็นเพียงแค่อุปกรณ์ชิ้นหนึ่งเท่านั้น ถ้าหากขาดสิ่งที่เราเรียกว่าระบบปฏิบัติการ และระบบปฏิบัติการนี้เองที่อาจจะเรียกได้ว่าเป็นผู้ที่ทำให้มนุษย์ติดต่อกับคอมพิวเตอร์ได้ และสั่งให้คอมพิวเตอร์ทำงานต่าง ๆ ตามที่ต้องการ ระบบปฏิบัติการที่ใช้กับเครื่อง IBM คอมแพคทิเบิลจะมีอยู่หลายแบบที่ใช้กันคือ ดอส, OS/2 และ ยูนิกซ์ แต่ที่นิยมใช้กันในเครื่อง พีซี คือดอส ในการศึกษาคอมพิวเตอร์สำหรับผู้เริ่มต้นจึงต้องทำความเข้าใจกับ ดอส ก่อนซึ่งในการศึกษา ดอส อาจจะมีปัญหาเกิดขึ้นหลายประการเช่น

1. การศึกษาใช้เวลามากเกินไป
2. การศึกษาค้นคว้าตนเองจะเข้าใจได้ยาก
3. ต้องใช้บุคคลากรที่มีความรู้เกี่ยวกับ ดอส คอยตอบปัญหา
4. มองไม่เห็นต่างๆ ของแต่ละคำสั่งอย่างชัดเจน
5. เกิดความเบื่อหน่ายในการศึกษา
6. ไม่มีโอกาสได้ใช้ความคิด ลองผิดลองถูกในการใช้คำสั่งต่าง ๆ ของดอส

ปัญหาต่าง ๆ ที่เกิดขึ้นจึงอาจแก้ไขได้ด้วยการใช้ ซอฟต์แวร์ ในการให้ความรู้เรื่อง ดอส ด้วยเหตุนี้จึงมีความคิดที่จะเขียนโปรแกรมช่วยสอน ดอส ขึ้นเพื่อให้ผู้ที่ต้องการศึกษาเกี่ยวกับระบบปฏิบัติการดอส มีความเข้าใจง่ายขึ้นโปรแกรมนี้มีประโยชน์สำหรับกลุ่มหรือบุคคลที่ต้องการที่จะศึกษาเรื่อง ดอส และยังช่วยให้ผู้ที่มีความรู้เรื่อง ดอส มาบ้างแล้วได้มีความแจ่มแจ้งมากขึ้นจึงหวังว่าปฏิญานี้ฉบับนี้คงจะเป็นประโยชน์ในการศึกษาทางด้านคอมพิวเตอร์มากยิ่งขึ้น

เนื้อหาในปฏิญานี้ฉบับนี้

บทที่ 1 บทนำ

กล่าวถึงการพัฒนาของระบบปฏิบัติการของดอส ความเป็นมาของโปรแกรมช่วยสอนดอส เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษเท่านั้น ไม่นิยญาติให้นำไปใช้ประโยชน์ด้านการค้า มีความสามารถของโปรแกรมช่วยสอนดอส ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุเปลี่ยนแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2 ทฤษฎีและหลักการ

ความหมายของดอส ความเป็นมาของดอส ส่วนประกอบของดอส คำสั่ง และการใช้งาน การแสดงผลภาษาไทยในกราฟฟิคโหมด

บทที่ 3 การสร้างและการออกแบบ

โปรแกรมคอมไพเลอร์ภาษาซี การใช้งานเทอร์โบซีในกราฟฟิคโหมด เขียนโปรแกรมแสดงผลภาษาไทยในโหมดกราฟฟิคโหมดด้วยเทอร์โบซี เขียนโปรแกรมแสดงส่วนให้ความช่วยเหลือ สร้างเท็กซ์ไฟล์ (text file) ที่ใช้แสดงข้อความช่วยเหลือ เขียนโปรแกรมเพื่อรับคำสั่งจากคีย์บอร์ด เขียนโปรแกรมจำลองการแสดงผลคำสั่งต่าง ๆ ของ ดอส

บทที่ 4 การทดลองและการใช้งาน

การใช้ และการทดสอบการแสดงผลของโปรแกรมช่วยสอนดอส

บทที่ 5 บทสรุปและวิจารณ์

สรุป ปัญหาที่พบในการทำโครงงาน การแก้ปัญหา ประโยชน์ที่ได้รับจากการทำโครงงาน แนวทางการพัฒนาโครงงาน

บทที่ 2

ทฤษฎีและหลักการ

โครงการนี้เป็นโครงการที่จัดทำขึ้นใหม่ซึ่งมีทฤษฎีที่ต้องศึกษา และทำความเข้าใจดังนี้

- 1) ความหมายของคอมพิวเตอร์
- 2) ความเป็นมาของคอมพิวเตอร์
- 3) ส่วนประกอบของคอมพิวเตอร์
- 4) คำสั่ง และการใช้งาน
- 5) การแสดงผลภาษาไทยในกราฟฟิคโหมด

2.1 ความหมายของคอมพิวเตอร์

เมื่อพูดถึงเครื่องคอมพิวเตอร์ สิ่งที่มีคนนึกถึงก็คือ ตัวเครื่อง, คีย์บอร์ด, ดิสก์ ฯลฯ ซึ่งเป็นของที่เห็นและสัมผัสได้แน่นอนเรียกว่าเป็นฮาร์ดแวร์ แต่ถ้ามีเพียงเท่านั้น คอมพิวเตอร์ก็ไม่สามารถทำงานได้ จึงจำเป็นต้องมีส่วนที่ควบคุมเครื่องให้ทำงานต่าง ๆ ตามที่เราต้องการได้ซึ่งเรียกว่าเป็นซอฟต์แวร์ และในบรรดาซอฟต์แวร์ทั้งหลาย ที่ใช้กันในการสั่งงานเครื่องคอมพิวเตอร์ ตัวที่สำคัญที่สุดก็คือ ระบบปฏิบัติการ ซึ่งเป็นตัวกลางในการรับคำสั่งจากซอฟต์แวร์ตัวอื่น ๆ แล้วมาควบคุมฮาร์ดแวร์ของเครื่องให้ทำตามความประสงค์

ตัวโปรแกรมหรือฮาร์ดแวร์ที่เรียกว่าระบบปฏิบัติการนี้จะเก็บอยู่ในดิสก์และทำงานเกี่ยวข้องกับการจัดการควบคุมดิสก์เป็นหลัก เพราะเป็นอุปกรณ์ที่ทำงานเร็ว มีความจุข้อมูลสูง และเก็บข้อมูลได้แม้จะปิดเครื่องแล้ว เมื่อจะใช้ซอฟต์แวร์นั้นก็ขอให้คอมพิวเตอร์อ่านเข้ามาในหน่วยความจำอีกทีหนึ่ง ดังนั้นเราจึงเรียกระบบปฏิบัติการนี้ว่า disk operating System หรือเรียกสั้นๆ ว่า DOS นั่นเอง

ไมโครคอมพิวเตอร์ก็เช่นเดียวกับคอมพิวเตอร์ก็จะต้องมีดอส เป็นตัวควบคุมเครื่องไมโครคอมพิวเตอร์รุ่นแรก ๆ นั้นมีขนาด 8 บิต (บิตเป็นหน่วยวัดปริมาณข้อมูลที่เครื่องนำไปคิดคำนวณในแต่ละครั้ง ดังนั้นเครื่องขนาด 8 บิตจึงมีขีดความสามารถน้อยกว่าเครื่อง 16 บิต) และ ดอสตัวที่แพร่หลายมากที่สุดก็คือ CP/M (Control Program for Microcomputer) ของบริษัทคิจิตอลรีเสิร์ช (digital Research) ต่อมาเมื่อเทคโนโลยีก้าวขึ้นสู่ระดับของเครื่อง 16 บิต ดอส ที่ว่านี้ก็คือ เอ็มเอส ดอส (MS-DOS) หรือ (PC-DOS) ของบริษัทไมโครซอฟต์ ซึ่งเป็นดอสที่ใช้กับเครื่องไอบีเอ็มพีซีนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านธุรกิจ
ตามปกติแล้ว โปรแกรม ดอส แต่ละตัวจะถูกเขียนขึ้นสำหรับใช้กับเครื่องเฉพาะแบบเท่านั้น
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังมีเหตุเปลี่ยนแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทั้งนี้เพราะภาษาเครื่องหรือ machine language ของแต่ละเครื่องนั้นแตกต่างกัน เอ็มเอสดอส หรือ พิกัดอส ก็เช่นกันเป็นดอสที่เขียนขึ้นสำหรับไมโครคอมพิวเตอร์ที่ใช้สำหรับไมโครโปรเซสเซอร์ เบอร์ 8086 หรือ 8088 เป็นหน่วยประมวลผลกลางหรือ ซีพียู (CPU-Central Processing Unit) ซึ่งทั้งสองเบอร์นี้เป็นของบริษัทอินเทล มีโครงสร้างภายในที่แตกต่างกันแต่ใช้ภาษาเครื่องที่เหมือนกันจึงใช้ดอส ร่วมกันได้ ส่วนคนที่เรียกว่า พิกัดอส เอ็มเอสดอส บ้างนั้น ความจริงเป็นโปรแกรมที่อยู่ในกลุ่มเดียวกัน คือเขียนโดยบริษัทไมโครซอฟท์ทั้งคู่แต่พิกัดอส นั้นหมายถึงโปรแกรมที่ไมโครซอฟท์ร่วมกับบริษัท ไอบีเอ็มจัดทำเป็นพิเศษ สำหรับเครื่องไอบีเอ็มพีซี ซึ่งใช้ไมโครคอมพิวเตอร์ 8088 เป็นซีพียูส่วน เอ็มเอสดอสหรือ ไมโครซอฟท์วินด์โดว์ ดอส นั้นเป็นชุดที่ทำขึ้นสำหรับเครื่องไมโครคอมพิวเตอร์ 16 บิต ทว่าไปหลายยี่ห้อที่ใช้ไมโครคอมพิวเตอร์เบอร์ 8086 หรือ 8088 เหมือนกันแต่การทำงานภายในอาจแตกต่างกันไปบ้างตามลักษณะของเครื่องแต่ละยี่ห้อ

ในการใช้งานคอมพิวเตอร์จะต้องเกี่ยวข้องกับ ดอส ไม่ทางตรงก็ทางอ้อม คืออาจจะใช้คำสั่งต่าง ๆ ในโปรแกรม ดอสให้เครื่องทำงานบางอย่างตามประสงค์ซึ่งส่วนมากจะเป็นการจัดการทั่ว ๆ ไปเกี่ยวกับระบบของเครื่อง เช่นการจัดที่ทางให้เหมาะสมสำหรับการเก็บข้อมูล หรือโปรแกรม การควบคุมเครื่องพิมพ์หรือพริ้นเตอร์ เป็นต้น นอกจากนี้ในการใช้งาน โปรแกรมอื่น ๆ ไม่ว่าจะเป็นตารางการคำนวณ หรือสเปรดชีต (Spreadsheet) เช่น โลดัส (Lotus) 1-2-3, Multiplan โปรแกรมเหล่านี้มักจะเรียกใช้ ดอส อีกต่อหนึ่งเพื่อให้จัดการระบบของเครื่อง เช่น การอ่าน หรือบันทึกข้อมูล ลงในแผ่นดิสก์ การพิมพ์ผลลัพธ์ออกจากเครื่องพิมพ์ ดังนั้นเรื่องของดอส จึงอาจกล่าวได้ว่าเป็นความรู้ที่จำเป็นสำหรับผู้ใช้งานคอมพิวเตอร์โดยทั่วไปถึงแม้จะไม่ถึงกับขนาดเชี่ยวชาญก็ตาม แต่ก็ช่วยให้ทำงานได้สะดวกรวดเร็ว และคล่องตัวมากยิ่งขึ้น และลดความจำเป็นที่จะต้องพึ่งพามืออาชีพ หรือผู้รู้ทางคอมพิวเตอร์ลง

โดยทั่วไปแล้วเมื่อพูดถึงโปรแกรมดอส ของเครื่องไอบีเอ็มพีซี หรือพิกัดอส ตลอดจนถึงเอ็มเอสดอส ด้วยนั้น มักจะระบุรุ่นของโปรแกรมหรือที่เรียกว่า เวอร์ชัน (Version) กำกับไปด้วย ทั้งนี้เพราะโปรแกรม ดอส นี้ถูกเขียนขึ้นตั้งแต่ปี 1981 พร้อมกับการวางตลาดของเครื่องไอบีเอ็มพีซี อันเป็นไมโครคอมพิวเตอร์รุ่นแรกของ บริษัทไอบีเอ็ม จากนั้นก็ได้มีการปรับปรุงแก้ไข ดอส มาหลายครั้ง ตั้งแต่รุ่นแรก ซึ่งมีเลขกำกับรุ่น คือ 1.0 โดยเมื่อมีการแก้ไข โปรแกรมดอส แต่ละครั้งก็จะมีทำให้เลขกำกับรุ่นตัวใหม่ที่มากกว่าเดิม

2.2 ความเป็นมาของดอส

ในช่วงปี ค.ศ. 1974 ไมโครคอมพิวเตอร์ได้เริ่มถือกำเนิดขึ้นเป็นครั้งแรก ดอสในระยะนั้น ไมโครโปรเซสเซอร์ที่ทำหน้าที่เป็นซีพียูตัวแรกที่ใช้กันแพร่หลาย คือ 8080 ของบริษัทอินเทล และไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังมีให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต่อมาก็เป็นเบอร์ Z-80 ของบริษัทไซลอกซึ่งทั้งสองตัวนี้เป็นไมโครโปรเซสเซอร์ระดับ 8 บิตโดย Z-80 ซึ่งออกมาภายหลังซึ่งมีขีดความสามารถสูงกว่า แต่ยังคงใช้ภาษาเครื่องของ 8080 ทั้งหมด ได้อยู่ โดยมีความพิเศษเพิ่มเติมอีกมาก ตั้งแต่ต้นจนถึงปี 1981 ระบบปฏิบัติการที่ใช้กันแพร่หลายกันมากที่สุดคือ CP/M หรือถ้าจะระบุกันให้ชัดเจนก็คือ CP/M-80 เพราะเขียนขึ้นด้วยภาษาเครื่องของ 8080 ถึงแม้ภายหลัง Z-80 จะได้รับความนิยมมาก และเข้ามาแทนที่ 8080 ก็ตาม แต่ก็ยัง CP/M อยู่

ต่อมาปี ค.ศ.1978 บริษัทอินเทลได้ผลิตไมโครโปรเซสเซอร์เบอร์ 8086 และ 8088 ขึ้น ทำให้มีความต้องการระบบปฏิบัติการตัวใหม่สำหรับซีพียูระดับ 16 บิต ที่มีความสามารถสูงขึ้นวงการอุตสาหกรรมไมโครคอมพิวเตอร์ ต่างก็พากันฝากความหวังให้กับ บริษัทดิจิทัล รีเสิร์ช เจ้าของ CP/M-80 ที่กำลังเร่งมือพัฒนา CP/M-86 เพื่อใช้กับ 8088 และ 8086 แต่โครงการดังกล่าวประสบกับปัญหาขณะเดียวกันบริษัทไมโครซอฟท์ผู้เขียนซอฟต์แวร์ตัวแปลภาษาเบสิก (BASIC) หรือ BASIC interpreter ซึ่งที่รู้จักกันดีคือ เอ็มเบสิก (MBASIC) ก็กำลังพัฒนาตัวแปลภาษาเบสิกเพื่อใช้กับ 8088 และ 8086 อยู่เช่นกัน โดยมีฮาร์นาร์ คือแพวงจรมิโครคอมพิวเตอร์ 8086 ที่ผลิตโดยบริษัท ซีแอตเติลคอมพิวเตอร์ ในการทดลอง

จนกระทั่ง ค.ศ.1980 ระบบปฏิบัติการ CP/M-86 ของบริษัทดิจิทัลรีเสิร์ช ก็ยังไม่เสร็จ แต่ความต้องการระบบปฏิบัติการเพื่อใช้กับ 8088 และ 8086 ได้เพิ่มขึ้นอย่างมากในที่สุดซีแอตเติลคอมพิวเตอร์ ก็ตัดสินใจที่จะสร้างระบบปฏิบัติการขึ้นใช้เองสำหรับซีพียูทั้งสอง โดยการนำของ ทิม แพตเตอร์สัน ระบบปฏิบัติการดังกล่าวเป็นรูปร่างขึ้นในปี ค.ศ.1980 โดยมีชื่อ .sh เรียกว่า QDOS (Quick and Dirty Operating System) ทั้งนี้เพราะเป็นโปรแกรม ที่ได้พัฒนาขึ้นอย่างรวดเร็ว

ในปลายปี ค.ศ.1980 QDOS ก็ได้เปลี่ยนชื่อเป็น 86-ดอส และให้บริษัทไมโครซอฟท์เป็นตัวแทนจำหน่ายโปรแกรมดังกล่าวแทน ขณะเดียวกันบริษัทดิจิทัลคอมพิวเตอร์ก็ได้ออก 86-ดอส เวอร์ชัน 1.00 ซึ่งใกล้เคียงกับ เอ็มเอสดอส หรือ ฟิซิคอส เวอร์ชัน 1.0 มาก และในปี 1981 ไมโครซอฟท์ก็ได้ซื้อลิขสิทธิ์ของ 86-ดอส จากซีแอตเติลคอมพิวเตอร์ และเปลี่ยนชื่อเป็น เอ็มเอสดอส เมื่อไอบีเอ็มออกเครื่องพีซีเป็นครั้งแรกในเดือนต่อมา ก็ได้เลือกให้ไมโครซอฟท์จัดทำระบบปฏิบัติการให้โดยดัดแปลงจาก เอ็มเอสดอส เรียกว่า ฟิซิคอส เวอร์ชัน 1.0 เป็นระบบปฏิบัติการมาตรฐาน ซึ่งทำให้ ดอส ดังกล่าวแพร่หลายออกไปอย่างกว้างขวาง และทางบริษัทไมโครซอฟท์ก็ได้พัฒนา เอ็มเอสดอส ไปจนถึงเวอร์ชัน 1.24 ซึ่งก็คือ ฟิซิคอส เวอร์ชัน 1.1 หรือ ดอส 1.1 นั้นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษานานาชาติ ไม่อนุญาตให้นำไปใช้โดยไม่ขออนุญาตจาก
 จุดมุ่งหมายแต่เดิมของ QDOS, 86-ดอส หรือ เอ็มเอสดอส ในยุคแรก ๆ นั้นต้องการจะใช้
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นระบบปฏิบัติการที่จะทดแทน CP/M-86 เพื่อให้โปรแกรมต่าง ๆ ที่ใช้งานภาษาได้ CP/M-80 สามารถแปลงมาใช้กับ 8086 ภาษาได้ QDOS, 86-ดอ หรือ เอ็มเอสดอส ได้ง่ายดังนั้นจึงพหุชาวมที่รักชาติศาสตร์ของ PC/M-80 ไว้มากพอสมควร ตั้งแต่เครื่องหมายพร้อมรับคำสั่ง (prompt) ก็ยังใช้ชื่อคำสั่งไครฟ์ตามด้วยเครื่องหมาย > เช่น A> เหมือนกับ CP/M หรือการทำงานภาษาใน เช่น CP/M หรือการทำงานภาษาใน เช่นการเรียกใช้โปรแกรมย่อยสำหรับการทำงานที่ต่าง ๆ ภาษาในดอส จากโปรแกรมภาษาเครื่องก็มีลักษณะคล้ายกันทำให้โปรแกรมที่เป็นที่นิยมแพร่หลายของ PC/M เช่น dBASEII, SuperCalc, WordStar และอื่น ๆ แปลงมาใช้กับ 8088 หรือ 8086 ได้โดยใช้เวลาไม่นานในระยะแรกที่เครื่องไอบีเอ็มพีซีออกมานั้น จึงมีโปรแกรมต่าง ๆ ออกตามมาอย่างรวดเร็ว เป็นปัจจัยหนึ่งที่ทำให้เครื่องในตระกูลไอบีเอ็มพีซีได้รับความนิยมแพร่หลายอย่างมากจนถึงปัจจุบัน

2.3 ส่วนประกอบของดอส

โครงสร้างภาษาในของโปรแกรม ดอส อาจแยกออกเป็นส่วนตัวต่าง ๆ ได้ 6 ส่วน ซึ่งมีหน้าที่แตกต่างกัน และถูกเก็บอยู่เป็นส่วน ๆ แยกออกกันเป็นดังนี้

2.3.1 รวม ไบออส (Rom-Bios) สืบมาจาก Basic Input Output System

คือโปรแกรมที่ควบคุมอุปกรณ์พื้นฐานของคอมพิวเตอร์ทั้งหมดโปรแกรมนี้ไบออสนี้จะเก็บอยู่ในหน่วยความจำแบบ รวม(ROM) สืบมาจาก Read Only Memory ซึ่งก็เก็บข้อมูลได้แม้จะปิดเครื่องและเนื่องจากมันถูกเก็บอยู่ในอุปกรณ์ทางฮาร์ดแวร์คือ รวมไม่ได้ถูกแจกจ่ายไปพร้อมกับดิสก์จึงเป็นส่วนที่คงที่ตายตัว ไม่เปลี่ยนแปลงไปตามโปรแกรม ดอส เวอร์ชันต่าง ๆ แทน

2.3.2 บุต แร็กคอร์ด (Boot Record)

เป็นโปรแกรมสั้น ๆ ที่เก็บอยู่ในเซกเตอร์ของแผ่นดิสก์ทุกแผ่นที่ใช้บูตได้ (boot) หน้าที่ของโปรแกรมนี้อีกคือ การสั่งให้เครื่องพีซีทำการ โหลด (Load) คืออ่านโปรแกรมสองส่วนถัดไปของ ดอสเข้ามาในหน่วยความจำ การโหลดนี้จะถูกทำทุกครั้งที่เครื่อง พีซีถูกเปิดขึ้นมาใหม่ หรือเมื่อทำการรีเซ็ต (Restart) หรือการบูตใหม่คือตั้งคอนโทรลการกดปุ่ม <Ctrl>, <Alt> และ พร้อมกันทั้งสามปุ่มบนคีย์บอร์ด ทั้งนี้เพราะเมื่อเปิดเครื่องใหม่นั้นในหน่วยความจำ แรม (RAM) สืบมาจาก Random Access Memory) ยังไม่มีโปรแกรม ดอสอยู่ จึงต้องโหลด เข้าใหม่จากดิสก์

2.3.3 IBKIO.COM และ IBMDOS.COM

คือโปรแกรมที่ถูกเก็บเป็นไฟล์ในดิสก์เกิดสองไฟล์ สองส่วนนี้จะถูกโหลดเข้าโดยการทำงานของ บุต แร็กคอร์ด ทั้งสองส่วนนี้จะคงยังอยู่ในหน่วยความจำตลอดเวลาที่ ดอส ทำงานอยู่ แต่เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เหตุที่แยกออกเป็นสองส่วนก็เพราะมีลักษณะการทำงานที่แตกต่างกัน คือ IBMIO.COM นั้นเป็นส่วนที่จะทำงานควบคู่กับ รอมไบออส ในการควบคุมอุปกรณ์ต่าง ๆ แต่ละตัวนอกจากนี้ถ้ามีการติดตั้งอุปกรณ์ใด ๆ เพิ่มเข้าไปในระบบตั้งแต่คอสเวอร์ชัน 2.00 ขึ้นไปก็จะสามารถ ติดตั้งโปรแกรมย่อยที่สามารถควบคุมอุปกรณ์ตัวนั้น ๆ ที่เรียกว่าไดร์ไครเวอร์เข้าเป็นส่วนหนึ่งของ IBMIO.COM นี้ได้อีกด้วยดังนั้นเราจึงกล่าวว่ทั้งรอมไบออส , IBMDOS.COM และ ไดรเวอร์ที่ติดตั้งเพิ่มนี้ เป็นส่วนที่จะทำหน้าที่ร่วมกันในการควบคุม ฮาร์ดแวร์ของพีซีโดยตรงคือรวมกันเป็นไบออสทั้งหมด ส่วนอีกไฟล์ก็คือ IBMDOC.COM นั้นจะเป็นตัวควบคุมในระดับสูงขึ้นไป คือแทนที่จะควบคุมอุปกรณ์แต่ละตัวก็จะเป็นการประสานงานการทำงานของอุปกรณ์ต่าง ๆ เข้าด้วยกันทั้งไฟล์ IBMBIO.COM และ IBMDOC.COM นี้จะเป็นไฟล์ที่ถูกซ่อนไม่ปรากฏให้เห็นเมื่อใช้คำสั่ง DIR เพื่อดูรายชื่อไฟล์บนแผ่นดิสก์ และจะเป็นที่ถูกจัดเป็นพิเศษบนแผ่นดิสก์ตั้งแต่การฟอร์แมต เพื่อจัดข้อมูลบนแผ่นดิสก์แผ่นนั้น ๆ ทั้งนี้เพื่อให้บูตแรกคอร์ค้นหาไฟล์ทั้งสองเจอ และโหลดเข้ามาได้สำหรับเอ็มเอสดอสไฟล์ทั้งสองของดอสจะเรียกชื่อต่างกันคือ IO.SYS ซึ่งจะเทียบได้กับ IBMBIO.COM และ MS-DOS.SYS เทียบได้กับ IBMDOC.COM

2.3.4 COMMAND.COM

ซึ่งเป็นไฟล์ที่มีลักษณะไฟล์เก็บโปรแกรมธรรมดาทั่วไปมีลักษณะเป็นแบบ .COM ไฟล์นี้จะถูกโหลดเข้าโดยการทำงานของ IBMBIO.COM อีกที่หนึ่งหน้าที่ของส่วนนี้เรียกได้ว่าเป็นตัวแปลคำสั่ง หรือ command interpreter ซึ่งจะรับคำสั่งที่ป้อนเข้าไปทางคีย์บอร์ด แปลความหมายว่าต้องการให้ดอสทำอะไรแล้วจึงเรียกใช้ส่วนอื่น ๆ ของดอส อีกที่หนึ่ง ส่วนที่เป็น COMMAND.COM นี้จะมีบางส่วนเท่านั้นที่อยู่หน่วยความจำตลอดเวลาที่ดอส ทำงาน แต่บางส่วนจะต้องไปอ่านเข้ามาเพิ่มจากดิสก์เป็นครั้งคราวตามความจำเป็น

คำสั่งในดอสนั้น อาจแบ่งได้เป็นสองพวกเช่นกันคือ คำสั่งภายใน (internal) ซึ่งจะถูกแปลและทำงานตามได้ทันทีจาก COMMAND.COM เลย กับ คำสั่งภายนอก (external) ซึ่งนับเป็นตัวสุดท้ายของดอส คำสั่งภายนอกเหล่านี้จะถูกเรียก หรือโหลด เข้าไปยังหน่วยความจำโดย COMMAND.COM เพื่อทำงานต่อไป คำสั่งภายนอกเหล่านี้แต่ละคำสั่งจะถูกเก็บแยกเป็นไฟล์ต่างหากจากกัน และมีฐานะคล้าย ๆ กันกับโปรแกรมใช้งานอื่น ๆ ที่ทำงานภายใต้ดอส เพียงแต่ว่าเรามองมันเป็นคำสั่งของดอส เพราะมันมาด้วยกัน และทำหน้าที่ในลักษณะใกล้เคียงกันคำสั่งเหล่านี้จะอยู่ในหน่วยความจำแต่เฉพาะเมื่อถูกเรียกใช้เท่านั้น

2.4 คำสั่งและการใช้งานดอส

2.4.1 ดอส และงานของ ดอส

ดอส หรือที่เรียกกันว่าระบบจัดการเป็นโปรแกรมซึ่ง มีหน้าที่ควบคุมการทำงานของคอมพิวเตอร์เป็นสื่อกลางระหว่างผู้ใช้กับคอมพิวเตอร์โดยตรงนอกจากนี้ดอสจะติดต่อและควบคุมการทำงานของอุปกรณ์ร่วมต่าง ๆ ที่ต่อกับตัวคอมพิวเตอร์ในการใช้ดอส ผู้ใช้จะต้องป้อนคำสั่ง ซึ่งดอสจะรับคำสั่งนั้นไว้ และทำงานตามคำสั่งนั้น

2.4.2 คำสั่งภายในและคำสั่งภายนอก

คำสั่งประเภทภายในคือคำสั่งที่ดอสสามารถทำงานได้โดยตรง เช่นคำสั่ง DIR, COPY คำสั่งประเภทภายนอกเป็นไฟล์ที่คำสั่งที่อยู่บนแผ่นดิสก์เมื่อดอสรับคำสั่ง และตรวจสอบพบว่าไม่ใช่คำสั่งใน ดอสจะถือว่าเป็นคำสั่งประเภทภายนอก และทำการค้นหาจากแผ่นดิสก์เมื่อพบก็จะโหลดไฟล์นั้นเข้าสู่หน่วยความจำและรันตามโปรแกรมของไฟล์นั้น ๆ ตัวอย่างของคำสั่งประเภทภายนอกคือ CHKDSK, FORMAT และ DISKCOPY เป็นต้น จากลักษณะที่ดอสตรวจสอบไฟล์จากแผ่นดิสก์และโหลดเข้าสู่หน่วยความจำ ทำให้เราสามารถสร้างไฟล์โปรแกรมที่มีลักษณะคล้ายคำสั่งประเภทภายนอก เพื่อที่ดอสจะสามารถรันไฟล์เหล่านั้นได้เนื่องจากไฟล์ที่ดอสสามารถโหลด และรันได้คือไฟล์ที่มีส่วนขยายเป็น .COM และ .EXE ดังนั้นไฟล์โปรแกรมที่เราสร้างขึ้นจะมีส่วนขยายเป็น .COM และ EXE ไฟล์เหล่านั้นจะเป็นไฟล์โปรแกรมภาษาเครื่องเท่านั้น

นอกจากไฟล์ .COM และ .EXE แล้วยังมีไฟล์อีกชนิดหนึ่งที่ดอสสามารถจะโหลด และรันตามคำสั่งได้คือแบตช์ไฟล์ซึ่งเป็นไฟล์ที่มีส่วนขยายเป็น .BAT แบตช์ไฟล์คือไฟล์ที่บรรจุคำสั่งต่าง ๆ ของดอสเอาไว้เป็นชุด และให้ดอสทำงานตามชุดคำสั่งเหล่านั้นเรียงตามลำดับ ซึ่งจะมีผลเท่ากับเราป้อนคำสั่งของดอสทีละคำสั่ง คำสั่งในแบตช์ไฟล์อาจจะป้อนคำสั่งประเภทภายในคำสั่งประเภทภายนอก หรือไฟล์โปรแกรมต่าง ๆ ที่ดอสสามารถโหลดและรันก็ได้

2.4.3 คำสั่งต่าง ๆ ที่ใช้งานของดอส

2.4.3.1 การเริ่มสตาร์ท (STARTUP)

เมื่อเริ่มเปิดเครื่อง คอมพิวเตอร์คอมพิวเตอร์จะทำการตรวจสอบชุดคำสั่งที่บันทึกไว้เป็นการถาวรซึ่งจะบอกคอมพิวเตอร์ ว่าจะต้องทำอะไรก่อน และอ่านคำสั่งเหล่านั้นไว้ในหน่วยความจำ จากนั้นคอมพิวเตอร์จะเริ่มตรวจสอบตัวเองพร้อมกับตรวจสอบ หน่วยความจำ ดิสก์ไดรฟ์ และเคียบอร์ดไปพร้อมๆ กัน

เมื่อคอมพิวเตอร์พบส่วนประกอบต่าง ๆ ครบทั้งหมด และเห็นว่าส่วนประกอบเหล่านั้นอยู่ในสภาพที่ทำงานได้แล้ว คอมพิวเตอร์จะตรวจสอบหา เอ็มเอสดอส นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แห่งแรกที่คอมพิวเตอร์ตรวจสอบ คือไดรฟ์ A แต่ถ้าไม่พบ คอมพิวเตอร์จะไปตรวจสอบในไดรฟ์ C ถ้าหากคอมพิวเตอร์นั้นมีฮาร์ดดิสก์ แต่ถ้าที่เครื่องคอมพิวเตอร์ไม่มีฮาร์ดดิสก์ก็จะแสดงข้อความว่า

Non-System disk or disk error

และขอให้ใส่แผ่นดิสก์ใหม่ที่ถูกต้องลงไดรฟ์ A เมื่อ MS-DOS พร้อมทั้งจะทำงานแล้ว จะมีเครื่องหมายแสดงดังนี้

C:\>

ถ้าเครื่องหมายหรือรับคำสั่งที่แตกต่างไปจากนี้แสดงว่าระบบปฏิบัติการของเครื่องแตกต่างกันแต่อย่างระก่ก็ตามก็จะมีข้อความเหมือนกัน

2.4.3.2 การเปลี่ยนไดรฟ์

แต่ละไดรฟ์ในคอมพิวเตอร์ จะกำหนดด้วยตัวอักษรแตกต่างกัน เช่น ไดรฟ์แรกเป็นตัวอักษร A ไดรฟ์ที่สองเป็น B หรือถ้าหากมีฮาร์ดดิสก์ก็จะใช้อักษร C

เมื่อสตาร์ทคอมพิวเตอร์จากฮาร์ดดิสก์ เครื่องหมายพร้อมจะอยู่ในรูป C:\> ถ้าต้องการเปลี่ยนไดรฟ์ทำได้โดยการพิมพ์ตัวอักษรของไดรฟ์ที่ต้องการลงไปแล้วตามหลังด้วยเครื่องหมาย

เช่น ถ้าต้องการเปลี่ยนจากไดรฟ์ C เป็นไดรฟ์ A

C:\> a:

2.4.3.3 คำสั่ง DATE

เป็นคำสั่งที่ใช้ในการเปลี่ยน วัน เดือน ปี ซึ่งในปกติคอมพิวเตอร์จะรักษาวันที่ให้ทันสมัยอยู่เสมอ

เช่น ถ้าต้องการเปลี่ยน หรือดูวันที่ปัจจุบัน

C:\> date

2.4.3.3 คำสั่ง TIME

เป็นคำสั่งที่ใช้เปลี่ยนแปลง เวลา ของคอมพิวเตอร์ซึ่งในปกติคอมพิวเตอร์จะรักษาวันที่ให้ทันสมัยอยู่เสมอ

เช่น ถ้าต้องการเปลี่ยน หรือเวลาปัจจุบัน

C:\> time

2.4.3.4 คำสั่ง CLS

เป็นคำสั่งใช้ในการทำความสะอาดจอภาพในกรณีที่จอภาพมีข้อมูลที่ไม่ต้องการอยู่ เช่น ถ้าไม่ต้องการข้อมูลในไดรฟ์ C ที่แสดงอยู่บนหน้าจอภาพ

```
C:\> cls
```

2.4.3.5 คำสั่ง FORMAT

การฟอร์แมตจะใช้เฉพาะในครั้งแรกที่ต้องการที่จะใช้แผ่นดิสก์เท่านั้นที่จำเป็นต้องใช้ เพราะไม่สามารถใช้ดิสก์ที่ยังไม่ได้ฟอร์แมตได้ และเราสามารถที่จะนำแผ่นดิสก์เก่ามาใช้ใหม่ได้โดยการฟอร์แมตใหม่

แบบฟอร์มคำสั่ง

```
format drive:
```

เมื่อ drive เป็นตัวอักษรของไดรฟ์ที่ต้องการใช้ฟอร์แมตดิสก์

- การฟอร์แมตแผ่นดิสก์และการตั้งชื่อแผ่นดิสก์

ปกติ เอ็มเอส ดอส จะขอให้ตั้งชื่อแผ่นดิสก์โดยถือเป็นส่วนหนึ่งของคำสั่งฟอร์แมต

แบบฟอร์มคำสั่ง

```
format drive: /v:label
```

เมื่อ drive เป็นตัวอักษรของไดรฟ์ที่ต้องการใช้ฟอร์แมตดิสก์

/v:label เป็นการบอก เอ็มเอส ดอส ให้ทราบว่าต้องการกำหนดชื่อแผ่นดิสก์

- การฟอร์แมตแผ่นดิสก์ที่มีความจุน้อยกว่าความจุของไดรฟ์

ซึ่งในบางครั้งจะต้องการฟอร์แมตแผ่นดิสก์โดยที่แผ่นดิสก์มีความจุน้อยกว่าความจุของไดรฟ์ตามมาตรฐาน

แบบฟอร์มคำสั่ง

```
format drive: /f:size
```

เมื่อ drive เป็นตัวอักษรไดรฟ์ที่ต้องการใช้ในการฟอร์แมต

/f:size เป็นการบอกให้ MS-DOS ทราบความจุของแผ่นดิสก์ที่ต้องการ

การฟอร์แมตสำหรับแผ่นดิสก์ขนาด 5 1/4 นิ้วสามารถทำได้

360KB ได้สำหรับแผ่นดิสก์ขนาด 3 1/2 นิ้ว สามารถใช้

720 kB หรือ 1.2MB ได้ /4 เป็นวิธีที่กำหนด /f:360
อย่างรวดเร็ว (ฟอร์แมตแผ่นดิสก์ 360kB ในไดรฟ์ 1.2MB)

- การฟอร์แมตอย่างรวดเร็ว

เป็นการประหยัดเวลาในการฟอร์แมต แต่จะใช้ได้เฉพาะแผ่นดิสก์ที่ผ่านการฟอร์แมต
มาแล้วเท่านั้นและไม่ต้องการเปลี่ยนความจุของดิสก์เท่านั้น

แบบฟอร์มคำสั่ง

`format drive:/q`

เมื่อ drive เป็นตัวอักษรของไดรฟ์ที่ต้องการใช้ฟอร์แมตแผ่นดิสก์

11

/q เป็นการบอกให้ เอ็มเอส ดอส ให้ทราบว่าการฟอร์แมต
อย่างรวดเร็ว

- การฟอร์แมตแผ่นที่บูตได้

เป็นการฟอร์แมตแบบหนึ่งที่ทำกากร้ายไฟล์ `command.com` ไฟล์ `msdos.sys` และ
`io.sys` ไปไว้ในแผ่นดิสก์ที่ฟอร์แมตใหม่ได้ ทำให้แผ่นดิสก์นี้กลายเป็นดิสก์ระบบปฏิบัติการ

แบบฟอร์มคำสั่ง

`format drive:/s`

เมื่อ drive เป็นตัวอักษรของไดรฟ์ ที่ต้องการใช้ฟอร์แมตแผ่นดิสก์

/s เป็นการบอก เอ็มเอส ดอส ให้ฟอร์แมตแผ่นดิสก์ให้เป็นดิสก์
ระบบปฏิบัติการ

2.4.3.6 คำสั่ง CHKDSK

ใช้เพื่อตรวจสอบทั้งสภาพของดิสก์จำนวนพื้นที่ทั้งหมดที่ใช้เก็บข้อมูลพื้นที่ที่ชำรุดจำนวน
หน่วยความจำในคอมพิวเตอร์ และจำนวนหน่วยความจำที่ยังเหลือให้รันโปรแกรม

แบบฟอร์มคำสั่ง

`chkdsk [drive:][file] /v /f`

เมื่อ drive เป็นตัวอักษรของไดรฟ์ ที่บรรจุแผ่นดิสก์ที่ต้องการตรวจสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งาน การศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
file เป็นชื่อไฟล์เดี่ยวหรือหลายไฟล์ ที่ต้องการตรวจสอบ
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- /v เป็นการบอก เอ็มเอส คอส ให้ทำการสร้างไฟล์ต่าง ๆ ที่อยู่ในดิสก์
- /f เป็นการบอก เอ็มเอส คอส ให้แก้ไขข้อผิดพลาดในการเก็บข้อมูลให้ถูกต้อง

2.4.3.7 คำสั่ง DISKCOPY

เป็นคำสั่งที่ทำการก๊อปปี้ ที่เหมือนกันทุกอย่างของแผ่นดิสก์ต้นฉบับ คำสั่ง ดิสก์ก๊อปปี้ จะทำงานกับแผ่นฟลอปปี้ดิสก์เท่านั้น

แบบฟอร์มคำสั่ง

```
diskcopy [disk1:][disk2:] /v
```

- เมื่อ disk1 เป็นตัวอักษรของไดรฟ์ที่บรรจุแผ่นดิสก์ที่ต้องการก๊อปปี้
- disk2 เป็นตัวอักษรของไดรฟ์ที่บรรจุแผ่นดิสก์ที่ต้องการทำก๊อปปี้
- /v เป็นการบอก เอ็มเอส คอส ให้พิสูจน์ว่าก๊อปปี้เหมือน

2.4.3.8 คำสั่ง DISKCOMPARE

เป็นการเปรียบเทียบโปรแกรมในแผ่นดิสก์สองแผ่นว่าเหมือนกันหรือไม่

แบบฟอร์มคำสั่ง

```
diskcomp [disk1:][disk2:]
```

- เมื่อ disk1 เป็นตัวอักษรของไดรฟ์ที่บรรจุแผ่นแรกที่ต้องการเปรียบเทียบ
- disk2 เป็นตัวอักษรของไดรฟ์ ที่บรรจุแผ่นที่สองที่ต้องการเปรียบเทียบ

2.4.3.9 คำสั่ง DIRECTORY

เป็นคำสั่งที่ใช้แสดงชื่อของไฟล์หรือค้นหาชื่อไฟล์ ซึ่งสามารถใช้ได้หลายวิธี ตัวอย่างรายชื่อไฟล์ในไดเรกทอรี

```
Volume in drive C is MIRANDA
```

```
Volume Serial Number is 16E4-9EF7
```

```
directory of C:\PROJECTS
```

. .	<DIR>	09-16-91
FILE1	1024	09-18-90
FILE2	32358	09-27-91
FILE3	DOC 27	10-04-90
FILE4	TXT 28160	11-11-91

6 file(s) 61559 bytes

24759552 bytes free

สองบรรทัดแรกจะระบุชื่อดิสก์และหมายเลขของดิสก์

ข้อความ Directory of จะบอกให้ทราบถึงโคเรกทอรีที่ใช้เก็บไฟล์ ต่าง ๆ จาก FILE1 ถึง FILE4.TXT จะบอกให้ทราบเรื่องของไฟล์ในบรรทัดที่อยู่ในดิสก์ เช่น

- FILE4 เป็นชื่อเฉพาะของไฟล์
- TXT เป็นนามสกุลของไฟล์
- 28160 ขนาดของไฟล์เป็นไบต์
- 11-11-91 เป็นวันที่สร้างไฟล์
- 8.34a เป็นเวลาที่สร้างไฟล์นี้เสร็จ

ในบรรทัดรองสุดท้ายจะบอกจำนวน ไฟล์ และ ไบต์ ที่ใช้งานไปแล้ว และบรรทัดสุดท้ายจะบอกไบต์ที่เหลือไว้สำหรับเก็บข้อมูลในดิสก์

- รายละเอียดธรรมดาของคำสั่ง โคเรกทอรี

ใช้ในการทำรายชื่อไฟล์ทั้งหมดหรือเฉพาะไฟล์ที่กำหนดชื่อลงไปเท่านั้น

แบบฟอร์มคำสั่ง

```
dir [drive:][filespec]/p
```

เมื่อ drive เป็นชื่อของไดรฟ์ ที่บรรจุดิสก์ที่ต้องการดูรายชื่อไฟล์ในโคเรกทอรี

filespec เป็นชื่อไฟล์หรือกลุ่มไฟล์ที่ต้องการให้อยู่ในรายชื่อ

/p เป็นสวิตช์ซึ่งทำให้ เอ็มเอส ดอส หยุดชั่วคราวเมื่อแสดงผล

เพิ่มเติมหน้าจอถ้าจะให้แสดงผลต่อให้กดคีย์ใดคีย์หนึ่งก็ได้:

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานที่ละรายซึ่งหาอ่านได้ไม่ละเมิดให้นำไปใช้ประโยชน์ด้านการค้า
- การแสดงผลรายชื่อไฟล์ในโคเรกทอรีในแบบกว้างเต็มจอ
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะแสดงไฟล์เป็นคอลัมน์ขวางจนเต็มจอภาพ โดยไม่แสดงขนาดไฟล์ วันที่ และเวลาแบบฟอร์มคำสั่ง

```
dir [drive:][filespec]/w
```

เมื่อ drive เป็นชื่อของไดรฟ์ที่บรรจุดิสก์ที่ต้องการดูรายชื่อไฟล์ในไดเร็กทอรี

filespec เป็นชื่อไฟล์ หรือกลุ่มไฟล์ที่ทำให้เกิดรายชื่อไฟล์ในไดเร็กทอรีแบบกว้างเต็มจอ

- การจัดลำดับรายชื่อไฟล์

การจัดลำดับรายชื่อไฟล์ในไดเร็กทอรีทำได้หลายวิธี

แบบฟอร์มคำสั่ง

```
dir [drive:][filespec]/o[:] sort-type
```

เมื่อ drive เป็นชื่อของไดรฟ์ ที่บรรจุดิสก์ที่ต้องการดูรายชื่อไฟล์ในไดเร็กทอรี

filespec เป็นชื่อของไฟล์ หรือกลุ่มไฟล์ที่ทำให้เกิดรายชื่อไฟล์ในไดเร็กทอรี

/o[:] sort-type เป็นการบอกให้เอ็มเอส ดอส ทราบชนิดของการจัดลำดับ จะต้องพิมพ์ /O ลงไปซึ่งสามารถกำหนดชนิดของการจัดลำดับได้ดังนี้

ชนิดของการจัดลำดับ

การเรียงลำดับที่ได้

N	เรียงลำดับตามตัวอักษรของชื่อไฟล์ (A ถึง Z)
E	เรียงลำดับตามตัวอักษรของนามสกุล (A ถึง Z)
S	ตามขนาด จากเล็กที่สุดไปหาใหญ่ที่สุด
D	ตามวันที่ จากวันที่ผ่านมาแล้วนานที่สุดไปวันที่ล่าสุด
G	ตามไดเร็กทอรีที่เรียงรายชื่อไว้ก่อน

- การแสดงผลตามลักษณะเฉพาะ

นอกจากขนาด วันที่ และเวลาแล้ว เอ็มเอส ดอส ยังคิดตามคุณลักษณะเฉพาะอย่าง

ของไฟล์ที่เรียกว่า ลักษณะเฉพาะซึ่งมีผลต่อการปฏิบัติของไฟล์อีก 5 อย่างดังนี้

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



- D สำหรับไดเรกทอรี (Directory)
 - H สำหรับไฟล์ที่ซ่อนไว้ (Hidden)
 - S สำหรับไฟล์ระบบ (System)
 - R สำหรับไฟล์อ่านเท่านั้น (Read-Only)
 - A สำหรับสถานการณ์เก็บข้อมูลไว้ในหน่วยความจำสำรอง
- แบบฟอร์มคำสั่ง

```
dir [drive:][filespec]/a[:]attribute
```

เมื่อ drive เป็นชื่อของไดรฟ์ที่บรรจุดิสก์ที่ต้องการดูรายชื่อไฟล์ในไดเรกทอรี

filespec เป็นชื่อไฟล์หรือกลุ่มไฟล์ ที่ทำให้เกิดรายชื่อไฟล์ในไดเรกทอรี

/o [:] attribute เป็นการบอกให้เอ็มเอส ดอส แสดงรายชื่อไฟล์ในไดเรกทอรีที่มีลักษณะ เฉพาะของไฟล์ตามที่กำหนดให้เครื่องหมายโคลอนจะมี หรือไม่มีก็ได้แต่จะต้องกำหนด attribute ลงไปว่าเป็น D,H,S,R หรือ A

2.4.3.10 คำสั่ง RENAME

เป็นคำสั่งที่ใช้ในการเปลี่ยนชื่อไฟล์เดี่ยวหรือหลายไฟล์ก็ได้

แบบฟอร์มคำสั่ง

```
ren [drive:] name1 name2
```

เมื่อ drive เป็นไดรฟ์ที่บรรจุแผ่นดิสก์ที่มีไฟล์หรือไฟล์ต่าง ๆ ที่ต้องการตั้งชื่อใหม่

name1 เป็นชื่อของไฟล์ที่จะตั้งชื่อใหม่

name2 เป็นชื่อที่ต้องการตั้งแทน name1

2.4.3.11 คำสั่ง ATTRIB

เป็นคำสั่งที่ใช้ในการป้องกันการแก้ไขไฟล์ถ้าใช้คำสั่งนี้เพื่อซ่อนไฟล์แล้วไฟล์ต่าง ๆ จะไม่ปรากฏในรายชื่อไฟล์ในไดเรกทอรีธรรมดา

แบบฟอร์มคำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
attrib[+attribute][drive:]filespec[/s]
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ + เป็นการเปิด attribute, - เป็นการปิด attribute สำหรับ attribute ให้ กำหนดอย่างใดอย่างหนึ่งดังนี้

R สำหรับ read-only (อ่านเท่านั้น) เพื่อป้องกันการแก้ไขไฟล์

A สำหรับ archive เพื่อแสดงว่าไฟล์ที่มีการแก้ไข หรือไม่นับตั้งแต่นั้นสำรองไว้ในดิสก์ อีกแผ่นหนึ่งแล้ว

H สำหรับ hidden เพื่อซ่อนไฟล์ไว้ไม่ให้มองเห็น

/s เป็นสวิตช์ที่เพิ่มเติมการปฏิบัติของคำสั่ง เป็นการบอกให้คอมพิวเตอร์ดำเนินการวิธีคือ ไฟล์ในไดเรกทอรีทั้งหมดภาษาในไดเรกทอรีที่ทำงานอยู่แทนที่จะดำเนินการวิธี คือไดเรกทอรีที่ทำงานอยู่แต่อย่างเดียว

2.4.3.12 คำสั่ง TYPE

คำสั่งนี้จะสั่งให้ เอ็มเอช คอส แสดงผลรายละเอียดภาษาในไฟล์ เพื่อดูว่าในไฟล์นั้นมีอะไรอยู่บ้างโดยไม่ต้องสตาร์ทโปรแกรมประยุกต์ที่ใช้สร้างไฟล์นั้นขึ้นมา

แบบฟอร์มคำสั่ง

```
type [drive:] filespec
```

เมื่อ drive เป็นตัวอักษรของไดรฟ์ ที่บรรจุจุดสักที่มีไฟล์ที่ต้องการให้แสดงผล

filespec เป็นชื่อของไฟล์ที่ต้องการแสดงผลสามารถกำหนดได้เพียงครั้งละหนึ่งไฟล์เท่านั้น

2.4.3.13 คำสั่ง MORE

คำสั่งนี้จะสั่งให้ เอ็มเอช คอส หยุดการแสดงผลที่ขารวมหลังจากที่เต็มแต่ละจอภาพแล้วและคอยจนกว่าจะคีย์เพื่อแสดงว่าต้องการจะดูต่อไป

แบบฟอร์มคำสั่ง

```
command : more
```

เมื่อ command เป็นคำสั่งของ เอ็มเอช คอส เช่น Dir

: more เป็นการบอก เอ็มเอช คอส ให้หยุดหลังจากการแสดงผลเต็มจอภาพแต่ละครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมีให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.3.14 คำสั่ง COPY

เป็นการสำเนาไฟล์ซึ่งไม่เหมือนกับการย้ายไฟล์ คือคำสั่ง ก๊อปปี้จะสร้างไฟล์ขึ้นมา แต่จะไม่เปลี่ยน หรือลบไฟล์เดิม ซึ่งจะสามารถจะก๊อปปี้จากดิสก์หนึ่งไปไว้ยังอีกดิสก์หนึ่ง หรือจาก ไดรฟ์ทอรัหนึ่งไปยังอีกไดรฟ์ทอรัหนึ่งได้

- การก๊อปปี้เพื่อสำเนาไฟล์

เป็นการก๊อปปี้ไฟล์หนึ่งหรือหลายไฟล์จากดิสก์ หรือไดรฟ์ทอรัไปไว้ยังอีกดิสก์หรืออีกไดรฟ์ทอรัหนึ่ง

แบบฟอร์มคำสั่ง

copy source [destination]

เมื่อ source เป็นไดรฟ์และการกำหนดชื่อไฟล์เดียว หรือหลายไฟล์ที่ต้องการก๊อปปี้

destination เป็นไดรฟ์ และการกำหนดชื่อไฟล์ของสถานที่ที่ต้องการให้ก๊อปปี้ไฟล์เดียวหรือหลายไฟล์

2.4.3.15 คำสั่ง DELETE

จะขอลดความหนาแน่นในแผ่นดิสก์ และลบไฟล์ที่ไม่ต้องการออกระบบปฏิบัติการจะไม่มีคำสั่ง Move อยู่ด้วย แต่ก็สามารถใช้คำสั่ง Delete พร้อมด้วยคำสั่ง Copy เพื่อย้ายไฟล์ไป

แบบฟอร์มคำสั่ง

del [drive:] filespec/p

เมื่อ drive และ filespec เป็นตัวอักษรของไดรฟ์และการกำหนดชื่อไฟล์

/p เป็นการบอก เอ็มเอส ดอส ให้มีเครื่องหมายพร้อมรับการยืนยันโดยแสดงผลชื่อของไฟล์ที่จะลบออกตามด้วย Delete (Y/N)?

2.4.3.16 คำสั่ง UNDELETE

เป็นคำสั่งที่ช่วยในการกู้ข้อมูลที่สูญหายไปจากการลบข้อมูล แต่การกู้ อาจจะมีผลผิดพลาดได้ในกรณีที่การลบข้อมูลนั้นเกิดขึ้นมานาน หรือเมื่อพื้นที่ของดิสก์ถูกเซฟเข้าตรงที่ที่ต้องการกู้ก็ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไม่สามารถที่จะกู้ได้

แบบฟอร์มคำสั่ง

`undelete [drive:][filespec]/list/all`

เมื่อ `drive` และ `filespec` คือตัวอักษรของไดรฟ์ และการกำหนดชื่อไฟล์ของไฟล์เดี่ยว หรือหลายไฟล์ที่ต้องการจะ `undelete` ถ้าไม่พิมพ์ไดรฟ์คำสั่ง `undelete` จะกู้ไฟล์ที่กำหนดให้จากไดรฟ์ และไดเรกทอรีที่ทำงานอยู่ถ้าไม่พิมพ์ชื่อ `filespec` คำสั่ง `undelete` จะสมมติว่า เราต้องการกู้ไฟล์ที่ลบออกไปทุกไฟล์ในไดรฟ์ หรือไดเรกทอรีที่กำหนดให้

`/list` เป็นสวิตช์ที่บอกคำสั่ง `undelete` ให้ทำรายชื่อไฟล์โดยไม่พยายามที่จะกู้กลับคืนไฟล์ที่ถูกลบออกไปทั้งหมดที่สามารถค้นหาได้ เมื่อใช้สวิตช์นี้คำสั่ง `undelete` จะใช้เครื่องหมายดอกจัน (*) ดอกเดี่ยวแทนไฟล์ ที่กู้กลับเข้ามาได้เป็นบางส่วน และ (**) แทนไฟล์ที่กู้กลับมาไม่ได้เลย

`/all` เป็นการบอกคำสั่ง `undelete` ให้กู้ไฟล์ที่ได้กำหนดทั้งหมดกลับคืนโดยไม่ต้องมีเครื่องหมายพร้อมรับการยืนยัน

2.4.3.17 คำสั่ง MAKE DIRECTORY

เป็นการบอกให้เอ็มเอส ดอส ให้สร้างไดเรกทอรีใหม่สามารถสร้างไดเรกทอรีในรากไดเรกทอรี หรือในไดเรกทอรีอื่นในดิสก์โดยไม่ต้องคำนึงถึงว่า ไดเรกทอรีที่ทำงานอยู่จะเป็นอย่างไร

แบบฟอร์มคำสั่ง

`md [drive:]path`

เมื่อ `drive` เป็นชื่อไดรฟ์ที่บรรจุดิสก์ที่ต้องการสร้างไดเรกทอรี

`path` เป็นชื่อทางเดินสารบบของไดเรกทอรีที่ต้องการสร้างขึ้น

2.4.3.18 คำสั่ง CHANGE DIRECTORY

เป็นคำสั่งที่ใช้ในการเปลี่ยนไดเรกทอรีที่ทำงานอยู่ไปยังไดเรกทอรีหนึ่ง

แบบฟอร์มคำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ drive เป็นชื่อไดรฟ์ ที่บรรจุดิสก์ที่ต้องการเปลี่ยน หรือแสดงผลของ
ไดเรกทอรีที่ทำงานอยู่
path เป็นชื่อทางเดินสารบบในไดรฟ์ที่ต้องการจะเปลี่ยน

2.4.3.19 คำสั่ง REMOVE DIRECTORY

เป็นคำสั่งที่ใช้ลบไดเรกทอรี

แบบฟอร์มคำสั่ง

```
rd [drive:]path
```

เมื่อ drive เป็นชื่อไดรฟ์ที่บรรจุดิสก์ที่มีไดเรกทอรีที่ต้องการลบ

path เป็นชื่อทางเดินสารบบของไดเรกทอรีที่ต้องการจะลบออกไป

2.4.3.20 คำสั่ง DIRECTORY ใช้ค้นหาไดเรกทอรี

เป็นคำสั่งใช้ในการค้นหาไฟล์ หรือไดเรกทอรีในฮาร์ดดิสก์ หรือในฟลอปปีดิสก์โดย
เฉพาะไฟล์และไดเรกทอรีที่มีจำนวนมาก ๆ โดยเฉพาะอย่างยิ่งเพื่อสร้างไดเรกทอรีอยู่ภายในได
เรกทอรีอื่น ๆ

แบบฟอร์มคำสั่ง

```
dir [drive:][path][filename]/s
```

เมื่อ drive เป็นชื่อไดรฟ์ที่ต้องการให้ค้นหา

path เป็นทางเดินสารบบที่ต้องการค้นหา

filename เป็นชื่อไฟล์ที่ต้องการค้นหา

/s เป็นตัวบอกเอ็มเอส ดอส ให้ค้นหาไดเรกทอรีที่กำหนด และ
ค้นไดเรกทอรีอื่น ๆ ที่อยู่ในไดเรกทอรีนั้น

2.4.3.21 คำสั่ง TREE

เป็นคำสั่งที่ใช้แสดงโครงสร้างไดเรกทอรี โดยย่อหน้าไดเรกทอรีเข้าไปใต้ไดเรก
ทอรีต้นกำเนิด และใช้เส้นระดับกับเส้นตั้งเพื่อต่อไดเรกทอรีที่สัมพันธ์กันเข้าด้วยกัน

แบบฟอร์มคำสั่ง

```
tree [drive][path]/f/a
```

เมื่อ drive เป็นชื่อไดรฟ์ที่บรรจุดิสก์ที่ต้องการจะดูโครงสร้างไดเรกทอรี

path เป็นชื่อทางเดินสารบบไปยังโครงสร้างไดเรกทอรีที่ต้องการดู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้ ไม่อนุญาตให้ไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- /f เป็นสวิตช์ที่บอกให้ เอ็มเอส ดอส แสดงผลทั้งโครงสร้างไดเรกทอรี และชื่อ ไฟล์ต่าง ๆ ในแต่ละไดเรกทอรี
- /a เป็นสวิตช์ที่บอกให้ เอ็มเอส ดอส เขียนโคแอสแกรมของโครงสร้างไดเรกทอรี ด้วยคาเรกเตอร์ของ ASCII ธรรมดา

2.4.3.22 คำสั่ง XCOPY

เป็นคำสั่งที่ใช้ในการก๊อปปี้ทั้งไฟล์ และไดเรกทอรีที่ใช้เก็บไฟล์ หรือเมื่อต้องการก๊อปปี้โครงสร้างไดเรกทอรีของดิสก์ไปไว้ในดิสก์อีกแผ่นหนึ่ง

แบบฟอร์มคำสั่ง

```
xcopy source [destination]/d:date/p/s/e/w
```

เมื่อ source เป็นไดรฟ์ทางเดินสารบบ และการกำหนดชื่อของไฟล์ หรือกลุ่มไฟล์ที่ ต้องการก๊อปปี้ไว้

deestination เป็นไดรฟ์ทางเดินสารบบ และการกำหนดไฟล์ของเป้าหมายสำหรับไฟล์ หรือไฟล์ต่าง ๆ ที่จะก๊อปปี้ไว้

/d:date ให้ก๊อปปี้จากไฟล์ต่าง ๆ ที่รายการในไดเรกทอรีของไฟล์เหล่านั้นแสดงวันที่เดียวกับ หรือหลังจากวันที่กำหนดไว้ให้เป็น date

/p แสดงเครื่องหมายพร้อม กับการยืนยันก่อนที่จะก๊อปปี้แต่ละไฟล์ที่กำหนดไว้ใน source

/s ก๊อปปี้ไดเรกทอรีทั้งหมดใน source

/e ใช้กับสวิตช์ /s เพื่อก๊อปปี้ไดเรกทอรีทั้งหมดใน source

แม้ว่าไดเรกทอรีนั้นจะว่างเปล่าก็ตาม

/w ทำให้คำสั่ง Xcopy คอชงนกว่าจะกดคีย์ก่อนเริ่มต้นก๊อปปี้

2.4.3.23 คำสั่ง REPLACE

เป็นคำสั่งที่เหมาะสมสำหรับการก๊อปปี้ไฟล์ที่เลือกไว้ และสามารถทำการทดแทนไฟล์ต่าง ๆ ในไดเรกทอรีหนึ่ง หรือมากกว่าสามารถเพิ่มเติมไฟล์ต่าง ๆ ให้ไดเรกทอรีได้

แบบฟอร์มคำสั่ง

```
replace source destination /a/p/r/s/w/u
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานานับ ไม่นอบอบตให้ไปใช้ประโยชน์อื่นใด
เมื่อ source เป็นไดรฟ์ทางเดินสารบบ และการกำหนดชื่อของไฟล์ หรือไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- กลุ่มไฟล์ต่างๆ ที่กำลังถือปี่ไปไว้ใน destination destination เป็นไครฟ์ และทางเดินที่กำหนดตำแหน่งไฟล์
- /a เพิ่มเติมไฟล์ จากดิสก์ต้นกำเนิดที่ไม่ได้อยู่ในดิสก์ปลายทาง
 - /p ทำให้คำสั่ง replace เกิดเครื่องหมายพร้อมรับการยืนยัน ก่อนที่จะถือปี่แต่ละไฟล์
 - /r ทดแทนไฟล์อ่านเท่านั้นที่ไครเรททอริปลายทาง
 - /s เพิ่มเติมการค้นหาไฟล์ที่ทดแทนได้ในเรททอริทั้งหมด ซึ่งไม่สามารถใช้ /s ร่วมกับ /a ได้
 - /w เป็นการทำให้คำสั่ง replac เพื่อคอยให้กดคีย์ก่อนทดแทนไฟล์ต่าง ๆ
 - /u Update ไฟล์ต่าง ๆ ที่ไครเรททอริปลายทางถ้าไฟล์เหล่านั้น เก่ากว่า ไฟล์เดียวกันในไครเรททอริ ต้นกำเนิดไม่สามารถใช้ /u ร่วมกับ /s ได้

2.4.3.24 คำสั่ง BACKUP

จะทำงานร่วมกับคำสั่งรีสทอร์ (Restore) เพื่อถือปี่ไฟล์จากดิสก์หนึ่งไปไว้ที่อีกดิสก์หนึ่ง คำสั่งแบ็คอัพ จะถือปี่ไฟล์ต่าง ๆ แล้วปล่อยต้นฉบับไว้ในดิสก์ต้นกำเนิด โดยทั่วไปแล้วจะใช้เก็บไฟล์เก่า หรือไฟล์ที่สำคัญไว้ชั่วคราวคำสั่งแบ็คอัพ ไม่สามารถใช้ถือปี่ไฟล์แล้วใช้คำสั่ง Copy หรือ xcopy เพื่อส่งไฟล์นั้นกลับคืนไปในดิสก์ได้

แบบฟอร์มคำสั่ง

```
backup source destination /s/m/a/f[:size]/l[:log]
```

เมื่อ source เป็นไครฟ์ทางเดินสารบบ และการกำหนดไฟล์ของไฟล์หรือไฟล์ต่าง ๆ เพื่อถือปี่ไว้

deestination คือไครฟ์ที่ต้องการทำสำเนาไฟล์ไว้จะต้องกำหนด destination ไว้เสมอ

/s เพิ่มเติมถือปี่ ให้กับไฟล์ในไครเรททอริต่าง ๆ อยู่ใต้ระดับไครเรททอริ ที่กำหนดไว้ใน source

/m สำเนาไฟล์เฉพาะไฟล์ที่มีการแก้ไขนับตั้งแต่ไฟล์สำเนาครั้งล่าสุดเท่านั้น

/a เพิ่มไฟล์ที่สำเนาไว้ให้กับดิสก์เป้าหมาย

/f[:size] ทำการฟอร์แมตแผ่นดิสก์ปลายทาง ถ้ายังไม่ได้ฟอร์แมต (size) คือการกำหนดขนาดความจุของดิสก์
/l[:log] ให้สร้างไฟล์ลงบันทึกรายชื่อทางเดินสารบบ และชื่อไฟล์ของแต่ละ ไฟล์ที่คำสั่งทำการสำเนาไว้

2.4.3.25 คำสั่ง RESTORE

จะใช้ร่วมกับคำสั่ง backup จะส่งไฟล์ที่คำสั่งแบ็คอัพที่เก็บไว้กลับไปยังดิสก์ และไดเรกทอรีที่สำเนาไฟล์นั้นมา

แบบฟอร์มคำสั่ง

restore source destination /s/p/n/d

เมื่อ source เป็นไดรฟ์ที่บรรจุดิสก์ สำหรับสำเนาซึ่งเป็นแผ่นดิสก์ที่ต้องการใช้ในการกลับคืนไฟล์ destination เป็นไดรฟ์ทางเดินสารบบ และการกำหนดไฟล์ของไฟล์ หรือไฟล์ต่าง ๆ ที่ต้องการกลับคืนไฟล์

/s

กลับคืนไฟล์ต่าง ๆ ไปยังไดเรกทอรีทั้งหมดที่อยู่ในระดับต่ำกว่าไดเรกทอรีกำหนด

/p

ทำให้คำสั่งรีสตอร์ ขอการยืนยันก่อนกลับคืนไฟล์

/n

กลับคืนไฟล์ก็ต่อเมื่อไฟล์เหล่านั้นอยู่ในดิสก์แผ่นสำเนาเท่านั้น

/d

ทำให้คำสั่งรีสตอร์ แสดงผลชื่อไฟล์ต่าง ๆ ในดิสก์แผ่นสำเนาที่ตรงกับการกำหนดชื่อไฟล์

2.4.3.26 คำสั่ง PROMPT

เป็นคำสั่งที่สามารถเลือกใช้เครื่องหมายพร้อมรับคำสั่งเป็นอย่างอื่น ตามความต้องการ โดยใช้คำสั่ง prompt

แบบฟอร์มคำสั่ง

prompt definition

เมื่อ definition เป็นชุดคานรแรกเตอร์ที่กำหนดเครื่องหมายพร้อมคำสั่ง

โดยใช้รหัสคือไปนี้ จะใช้ตัวเดียว หรือหลายตัวก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

%h	แสดงผลเป็น
%b	!
%d	วันที่ในขณะนั้น
%e	ค่าแรกเตอร์ EEscape
%g	>
%h	ค่าแรกเตอร์ Backspace
%l	<
%n	ไครฟ์ที่ทำงานอยู่
%p	ไครฟ์ และไครเรกทอรี ที่ทำงานอยู่
%q	=
%t	เวลาในขณะนั้น
%v	เวอร์ชันของ MS-DOS
%-	บรรทัดใหม่

2.4.3.27 คำสั่ง Set

เป็นคำสั่งที่ใช้ไว้ในไฟล์ AUTOEXE.BAT เพื่อกำหนดตัวแปรของสภาพแวดล้อมที่ใช้ อยู่เป็นประจำในแต่ละครั้งที่สแตร์ตคอมพิวเตอร์

แบบฟอร์มคำสั่ง

```
set variable = definition
```

เมื่อ variable เป็นชื่อตัวแปรของสภาพแวดล้อมที่ต้องการกำหนด

definition เป็นคำอธิบายที่ต้องการกำหนดให้กับตัวแปรของสภาพแวดล้อม

2.4.3.28 คำสั่ง MEM

เป็นคำสั่งที่ใช้ตรวจสอบหน่วยความจำของคอมพิวเตอร์ เช่น มีหน่วยความจำเหลือไว้ให้ใช้งานได้เท่าไร หน่วยความจำให้ใช้ในการโหลดโปรแกรม ทัชบุปรแกรม และโปรแกรมอื่น ๆ ได้เท่าไร และมีหน่วยความจำให้เท่าไรที่จะใช้กับโปรแกรมประยุกต์

แบบฟอร์มคำสั่ง

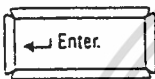
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 mem/c
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ /c จะทำรายงานที่บอกให้ผู้ใช้ทราบชื่อ และขนาดของโปรแกรมในหน่วยความจำ

2.4.4 คีย์สำคัญต่าง ๆ ในการควบคุมเอ็มเอส ดอส

คีย์ (หรือคีย์ผสม)

ผลลัพธ์ที่ได้สำหรับ MS-DOS



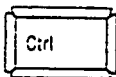
บอกให้เอ็มเอส ดอส ทราบว่าได้พิมพ์คำสั่งเสร็จแล้ว และต้องการให้คำสั่งทำงานเดิยวนเลขเอ็มเอส ดอส จะคอยให้ทำงานอย่างใดอย่างหนึ่ง คือ คอยจนกว่าจะกดคีย์ Enter



(ชื่อมาจาก Escap) เป็นคีย์ "ยกเลิก" สำหรับเอ็มเอส ดอส ก่อนที่จะคีย์ Enter คีย์ Esc จะยกเลิกคำสั่งที่เพิ่งพิมพ์ลงไป เครื่องหมายพร้อมรับคำสั่งของเอ็มเอส ดอสและเริ่มต้นใหม่อีก



ชกคีย์ตัวอักษร และคีย์ตัวเลขจากตัวพิมพ์เล็กเป็นตัวพิมพ์ใหญ่สำหรับเอ็มเอส ดอส ปกติตัวพิมพ์ใหญ่และตัวพิมพ์เล็กจะไม่มี ความแตกต่างกันเลย แต่คีย์นี้มีความสำคัญต่อคานรกดเตอร์ดอกจัน(*) และ เครื่องหมายคำถาม (?) เพราะว่าทั้งสองคานรกดเตอร์นี้ จะต้องพิมพ์ด้วยคีย์ Shift



(ชื่อมาจาก Control) เป็นคีย์ที่ใช้ "ชก" บางคีย์เพื่อให้เกิดความหมายแตกต่างออกไป โดยทั่วไปแล้วคีย์ Ctrl ตามชื่อคีย์ จะส่งสัญญาณควบคุมโปรแกรมบางชนิดไปให้เอ็มเอส ดอส หรือไปให้โปรแกรมอื่นถ้ากดคีย์ Ctrl ตามลำพังจะไม่เกิดอะไรขึ้น สำหรับเอ็มเอส ดอส การกดคีย์ Ctrl จะต้องทำการกดกับคีย์ ฟังก์ชันอื่นด้วย เช่น Ctrl Break และ Ctrl-Alt-Del

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คีย์ (หรือคีย์ผสม)

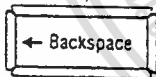
ผลลัพธ์ที่ได้สำหรับ MS-DOS



ทำให้ เอ็มเอส ดอส หยุดการแสดงผลที่ขาว ๆ ไว้ชั่วคราวเพื่อให้มีเวลาอ่านสิ่งที่ปรากฏในจอภาพได้ทัน เมื่อกดคีย์ใด ๆ ก็จะทำให้สัญญาณแก่ เอ็มเอส ดอส ให้กลับไปแสดงผลข้อมูลที่ยังอยู่ต่อไปคีย์นี้ปกติจะเขียนว่า Pause ไว้ด้านบนและเขียนว่า Break ไว้ด้านล่างของแป้นคีย์การทำหน้าที่หยุดในเอ็มเอส ดอส จะทำต่อเมื่อกดคีย์ Ctrl ด้วยเท่านั้นดังอธิบายไว้ข้างล่างนี้



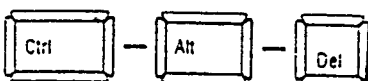
บอกให้ เอ็มเอส ดอส หยุดและยกเลิกการปฏิบัติของคำสั่งเป็นการกดคีย์ผสมด้วยการกดคีย์ Ctrl ไว้แล้วกดคีย์ Break การกดคีย์ Ctrl-Break จะยกเลิกคำสั่งเหมือนคีย์ Esc แต่จะหยุดและยกเลิกคำสั่งที่ได้บอก เอ็มเอส ดอส ไปให้ปฏิบัติแล้วได้คีย์ Esc นั้นเพียงแต่ยกเลิกคำสั่งที่ได้พิมพ์ลงไปแล้วเท่านั้น แต่ยังไม่ได้ออก เอ็มเอส ดอส ให้ปฏิบัติตามคำสั่งอีกด้วย



เลื่อนเคอร์เซอร์ไปทางซ้ายหนึ่งช่อง ในขณะที่เคียวกันก็ลบ คาแรกเตอร์ใด ๆ ที่เคอร์เซอร์เคลื่อนที่ผ่านไปด้วย



(ชื่อมาจาก Numeric Lock) เป็นคีย์ปิด เปิดเหมือนสวิตช์ไฟซึ่งใช้เปลี่ยนชุดแป้นคีย์ตัวเลขไปมาระหว่างแป้นป้อนตัวเลขแบบเป็นวิธีเร็วที่สุดในการสคาร์ตใหม่ในบางสถานการณ์โดยเฉพาะอย่างยิ่งถ้าเกิดมีการทำงานของโปรแกรมประยุกต์ ในคอมพิวเตอร์ เครื่องคิดเลขกับแป้นเลื่อนเคอร์เซอร์ ดังแสดงไว้ด้วยลูกศรและคำ เช่น Home และ End ที่เขียนไว้บนแป้นคีย์



สคาร์ตคอมพิวเตอร์ใหม่การกดคีย์ผสมนี้หากที่จะเป็นการกดโดยไม่ได้ตั้งใจ เพราะหามาความว่าจะต้องกดคีย์ทั้งสามนี้พร้อม ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คีย์ (หรือคีย์ผสม)

ผลลัพธ์ที่ได้สำหรับ MS-DOS

กันซึ่งมีเหตุผลสำคัญจะต้องจดจำไว้สองอย่างคือ อย่างแรกการ
 สตาร์ทคอมพิวเตอร์ใหม่ หมายความว่าเราจะสูญเสียข้อมูลทั้ง
 หมดที่ตัวมิได้เซฟไว้ในดิสก์ไม่มีทางเลยที่จะกู้ข้อมูลที่สูญหายไปแบบ
 นี้กลับคืนมาได้ ข้อดีของการกดคีย์ผสม Ctrl -Alt -Del คือ
 เป็นวิธีเร็วที่สุดในการสตาร์ทใหม่ในบางสถานการณ์โดยเฉพาะ
 อย่างยิ่งถ้าเกิดมีการทำงานของโปรแกรมประยุกต์ ในคอมพิวเตอร์
 ของเราแฮงค์ (hang) ซึ่งเป็นเหตุการณ์ไม่ค่อยจะเกิด
 บ่อยนักแต่ก็จะมีโอกาสเกิดขึ้นได้ และไม่มีทางอื่นที่จะหยุด และ
 ออกจากโปรแกรมนั้นมาได้ Ctrl-Alt-Del จะเป็นวิธีเดียวที่
 สามารถหนีออกจากเหตุการณ์ดังกล่าวนี้ได้ แม้จะต้องสูญเสียงาน
 ที่ทำไว้โดยที่ยังไม่ได้เซฟไว้ก็ตาม

2.5. การแสดงผลภาษาไทยในกราฟฟิคโหมด ด้วยเทอร์โบซี

ในการแสดงผลอักษรภาษาไทยลงในจอคอมพิวเตอร์ในลักษณะกราฟฟิคโหมดนั้นมีอยู่ด้วยกัน
 หลายวิธี แต่ก่อนที่จะกล่าวถึงวิธีการแสดงผล เรามาทำความเข้าใจกับรหัสภาษาไทยกันก่อน ตา
 รางรหัสภาษาไทยที่นิยมใช้กันอยู่ มีอยู่ด้วยกัน 2 ตารางคือ

- มาตรฐาน ส.ม.อ. หรือสถาบันมาตรฐานอุตสาหกรรม
- มาตรฐานเกษตร

ดังจะเห็นได้จากตาราง จะเห็นว่าลำดับของตัวอักษรบางตัวไม่ตรงกัน

	8	9	A	B	C	D	E	F
0	ร			๑	๒	๓	๔	๕
1	๖		๗	๘	๙	๐	๑	๒
2	๓	๔	๕	๖	๗	๘	๙	๐
3	๔	๕	๖	๗	๘	๙	๐	๑
4	๕	๖	๗	๘	๙	๐	๑	๒
5	๖	๗	๘	๙	๐	๑	๒	๓
6	๗	๘	๙	๐	๑	๒	๓	๔
7	๘	๙	๐	๑	๒	๓	๔	๕
8	๙	๐	๑	๒	๓	๔	๕	๖
9	๐	๑	๒	๓	๔	๕	๖	๗
A		๑	๒	๓	๔	๕	๖	๗
B	๑	๒	๓	๔	๕	๖	๗	๘
C	๒	๓	๔	๕	๖	๗	๘	๙
D	๓	๔	๕	๖	๗	๘	๙	๐
E	๔	๕	๖	๗	๘	๙	๐	๑
F	๕	๖	๗	๘	๙	๐	๑	๒

ตารางที่ 2.1 ตารางอักษร สมอ.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	8	9	A	B	C	D	E	F
0		๐		๒	๒	๒	.	๒
1		๑	๓	๓	๓	๓	.	๓
2		๒	๔	๔	๔	๔	.	๔
3		๓	๕	๕	๕	๕	.	๕
4		๔	๖	๖	๖	๖	.	๖
5		๕	๗	๗	๗	๗	.	๗
6		๖	๘	๘	๘	๘	.	๘
7		๗	๙	๙	๙	๙	.	๙
8		๘	๐	๐	๐	๐	.	๐
9	๙	๑	๑	๑	๑	๑	.	๑
A	๐	๒	๒	๒	๒	๒	.	๒
B	๑	๓	๓	๓	๓	๓	.	๓
C	๒	๔	๔	๔	๔	๔	.	๔
D	๓	๕	๕	๕	๕	๕	.	๕
E	๔	๖	๖	๖	๖	๖	.	๖
F	๕	๗	๗	๗	๗	๗	.	๗

ตารางที่ 2.2 ตารางอักขระ เกษตร

เมื่อเราทราบถึงลักษณะของตารางอักขระแล้ว เราจะได้มาพิจารณาถึงวิธีในการนำอักขระออกจากรูปภาพในลักษณะกราฟิคโหมด

ในการทำให้ตัวอักขระปรากฏบนหน้าจอภาพในกราฟิคโหมด จะต้องใช้คำสั่งต่าง ๆ ในกราฟิคโหมด เช่น Line (คำสั่งในการลากเส้น) Putpixel (คำสั่งลงจุด) bar (คำสั่งวาดรูปสี่เหลี่ยม) ซึ่งคำสั่งต่าง ๆ มีวิธีการใช้อย่างไรได้กล่าวถึงไปแล้ว

ในการแสดงรูปแบบของอักขระออกที่จอภาพ คือการวาดรูปตัวอักขระนั้นลงบนจอภาพโดยนำข้อมูลมาจากไฟล์ต่างหาก

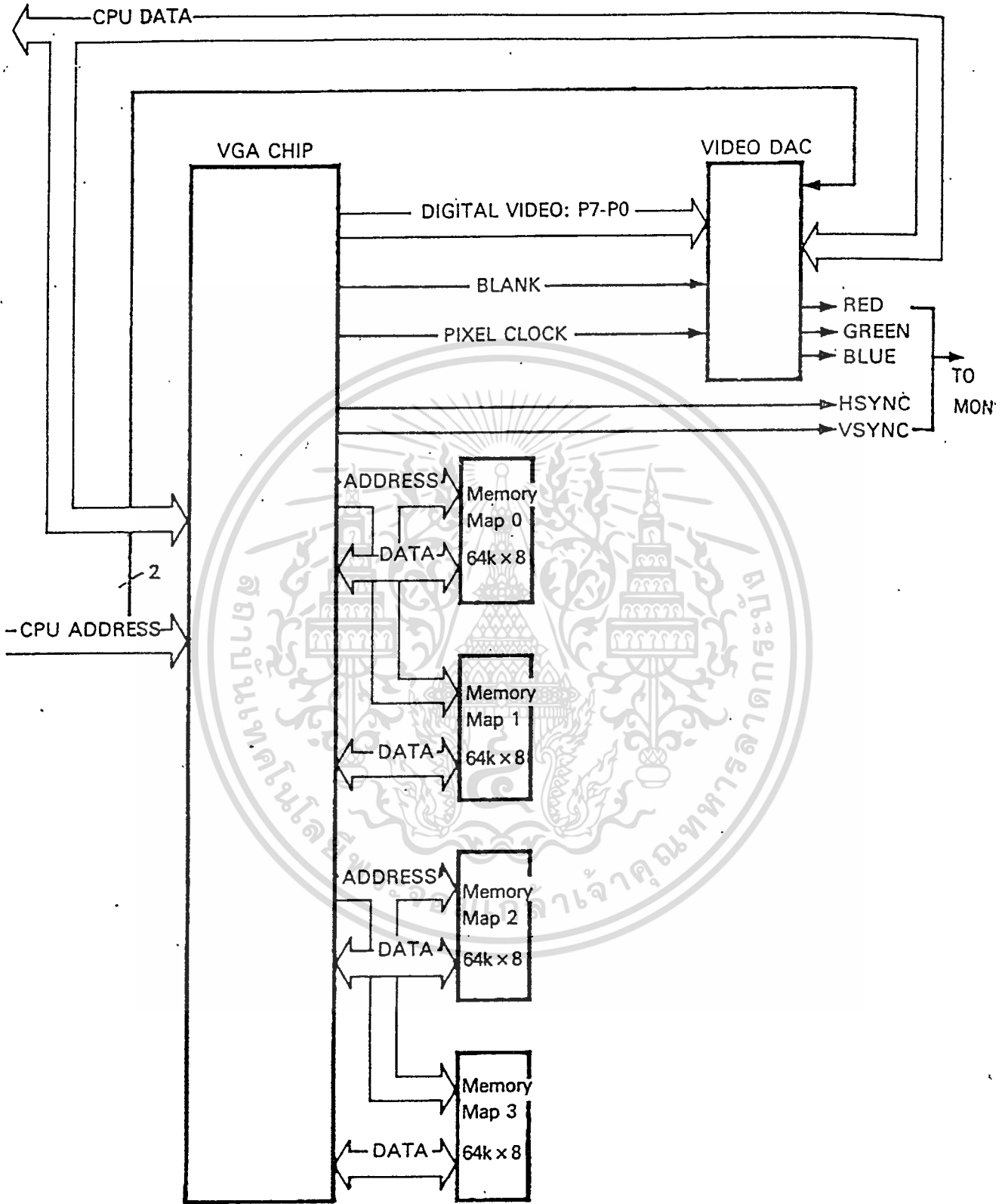
2.5.1 โครงสร้างของการ์ด VGA

เทคโนโลยีการแสดงผลบนเครื่องคอมพิวเตอร์ส่วนบุคคลนับว่ามีการพัฒนาขึ้นมา มากที่เดียวจากเดิมซึ่งเป็นการแสดงผลบนจอโมโนโครมซึ่งใช้การ์ดโมโนโครมอะแดปเตอร์ หรือการ์ด เซอร์คิลิส แล้วพัฒนาต่อมาเป็นการแสดงผลบนจอสีที่เรียกว่า Color Graphics Adapter (CGA) และ Enhanced Graphic Adapter (EGA) ซึ่งล้วนแล้วแต่เป็นระบบแสดงผลที่เป็นสัญญาณดิจิทัล

นั่นก็คือ สัญญาณที่ออกจากการ์ดแสดงผลจะไปควบคุมจอภาพนั้นเป็นสัญญาณดิจิทัลในระบบดิจิทัลการ์ด EGA จะมีความละเอียดสูงสุดคือมีจำนวนจุด (pixel) เท่ากับ 640 x 350 จุด และสามารถแสดงสีได้ทั้งหมด 64 สี ต่อมาไอบีเอ็มได้พัฒนาระบบแสดงผลแบบใหม่ขึ้นมาเรียกว่า Video Graphic Array (VGA) ซึ่งใช้สัญญาณอะนาลอกควบคุมจอแสดงผล ทำให้แสดงจำนวนสีได้มากขึ้นเพราะว่าคอมพิวเตอร์นั้นของสัญญาณอะนาลอกมีมากกว่าสัญญาณดิจิทัล ระบบ VGA จึงสามารถแสดงภาพสีได้ถึง 256 สีพร้อมกันจากจำนวนสีที่เป็นไปได้ทั้งหมด 256 x 1024 สี และ ยังมีความละเอียดสูงถึง 640 x 480 จุด ทำให้สามารถแสดงภาพได้เหมือนจริงยังมีประโยชน์อย่างมากในการใช้เครื่องคอมพิวเตอร์กับงานด้านต่าง ๆ อาทิเช่น งานออกแบบด้านอุตสาหกรรม (CAD/CAM) งานทางการแพทย์งานทางด้านการประมวลผลของรูปภาพ (Image Processing) ซึ่งสามารถที่ประยุกต์ใช้กับงานต่าง ๆ ได้อย่างมากมายเนื่องจาก ข้อดีของการแสดงผลแบบ VGA เครื่องไอบีเอ็มจึงรวบรวมการ์ดแสดงผลระบบ VGA ลงบนเมนบอร์ดของเครื่อง IBM PS/2 เช่น PS/2 model 50, 60, 70 จึงถือได้ว่าเป็นมาตรฐานการแสดงผลของเครื่อง IBM PS/2

ในปัจจุบันมีผู้ผลิตตราอื่นที่ทำการ์ด VGA ให้มีความละเอียดสูงกว่า VGA ของไอบีเอ็ม โดยเรียกว่า super VGA ซึ่งมีความละเอียดถึง 800 x 600 จุดหรือ 1024x768 จุดซึ่งจะต้องใช้กับจอแสดงผลแบบมัลติซิงค์ หรือจอ VGA ความละเอียดสูงระบบแสดงผลแบบ VGA หรือที่รู้จักกันคือว่าการ์ด VGA นั้นประกอบด้วยส่วนสำคัญอยู่ 3 ประการด้วยกันคือ

- หน่วยความจำของส่วนแสดงผล (Video Memory)
- ชิพ VGA
- ตัวเปลี่ยนสัญญาณดิจิทัลให้เป็นสัญญาณอะนาลอก (Video DAC)



รูปที่ 2.1 แสดงส่วนประกอบหลักของการ์ด VGA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
จากรูปที่ 2.1 แสดงส่วนประกอบหลักของการ์ด VGA ซึ่งมีส่วนประกอบดังนี้
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.2 หน้าที่ความจำแสดงผล

หน้าที่ความจำของการ์ด VGA ตามมาตรฐานไอบีเอ็มจะมีจำนวน 256 กิโลไบต์เป็นไดนามิกแรมที่ถูกแบ่งออกเป็น 4 plane ข้อมูลที่ถูกใช้แสดงผลจะถูกเก็บอยู่ในหน่วยความจำนี้ ซึ่งรูปแบบของการเก็บจะแตกต่างกันไปขึ้นอยู่กับโหมดการแสดงผล ตัวอย่างเช่น การแสดงผลในเท็กซ์โหมดเพลน 0 ซึ่งเป็นหน่วยความจำแอสเคอร์สคูจะเก็บข้อมูลที่เป็นรหัสแอสกี หน่วยความจำแอสเคอร์สคูซึ่งอยู่ในเพลน 1 จะเก็บค่าแอสเคอร์สคู ส่วนหน่วยความจำเพลน 2 จะสำรองไว้ใช้สำหรับเก็บค่าคาแรกเตอร์เซนเนอเรเตอร์ หน่วยความจำเพลน 3 จะไม่ถูกใช้ ส่วนโหมดกราฟฟิกก็จะแตกต่างกันไปในปัจจุบันการ์ดของ VGA ที่ถูกผลิตขึ้นนั้น มักจะมีความละเอียดการแสดงผลสูงกว่ามาตรฐานไอบีเอ็ม ซึ่งจำเป็นจะต้องใช้หน่วยความจำเพิ่มมากขึ้นเพื่อเก็บข้อมูลภาพที่มากขึ้นนั่นเอง

2.5.3 หน้าที่ความจำในเท็กซ์โหมด

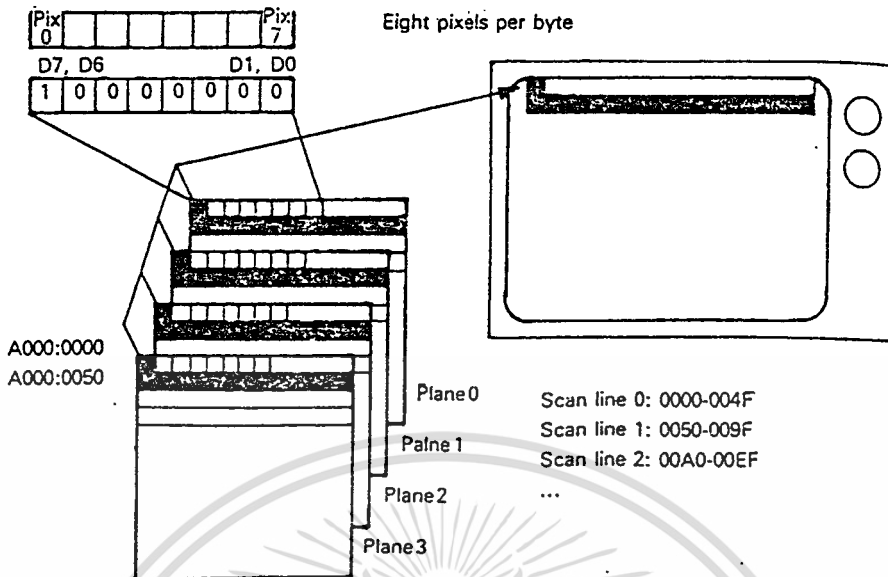
การแสดงผลในเท็กซ์โหมดจะ มีความซับซ้อนน้อยกว่าการแสดงผล ในโหมดกราฟฟิกมาก เพราะว่าการจัดการกับรหัสแอสกีมีใช้จัดการกับจุดใดจุดหนึ่งบนจอภาพมาตรฐานของเท็กซ์โหมดแบ่งออกเป็น 25 บรรทัด 40 คอลัมน์ หรือ 80 คอลัมน์ต่อบรรทัด ในกรณีที่เป็น 80 คอลัมน์ต่อบรรทัดใน 1 จอภาพสามารถแสดงตัวอักษรได้ทั้งสิ้น 2000 ตัว แต่การแสดงผลของตัวอักษร 1 ตัว ต้องใช้หน่วยความจำ 2 ไบต์ ดังนั้นใน 1 จอภาพจะต้องใช้หน่วยความจำทั้งสิ้น 4000 ไบต์ แต่หน่วยความจำแสดงผลแบ่งออกเป็นเพจ ๆ ละ 4096 ไบต์ซึ่งจะเหลือที่ว่าง 96 ไบต์ที่ไม่ถูกใช้งาน

2.5.4 หน้าที่ความจำในกราฟฟิกโหมด

ในโหมดของกราฟฟิกจุด ๆ หนึ่งบนจอภาพจะแทนด้วยข้อมูลที่มีจำนวนบิตต่าง ๆ กัน เช่น ในโหมด CGA 2 สี (mode 6) จะใช้ข้อมูล 1 บิตในการแสดงจุดหนึ่งจุด (8 จุดต่อไบต์) ซึ่งในที่สุดก็จะขอลงมาถึงโหมด Enhanced Color Graphics (mode 10) เท่านั้นในโหมด 10H เป็นที่นิยมสำหรับการใช้งานต่าง ๆ เพราะว่ามีขนาดละเอียดสูงถึง 640 x 350 จุด และแสดงสีได้ 16 สี พร้อมกันในการแสดงจุด ๆ หนึ่งบนจอภาพจะใช้ข้อมูล 4 บิตโดยแต่ละบิตมาจากแต่ละเพลนของหน่วยความจำรูปที่ 2.2 แสดงการจัดการของหน่วยความจำการแปลงจากตำแหน่งของจุดบนจอภาพไปเป็นตำแหน่งของบิตในหน่วยความจำ โดยที่ x คือ โคออร์ดิเนตตามแนวนอนช่วงจาก 0-639 และ y เป็นโคออร์ดิเนตตามแนวตั้งช่วงจาก 0-349 จะแปลงได้โดยการใช้สูตร

$$\text{ตำแหน่งไบต์ที่} = y * 80 + x / 8$$

$$\text{ตำแหน่งบิตที่ (0-7)} = y - (x \bmod 8)$$



รูปที่ 2.2 รูปแบบการเก็บข้อมูลในหน่วยความจำสำหรับโหมดกราฟฟิก

2.5.5 ชิพ VGA

ระบบการแสดงผลแบบ VGA มาตรฐานของไอบีเอ็มนั้นใช้ชิพ 82706 Video Graphics Array ซึ่งคอมแพคติเบิ้ลระดับไบออส กับระบบเดิมคือ EGA, CGA และ MDA บล็อกไดอะแกรมของชิพ VGA ดังแสดงดังรูปที่ 2.3

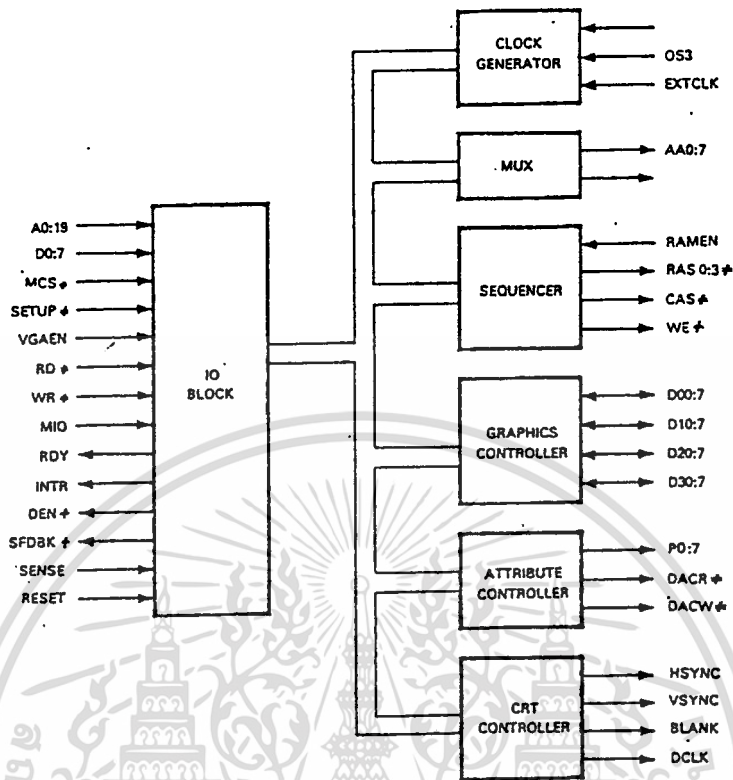
82706 VGA Controller เป็นตัวอินเทอร์เฟสระหว่างชิพซึ่งในที่นี้อาจจะเป็น 8088, 80286 หรือ 80386 กับหน่วยความจำของการแสดงผล (video memory) และเป็นตัวส่งข้อมูลไปยัง Video DAC (Digital to Analog Converter) ทำการแปลงข้อมูลขนาด 8 บิต ซึ่งเป็นสัญญาณดิจิทัลไปเป็นสัญญาณอะนาลอก เพื่อส่งให้จอแสดงผลต่อไป

การเข้าถึงหน่วยความจำของการแสดงผลจะต้องผ่านตัว 82706 เสมอซึ่งมีข้อดีก็คือชิพสามารถเข้าถึงหน่วยความจำขณะที่ทำการรีเฟรชหน่วยความจำซึ่งทำให้ชิพเขียนหรืออ่านจากหน่วยความจำได้ โดยไม่จำเป็นต้องรอให้ถึงช่วงเวลาการย้อนกลับ (Retrace) ของจอภาพ

82706 ใช้กับหน่วยความจำได้ถึง 256 กิโลไบต์ โดยสามารถโปรแกรมแอดเดรสได้ 3 ตำแหน่งซึ่งโดยทั่วไปมักจะขึ้นกับโหมดการแสดงผลจากรูปที่ 2.11 จะเห็นได้ว่าภายในชิพ 82706 จะมีตัวควบคุมอยู่ 4 ตัว คือ

2.5.6 CRT controller

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรู้ใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ทำหน้าที่กำเนิดสัญญาณที่ใช้ควบคุมการทำงานของจอภาพ เช่น สัญญาณซิงค์ตามแนวนอน
ไม่ว่ากรณีใดๆทั้งสิ้น ยกเว้นห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีกรู้ ปรินต์



รูปที่ 2.3 บล็อกโดยะแกรมของชิพ VGA

สัญญาณซึ่งก็ตามแนวตั้งสัญญาณแบดิ่งกึ่ง และแอดเดรสสำหรับกการรีเฟรชหน่วยความจำควบคุม CRT controller มีรีจิสเตอร์ 25 ตัวซึ่งมีบางตัวที่คอมแพคติเบิ้ลกับ 6845 CRT controller ที่มีอยู่บนการ์ดแสดงผลแบบโมโนโครมรีจิสเตอร์ต่าง ๆ ของ CRT controller แสดงดังตารางในตารางที่ 2.2

Register Name	R/W	Index	Read Port	Write port
CRT controller Address	R/W		0374	0374
Horizontal Total	R/W	00	0375	0375

Register Name	R/W	Index	Read Port	Write port
Horizontal Display Enable	R/W	01	0375	0375
End Horizontal Blanking	R/W	03	0375	0375
Start Horizontal Blanking	R/W	02	0375	0375
End Horizontal Blanking	R/W	03	0375	0375
Start Horizontal Retrace Pulse	R/W	04	0375	0375
End Horizontal Retrace	R/W	05	0375	0375
Vertical Total	R/W	06	0375	0375
Overflow	R/W	07	0375	0375
Preset Row Scan	R/W	08	0375	0375
Maximum Scan Line	R/W	09	0375	0375
Cursor Start	R/W	0A	0375	0375
Cursor End	R/W	0B	0375	0375
Start Address High	R/W	0C	0375	0375
Start Address Low	R/W	0D	0375	0375
Cursor Location High	R/W	0E	0375	0375
Cursor Location Low	R/W	0F	0375	0375
Vertical Retrace Start	R/W	10	0375	0375
Vertical Retrace End	R/W	11	0375	0375
Offset	R/W	13	0375	0375
Underline Location	R/W	14	0375	0375
Start Vertical Blank	R/W	15	0375	0375
End Vertical Blank	R/W	16	0375	0375
CRTC Mode Control	R/W	17	0375	0375

2.5.7 Graphics Controller

เป็นตัวกลางเชื่อมทางเดินข้อมูล ระหว่างหน่วยความจำของการแสดงผลกับตัวโปรเซสเซอร์หลัก และยังเป็นที่ที่จะเชื่อมระหว่างตัวโปรเซสเซอร์หลัก กับแอดทริบิวต์คอนโทรลเลอร์ในสภาวะปกติ ข้อมูลจากโปรเซสเซอร์จะสามารถถูกส่งไปยังหน่วยความจำ โดยการทะลุกราฟฟิคคอนโทรลเลอร์ได้โดยตรงแต่ในกรณีอื่นกราฟฟิคคอนโทรลเลอร์มีฟังก์ชันช่วยในการวาดรูปกราฟฟิค โดยนำข้อมูลที่ผ่านตัวมันมากระทำฟังก์ชันทางลอจิก ก่อนที่จะเขียนลงไปยังหน่วยความจำภายในกราฟฟิคคอนโทรลเลอร์มีรีจิสเตอร์อยู่ 9 รีจิสเตอร์ ดังแสดงในตารางที่ 2.3 ซึ่งสามารถสรุปการทำงานได้ดังรูปที่ 2.4 จากรูปจะเห็นว่าข้อมูลที่เราเรียกว่า Latched Data มีอยู่ 4 ไบต์ (แต่ละไบต์มาจากแต่ละเฟรม) ข้อมูลเหล่านี้ถูกเก็บอยู่ในที่ที่หนึ่ง ซึ่งจะเกิดขึ้นทุกครั้งที่มีการอ่านข้อมูลจากหน่วยความจำแสดงผล ข้อมูลจากโปรเซสเซอร์ (Processor Data) จะผ่านการหมุน (Rotate) ซึ่งอาจจะเป็นการหมุนตั้งแต่ 0 บิต (ไม่มีการหมุน) ถึง 7 บิต ทั้งนี้ขึ้นอยู่กับค่าในรีจิสเตอร์ Data Rotate (บิตที่ 0-2) จากนั้นจะถูกนำมากระทำฟังก์ชันทางลอจิก เช่น OR, AND หรือ XOR กับ Latched Data การทำฟังก์ชันใดนั้นขึ้นอยู่กับค่าในรีจิสเตอร์ Data Rotate (บิตที่ 3-4) ฟังก์ชันทางลอจิกแสดงดังตารางที่ 2.4

จากนั้นข้อมูลจะผ่านรีจิสเตอร์ 2 ตัวคือ Set/Reset และ Set/Resetenable แต่ละรีจิสเตอร์จะใช้เพียง 4 บิต แต่ละบิตแทนแต่ละเฟรมจากรูปบิตที่ 2 และที่ 3 ของรีจิสเตอร์ Set/Reset Enable เป็น 0 ดังนี้ ข้อมูลเฟรมที่ 2 และ 3 จะผ่านไปได้โดยตรง ส่วนบิตที่ 0 และ 1 มีค่าเป็น 1 ต้องพิจารณาในรีจิสเตอร์ Set/Reset เช่น บิตที่ 0 มีค่าเป็น 1 ดังนั้นเฟรมที่ 0 จึงมีค่าเป็น 1 ทั้ง 8 บิต ส่วนบิตที่ 1 มีค่าเป็น 0 เฟรมที่ 1 จึงมีค่าเป็น 0 ทั้ง 8 บิตจากนั้นข้อมูลจะผ่านรีจิสเตอร์ Bit Mask ถ้าค่าในรีจิสเตอร์เป็น 1 ก็จะไม่มีการเปลี่ยนแปลง ค่าของข้อมูล แต่ถ้าค่าในรีจิสเตอร์เป็น 0 Latched Data แทนที่จะเป็นข้อมูลเดิมรีจิสเตอร์ Bit Mask เป็นรีจิสเตอร์สุดท้ายของกราฟฟิคคอนโทรลเลอร์ที่จะทำให้ข้อมูลมีการเปลี่ยนแปลง หลังจากนั้นแล้วข้อมูลนี้ยังมิได้ถูกเขียนลงไปยังหน่วยความจำโดยตรง ซึ่งจะต้องผ่านตัวควบคุมอีกตัวหนึ่งที่เรียกว่า Sequencer ซึ่งจะได้อีกต่อไป ตัวอย่างของการกระทำฟังก์ชันทางลอจิกของกราฟฟิคคอนโทรลเลอร์แสดงดังรูปที่ 2.5 จากรูปจะเห็นว่าเดิมมีรูปร่างของ A และ B อยู่เราสามารถทำให้เกิดรูปร่างอื่น ๆ ขึ้นได้ ซึ่งมีประโยชน์ในการทำให้เกิดภาพต่าง ๆ ในโหมดกราฟฟิค

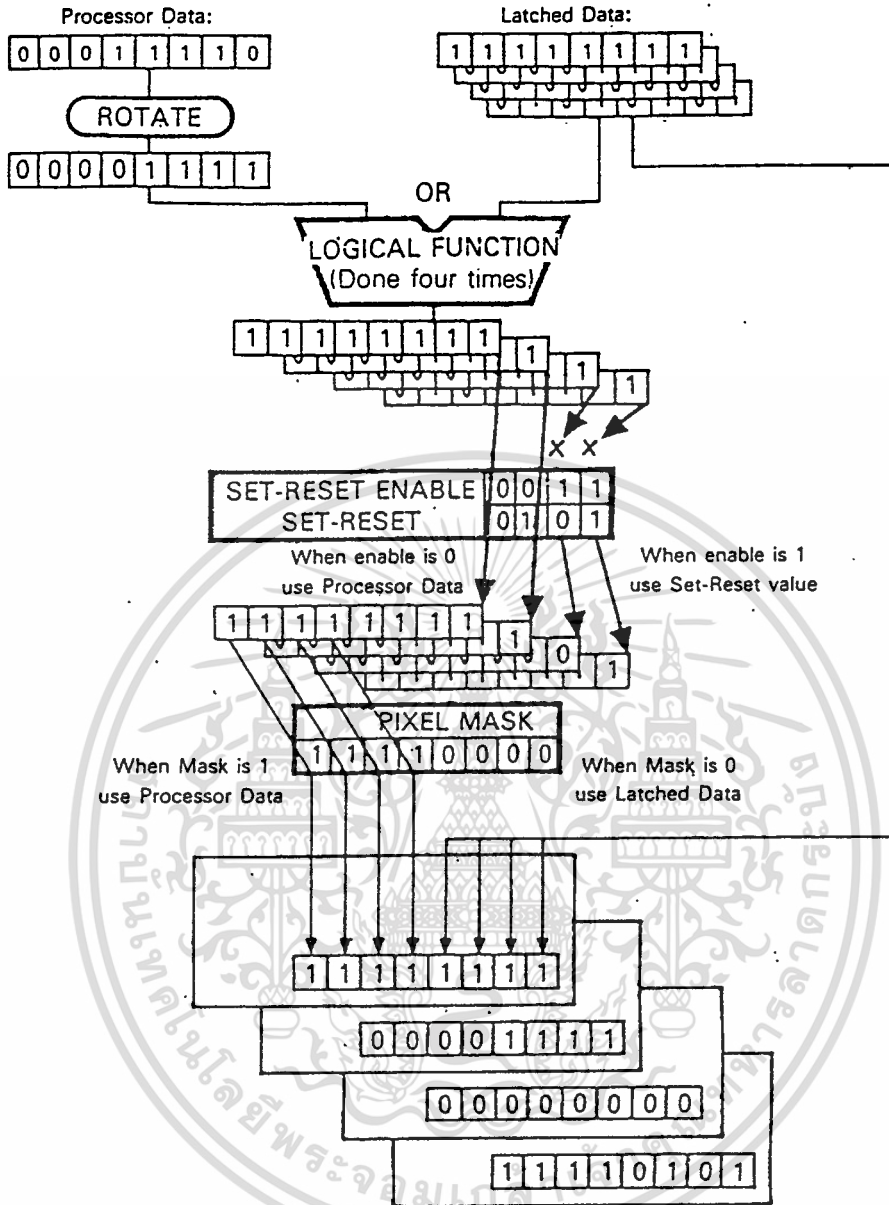
Register Name	R/W	Index	Read Port	Write Port
Graphics Address	R/W		03CE	03CE
Set/Reset	R/W	00	03CE	03CE
Enable Set/Reset	R/W	01	03CE	03CE
Color Compare	R/W	02	03CE	03CE
Data Rotate	R/W	03	03CE	03CE
Read Map Select	R/W	04	03CE	03CE
Graphics Mode	R/W	05	03CE	03CE
Miscellaneous	R/W	06	03CE	03CE
Color Don't Care	R/W	07	03CE	03CE

ตารางที่ 2.4 รีจิสเตอร์ในกราฟฟิกคอนโทรลเลอร์

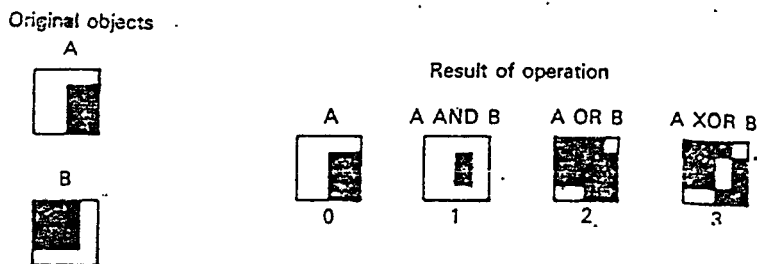
D4	D3	ฟังก์ชัน
0	0	ข้อมูลจากโปรเซสเซอร์ไม่เปลี่ยนแปลง
0	1	ข้อมูลจากโปรเซสเซอร์ AND Latched Data
0	1	ข้อมูลจากโปรเซสเซอร์ OR Latched Data
1	1	ข้อมูลจากโปรเซสเซอร์ XOR Latched Data

ตารางที่ 2.5 ตารางแสดงฟังก์ชันลอจิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 บล็อกไดอะแกรมของกราฟิกคอนโทรลเลอร์



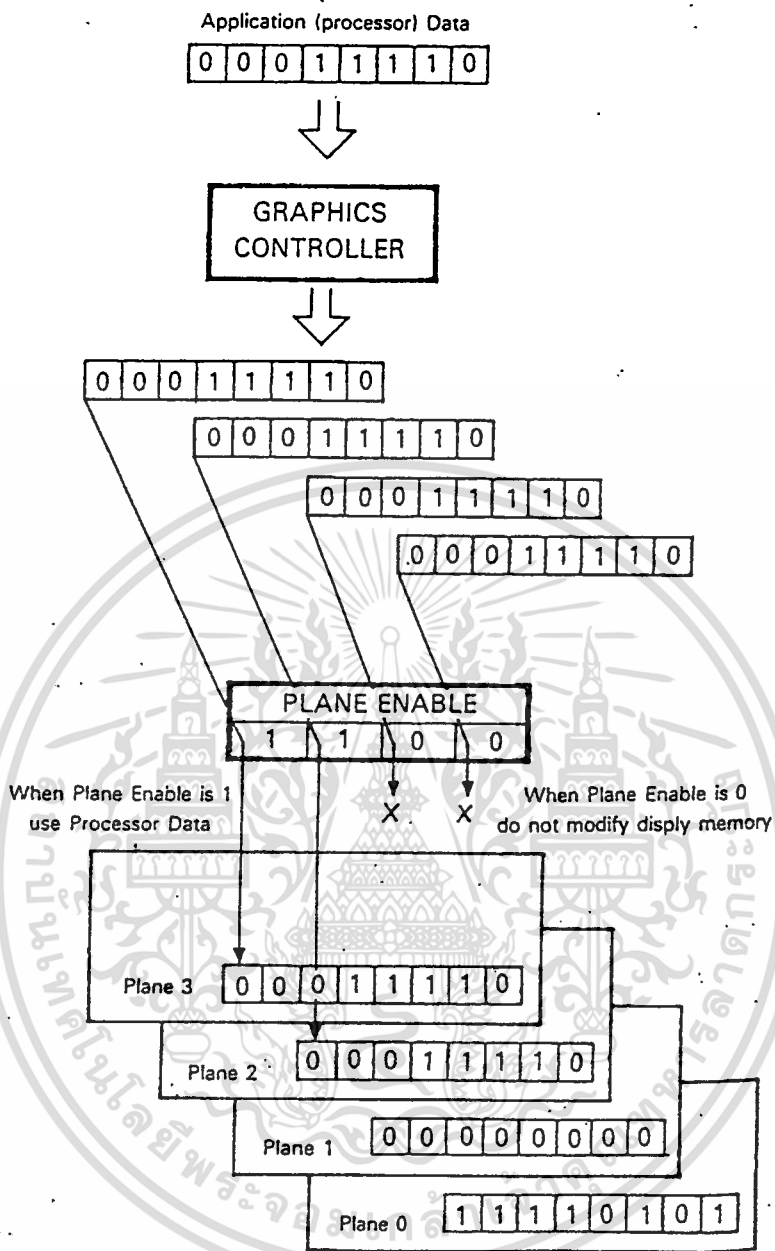
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 รูปที่ 2.5 การกระทำฟังก์ชันทางลอจิกระหว่างข้อมูล 2 ข้อมูล
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.8 Sequencer

ทำหน้าที่กำเนิดสัญญาณนาฬิกาที่ควบคุมการรีเฟรช หน่วยความจำควบคุมช่วงเวลาในการเขียน และการอ่านกับหน่วยความจำ และยังมีส่วนควบคุมการยอม หรือไม่ยอมให้โปรเซสเซอร์กระทำกับหน่วยความจำเฟลนใดเฟลนหนึ่งซึ่งดังแสดงในรูปที่ 2.7 และตารางที่ 2.5 แสดงรีจิสเตอร์ต่าง ๆ ของ Sequencer จากรูปจะเห็นว่าถ้าค่าในรีจิสเตอร์ Map mask(plane enable) เป็น 0 (D0บนเฟลน 0) เฟลนนั้นจะไม่สามารถถูกเปลี่ยนข้อมูลได้

Register Name	R/W	Index	Read Port	Write Port
Sequencer Address	R/W		03C4	03C4
RESET	R/W	00	03C5	03C5
Clocking Modeet	R/W	01	03C5	03C5
Map Mask	R/W	02	03C5	03C5
Character Map Select	R/W	03	03C5	03C5
Memory Mode	R/W	04	03C5	03C5

ตารางที่ 2.6 แสดงรีจิสเตอร์ใน SEQUENCE



รูปที่ 2.6 การทำงานของรีจิสเตอร์ Map mask (plane enable)

2.5.9 Attribute Controller

ทำหน้าที่ควบคุมแอดทริบิวต์ของการแสดงผล เช่นแอดทริบิวต์ที่เป็นสีต่าง ๆ แอดทริบิวต์ที่แสดงการกะพริบ (blinking) หรือการขีดเส้นใต้ (underline) แอดทริบิวต์คอนโทรลเลอร์ประกอบด้วยรีจิสเตอร์ 20 รีจิสเตอร์ดังแสดงในตารางที่ 2.6 ในการกระทำ (เขียนหรืออ่าน) กับรีจิสเตอร์ของ VGA นั้นจะใช้คำสั่งที่กระทำกับพอร์ตคือ คำสั่ง In หรือ Out แต่ละกลุ่มของเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำมาใช้ในเชิงพาณิชย์ ค่ารีจิสเตอร์จะมีแอดเดรสพอร์ตประจำกลุ่ม เช่นกราฟฟิกคอนโทรลเลอร์ จะมีแอดเดรสพอร์ตที่ 3CE ไม่มีการันตีว่าหนังสือนี้ถูกต้องและต้องอยู่ใต้อาณัติของเอกสารที่สงวนไว้เพื่อใช้

เป็นอินเด็กซ์รีจิสเตอร์ และ 3CF เป็นดาต้ารีจิสเตอร์ ตัวอย่างเช่น ถ้าต้องการ out ค่า 3EH (ฐานสิบหก) ไปที่รีจิสเตอร์ Data Rotate ก็ทำได้โดย out ค่า 03 ไปที่พอร์ต 3CE (03 ซึ่งเป็นอินเด็กซ์ของ Data Rotate) จากนั้นจึง out ค่า 3EH ไปที่พอร์ต 3CF สำหรับรีจิสเตอร์อื่นก็เช่นเดียวกันแต่สำหรับแอดทริบิวต์คอนโทรลเลอร์ซึ่งมีพอร์ตสำหรับการเขียนอยู่แอดเดรสเดียวกัน ไซเคิลของการเขียนจะทำให้มีการเปลี่ยนแปลงไปมา ระหว่างอินเด็กซ์รีจิสเตอร์กับดาต้ารีจิสเตอร์ ซึ่งสามารถเริ่มต้นด้วยคำสั่ง In ที่พอร์ต 3DA หลังจากนั้นคำสั่งแรกที่ out ไปที่พอร์ต 3C0 จะถูกส่งไปที่อินเด็กซ์รีจิสเตอร์และคำสั่ง out คำสั่งต่อไปจะถูกส่งไปที่ดาต้ารีจิสเตอร์

ส่วนสำคัญของแอดทริบิวต์คอนโทรลเลอร์อยู่ที่ตารางค้นหาสี (Color lookup table) ซึ่งจะทำการแปลงข้อมูลขนาด 4 บิต ที่เก็บบนหน่วยความจำของการแสดงผลไปเป็นข้อมูลสีขนาด 6 บิตซึ่งจะรวมกับข้อมูลรีจิสเตอร์ Color Select อีก 2 บิตรวมเป็น 8 บิต ที่จะส่งให้กับ Video DAC ต่อไปแต่สำหรับ EGA ข้อมูล 6 บิตจะถูกส่งไปยังจอแสดงผลโดยตรงซึ่งมีข้อแตกต่างระหว่างตัวอักษรเท็กโหมด กับกราฟิกโหมด รูปที่ 2.7 แสดงตารางการค้นหาสีของแอดทริบิวต์คอนโทรลเลอร์ในรูป 2.7(ก) นั้นเป็นการแสดงในโหมดกราฟิกค่าของสีจุด ๆ หนึ่งมีค่าเป็น 011 ค่าของสีนี้จะถูกใช้เป็นแอดเดรสซีไปรีจิสเตอร์ 7 ของตารางค้นหาสีซึ่งมีค่าเป็น 001001 บิตที่ 0 และบิตที่ 3 แทนสีฟ้าที่มีความเข้มแตกต่างกัน (pixel) จุดแรกจึงมีสีฟ้าส่วนจุดต่อไปมีค่าเป็น 0000 ซึ่งซีไปรีจิสเตอร์ 0 และค่าภายในเป็น 0 ดังนั้นจุด นี้จึงเป็นสีดำในรูปที่ 17(ข) เป็นการแสดงในโหมดอักษรจะเห็นว่าเพลน 0 เก็บค่า 41 ซึ่งเป็นรหัสสีของตัวอักษร A เพลน 1 เก็บค่า 07 ซึ่งเป็นแอดทริบิวต์ของตัวอักษร A 4 บิตบนของแอดทริบิวต์เป็นตัวกำหนดสีของแบกราวนด์ และ 4 บิตล่างเป็นตัวกำหนดสีของไฟร์กราวนด์

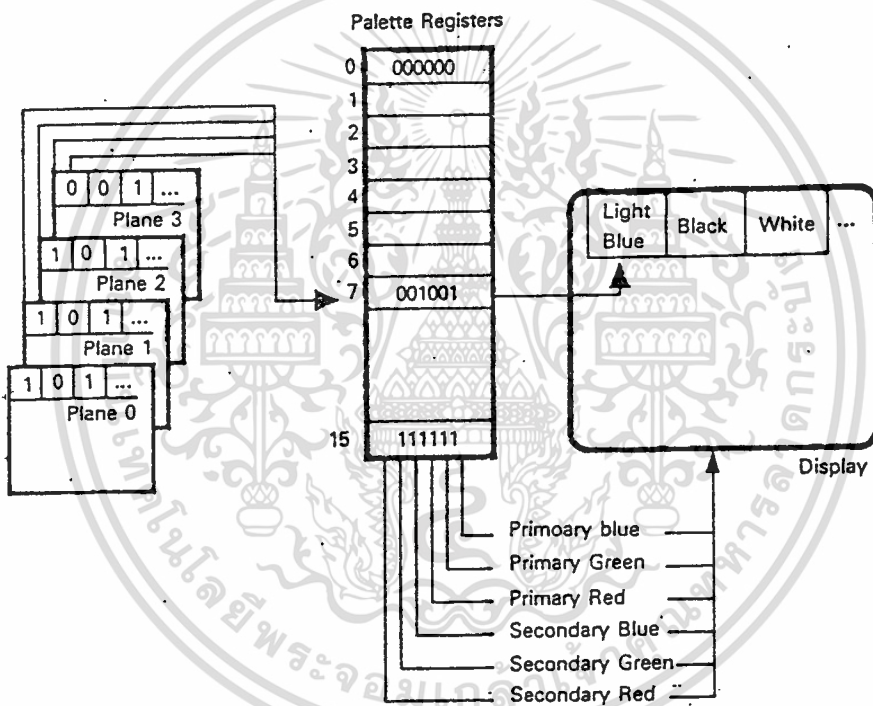
Register Name	R/W	Index	Read Port	Write Port
Address	R/W		03C0	03C0
Palette registers	R/W	00	03C1	03C0
Attribute Mode Control	R/W	01	03C1	03C0
Overscan Color	R/W	02	03C1	03C0
Color Plane Enable	R/W	03	03C1	03C0
Horizontal PEL Panning	R/W	04	03C1	03C0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

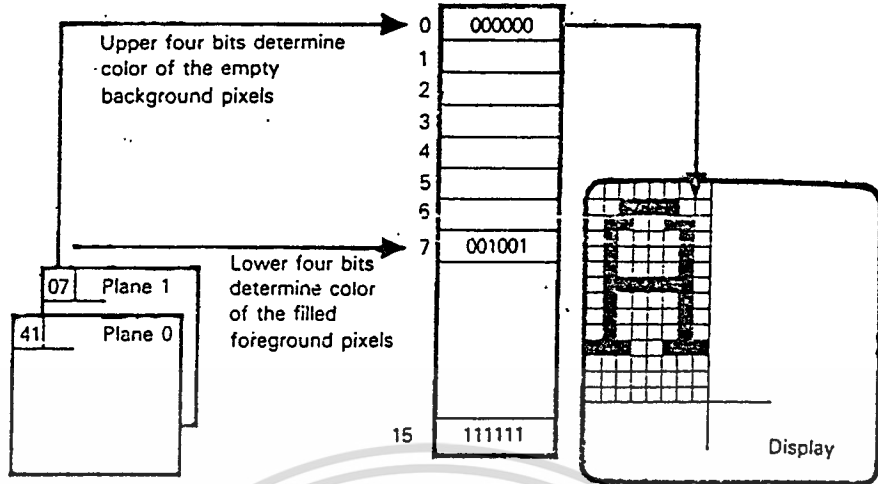
Register Name	R/W	Index	Read Port	Write Port
Colr Select	R/W	04	03C1	03C0

ตารางที่ 2.7 แสดงวิธีสแตร์ในแอดทรีบิตคอนโทรลเลอร์



รูปที่ 2.7(ก) แสดงการแปลงข้อมูลจากหน่วยความจำไปเป็นรูป (จุด) ในโหมดกราฟฟิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 (ข) การแปลงรหัสไปเป็นตัวอักษรพร้อมทั้งแอตทริบิวต์

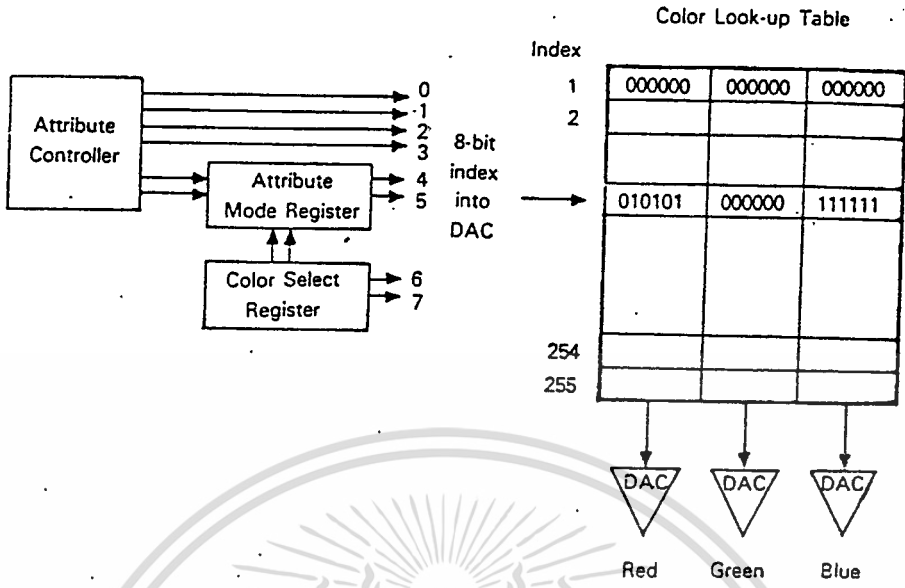
2.5.10 VGA VIDEO DAC

ส่วนประกอบตัวสุดท้ายของระบบแสดงผล VGA คือ VIDEO DAC (Digital to Analog Converter) ตัวแปลงข้อมูลดิจิทัล ให้เป็นสัญญาณอะนาลอก เพื่อจะขับจอแสดงผลไอบีเอ็มใช้ VIDEO DAC ของบริษัท Imos เบอร์ IM5G-171 ซึ่งภายในประกอบด้วยตัวแปลงสัญญาณ 3 ชุด สำหรับ 3 สีคือ แดง, เขียว และน้ำเงิน และตารางการค้นหาลี (Color Look-up Table) ที่รับข้อมูลขนาด 8 บิตจากแอตทริบิวต์คอนโทรลเลอร์มาเป็นตัวชี้ชี้สเตอร์ขนาด 18 บิต 1 ใน 256 รีจิสเตอร์ เพื่อเลือกข้อมูลที่จะนำไปแปลงเป็นสัญญาณอะนาลอกดังแสดงในรูปที่ 2.7

จากรีจิสเตอร์จำนวน 256 รีจิสเตอร์ ทำให้สามารถแสดงสีได้พร้อม 256 สี แต่ภายใน 1 รีจิสเตอร์ ซึ่งมีขนาด 18 บิต จะมีสีแตกต่างกันทั้งสิ้น $2^{18} = 256 \text{ K}$ สี และนี่คือเหตุผลที่ว่า ระบบ VGA สามารถแสดงสีได้ 256 สีพร้อมกันจากจำนวนสีทั้งสิ้น 256 K สี

ที่กล่าวมาทั้งหมด คือโครงสร้างทางฮาร์ดแวร์ของการ์ด VGA จะเห็นว่าภายใน VGA มี รีจิสเตอร์ต่าง ๆ มากมายซึ่งแต่ละตัวก็มีหน้าที่แตกต่างกัน การที่จะแก้ไขค่าในรีจิสเตอร์นั้น ควรที่จะศึกษาก่อนว่ารีจิสเตอร์ตัวนั้นถ้าแก้ไขแล้ว จะทำให้เกิดความเสียหายหรือไม่ อย่างเช่น รีจิสเตอร์บางตัวของ CRT Controller ถ้ามีค่าไม่เหมาะสมจะทำให้จอภาพเสียหายได้ การใช้ ฟังก์ชันต่าง ๆ ของการ์ด VGA นอกจากการกระทำกับรีจิสเตอร์โดยตรงสามารถทำได้โดยที่เรียก ใช้บริการของรอมไบออสบนการ์ดฟังก์ชันเหล่านี้ ได้แก่ เซตโหมดของการแสดงผลอ่านโหมดการแสดงผล เซตพาร์เล็ตรีจิสเตอร์ โหลดค่าลงในตารางแปลงตัวอักษร เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ตารางที่ 2.8 แสดงการแปลงข้อมูลสีของ DAC



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การสร้างและการออกแบบ

โครงการนี้เป็น การนำคอมพิวเตอร์มาใช้เป็นสื่อช่วยสอน หรือเรียกว่า CAI (Computer Aids Intruction) โดยโครงการชุดนี้จัดทำตามหัวข้อต่อไปนี้

- 1) โปรแกรมคอมพิวเตอร์ภาษา ซี
- 2) การใช้งานเทอร์โบซีในกราฟฟิกโหมด
- 3) เขียนโปรแกรมแสดงผลภาษาไทยในโหมดกราฟฟิกโหมดด้วยเทอร์โบซี
- 4) เขียนโปรแกรมแสดงส่วนให้ความช่วยเหลือ (help)
- 5) สร้างเท็กซ์ไฟล์ (text file) ที่ใช้แสดงข้อความช่วยเหลือ
- 6) เขียนโปรแกรมเพื่อรับคำสั่งจากคีย์บอร์ด
- 7) เขียนโปรแกรมจำลองการแสดงผลคำสั่งต่าง ๆ ของ คอส

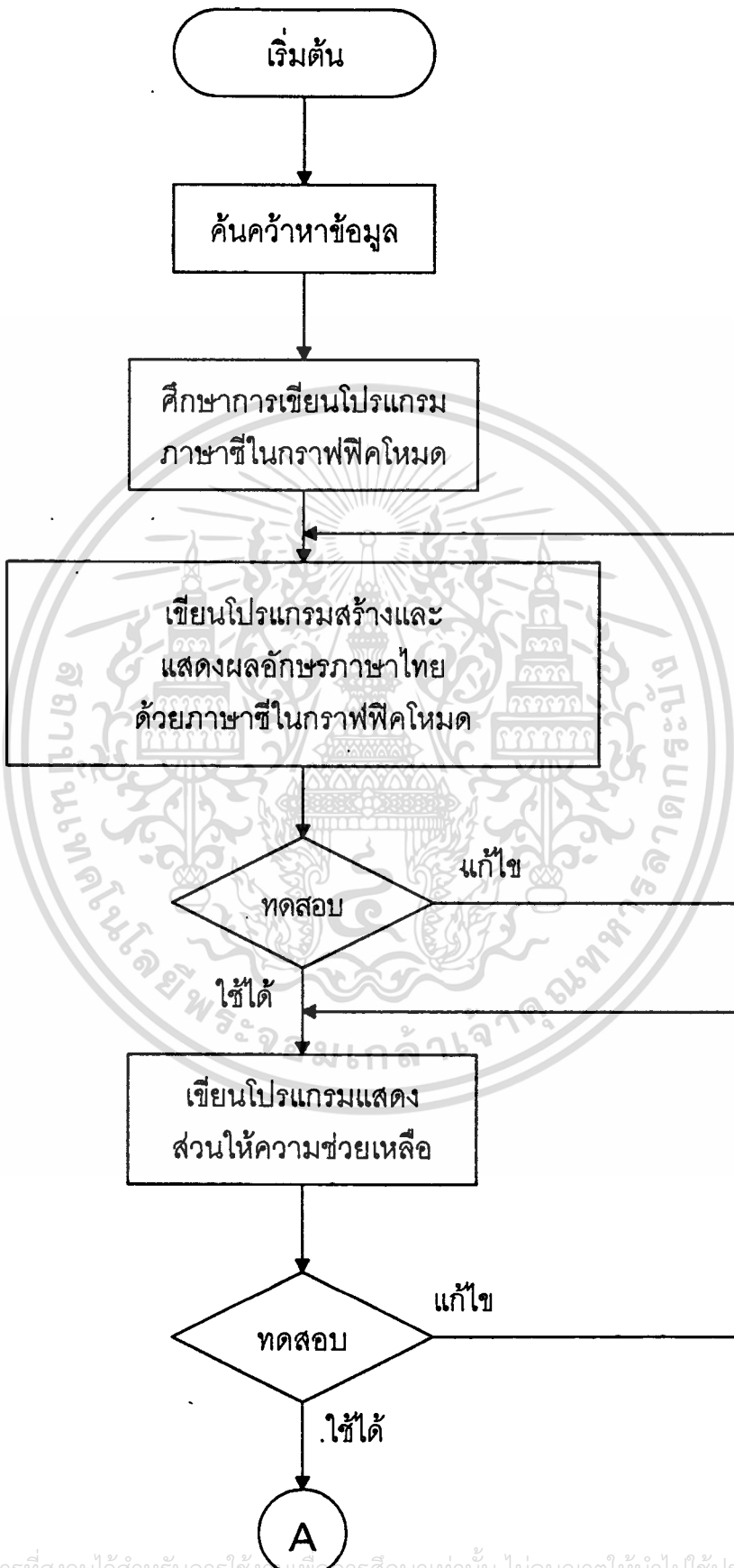
ในการเขียนโปรแกรมช่วยสอนนี้มีหลักการออกแบบ ดังโพลีชาร์ตรูปที่ 3.1

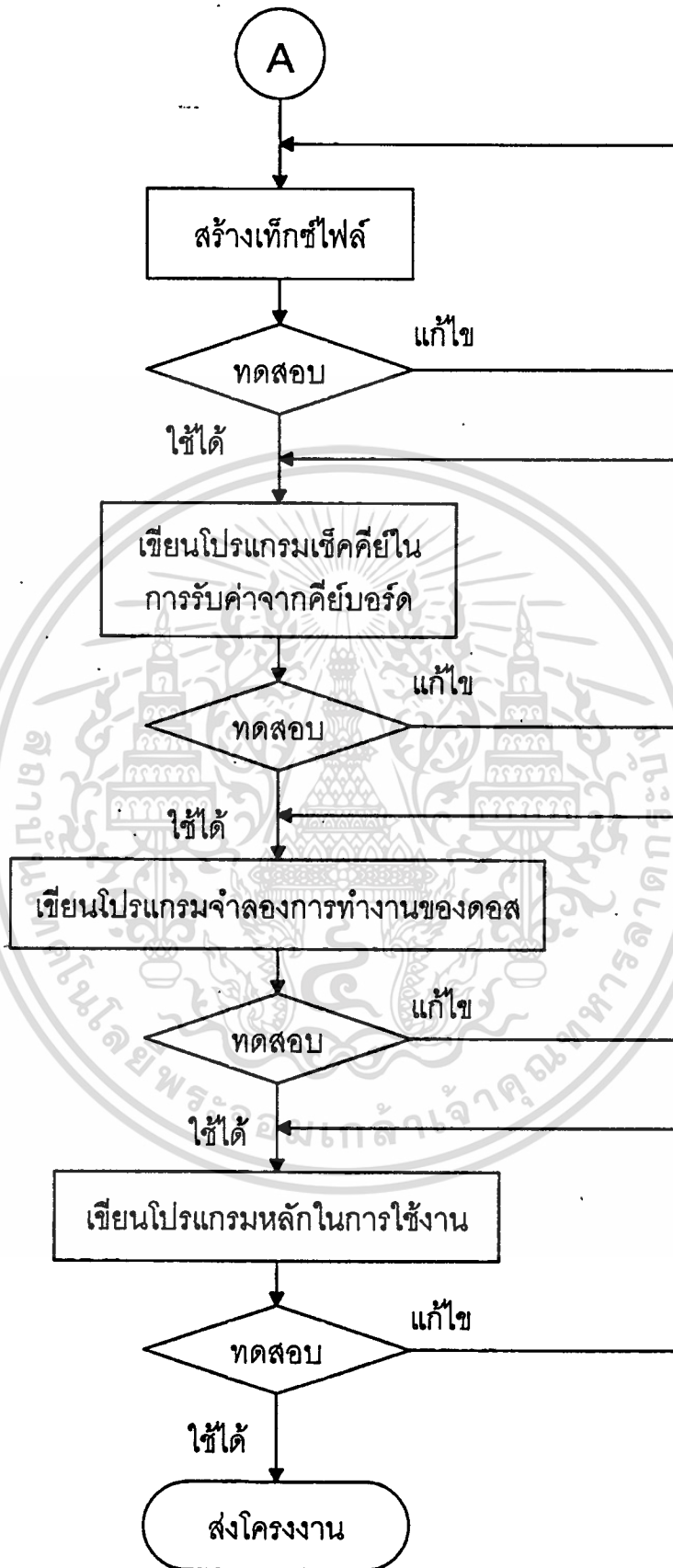
3.1 โปรแกรมคอมพิวเตอร์ภาษาซี (C Language)

ภาษา ซี จัดเป็นภาษาระดับกลาง (Middle Language) ที่นำมาเขียนโปรแกรมระบบปฏิบัติการยูนิกซ์ (U-Nix) ภาษา C มีความสามารถเหมือนภาษาแอสเซมบลี คือมีคำสั่งที่สามารถเข้าถึงบิต, ไบต์และตำแหน่งของหน่วยความจำในเครื่องคอมพิวเตอร์ได้แต่เขียนได้ง่ายเหมือนกับภาษาระดับสูงทั่วไป เช่น ภาษาเบสิก (Basic) หรือภาษาปาสคาล (Pascal) นอกจากนี้โปรแกรมที่เขียนด้วยภาษาซี ยังมีข้อดีที่สำคัญอีกประการหนึ่งคือภาษาซี ที่ใช้บนเครื่องหนึ่งสามารถไว้บนเครื่องอื่น ๆ ได้โดยไม่ต้องดัดแปลง และมีลักษณะเป็นภาษาโครงสร้างที่เหมาะสมจะใช้ในการพัฒนาโปรแกรม และมีความเร็วสูงในการทำงานจึงทำให้ภาษา ซี ถูกนำมาใช้ในการเขียนโปรแกรมสำเร็จรูปต่าง ๆ เช่น DBASRTV, Microsoft Exel, Sidekick, Oracal, ASA, Cwriter เป็นต้นบริษัทที่มีชื่อเสียงในการผลิตซอฟต์แวร์ทางด้านคอมพิวเตอร์ต่าง ๆ จึงพยายามผลิตตัวแปลชุดคำสั่งภาษาซี (C compiler) ของบริษัทตัวเองเพื่อใช้กับเครื่องขนาดต่าง ๆ ตั้งแต่ไมโครคอมพิวเตอร์จนถึงคอมพิวเตอร์ขนาดใหญ่ ได้แก่เทอร์โบซี ไมโครซอฟท์ซี ที่ใช้กับระบบปฏิบัติการบนยูนิกซ์ของแต่ละบริษัท เป็นต้น

สำหรับปริมาณในฉบับนี้จะใช้เทอร์โบซีเป็นหลัก เพราะสามารถไว้บนเครื่องไมโครคอมพิวเตอร์ได้ทั้งรุ่น XT, AT และ PS/2 ที่มีหน่วยขับแผ่นบันทึก (Floppy Disk Drive)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ตั้งแต่ 1 หน่วยขับ (Drive) ขึ้นไป แต่ควรรู้ใช้ตั้งแต่ 2 หน่วยขับ หรือมีงานบันทึกแบบแข็ง ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 3.1 ฝ่าฝืนขั้นตอนการออกแบบโปรแกรม
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามเผยแพร่ต่อผู้อื่น และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Hard Disk) จะสะดวกและรวดเร็วในการทำงานและขนาดหน่วยความจำในเครื่องคอมพิวเตอร์ไม่ควรต่ำกว่า 488 KB สำหรับการใส่เทอร์โบซี เวอร์ชัน 2 โดยอุปกรณ์ที่กล่าวมานี้ เป็นอุปกรณ์ที่มีอยู่ทั่วไปในไมโครคอมพิวเตอร์ที่ใช้อยู่ในบ้านเรา และนอกจากนั้นโปรแกรมเทอร์โบซีซึ่งหาได้ง่ายในการศึกษาจากปริทัศน์านยนต์เล่มนี้ผู้เขียนจึงมุ่งหวังให้ผู้อ่านสามารถทำความเข้าใจได้ดียิ่งขึ้น

ภาษาในเทอร์โบซีได้รวบรวมโปรแกรมต่าง ๆ ไว้ในตัวเองเพื่อสะดวกสำหรับผู้ที่ต้องการใช้

เช่น

- โปรแกรม Editor มีหน้าที่สำหรับสร้างหรือแก้ไข Source File
- โปรแกรม compiler มีหน้าที่แปล Source File เป็น Object File
- โปรแกรม Linker มีหน้าที่ Link Object File ให้ได้เป็น Executable file

เป็นต้น

3.1.1 การจัดระดับของภาษา

ในสมัยก่อน มีการจัดแบ่งภาษาที่ใช้ในทางคอมพิวเตอร์ออกเป็น 2 ระดับคือ

3.1.1.1 ภาษาระดับต่ำ (Low Level Language) ซึ่งจะเป็นลักษณะภาษาที่มีลักษณะใกล้เคียงกับภาษาเครื่อง (Machine Language) มากจึงทำให้มีข้อดีในแง่ของความเร็วในการทำงานสูง แต่เขียนและเรียนรู้ได้ยาก เพราะมีลักษณะเฉพาะของเครื่องจึงทำให้ไม่สามารถใช้บนเครื่องอื่น ๆ ได้ และไม่เหมาะที่จะใช้ในการพัฒนาโปรแกรม

3.1.1.2 ภาษาระดับสูง (High Level Language) เป็นภาษาที่เขียนง่ายเรียนได้ง่ายสะดวกในการใช้งาน แต่เวลาในการทำงานจะช้ากว่าภาษาระดับต่ำ

แต่เมื่อมีภาษา C เกิดขึ้นโดยเดนนิส ริทชี (Denis Ritchie) ได้พัฒนามาจากภาษา B (B Language) ที่เคน ทอมสัน (Ken Thomson) เป็นผู้พัฒนาขึ้นริทชีได้รวมเอาข้อดีของภาษาทั้งสองระดับเข้าไว้ด้วยกัน จึงทำให้ภาษา C ถูกจัดระดับใหม่เป็นภาษาระดับกลาง ดังตารางที่ 3.1

ภาษาระดับต่ำ	ภาษาระดับกลาง	ภาษาระดับสูง
Assembler Macro-Assembler	C Forth	Basic Cobol Fortran Pascal Modula-2

ตารางที่ 3.1 การจัดระดับของภาษา

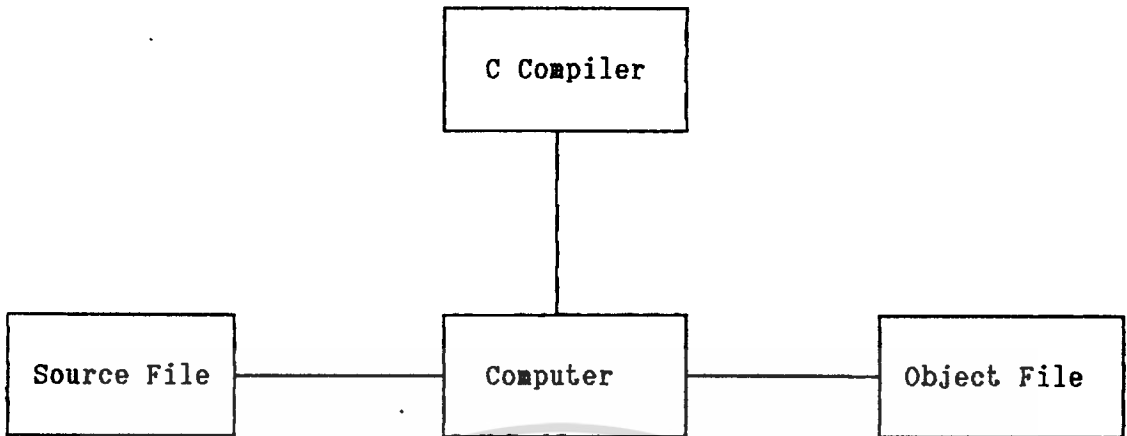
3.1.2 การโปรแกรมคอมพิวเตอร์ภาษาคิวเทอร์โบซี

ในการพัฒนาโปรแกรมคอมพิวเตอร์ภาษาซีให้ได้ประสิทธิภาพนั้น มีหลายวิธีโดยอาจใช้คอมมานด์ไลน์คอมไพล์ (Command Line Compile) หรือการคอมไพล์โดยโปรแกรมสำเร็จรูป และในภาษาซีนั้นยังแบ่งออกเป็น 2 คำสใหญ่ ๆ คือของบริษัทไมโครซอฟท์ และบริษัทบอแลนด์ซึ่งทั้งสองคำสมีความคล้ายคลึงกันในการใช้งาน ดังนั้นในการพัฒนาสามารถที่จะใช้ได้ทั้งสองบริษัท ในที่นี้ได้เลือกภาษาซีของบริษัทบอแลนด์เนื่องมาจากมีพื้นฐานมาจากการเรียนวิชา Computer Programming II มาแล้วจึงมีความคล่องตัวในการใช้งานได้มากกว่า และในการใช้คอมไพเลอร์ภาษาซีของบริษัทบอแลนด์นั้น ถ้าจะให้มีความคล่องตัวจึงควรมีความเข้าใจในการใช้งานดังนี้

3.1.3 ขั้นตอนการทำโปรแกรมโดยใช้เทอร์โบซี

เมื่อเขียนโปรแกรมในภาษา ซี เสร็จเรียบร้อยแล้ว เทอร์โบซีจะมีขั้นตอนในการทำงาน เพื่อให้ได้ผลลัพธ์ออกมา 2 ขั้นตอน คือ

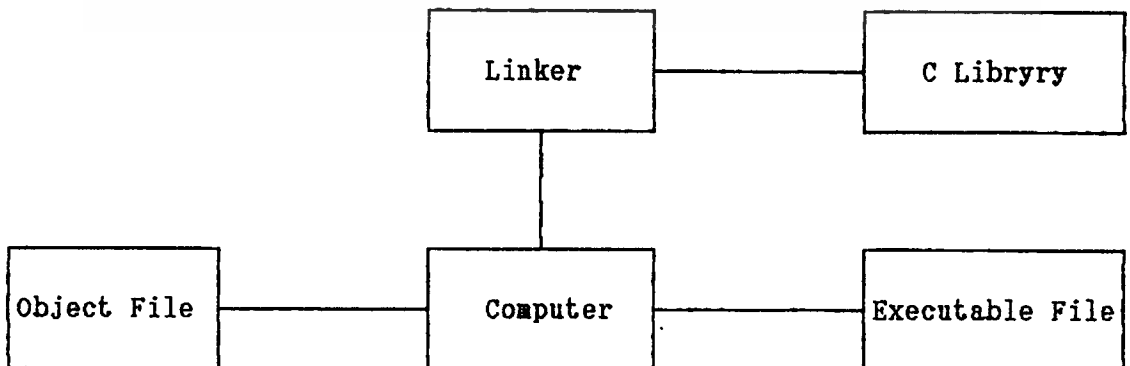
ขั้นตอนที่ 1 การแปลโปรแกรม (C Compile) เป็นการนำเอาโปรแกรมที่ได้เขียนขึ้น (source File) มาทำการแปลเป็นโปรแกรมภาษาเครื่อง (Object File) ดังรูปที่ 3.2



รูปที่ 3.2 การแปลโปรแกรม (Compile)

Source File เป็นโปรแกรมที่เขียนขึ้น ซึ่งมีกฎเกณฑ์การตั้งชื่อเหมือนกับการตั้งชื่อแฟ้มข้อมูลใน DOS แต่จะมีชนิดของแฟ้มข้อมูลเป็น C เท่านั้นเมื่อโปรแกรมผ่านการแปลโปรแกรมแล้วจะได้เป็นโปรแกรมภาษาเครื่อง (Object File) ที่มีชนิดของแฟ้มข้อมูลเป็น .OBJ เช่น TOP.OBJ

ขั้นตอนที่ 2 เป็นการเชื่อมโยง (Link) เป็นการเอาโปรแกรมภาษาเครื่อง (Object File) มาทำการเชื่อมโยงกับฟังก์ชันที่ต้องการจาก Library เพื่อสร้างให้เป็น Executable File ดังรูปที่ 2.2



Excutable File จะเป็นแฟ้มข้อมูลที่สามารถสั่งให้ปฏิบัติงานได้โดยตรงในขณะที่อยู่ที่ DOS แฟ้มข้อมูลนี้จะเป็นชนิด .EXE

3.1.4 การเข้าสู่เทอร์โบซี

การเข้าสู่เทอร์โบซีมีขั้นตอนในการทำดังนี้

3.1.4.1 นำแผ่นโปรแกรมเทอร์โบซีแผ่นที่ 1 ใส่ใน Drive A: และแผ่นที่ 2 ใส่ใน Drive b:

3.1.4.2 เปิดเครื่อง

3.1.4.3 สังเกตจอภาพ จะมีข้อความให้เติมวันที่กด Enter จะมีข้อความให้เติมเวลา กด Enter จะได้เครื่องหมาย



A>

ปรากฏบนจอภาพแสดงว่าอยู่ใน MS.DOS เรียบร้อยแล้ว

3.1.4.4 พิมพ์ TC แล้วกดปุ่ม Enter สักครู่หนึ่งจอภาพจะปรากฏเมนูขึ้นดังรูป

File Edit Run Compile Project Options Debug Break/watch

— Edit —

Line 1 Col 1 Insert Ident Tab Fill Unindent C:noname.c



— Message —

F1-Help F2-Zoom F6-Switch F7-Trace F8-Step F9-Make F10-Menu

รูปที่ 3.4 แสดงรูปเมนู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

File Edit Run Compile Project Options Debug Break/watch

Edit

Line 4 Col 6 Insert Ident Tab Fill Unindent C:\noname.c

```
main()
```

```
{
```

```
printf ("How are you?\n");
```

```
}
```



Message

F1-Help F2-Zoom F6-Switch F7-Trace F8-Step F9-Make F10-Menu

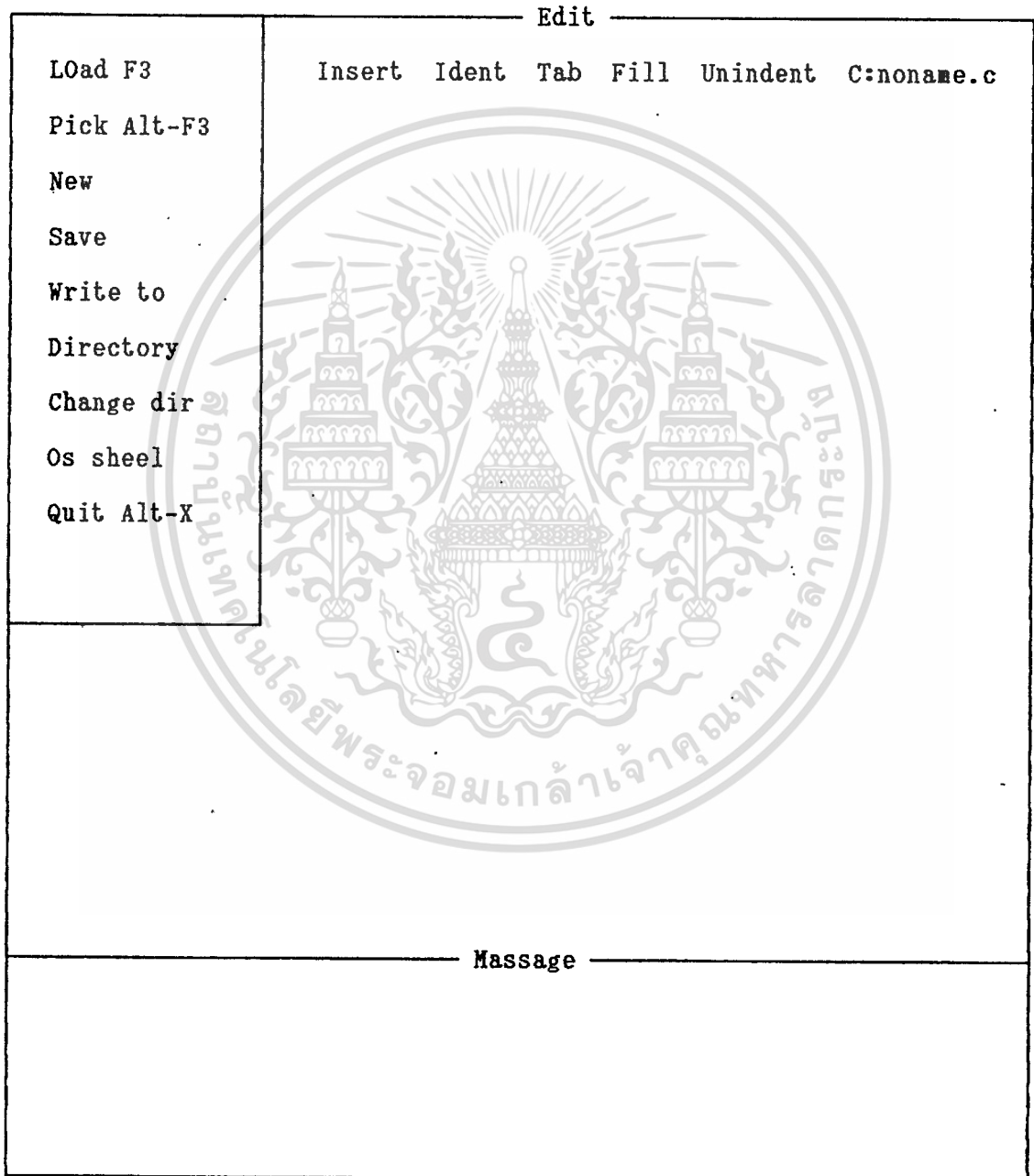
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 3.5 การโปรแกรมในโหมดการแก้ไข (EDIT MODE)
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.8 การใช้คำสั่งในคำสั่ง File

เมื่อเข้าสู่คำสั่ง File จอภาพจะปรากฏเมนูย่อย ซึ่งเป็นคำสั่งที่มีให้ใช้คำสั่ง File

ดังรูป

.File Edit Run Compile Project Options Debug Break/watch



F1-Help F2-Zoom F6-Switch F7-Tracr F8-Step F9-Make F10-Menu

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 3.6 การใช้งานเมนูหัวขั้วไฟล์
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.8.1 คำสั่ง Load

เป็นคำสั่งให้ดึงแฟ้มข้อมูลที่มีอยู่แล้วมาปรากฏในบริเวณ Edit

3.1.8.2 คำสั่ง Pick

เป็นคำสั่งใช้แสดงชื่อแฟ้มของข้อมูลที่เคยแก้ไขมาแล้วให้มาปรากฏบนหน้าจอแล้วให้สามารถดึงกลับมาปรากฏในบริเวณ Edit ได้ใหม่อีกครั้งหนึ่ง

3.1.8.3 คำสั่ง New

ใช้สร้างแฟ้มข้อมูลใหม่โดยแฟ้มข้อมูลที่สร้างใหม่นี้เครื่องจะกำหนดให้ชื่อ NONAME

3.1.8.4 คำสั่ง Save

เป็นคำสั่งให้นำข้อมูลที่อยู่ในบริเวณ Edit เข้าไปเก็บในแฟ้มข้อมูลบนแผ่นบันทึก (Diskette) ที่กำหนด

3.1.8.5 คำสั่ง Write to

ทำงานเหมือนกับคำสั่ง Save แต่จะใช้ในกรณีที่มีการ ก๊อปปี้ไฟล์

3.1.8.6 คำสั่ง Directory

เป็นคำสั่งกำหนดหน่วยขับแผ่นบันทึกพร้อมทั้งแสดงรายชื่อแฟ้มข้อมูลที่อยู่ในแผ่นบันทึกที่อยู่ในหน่วยขับนั้นออกมาให้ปรากฏบนจอภาพ และสามารถนำข้อมูลจากแฟ้มที่ต้องการไปแสดงในบริเวณ Edit ได้

3.1.8.7 คำสั่ง Change directory

เป็นคำสั่งกำหนด Directory ของการปฏิบัติงานสำหรับคำสั่ง file

3.1.8.8 คำสั่ง OS shell

เป็นคำสั่งให้ออกจากเทอร์มินัลไปยังเอ็มเอสดอสชั่วคราวเมื่อต้องการจะกลับเข้าสู่เทอร์มินัลให้พิมพ์ exit แล้วกด enter

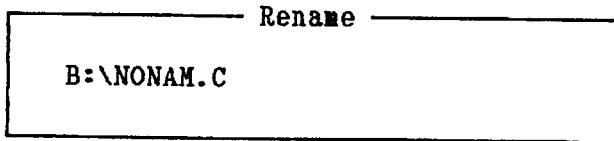
3.1.8.9 คำสั่ง Quit

เป็นคำสั่งให้ออกจากเทอร์มินัลไปยัง MS-DOS แบบถาวรในกรณีที่เป็นการสร้างแฟ้มข้อมูลใหม่โดยใช้คำสั่ง New จะปรากฏตัวชี้ตำแหน่งในส่วนของ Edit แสดงว่าพร้อมที่จะพิมพ์โปรแกรมที่เขียนขึ้นได้แล้ว

3.1.9 การเก็บแฟ้มข้อมูลลงบนแผ่นบันทึก

หลังจากพิมพ์โปรแกรมที่เขียนเองลงไปเรียบร้อยแล้วให้ออกจากการ Edit โดยการกดปุ่ม F10 การทำงานจะกลับมาสู่เมนเมนูแล้วใช้คำสั่ง File จากนั้นเลือกคำสั่งย่อย Save จอภาพจะปรากฏข้อความว่าต้องการเปลี่ยนชื่อแฟ้มข้อมูลหรือไม่ดังนี้

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ถ้าไม่ต้องการเปลี่ยนชื่อ ให้กดปุ่ม Enter ได้เลยถ้าต้องการเปลี่ยนชื่อหรือเปลี่ยนหน่วยขับ (Drive) ให้พิมพ์หน่วยขับและชื่อแฟ้มข้อมูลใหม่ลงไปได้เลยแล้วกด Enter

3.1.10 การใช้คำสั่งใน Edit

เมื่ออยู่ใน Edit จะสังเกตได้จากตัวชี้ตำแหน่งที่ปรากฏอยู่ในส่วนของ Edit แสดงว่าพร้อมที่จะทำการแก้ไขโปรแกรม หรือสร้างโปรแกรมใหม่ได้แล้ว โดยใน Edit มีคำสั่งที่สำคัญ ๆ เท่านั้น

3.1.10.1 การเลื่อนตัวชี้ตำแหน่ง

เราสามารถเลื่อนตัวชี้ตำแหน่งไปยังทิศทางต่าง ๆ ที่ต้องการได้โดยใช้ปุ่มลูกศร

3.1.10.2 การลบข้อมูล

ทำได้โดยเลื่อนตัวชี้ตำแหน่งไปยังตำแหน่งที่ต้องการจะลบใช้ปุ่ม Del หรือ Ctrl+Y เพื่อลบข้อมูลที่ละ 1 ตัวอักษร Ctrl+Y เพื่อลบข้อมูลที่ละ 1 บรรทัด

3.1.10.3 การแทรกข้อมูล

สามารถทำได้โดยให้เลื่อนตัวชี้ตำแหน่งไปยังตำแหน่งของตัวอักษรที่อยู่หลังตำแหน่งที่ต้องการแทรก แล้วสังเกตคำว่า Insert ที่อยู่ในบรรทัดแรกของบริเวณ Edit ถ้าปรากฏว่ามีอยู่แล้วให้พิมพ์ข้อความหรือตัวอักษรที่ต้องการจะแทรกได้เลยถ้าไม่ปรากฏอยู่ให้กดปุ่ม Insert ให้เข้าสู่ Insert Mode ก่อน

3.1.10.4 การกำหนดบล็อก

ข้อมูลที่ต้องการแก้ไขในบริเวณ Edit สามารถกำหนดขอบเขตเป็นบล็อกขึ้นได้โดยทำดังนี้

- ให้เลื่อนตัวชี้ตำแหน่งไปยังตำแหน่งที่ต้องการกำหนดเป็นจุดเริ่มต้นบล็อกแล้วกดปุ่ม Ctrl-K-B
- ให้เลื่อนตัวชี้ตำแหน่งไปยังตำแหน่งที่ต้องการกำหนดเป็นจุดจบของบล็อกแล้วกดปุ่ม Ctrl-K-K จะปรากฏแถบแสงขึ้นในบริเวณที่กั้นบล็อกจากนั้นสามารถทำงานต่าง ๆ เกี่ยวกับบล็อกได้ดังนี้

3.1.10.5 การลบข้อมูลภาษาในบล็อก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นการลบข้อมูลที่อยู่ในบล็อกทั้งไปทั้งหมด โดยใช้การกดปุ่ม Ctrl-K-Y

3.1.10.6 การย้ายข้อมูลภายในบล็อก

เป็นการย้ายข้อมูลที่อยู่ในบล็อกที่กำหนดทั้งหมด ไปไว้ยังอีกที่หนึ่งซึ่งทำได้โดยการเลื่อนตัวชี้ตำแหน่ง ไปยังตำแหน่งที่ต้องการย้ายข้อมูลไป แล้วกดปุ่ม Ctrl-K-V

3.1.10.7 การตัดลอกข้อมูลภายในบล็อก

เป็นการตัดลอกข้อมูลที่อยู่ในบล็อกที่กำหนดทั้งหมดไปไว้ยังอีกที่หนึ่ง ที่ต้องการซึ่งทำได้โดยเลื่อนตัวชี้ตำแหน่งไปยังตำแหน่งที่ต้องการตัดลอกบล็อกไปแล้วกดปุ่ม Ctrl-K-C

3.1.10.8 การยกเลิกบล็อก

เป็นการยกเลิกบริเวณที่กั้นบล็อกไว้ซึ่งทำได้โดยเลื่อนตัวชี้ตำแหน่ง ไปยังจุดเริ่มต้นของบล็อกใหม่แล้วกดปุ่ม Ctrl-K-K

3.1.10.9 การซ่อนแถบแสงของบล็อก

เมื่อไม่ต้องการให้บริเวณบล็อกมีแถบแสงสว่างปรากฏขึ้นมาสามารถทำได้โดยกดปุ่ม Ctrl-K-H ถ้าต้องการให้มีแถบแสงกลับมาอีกทำได้โดยกดปุ่ม Ctrl-K-H อีกครั้งหนึ่ง

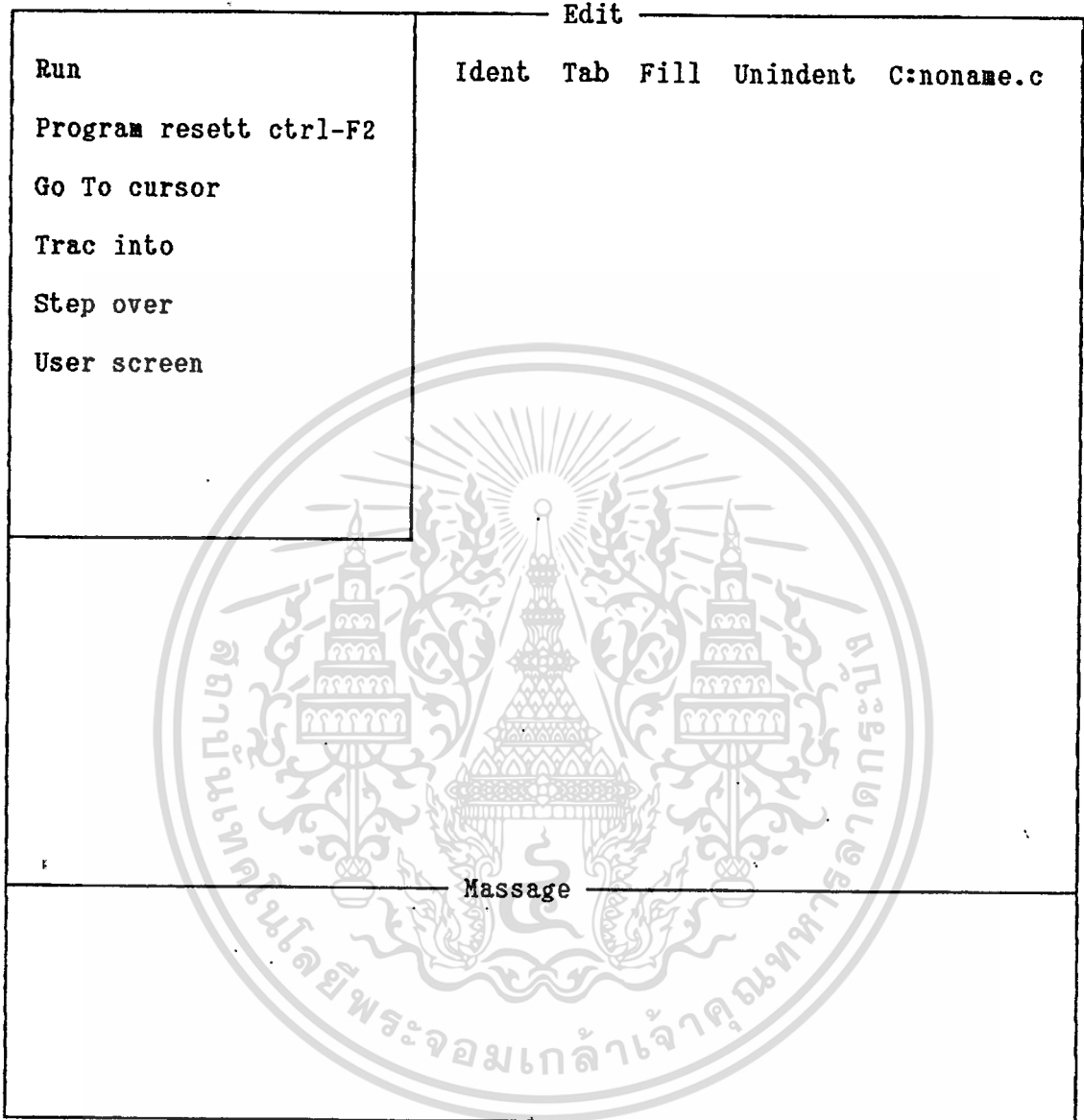
3.1.10.10 การพิมพ์ข้อมูลใน Edit

ขณะที่ข้อมูลอยู่ในบริเวณ Edit สามารถนำข้อมูลออกมาพิมพ์ทางเครื่องพิมพ์ได้โดยกดปุ่ม Ctrl-K-P

3.1.11 การสั่งให้โปรแกรมปฏิบัติงาน

สามารถสั่งให้โปรแกรมปฏิบัติงานได้โดยใช้คำสั่ง RUN ขณะที่อยู่เมนูจะปรากฏคำสั่งย่อยของข้อมูล RUN ขึ้นอีก 6 คำสั่งดังรูปที่ 3.7

File Edit Run Compile Project Options Debug Break/watch



F1-Help F2-Zoom F6-Switch F7-Tracr F8-Step F9-Make F10-Menu

รูปที่ 3.7 เมนูโปรแกรมปฏิบัติงาน

เพื่อต้องการให้โปรแกรมปฏิบัติงานให้ใช้คำสั่งข้อ RUN เครื่องจะทำการปฏิบัติงานตามโปรแกรมที่สั่งให้เลขโดยถ้าโปรแกรมนั้น ยังไม่คอมไพล์ (Compile) มาก่อนเครื่องจะทำการแปลโปรแกรมและลิงค์โปรแกรมให้ด้วยทำให้เกิดเพิ่มข้อมูลเกิดใหม่ขึ้น 2 ชนิดคือ .obj และ .EXE

ถ้าโปรแกรมที่เขียนไม่มีการผิดกฎเกณฑ์ของภาษา จะได้ผลของการปฏิบัติงานแล้วโปรแกรมเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าจะกลับมามี Edit อีกครั้ง ซึ่งบางครั้งจะไม่เห็นผลของการปฏิบัติงาน เนื่องจากใช้เวลาในการไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปฏิบัติงานเร็วมากถ้าต้องการคุณจะสามารถขุดได้โดยใช้คำสั่ง Userscreen หรือกดปุ่ม Alt กับปุ่ม F5 ถ้าโปรแกรมมีข้อผิดพลาดต้องแก้ไขโปรแกรมให้ถูกต้องก่อน จึงจะลงมือปฏิบัติงาน สำหรับการใช้น้ำสั่งย่ออื่น ๆ สามารถศึกษาเพิ่มเติมได้จากหนังสือคู่มือเทอร์โบชิของบริษัท Borland

3.2 การใช้งานเทอร์โบชิในกราฟิคโหมด

ในการแสดงผลบนจอคอมพิวเตอร์ในกราฟิคโหมดนั้น เราต้องมาทำความเข้าใจในส่วนของ ฮาร์ดแวร์ของจอภาพคอมพิวเตอร์กันก่อน ความละเอียดของจอภาพคอมพิวเตอร์นั้นเมื่ออยู่หลายระดับ การแสดงผลด้วยกัน ดังตารางที่ 3.2

Graphic Driver	Mode	ค่าคงที่	ความละเอียด
CGA	CGAO	0	320 X 200
	CGAC1	1	320 X 200
	CGAC2	2	320 X 200
	CGAC3	3	320 X 200
	CGAHI	4	640 X 200
MCGA	MCGAC0	0	320 X 200
	MCGAC1	1	320 X 200
	MCGAC2	2	320 X 200
	MCGAC3	3	320 X 200
	MCGAHI	5	640 X 480
EGA	EGALO	0	640 X 200
	EGAHI	1	640 X 350
EGA64	EGA64LO	0	640 X 200
	EGA64HI	1	640 X 350

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ตารางที่ 3.2 ความละเอียดของจอภาพคอมพิวเตอร์
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งยังมีเหตุเปลี่ยนแปลงเนื้อหา และตงยอ ึ่งองตงเงงของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งจากระดับการแสดงผลที่กล่าวข้างต้นยังแยกปลีกย่อยลงไปได้อีก ซึ่งจะต่างกันที่ความละเอียดของจุดการแสดงผล

ที่ต้องมาศึกษากันถึงจำนวนจุดความละเอียดของจอภาพนั้น เนื่องจากในกรณีที่แสดงผลทางด้านกราฟิกนั้นที่กักจุดในการวางตำแหน่งรูปมีความสำคัญมาก ตัวอย่างเช่นจอภาพ MGA ชนิดเฮอริคิวลิสต์นั้นมีความละเอียด 720 x 348 จุด ส่วนจอภาพระดับ VGA มีความละเอียด 640 x 480 จุดดังนั้นที่กักจุดในการลงตำแหน่งรูปภาพของ 2 จอภาพจะไม่ตรงกัน ถ้าจอ MGA วางรูปที่กัก 700x300 จะเห็นว่าแกน X นั้นภาพได้หลุดออกจากจอภาพของกักแกน X ของจอภาพ VGA แล้ว ดังนั้นในการเขียนโปรแกรมในกราฟิกโหมด ต้องพิจารณาถึงส่วนนี้เป็นอย่างมาก

3.2.1 คำสั่งในกราฟิกโหมด

คำสั่งต่าง ๆ ของภาษา C ที่ใช้ในกราฟิกโหมด จะบรรจุอยู่ในไฟล์ GRAPHIC.H และ GRAPHICS.LIB

3.2.1.1 ฟังก์ชัน Initgraph()

เป็นฟังก์ชันที่ใช้เพื่อต้องการเข้าสู่สภาวะกราฟิกโดยฟังก์ชันนี้จะทำหน้าที่โหลดเพิ่มข้อมูลที่ เป็น Graphics Driver ที่อยู่ในดิสก์มาเก็บไว้ในหน่วยความจำรูปแบบ

```
intergraph (*driver, *mode, *part);
```

driver เป็นตำแหน่งของตัวแปรจำนวนเต็ม ที่จะทำหน้าที่กำหนดชนิดของแฟ้มข้อมูลที่ เป็น Graphics Driver การกำหนดชนิดของแฟ้มข้อมูลนี้ สามารถกำหนดเป็นชื่อแอมคโตรหรือค่าคงที่ได้ดังนี้ถ้าชื่อแอมคโตรเป็น Detect ทำการตรวจสอบชนิดของวงจรที่แสดงผลให้เองพร้อมกับจะโหลดเพิ่มข้อมูล Graphics Driver สำหรับวงจรนั้นเข้ามาเองโดยอัตโนมัติ

mode เป็นการกำหนดขนาดของความละเอียดในการแสดงผลของแต่ละ Graphics driver ซึ่งจะสามารถกำหนดเป็นค่าแอมคโตรหรือค่าคงที่

path คือการกำหนดเส้นทางเพื่อค้นหาแฟ้มข้อมูล Graphic Driver (แฟ้มข้อมูล

ถือว่าเพิ่มข้อมูล Graphic Driver นั้นอยู่ที่ Working Directory

3.2.1.2 ฟังก์ชัน Closegraph(), Restorecrtmode()

เป็นฟังก์ชันที่ยกเลิกภาวะกราฟฟิกกลับไปสู่ภาวะข้อความ

รูปแบบ

```
closegraph();
```

หรือ

```
restorecrtmode
```

3.2.1.3 ฟังก์ชัน Line()

เป็นฟังก์ชันที่ใช้ในการวาดรูปเส้นตรงโดยต้องกำหนดจุดเริ่มต้น และจุดสุดท้ายของ

เส้น

รูปแบบ

```
line (startx,starty,endx, endy);
```

startx,starty เป็นการกำหนดจุดเริ่มต้นของเส้น

endx, endy เป็นการกำหนดจุดสุดท้ายของเส้น

3.2.1.4 ฟังก์ชัน Circle()

เป็นฟังก์ชันที่ใช้ในการวาดรูปวงกลมโดยต้องมีการกำหนดจุดศูนย์กลางวงกลม และความยาวของรัศมีเป็นจำนวนจุด

รูปแบบ

```
circle (x,y,radius)
```

x,y เป็นตำแหน่งของจุดศูนย์กลางของวงกลม
radius ความยาวรัศมี

3.2.1.5 ฟังก์ชัน Ellipse()

เป็นฟังก์ชันที่ใช้ในการวาดรูปวงรี

รูปแบบ

```
ellipse (x,y,angle,endangle,xradius,yradius);
```

x,y เป็นตำแหน่งจุดศูนย์กลางของวงรี
angle มุมเริ่มต้นของอาร์ค (arc) ของวงรี
endangle มุมสุดท้ายของอาร์ค (arc) ของวงรีถ้ามุมเริ่มต้นเป็น 0 มุมสุดท้ายเป็น 360 จะได้รูปวงรีทั้งรูป
xradius รัศมีทางแกน x
yradius รัศมีทางแกน y

3.2.1.6 ฟังก์ชัน rectangle()

เป็นฟังก์ชันที่ใช้สร้างรูปสี่เหลี่ยม

รูปแบบ

```
rectangle (left,top,right,bottom);
```

left, top เป็นการกำหนดจุดเริ่มต้น ของมุมซ้ายด้านบนของรูปสี่เหลี่ยมโดย
left เป็นตำแหน่งคอลัมน์ top เป็นตำแหน่งแถว
right, bottom เป็นการกำหนดจุด ที่เป็นมุมที่อยู่ขวาด้านล่างของรูปสี่เหลี่ยมโดย
right เป็นตำแหน่งคอลัมน์ bottom เป็นตำแหน่งแถว

3.2.1.7 ฟังก์ชัน Putpixel()

เป็นฟังก์ชันที่ให้สีกับจุดที่ตำแหน่งต่าง ๆ ตามต้องการ

รูปแบบ

```
putpixel (x,y,color);
```

x,y เป็นตำแหน่งของจุดที่ต้องการระบายสี
color เป็นสีที่ต้องการระบาย ถ้าไม่กำหนดจะระบายสีขาวให้

3.2.8.8 ฟังก์ชัน Setcolor()

เป็นฟังก์ชันที่ใช้กำหนดสี ที่จะใช้ในการวาดรูป

รูปแบบ

```
set (color);
```

color เป็นสีที่ต้องการใช้วาดรูป

BLACK = 0

BLUE = 1

GREEN = 2

CYAN = 3

RED	=	4
MAGENTA	=	5
BROWN	=	6
LIGHT	=	7
DARKGRAY	=	8
LIGHTBLUE	=	9
LIGHTGREEN	=	10
LIGHTCYAN	=	11
LIGHTRED	=	12
LIGHTMAGENTA	=	13
YELLOW	=	14
WHITE	=	15

3.2.1.9 ฟังก์ชัน Setfillstyle()

เป็นฟังก์ชันที่ใช้กำหนดรูปแบบของการระบายสีในบริเวณที่ต้องการ
รูปแบบ

```
setfillstyle (pattern,color);
```

pattern เป็นรูปแบบที่ต้องการระบายสี ซึ่งสามารถกำหนดได้เป็นชื่อแมคโคร หรือ
ค่าคงที่ดังตัวอย่างตามตารางที่ 2.3

ชื่อแมคโคร	ค่าคงที่	ลักษณะรูปแบบ
EMPTY_FILL	0	ระบายด้วยจุดสีพื้น
SOLID_FILL	1	ระบายภาพด้วยลายเส้นทึบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อแมคโคร	ค่าคงที่	ลักษณะรูปแบบ
LINE_FILL	2	ระบายภาพด้วยเส้นในแนวนอน
LTSLASH_FILL	3	ระบายด้วยเส้นเอียงขวา
SLASH_FILL	4	ระบายด้วยเส้นเอียงขวา(แบบหนา)
BKSLASG_FILL	5	ระบายด้วยเส้นเอียงซ้าย(แบบหนา)
LTBKSLASH_FILL	6	ระบายด้วยเส้นเอียงซ้าย
HATCH_FILL	7	ระบายด้วยลวดเส้นตาข่าย
XHATCH_FILL	8	ระบายด้วยลวดเส้นตาข่ายแบบหนา
INTERLEAVE_FILL	9	ระบายด้วยลวดเส้นตาข่ายเอียง

ตารางที่ 3.3 ชื่อแมคโคร และค่าคงที่ในการเติม

3.2.1.10 ฟังก์ชัน floodfill()

เป็นฟังก์ชันที่ใช้ระบายสีภายในรูปที่วาดขึ้น

รูปแบบ

```
floodfill (x,y,bordercolor);
```

x, y เป็นการกำหนดจุดที่อยู่ในรูปที่ต้องการระบายสี ถ้ากำหนดจุดอยู่นอกรูปจะระบายสีรอบนอกทั้งหมด ยกเว้นบริเวณภายในรูป

$bordercolor$ เป็นขอบเขตของสีที่ต้องการให้ระบาย

3.2.1.11 ฟังก์ชัน outtextxy()

เป็นฟังก์ชันที่ใช้เมื่อต้องการนำเอาข้อความที่เป็นภาวะข้อความมาแสดงในสภาวะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

กราฟิก

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบ

```
outtextxy (x,y,str);
```

x, เป็นจุดเริ่มต้นที่จะพิมพ์ข้อความ

st เป็นข้อความ หรือตัวแปรชุดที่ต้องการนำมาแสดงผลในภาวะ กราฟพิกถ้าเป็นข้อความ ต้องเขียนอยู่ในเครื่องหมายคำพูด (".....")

3.2.1.12 ฟังก์ชัน arc()

เป็นฟังก์ชันที่ใช้เมื่อต้องการเขียนส่วนของวงกลม

รูปแบบ

```
arc (x,y,stangle,endangle,radius);
```

x,y เป็นตำแหน่งจุดศูนย์กลางของส่วนโค้ง

stangle มุมเริ่มต้นของอาร์ค (arc)

endangle มุมสุดท้ายของอาร์ค (arc)

radius รัศมีทางจากจุดศูนย์กลางถึงเส้นโค้ง

3.2.1.13 ฟังก์ชัน bar()

ระบายพื้นที่ภายในสี่เหลี่ยม

รูปแบบ

```
bar (left,top,right,bottom);
```

left,top เป็นการกำหนดจุดเริ่มต้น ของมุมซ้ายด้านบนของพื้นที่ สีเหลี่ยมโดย
 left เป็นตำแหน่งคอลัมน์ top เป็นตำแหน่งแถว
 right,bottom เป็นกหนดจุดที่เป็นมุมขวาด้านล่างของพื้นที่สีเหลี่ยมโดย right เป็น
 ตำแหน่งคอลัมน์ bottom เป็นตำแหน่งแถว

3.2.1.14 ฟังก์ชัน bar3d()

เป็นฟังก์ชันที่ใช้เขียนกราฟแท่ง 3 มิติ มีลักษณะเป็นสีเหลี่ยมเช่นเดียวกับฟังก์ชัน
 bar() สิ่งที่แตกต่างกันก็คือการแสดงความลึกด้วย

รูปแบบ

```
bar3d (left,top,right,bottom,depth,topflag);
```

left,top เป็นการกำหนดจุดเริ่มต้นของมุมซ้ายด้านบน ของพื้นที่สีเหลี่ยมโดย
 left เป็นตำแหน่งคอลัมน์ top เป็นตำแหน่งแถว
 right,bottom เป็นการกำหนดจุดที่เป็นมุมขวาด้านล่าง ของพื้นที่สีเหลี่ยมโดย
 depth เป็นการกำหนดความลึก
 topflag เป็นพารามิเตอร์ แบบบูลีนในกรณีที่ เป็น 1 จะมีเส้นส่วนบนของ
 กราฟแท่งถ้าเป็น 0 จะไม่มีเส้นส่วนบน ซึ่งใช้สำหรับการเขียน
 กราฟซ้อนกัน

3.2.1.15 ฟังก์ชัน cleardevice()

ทำหน้าที่ล้างจอภาพให้ว่าง โดยระบายสีทั้งจอให้เป็นสีตามที่ระบุไว้ในฟังก์ชัน
 setbkcolor() แล้ววางตำแหน่งเคอร์เซอร์ไว้ที่มุมบนซ้าย

รูปแบบ

```
cleardevice();
```

3.2.1.16 ฟังก์ชัน clearviewport()

ทำหน้าที่ล้างช่องหน้าต่างที่กำลังทำงานอยู่ ให้ว่างโดยทำให้เป็นสีพื้นตามที่สั่งไว้แล้ว แล้ววางเคอร์เซอร์ไว้ที่มุมบนซ้ายของช่องหน้าต่างนั้น

รูปแบบ

```
clearviewport();
```

3.2.1.17 ฟังก์ชัน fillellipse()

เป็นฟังก์ชันที่ใช้เขียนรูปวงรีครอบวงหรือระบายสีตามที่สั่งไว้แล้ว

รูปแบบ

```
filellipse (x,y,xradius,yradius);
```

x,y เป็นตำแหน่งจุดศูนย์กลางของวงรี

xradius เป็นรัศมีจากจุดศูนย์กลางของวงกลม ถึงส่วนโค้งตามแนวนอน x

yradius เป็นรัศมีจากจุดศูนย์กลางของวงกลม ถึงส่วนโค้งตามแนวนอน y

3.2.1.18 ฟังก์ชัน drawpoly()

เป็นฟังก์ชันที่ลากเส้นตามจำนวนที่เราต้องการ

รูปแบบ

```
drawpoly (numpoint, polypoints);
```

`numpoint` เป็นจำนวนเส้นโดยลากจากจุดที่ 1 ไปยังจุดที่ 2 จากจุดที่ 2 จะลากต่อไปเรื่อย ๆ จนถึงจุดสุดท้าย

`polypoints` เป็นข้อมูลแบบตัวแปรชุด (array) และเป็นคู่ ๆ ประจำตำแหน่ง x และ y

3.2.1.19 ฟังก์ชัน `fillpoly()`

เป็นฟังก์ชันที่ใช้ระบายพื้นที่ภายในรูปหลายเหลี่ยมที่ลาก โดยคำสั่ง `drawpoly` ในกรณีที่เส้นสุดท้ายไม่บรรจบกัน การระบายจะระบายโดยให้จุดเริ่มต้นกับจุดสุดท้ายเชื่อมต่อกัน

รูปแบบ

```
fillpoly (numpoint, polopoint);
```

`numpoint` เป็นจำนวนเส้นโดยลากจากจุดที่ 1 ไปยังจุดที่ 2 จากจุดที่ 2 จะลากต่อไปเรื่อย ๆ จนถึงจุดสุดท้าย

`polypoints` เป็นข้อมูลแบบตัวแปรชุด และเป็นคู่ ๆ ประจำตำแหน่ง x และ y

3.2.1.20 ฟังก์ชัน `settextstyle()`

เป็นฟังก์ชันที่ใช้กำหนดแบบตัวอักษร (font) ทิศตั้ง หรือนอน (direction) และขนาด (size)

รูปแบบ

```
settextstyle (font,direction,size);
```

`font` คือแบบตัวอักษรตั้งแต่ 1 ถึง 4

`direction` คือแนวตัวอักษร 0 คือการเขียนแนวนอน 1 คือการเขียนแนวตั้ง

`size` คือขนาดตัวอักษร ตั้งแต่ 1 ถึง 10 (จากใหญ่ไปเล็ก) การใช้ฟังก์ชันนี้

ในแผ่นดิสก์ต้องมีไฟล์ *.chr อยู่ด้วย

3.2.2 คำสั่งในการจัดการแฟ้มข้อมูล

ในโครงการนี้มีความจำเป็นที่จะต้องยุ่งกับข้อมูล (DATA) ของผู้เขียนจะต้องมีการศึกษาถึงการจัดเก็บข้อมูล และการนำกลับมาใช้ใช้นั้นจึงต้องพิจารณาถึงคำสั่งที่เกี่ยวกับการจัดการแฟ้มข้อมูล ซึ่งฟังก์ชันเหล่านี้บรรจุอยู่ในแฟ้มข้อมูล (*.h และ *.lib)

3.2.2.1 ความหมายและลักษณะการประมวลผลแฟ้มข้อมูล

แฟ้มข้อมูลหมายถึง การรวบรวมข้อมูลที่สัมพันธ์กัน หรือเกี่ยวข้องกันมาไว้ที่เดียวกัน เพื่อให้สะดวกในการเรียกใช้ และการค้นหาข้อมูลดังนั้นในภาษา ซี จะมองข้อมูลที่เก็บอยู่ในแฟ้มมีลักษณะเป็นข้อมูลที่อยู่ต่อเนื่องกัน ตั้งแต่ต้นแฟ้มของข้อมูลจนจบแฟ้มข้อมูล โดยไม่มีการแบ่งระเบียบ (record) เหมือนกับภาษาระดับสูงทั่วไปในการประมวลผลแฟ้มข้อมูลแต่ละครั้ง ผู้เขียนโปรแกรมจะเป็นผู้กำหนดขอบเขตในการนำข้อมูลไปประมวลผลเอง

ลักษณะการประมวลผลแฟ้มข้อมูลโดยทั่วไป จะแบ่งออกเป็น 2 แบบใหญ่ ๆ คือ

- การบันทึกข้อมูลลงในแฟ้ม
 - การอ่านข้อมูลที่เก็บอยู่ในแฟ้มข้อมูลออกมาทำงาน
- การทำงานทั้ง 2 แบบนี้ จะมีขั้นตอนการทำงานดังนี้

การบันทึกข้อมูล	การอ่านข้อมูล
1. เปิดแฟ้มข้อมูล (Open file) 2. การบันทึกข้อมูลลงในแฟ้มข้อมูล (write file)	1. เปิดแฟ้มข้อมูล (open file) 2. การอ่านข้อมูลจากแฟ้มข้อมูล (Read file)

ตารางที่ 3.4 ขั้นตอนการเปิดและปิดแฟ้มข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2.2 ฟังก์ชันที่ใช้ในการประมวลผลเพิ่มข้อมูล

ฟังก์ชันที่ใช้ในการประมวลผลเพิ่มข้อมูล แบ่งออกได้เป็น 2 ระดับใหญ่ ๆ คือ

- ฟังก์ชันมาตรฐานในการรับข้อมูล และแสดงผลข้อมูล (Standard I/O) เป็นฟังก์ชันที่เหมาะสมกับการเขียนโปรแกรมที่มีการประมวลผ่านบัฟเฟอร์
- ฟังก์ชันระดับระบบที่ใช้ในการรับข้อมูล และแสดงผลข้อมูล (System level I/O) เป็นฟังก์ชันที่เหมาะสมกับการเขียนโปรแกรมโดยตรงไม่ต้องใช้บัฟเฟอร์ ฟังก์ชันต่าง ๆ เหล่านี้จะเข้าถึง (access) โดยตรงกับโปรแกรมระบบดำเนินการ (Operating System) ทำให้มีความเร็วในการประมวลผลสูง และประหยัดขนาดของหน่วยความจำ

3.2.2.3 ชนิดของแฟ้มข้อมูล

เนื่องจากเทอร์โบซี เป็นคอมไพเลอร์ที่ใช้งานโดย เอ็มเอส ดอส รหัสพิเศษบางอย่าง เช่น \n (new line) มีความหมายถึงการขึ้นบรรทัดใหม่ในภาษา ซี แต่เมื่อนำมาบันทึกลงแฟ้มข้อมูลที่จะนำไปใช้บน เอ็มเอส ดอส ความหมายจะเปลี่ยนไป ดังนั้นเพื่อให้ใช้แฟ้มข้อมูลที่สร้างขึ้นด้วยภาษา ซี สามารถใช้ร่วมกับซอฟต์แวร์อื่น ๆ บนเอ็มเอส ดอสได้จึงแบ่งชนิดแฟ้มข้อมูลออกเป็น 2 ชนิดคือ

- Text File
- Binary File
- Text File ในการกำหนดแฟ้มข้อมูลเป็นชนิดเท็กซ์ไฟล์ ถ้ามีการบันทึกลงแฟ้มข้อมูลแล้วพบรหัส \n (new line) จะถูกเปลี่ยนเป็น carriage return และ linefeed ให้แล้วบันทึกลงในแฟ้มข้อมูลนั้นให้ในกรณีที่เป็นการอ่านข้อมูลจากแฟ้มข้อมูล ถ้าพบรหัส carriage return และ linefeed จะเปลี่ยนกลับเป็น new line ให้
- Binary File ในการที่กำหนดแฟ้มข้อมูลเป็นชนิด binary file ถ้าพบรหัส \n (new line) จะไม่มีการเปลี่ยนแปลงเป็น carriage return และ linefeed ให้

3.2.2.4 การเปิดแฟ้มข้อมูล (Open File)

การเปิดแฟ้มข้อมูลเป็นการประกาศว่า จะมีการใช้แฟ้มข้อมูลพร้อมกับกำหนดตัวชี้ตำแหน่งข้อมูลในแฟ้มให้กับแฟ้มข้อมูลที่ต้องการใช้นั้นการเปิดแฟ้มข้อมูลทำได้โดยใช้ฟังก์ชัน fopen() ซึ่งอยู่ในแฟ้มข้อมูล stdio.h และตัวชี้ตำแหน่งของข้อมูลในแฟ้มจะถูกกำหนดให้เป็นตัวแปรชนิด File โดยประกาศได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปแบบ
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
FILE *fp
```

fp เป็นตัวแปรชนิด FILE ที่ทำหน้าที่เป็นตัวชี้ตำแหน่งข้อมูลในแฟ้ม

สำหรับฟังก์ชันที่ใช้ในการเปิดแฟ้มข้อมูลจะมีรูปแบบดังนี้

รูปแบบ

```
FILE *fp
fp = Fopen(filename, mode);
```

filename เป็นชื่อของแฟ้มข้อมูลที่ต้องการเปิด

mode เป็นลักษณะของแฟ้มข้อมูลที่จะเปิดว่า ต้องการเปิดขึ้นมาเพื่อทำอะไรซึ่ง

มีหลายลักษณะดังนี้

mode		ความหมาย
text file	binary file	
"r" หรือ "rt"	"rb"	เปิดแฟ้มข้อมูลสำหรับการอ่านข้อมูล
"w" หรือ "wt"	"wb"	สร้างแฟ้มข้อมูลสำหรับการบันทึก
"a" หรือ "at"	"ab"	ใช้สำหรับบันทึกแฟ้มข้อมูลต่อจากแฟ้มข้อมูลเดิม
"r+" หรือ "r+t"	"r+b"	เปิดแฟ้มข้อมูลสำหรับการอ่าน / บันทึกข้อมูล

3.2.2.5 การปิดแฟ้มข้อมูล (Close File)

การปิดแฟ้มข้อมูลจะทำหลังจากที่สิ้นสุดการใช้แฟ้มข้อมูลนั้น ๆ แล้ว เพื่อให้หน้าข้อมูลที่เหลือในบัฟเฟอร์ทั้งหมดเข้าไปบันทึกในแฟ้มข้อมูล ดังนั้นแฟ้มข้อมูลที่มีการเปิดแฟ้มข้อมูลทุกแฟ้มจะต้องมีการปิดแฟ้มข้อมูลก่อนสิ้นสุดโปรแกรมเพื่อป้องกันข้อมูลสูญหาย

รูปแบบ

```
fclose (fp);
```

fp เป็นตัวชี้ตำแหน่งของข้อมูลในแฟ้มที่ต้องการปิด ถ้าการปิดแฟ้มข้อมูลเรียบร้อย
fclose จะให้ค่าศูนย์แต่ถ้าการเปิดแฟ้มข้อมูลไม่เรียบร้อยจะให้ค่าไม่ใช่ศูนย์ออกมา

3.2.2.6 การประมวลผลโดยใช้ฟังก์ชันมาตรฐานในการรับข้อมูลและแสดงผล (Standard I/O)

จะแบ่งการประมวลผลออกได้เป็น 3 ระดับคือ

- การประมวลผลระดับตัวอักษร (character level)
- การประมวลผลระดับข้อความ (String level)
- การประมวลผลระดับระเบียน (Record level)

3.2.3 การประมวลผลระดับตัวอักษร (character level)

เป็นการประมวลผลทีละ 1 ตัวอักษร โดยฟังก์ชันต่อไปนี้

- putc(), fputc()
- getc(), fgetc()

ฟังก์ชัน putc(), fputc() เป็นฟังก์ชันที่ใช้สำหรับบันทึกข้อมูลที่ละ 1 ตัวอักษร

เข้าไปเก็บไว้ในแฟ้มข้อมูล ตรงตำแหน่งที่ตัวชี้ตำแหน่งของข้อมูลในแฟ้มข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบ

```
put (ch,fp);
```

ch เป็นค่าคงที่ ตัวแปรชนิดอักขระที่ต้องการนำข้อมูลเข้าไปบันทึกไว้ในแฟ้มข้อมูล

fp เป็นตัวชี้ตำแหน่งข้อมูลในแฟ้ม

ฟังก์ชัน `getc()`, `fgetc()` เป็นฟังก์ชันที่ใช้สำหรับอ่านข้อมูลจากแฟ้มข้อมูลทีละ 1 ตัว อักขระ จากตำแหน่งที่ตัวชี้ตำแหน่งของข้อมูลในแฟ้มข้อมูลออกมา

รูปแบบ

```
char ch;
ch = getc(fp);
```

fp เป็นตัวชี้ตำแหน่งข้อมูลในแฟ้มของแฟ้มข้อมูลที่จะอ่าน การอ่านข้อมูลจากแฟ้มข้อมูลถ้า เป็นเท็กซ์ไฟล์จะอ่านจนจบแฟ้มข้อมูลแล้วให้ค่า EOF ออกมาเป็นการบอกว่าจบแฟ้มข้อมูลแต่ถ้าเป็น ไบนารี ไฟล์ จะต้องใช้ฟังก์ชัน `feof()` ตรวจสอบการจบของแฟ้มข้อมูล ในการตรวจสอบโดยใช้ ฟังก์ชัน `feof()` ถ้ายังไม่จบแฟ้มข้อมูล จะให้ค่าเป็นศูนย์ถ้าจบแฟ้มข้อมูลจะให้ค่าไม่ใช่ศูนย์ออกมา จากฟังก์ชัน

3.2.4 การประมวลผลระดับข้อความ (String level)

เป็นการประมวลผลทีละข้อความโดยใช้ฟังก์ชันต่อไปนี้

- `fputs()`, `fprint()`

- `fgets()`, `fscanf()`

ฟังก์ชัน `fputs()` เป็นฟังก์ชันที่ใช้บันทึกข้อมูลชนิดข้อความ (String) ลงไปในแฟ้มข้อมูล โดยไม่มีการบันทึกรหัส \0 ซึ่งแสดงการจบข้อความลงในแฟ้มข้อมูลด้วย ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบ

```
fputs (str,*fp);
```

str เป็นข้อความหรือตัวแปรชุดที่ต้องการนำข้อมูลไปบันทึกในแฟ้มข้อมูล

fp เป็นตัวชี้ตำแหน่งข้อมูลในแฟ้มผลที่ได้จากฟังก์ชันถ้าการบันทึกข้อมูลถูกต้องจะให้ค่าศูนย์ ถ้าการบันทึกข้อมูลไม่ถูกต้อง จะให้ค่าที่ไม่ใช่ศูนย์ออกมา

ฟังก์ชัน fprintf() เป็นฟังก์ชันทำงานเหมือนกับฟังก์ชัน printf() แต่เป็นการบันทึกข้อมูลลงแฟ้มข้อมูลเท่านั้น

รูปแบบ

```
fprintf (*fp,"control",arg-list);
```

fp เป็นตัวชี้ตำแหน่งข้อมูลในแฟ้ม

control เป็นรหัสรูปแบบของการบันทึกข้อมูลโดยมีรูปแบบ และการใช้เหมือนกันกับรหัสรูปแบบที่ใช้ในฟังก์ชัน printf()

arg-list เป็นค่าคงที่ ตัวแปรหรือนิพจน์ที่จะบันทึกลงในแฟ้มข้อมูลซึ่งเป็นชนิดตัวเลขหรือตัวอักษรก็ได้

การใช้ฟังก์ชัน fprintf() จะทำให้ค่าที่ลงในแฟ้มข้อมูลแต่ละค่าแยกจากกันได้โดยการเว้นระยะของรหัสรูปแบบทำให้ง่ายต่อการอ่านข้อมูล

ฟังก์ชัน fgets() เป็นฟังก์ชันที่ใช้อ่านข้อมูลชนิดข้อความจากแฟ้มข้อมูลมาเก็บไว้ในตัวแปรชุดที่กำหนด โดยสามารถกำหนดจำนวนอักขระที่อ่านเข้ามาได้

รูปแบบ

```
fgets (str,num,*fp);
```

str เป็นตัวแปรชุดที่จะเก็บข้อมูลที่อ่านมาจากแฟ้มข้อมูล

num เป็นจำนวนไบต์ของข้อมูลที่จะอ่านมาแต่ละครั้งเท่ากับ num-1 ตัว หรืออ่านจนจบพบเครื่องหมาย new line หรือ EOF

fp เป็นตัวชี้ตำแหน่งข้อมูลในแฟ้ม

ฟังก์ชัน fscanf() เป็นฟังก์ชันที่ใช้อ่านข้อมูลเข้าเหมือนกับฟังก์ชัน scanf() ต่างกันที่ fscanf() จะอ่านข้อมูลมาจากแฟ้มข้อมูลค่าที่ได้จากฟังก์ชันจะเท่ากับจำนวนค่าที่ต้องการอ่านเข้ามา ถ้าอ่านจนจบแฟ้มข้อมูลจะได้ค่า EOF เป็นค่าของฟังก์ชัน

รูปแบบ

```
fscanf (*fp,"control",arg-list);
```

arg-list เป็นตัวแปรที่ใช้ในการเก็บข้อมูลที่อ่านมาจากแฟ้มข้อมูล

control เป็นรหัสรูปแบบข้อมูลที่ใช้ในการอ่านข้อมูล

fp เป็นตัวชี้ตำแหน่งข้อมูลในแฟ้ม

3.2.5 การประมวลผลระดับระเบียน (Record Level)

เป็นการประมวลผลทีละระเบียน (record) โดยแต่ละระเบียนต้องมีความยาวคงที่ สำหรับการประมวลผลทีละระเบียนมีความยาวไม่คงที่จะยังไม่กล่าวถึง วิธีการประเมินผลในขั้นการประมวลผลทีละระเบียนจะใช้ฟังก์ชันในการประมวลผลดังต่อไปนี้

- fwrite()

- fread()

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับวารใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ซึ่งส่วนมากจะใช้กับแฟ้มข้อมูลชนิด ไบนารี ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน `fwrite()` ใช้ในการบันทึกข้อมูลจากหน่วยความจำไปเก็บไว้ใน
แฟ้มข้อมูลในตำแหน่งที่กำหนด

รูปแบบ

```
fwriter (*ptr, size, num, *fp);
```

`ptr` เป็นตำแหน่งของข้อมูลภายในหน่วยความจำที่เราต้องการนำไปบันทึกลงในแฟ้ม
ข้อมูล
`size` เป็นขนาดของข้อมูลที่จะบันทึกลงในแฟ้มข้อมูลว่าเป็นขนาดกี่ไบต์
`num` ใช้แสดงจำนวนระเบียนที่ต้องการจะบันทึก
`fp` เป็นตัวชี้ตำแหน่งข้อมูล

นอกจากนั้นฟังก์ชัน `fwrite()` ยังสามารถใช้บันทึกข้อมูลที่เป็นชนิดโครงสร้างได้โดยค่า
ที่ได้จากฟังก์ชัน `fwrite()` จะเป็นจำนวนครั้งของการบันทึกซึ่งถ้าไม่ตรงกับที่กำหนดไว้ `num` จะ
เกิดการผิดพลาดขึ้น

ฟังก์ชัน `fread()` เป็นฟังก์ชันที่ใช้อ่านข้อมูลจากแฟ้มข้อมูลเข้ามาเก็บไว้ในหน่วยความจำ
ตามตำแหน่งที่กำหนด

รูปแบบ

```
fread (*ptr, size, num, *fp);
```

`ptr` เป็นตำแหน่งของหน่วยความจำที่ต้องการนำข้อมูลที่อ่านจากแฟ้มข้อมูลมาเก็บไว้
`size` เป็นขนาดของข้อมูลที่อ่านเข้ามาว่ามีขนาดกี่ไบต์
`num` เป็นจำนวนระเบียนที่จะอ่านเข้ามาเก็บไว้ในหน่วยความจำ
`fp` เป็นตัวชี้ตำแหน่งข้อมูลในแฟ้ม

ค่าที่ได้จากฟังก์ชันจะเป็นจำนวนของค่าที่อ่านเข้ามาตามที่กำหนดไว้ใน `num` ถ้าการอ่านไม่ได้ตามที่กำหนดหรือพบ EOF ก่อนครบจำนวนที่กำหนดจะเกิดความผิดพลาดขึ้น ดังนั้นจึงควรใช้ฟังก์ชัน `feof()` ตรวจสอบการจบเพิ่มข้อมูลด้วยในการอ่านข้อมูล

3.2.5.1 การใช้อุปกรณ์มาตรฐาน

ในภาษา C ผู้เขียนโปรแกรมสามารถใช้ฟังก์ชันสำหรับการประมวลผลเพิ่มข้อมูลในการติดต่อกับอุปกรณ์ภายนอก เช่น การรับข้อมูลจากแป้นพิมพ์ หรือแสดงผลออกทางจอภาพ หรือเครื่องพิมพ์ได้โดยใช้ `stdin`, `stdout`, `stderr`, `stdaux`, `stdprn` ซึ่งไม่ใช่ตัวแปร แต่จะใช้เป็นค่าคงที่ที่ใช้แทนตัวชี้ตำแหน่งข้อมูลในแฟ้ม (file pointer) ในฟังก์ชันของแฟ้มข้อมูล

ค่าคงที่	ความหมาย
<code>stdin</code>	ใช้รับข้อมูลจากแป้นพิมพ์
<code>stdout</code>	ใช้แสดงผลทางจอภาพ
<code>stderr</code>	ใช้แสดงข้อผิดพลาด
<code>stdux</code>	ใช้ auxiliary port ของเครื่องสำหรับติดต่อกับอุปกรณ์ภายนอก
<code>stdprn</code>	ใช้แสดงผลทางแป้นพิมพ์

ตารางที่ 3.6 ค่าคงที่ที่ใช้ทำงานกับอุปกรณ์ Input / Output

3.2.5.2 บัฟเฟอร์ (Buffer).

ในการประมวลผลเพิ่มข้อมูลทุกครั้งเมื่อมีการเปิดแฟ้มข้อมูล (open file) เครื่องจะเตรียมบัฟเฟอร์ไว้ สำหรับใช้ในการประมวลผลทันที โดยถ้าเป็นการอ่านข้อมูลเครื่องจะนำข้อมูลจากแฟ้มข้อมูลเข้ามาเก็บไว้ในบัฟเฟอร์ก่อนนำไปทำการประมวลผลส่วนการบันทึกข้อมูลก็เช่นกัน เครื่องจะนำข้อมูลไปบันทึกลงในบัฟเฟอร์จนกว่าจะเต็มก่อนจึงจะนำข้อมูลในบัฟเฟอร์นั้นไปบันทึกลงแฟ้มข้อมูลอีกทีหนึ่งปัญหานี้มักเกิดขึ้นเสมอเมื่อมีการใช้บัฟเฟอร์ คือเมื่อมีการบันทึกข้อมูลลงแฟ้มข้อมูลอาจสูญหายได้ถ้าเกิดอะไรขึ้นกับโปรแกรม เช่น ไฟดับ และ ข้อมูลที่อยู่ในบัฟเฟอร์ที่ยังไม่ได้นำไปบันทึกลงในแฟ้มข้อมูลก็จะสูญหายไปได้ในกรณีที่ข้อมูล ยังไม่เต็มบัฟเฟอร์แล้วมีการปิดแฟ้มข้อมูล

เอกลีปิณีฉบับนี้จัดทำขึ้นเพื่อแจกจ่ายให้แก่นักเรียน นักศึกษา และบุคลากรในสังกัดสำนักงานคณะกรรมการการอุดมศึกษา กระทรวงศึกษาธิการ โดยไม่คิดค่า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังมีให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือเลิกการทำงานของโปรแกรมโดยใช้ฟังก์ชัน `exit()` จะทำให้ข้อมูลในบัฟเฟอร์ถูกบันทึกลงในแฟ้มข้อมูลให้เองโดยอัตโนมัติ

ดังนั้น ในเทอร์มินัลจึงมีฟังก์ชันที่ใช้สำหรับควบคุมการทำงานของบัฟเฟอร์ ให้เลือกใช้ได้ตามต้องการ

- `Setbuff()`
- `Setvbub()`, `fflushall()`

ฟังก์ชัน `Setbuff()` เป็นฟังก์ชันที่ใช้สำหรับให้ผู้เขียนโปรแกรมสามารถที่จะกำหนดขนาดของบัฟเฟอร์ขึ้นเอง โดยจะเขียนฟังก์ชันนี้ไว้หลังจากทำการเปิดแฟ้มข้อมูลแล้ว และถ้ามีการกำหนดขนาดของบัฟเฟอร์เป็นชื่อของ Macro เช่น `BUFSIZ` เครื่องจะเตรียมขนาดของบัฟเฟอร์ไว้ 512 ไบต์ ไว้แต่ถ้าชื่อ `NULL` จะไม่มีขนาดของบัฟเฟอร์ให้

รูปแบบ

```
setbuff (*fp, buff);
```

`fp` เป็นตัวชี้ตำแหน่งข้อมูลในแฟ้ม

`buf` เป็นขนาดของบัฟเฟอร์กำหนดเป็นไบต์

ฟังก์ชัน `Setvbub()` เป็นฟังก์ชันที่ใช้สำหรับให้ผู้เขียนโปรแกรมสามารถกำหนดขนาดของบัฟเฟอร์ขึ้นเองได้เหมือนกับฟังก์ชัน `setbuff()`

รูปแบบ

```
setvbub (*fp, buf, mode, size);
```

`buf` เป็นขนาดของตัวแปรชนิดอักขระที่จะใช้ในการประมวลผล

`size` เป็นขนาดของบัฟเฟอร์ที่ต้องกำหนดให้มีค่ามากกว่าศูนย์

`mode` เป็นวิธีการใช้บัฟเฟอร์ ซึ่งมีให้เลือกใช้ได้ 3 แบบคือ

- `_IONBF` แสดงว่าไม่มีบัฟเฟอร์สำหรับการประมวลผล
- `_IOFBF` แสดงว่าเครื่องจะเป็นผู้กำหนดขนาดของบัฟเฟอร์
- `_IOLBF` แสดงว่าจะมีการเตรียมบัฟเฟอร์ให้แบบ Line buffer คือเมื่อพบรหัสข้อมูลที่เป็น `\n` เครื่องจะนำข้อมูลออกจากบัฟเฟอร์ไปบันทึกลงในแฟ้มข้อมูล ถ้าเป็นการอ่านข้อมูลจะอ่านข้อมูลจากแฟ้มข้อมูลเข้ามา เก็บไว้ในบัฟเฟอร์จนพบรหัส `\n`

ฟังก์ชัน `fflush()` และฟังก์ชัน `fflushall()` เป็นฟังก์ชันที่ใช้สำหรับนำข้อมูลที่อยู่ในบัฟเฟอร์ไปบันทึกลงในแฟ้มข้อมูล ในกรณีที่มีการเปิดแฟ้มข้อมูลนั้นไว้เป็นแฟ้มข้อมูลใช้สำหรับ บันทึก แต่ถ้าเปิดแฟ้มข้อมูลไว้สำหรับการอ่านจะเป็นการลบข้อมูลในบัฟเฟอร์

รูปแบบ

```
fflush (*fo);
```

`fp` เป็นตัวชี้ตำแหน่งข้อมูลในแฟ้ม

ฟังก์ชัน `fflushall()` จะทำงานเหมือนกับฟังก์ชัน `fflush()` แต่จะเอาข้อมูลที่อยู่ในทุก ๆ บัฟเฟอร์ที่เตรียมไว้ บันทึกลงในแฟ้มข้อมูลให้หมดทุกแฟ้มข้อมูล

3.2.5.3 การประมวลผลแฟ้มข้อมูลแบบสุ่ม (Random Access To Files)

การประมวลผลที่กล่าวมาตำแหน่งของตัวชี้ตำแหน่งข้อมูลในแฟ้ม (file pointer) จะเปลี่ยนแปลงไปตามขนาดของข้อมูลที่ทำกรประมวลผลแต่ละครั้ง โดยเปลี่ยนแปลงอย่างเรียงลำดับซึ่งเราสามารถทำการเคลื่อนย้ายตำแหน่งของตัวชี้ตำแหน่งข้อมูลในแฟ้มไปยังตำแหน่งต่าง ๆ ในแฟ้มข้อมูลได้ตามต้องการโดยใช้ฟังก์ชันต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
`rewind(), fseek(), ftell()`
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน `rewind()` เป็นฟังก์ชันที่ใช้เมื่อต้องการย้ายตำแหน่งของตัวชี้ตำแหน่งข้อมูลในแฟ้มไปยังตำแหน่งเริ่มต้นของแฟ้มข้อมูล

รูปแบบ

```
rewind (*fp);
```

`fp` เป็นตัวชี้ตำแหน่งข้อมูลในแฟ้มของข้อมูลที่ต้องการย้ายไปยัง ตำแหน่งเริ่มต้นของแฟ้มข้อมูล

ฟังก์ชัน `fseek()` เป็นฟังก์ชันที่ใช้สำหรับย้ายตัวชี้ตำแหน่งข้อมูลในแฟ้ม ไปยังตำแหน่งที่ต้องการภายในแฟ้มข้อมูล การคำนวณหาตำแหน่งของตัวชี้ตำแหน่งข้อมูลในแฟ้มจะได้จากตำแหน่ง `origin` บวกด้วยค่าของ `offset` ที่กำหนดขึ้น

รูปแบบ

```
fseek (*fp, offset, origin);
```

`fp` เป็นตัวชี้ตำแหน่งข้อมูลในแฟ้มที่ต้องการย้ายตำแหน่ง

`origin` เป็นตำแหน่งที่ตัวชี้ตำแหน่งข้อมูลในแฟ้มย้ายไป โดยผู้เขียนโปรแกรมสามารถ

กำหนดตำแหน่งหลักในการย้ายได้ 3 แบบ คือ

- ตำแหน่งเริ่มต้นของแฟ้มข้อมูล (beginner of files)
- ตำแหน่งปัจจุบัน (current position)
- ตำแหน่งสุดท้ายของแฟ้มข้อมูล (End of file)

การกำหนดค่า `origin` ทำได้ 2 แบบ คือ กำหนดค่าด้วยชื่อ `macro` หรือค่าคงที่ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อ แมคโคร	ค่าคงที่	ความหมาย
SEEK-SET	0	ย้ายตัวชี้ตำแหน่งข้อมูลในแฟ้มไปยังตำแหน่งเริ่มต้นข้อมูล
SEEK-CUR	1	กำหนดให้ตัวชี้ตำแหน่งข้อมูลในแฟ้มที่อยู่ตำแหน่งปัจจุบัน
SEEK-END	2	ให้ย้ายตัวชี้ตำแหน่งข้อมูลในแฟ้มไปยังตำแหน่ง สิ้นสุดท้ายของแฟ้มข้อมูล

ตารางที่ 3.7 ค่าแมคโคร ค่าต่าง ๆ

offset คือ ระยะห่างจาก origin ไปยังตำแหน่งที่ต้องการมีขนาดเป็นไบต์ถ้าค่าของ offset กำหนดมากกว่า 64 k ไบต์ ต้องกำหนดชนิดของค่าเป็น long

ฟังก์ชัน ftell() เป็นฟังก์ชันที่ใช้บอกตำแหน่งของตัวชี้ตำแหน่งข้อมูลในแฟ้มที่ให้อยู่ในขณะนั้น ถ้าเป็นแฟ้มข้อมูลชนิดไบนารี ค่าที่ได้จากฟังก์ชัน คือ จำนวนไบต์จากตำแหน่งเริ่มต้นของแฟ้มข้อมูลจนถึงตำแหน่งที่ตัวชี้ตำแหน่งข้อมูลในแฟ้มให้อยู่ในขณะนั้นส่วนแฟ้มข้อมูลชนิด เท็กซ์ ค่าที่ได้จากฟังก์ชันอาจไม่ถูกต้องเพราะ จะมีการเปลี่ยนแปลงข้อมูลระหว่าง carriage return และ line feed กับ new line ทำให้จำนวนไบต์สำหรับการอ่าน และบันทึกต่างกัน ถ้าเกิดการผิดพลาดฟังก์ชันนี้จะให้ค่า -1 ออกมา

รูปแบบ

```
ftell (*fp);
```

fp เป็นตัวชี้ตำแหน่งข้อมูลในแฟ้มที่จะขอค่าตำแหน่ง

การลบข้อมูลออกจากแผ่นดิสก์จะทำได้โดยใช้ฟังก์ชัน remove() ถ้าการลบแฟ้มข้อมูลเสร็จเรียบร้อย ฟังก์ชันนี้จะให้ค่า 0 ออกมา ถ้าการลบมีข้อผิดพลาดจะให้ค่า -1 ออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบ

```
remove (fname);
```

fname เป็นชื่อแฟ้มข้อมูลที่ต้องการลบออกจากแผ่นดิสเกต

ในการที่จะเปลี่ยนชื่อแฟ้มข้อมูลสามารถทำได้ โดยใช้ฟังก์ชัน rename() ถ้าการเปลี่ยนชื่อแฟ้มข้อมูลเสร็จเรียบร้อย ฟังก์ชันจะให้ค่า 0 ออกมา ถ้าการเปลี่ยนชื่อเกิดการผิดพลาดฟังก์ชันนี้จะให้ค่าที่ไม่ใช่ 0 ออกมา

รูปแบบ

```
rename (old name, newname);
```

oldfname เป็นชื่อแฟ้มข้อมูลที่ต้องการเปลี่ยนชื่อใหม่

newfname เป็นชื่อแฟ้มข้อมูลใหม่

3.2.6 คำสั่งในการควบคุมทิศทางโปรแกรม

ในการเขียนโปรแกรมคอมพิวเตอร์นั้น ต้องมีการควบคุมทิศทางการทำงานของโปรแกรม ซึ่งก็คือ คำสั่งในการวนลูป คำสั่งเงื่อนไขตัดสินใจ ซึ่งพอสรุปได้ดังนี้

- คำสั่งให้ไปทำงานโดยไม่มีเงื่อนไข (Unconditional branch statement) ซึ่งได้แก่คำสั่ง goto
- คำสั่งให้ไปทำงานโดยมีเงื่อนไข (ondition branch statement) ซึ่งได้แก่คำสั่ง if, switch
- คำสั่งให้ไปทำงานเป็นวงจร (loop Control statement)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.6.1 คำสั่ง goto

เป็นคำสั่งให้ไปทำงานยังคำสั่งหนึ่งคำสั่งใดโดยไม่มีเงื่อนไข

รูปแบบ

```
goto statement-label
```

statement-label หมายถึง ชื่อประจำคำสั่ง (label) ที่ต้องการไปทำงาน

3.2.6.2 คำสั่ง IF

เป็นคำสั่งที่ให้ไปทำงานยังคำสั่งใดคำสั่งหนึ่ง โดยมีเงื่อนไขให้ตัดสินใจอย่างหนึ่งอย่างใดก่อนแล้วจึงไปทำงาน โดยผลของการตัดสินใจจะมีโอกาสเป็นไปได้ 2 ทางคือ

- จริงหมายความว่า มีค่าไม่เท่ากับศูนย์
- ไม่จริงหมายความว่า มีค่าเท่ากับศูนย์

รูปแบบ

```
if (condition)
    statement-1
else
    statement-2
```

condition หมายถึง เป็นเงื่อนไขที่สร้างขึ้นให้คอมพิวเตอร์ตัดสินใจอย่างใดอย่างหนึ่งก่อนที่จะไปทำงานโดยถ้าเงื่อนไขเป็นจริงจะไปทำงานยัง statement-1 แต่ถ้าเงื่อนไขไม่เป็นจริงมันจะไปทำงานยัง statement-2

statement-1 หมายถึง คำสั่งใด ๆ ในภาษา C หนึ่งคำสั่ง

statement-2 หมายถึง คำสั่งใด ๆ ในภาษา C หนึ่งคำสั่งนำไปใช้ประโยชน์ด้านการคำนวณว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากนั้นใน statement-1 และ statement-2 อาจจะเป็น if ที่ซ้อนกันอีกได้ซึ่ง เรียกว่า Nested If

รูปแบบ

```
if (condition-1)
    statement-1;
else if (condition-2)
    statement-2;
```

หมายความว่าถ้า Condition-1 จริงจะทำ statement-1 แล้วออกไปทำคำสั่งที่อยู่ ต่อจากคำสั่ง if เสร็จ

- ถ้า Condition-1 ไม่จริงจะทำ Condition-2
- ถ้า Condition-2 จริงจะทำ Statement-2 แล้วออกไปทำคำสั่งที่อยู่ต่อจากคำสั่ง if เสร็จ
- ถ้า Condition-2 ไม่จริงจะทำ Condition-3

ในรูปแบบคำสั่ง if เงื่อนไขสามารถมีมากกว่า 1 เงื่อนไขได้ ซึ่งเรียกว่า Compound Condition โดยใช้เครื่องหมายตรรก && (AND), || (OR), ! (NOT) เป็นตัวเชื่อม

3.2.6.3 คำสั่ง Switch

เป็นคำสั่งให้ไปทำคำสั่งใดคำสั่งหนึ่งตามที่ต้องการ โดยมีทางเลือกให้ไปทำคำสั่งได้ หลาย ๆ ทางเลือก และเลือกไปทำคำสั่งใดก็ขึ้นอยู่กับค่าของตัวแปรที่ทำหน้าที่ควบคุมการทำงานของคำสั่ง Switch นั้น

รูปแบบ

```

switch (variable) {
    case constant-1;
        statement sequent
        break;
    case constant-2;
        statement sequent
        break;
    case constant-n;
        statement sequent
        break;
    default;
        statement sequent
}

```

Variable หมายถึงตัวแปรที่ทำหน้าที่ควบคุมการทำงานของคำสั่ง

Switch ซึ่งจะเป็นตัวแปรชนิด int, char หรือนิพจน์ก็ได้โดยค่าของตัวแปรนี้ถ้าตรงกับ Constant ที่อยู่ใน case ใดก็ตาม ก็จะทำคำสั่งที่อยู่ใน case นั้นนั่นเอง

Constant-1, Constant-2, ..., Constant-n เป็นเงื่อนไขที่จะให้เลือกไปทำงานตามที่ต้องการ โดยจะต้องมีค่าเป็นค่าคงที่ชนิด int, char เท่านั้น

Statement sequent หมายถึง กลุ่มของคำสั่งที่ต้องการให้ทำงานตามเงื่อนไขแต่ละ case

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Break** เป็นคำสั่งให้ออกจากบล็อกของ case แต่ละ case แล้วไปทำคำสั่งที่อยู่ต่อจากคำสั่ง switch
- Default** เป็นกรณีที่ไม่ใช่ค่าของ Variable ตรงกับเงื่อนไขใดใน case เลข จะลงมาทำตามคำสั่งใน default นี้ทั้งหมดแล้วออกจากคำสั่ง switch ไปทำคำสั่งที่อยู่ต่อจากคำสั่ง switch นั้นเอง

3.2.6.4 คำสั่ง While

เป็นคำสั่งที่เขียนขึ้น เมื่อต้องการให้คอมพิวเตอร์ทำงานเป็นวงจร (loop) มี 2 รูปแบบดังหน้าถัดไป

รูปแบบที่ 1.

```
while (condition)
    statement;
next statement;
```

รูปแบบที่ 2.

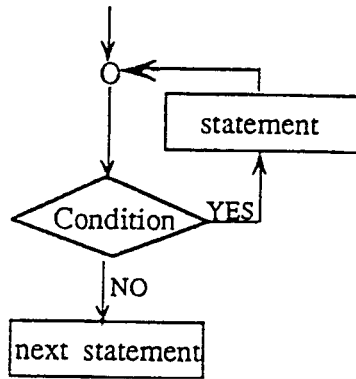
```
while (condition)
{
    statement;
    statement;
}
next statement;
```

Condition หมายถึง เงื่อนไขที่ต้องการให้คอมพิวเตอร์ตัดสินใจถ้าเงื่อนไขจริงจะทำคำสั่งที่อยู่ใน while ถ้าเงื่อนไขไม่จริงจะออกไปทำคำสั่งที่อยู่ต่อจากคำสั่ง while

Statement หมายถึง คำสั่งใด ๆ ในภาษา C ถ้ามีมากกว่าหนึ่งคำสั่งต้องเขียนอยู่ในลักษณะของบล็อกดังรูป

Next Statement หมายถึง คำสั่งใด ๆ ที่อยู่ต่อจากคำสั่ง while ลักษณะการทำ

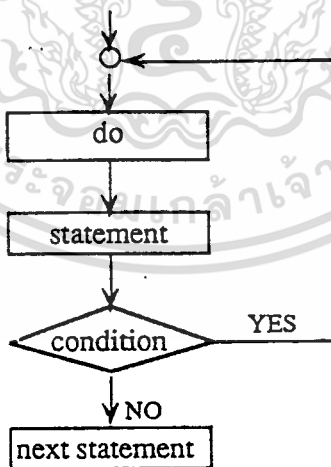
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่สามารถนำไปใช้ประโยชน์อื่น ๆ ได้
งานของ while สามารถอธิบายในลักษณะผังงานได้ดังนี้
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 ผังการทำงานของคำสั่ง While

3.2.6.5 คำสั่ง do-while

เป็นคำสั่งที่เขียนกันเมื่อต้องการให้คอมพิวเตอร์ทำงานเป็นวงจรเหมือนกับ while แต่คำสั่ง do-while จะทำคำสั่งที่ต้องการก่อนแล้วจึงตรวจสอบเงื่อนไข ถ้าเป็นจริงจะกลับขึ้นมาทำคำสั่งที่ต้องการใหม่ จนกว่าเงื่อนไขจะไม่เป็นจริงจะออกจาก loop ของ do-while ลักษณะการทำงานเขียนเป็นผังงานดังนี้



รูปที่ 3.9 ผังการทำงานของคำสั่ง do-while

รูปแบบที่ 1.

```
do
    statement;
while (condition);
```

รูปแบบที่ 2.

```
do
{
    statement;
    statement;
    .
    .
}
while (condition);
```

Statement หมายถึง คำสั่งใด ๆ ในภาษา ซี ถ้ามีมากกว่า 1 คำ

Condition หมายถึง เงื่อนไขที่ต้องการให้คอมพิวเตอร์ตัดสินใจ ถ้าเงื่อนไขจริงจะ
กลับไปทำตามคำสั่ง do ใหม่ถ้าเงื่อนไขไม่จริงจะออกนอก
วงจรถือ ลงมาทำคำสั่งที่อยู่ต่อจาก while เลย

3.2.6.6 คำสั่ง for

เป็นคำสั่งให้มีการทำงานอย่างเป็นวงจรรวนซ้ำเหมือนกับ do-while โดยมีรูปคำสั่ง

ดังนี้

รูปแบบ

```
for (expression;condition;increment)statement
```

Expression หมายถึง คำสั่งที่ใช้สำหรับ กำหนดค่าเริ่มต้นให้กับตัวแปรที่จะทำหน้าที่

ควบคุมวงจรถงคำสั่ง for

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมอนุญาตให้นำไปใช้ประโยชน์ด้านธุรกิจ
Condition หมายถึง เงื่อนไขที่ใช้ในการควบคุม การทำงานของตัวแปรที่ทำหน้าที่
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหา และต้องอ้างอิงเจ้าของเอกสารทุกครั้งที่มีการนำใบใช้

- ควบคุมวงจร โดยถ้าเงื่อนไขที่เป็นจริงจะทำในวงจร แต่ถ้าเงื่อนไขไม่จริงจะออกจากวงจรถัดที่
- Increment หมายถึง ค่าที่จะกำหนดให้เพิ่มขึ้นของตัวแปรที่จะทำหน้าที่ควบคุมวงจรในแต่ละรอบ
- Statement หมายถึง คำสั่งใด ๆ ในภาษา C ถ้ามีมากกว่า 1 คำต้องกันบล็อกรหัสกับกลุ่มคำสั่งเหล่านั้นด้วย

3.2.6.7 คำสั่ง Break

เป็นคำสั่งที่ให้ออกจากการทำงาน ของคำสั่ง case หรือ ออกจากการทำงานเป็นวงจร เพื่อให้ไปทำงานยังคำสั่งที่อยู่ต่อจากคำสั่งที่ให้ทำงานเป็นวงจร นั่นคือจะเป็นคำสั่งที่ใช้ทำงานในคำสั่ง switch, for, while, do while นั้นเอง

3.2.6.8 คำสั่ง Continue

เป็นคำสั่งที่ให้กลับไปทำยังคำสั่งแรกของคำสั่งควบคุม while ,do while ,for

3.2.6.9 ฟังก์ชัน scanf()

เป็นฟังก์ชันที่ใช้ในการรับข้อมูล จากแป้นพิมพ์ (key-board) มาเก็บไว้ในตัวแปรรูปแบบ

```
scanf (control,argument list);
```

Argument List เป็นตัวแปรที่จะรับค่าจากแป้นพิมพ์มาเก็บไว้ ถ้ามีมากกว่า 1 คำจะต้องแยกกันด้วยเครื่องหมายคอมม่า (,) และตัวแปรทุกตัวที่ใช้จะต้องนำหน้าด้วยเครื่องหมาย & ทุกตัว ยกเว้นตัวแปรชุด

Control เป็นรหัสรูปแบบ ที่ใช้ในการรับข้อมูลซึ่งต้องเขียนอยู่ภายในเครื่องหมาย " " และมีรหัสรูปแบบเหมือนกับรหัสรูปแบบที่ใช้ในคำสั่ง printf

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.6.10 ฟังก์ชัน getchar()

เป็นฟังก์ชันที่รับข้อมูลเข้ามาทางแป้นพิมพ์ทีละ 1 ตัวอักษรโดยต้องกด enter ทุกครั้ง เมื่อสิ้นสุดข้อมูลและข้อมูลที่ป้อนจะปรากฏให้เห็นบนหน้าจอภาพด้วยรูปแบบ

```
getch();
```

3.2.6.11 ฟังก์ชัน getch()

เป็นฟังก์ชันที่รับข้อมูลเข้ามาทางแป้นพิมพ์ ที่เป็นตัวอักษร 1 ตัวอักษรโดยไม่ต้องกด enter เมื่อป้อนข้อมูล และตัวอักษรที่ป้อนเข้ามาจะไม่ปรากฏบนหน้าจอภาพรูปแบบ

```
get();
```

3.2.6.12 ฟังก์ชัน gets()

เป็นฟังก์ชันที่รับข้อมูลที่เป็นข้อความจากแป้นพิมพ์ เข้ามาเก็บไว้ในตัวแปรชุดรูปแบบ

```
gets (n);
```

n เป็นชื่อตัวแปรชุดที่ต้องการรับค่าที่เป็นข้อความจากแป้นพิมพ์ เข้ามาเก็บไว้โดยตัวสุดท้ายจะเก็บ \0 เอาไว้เพื่อแสดงการสิ้นสุดของ

ข้อความที่รับเข้ามาเมื่อผู้ใช้กดปุ่ม enter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.6.13 ฟังก์ชัน putchar()

เป็นฟังก์ชันที่ใช้ให้คอมพิวเตอร์แสดงผลบนจอภาพทีละตัวอักษร

รูปแบบ

```
putchar();
```

3.2.6.14 ฟังก์ชัน puts()

เป็นฟังก์ชันที่ใช้แสดงผลที่เป็นข้อความที่เก็บไว้ในตัวแปรชุดออกมาบนจอภาพ

รูปแบบ

```
put (n);
```

n เป็นชื่อตัวแปรชุดที่ต้องการนำค่ามาแสดงผล

3.3 ค้นคว้าหาข้อมูล

ข้อมูลเกี่ยวกับการแสดงผลภาษาไทยโดยใช้เทอร์โบซีในโหมดกราฟฟิกบนจอ VGA โดยค้นคว้าจาก วารสารไมโครคอมพิวเตอร์ เล่มที่ 98 และ 99 หนังสือการโปรแกรมภาษาซี ของมหาวิทยาลัยขอนแก่น และ การเขียนโปรแกรมใช้งาน EGA/VGA ซึ่งจากการศึกษาข้อมูลต่าง ๆ ผู้เขียนได้เลือกโหมด 10H ในการแสดงผลเพราะโหมด 10H เป็นที่นิยมใช้กันมากในการแสดงผลกราฟฟิคที่มีความละเอียด 640x350 จุดการแสดงผลใช้ 4 เพลน ทำให้การแสดงผลได้ถึง 16 สีพร้อมกันโดยโหมด 10H นี้ยังสามารถที่จะแสดงผลได้ 2 เพลจ คือ เพลจ 0 และ เพลจ 1 โดยเพลจ 0 อยู่ที่ Address A000:0000 และ เพลจ 1 อยู่ที่ Address A000:8000

3.3.1 โหมดการแสดงผลมาตรฐานของ EGA/VGA

เอกสารนี้เป็นเอกสารทูลสงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ก่อนอื่นต้องทำความรู้จักกับโหมดการแสดงผลที่เป็นมาตรฐาน ซึ่งการ์ด VGA สามารถแสดง ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้ในตารางที่ 3.8 แสดงให้เห็นโหมดต่าง ๆ ระบบ VGA สลับส่นโหมดการแสดงผลของการ์ดแสดงผล MDA และ CGA ในระบบเก่าด้วยซึ่งในโครงการนี้จะใช้โหมดการแสดงผลในโหมด 10 เท่านั้น

Mode	Type	Colors	Resolution	Compatible Display
0,1	Color text	16	40x25 8x8 char cell	CD, ED, VGA Mutifrequency
0*,1*	Color text	16	40x25 8x14 char cell	ED, VGA Mutifrequency
0+,1+	Color text	16	40x25 9x16 char cell	VGA Mutifrequency
2,3	Color text	16	80x25 8x8 char cell	CD, ED, VGA Mutifrequency
2*,3*	Color text	16	80x25 8x14 char cell	ED, VGA Mutifrequency
2+,3+	Color text	16	80x25 9x16 char cell	VGA Mutifrequency
4,5	Color graph	4	320x200	CD, ED, VGA Mutifrequency

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Mode	Type	Colors	Resolution	Compatible Display
6	Color graph	2	640x200	CD, ED, VGA Mutifrequency
7	Monochrom text	2	80x25 8x14 char cell	Monochrom VGA
7	Monochrom text	2	80x25 9x16 char cell	VGA only
D	Color graph	16	320x200	CD, ED, VGA Mutifrequency
E	Color graph	16	640x200	CD, ED, VGA Mutifrequency
F	Mono graphs		640x350	Monochrom VGA
10	Color graph	16	640x350	ED, VGA Mutifrequency
11	Color graph	2	640x480	VGA Mutifrequency

ตารางที่ 3.8 โหมดแสดงผลมาตรฐานของระบบวีดิโอไลปี้เต็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 การแสดงผลโหมด 10H (โหมดกราฟิกแบบ Enhanced ขนาด 640x350 จุด)

ในโหมด 10H เป็นโหมดที่เพิ่มขึ้นใน EGA และ VGA หนึ่งเป็นโหมดที่นิยมใช้กับสำหรับโปรแกรมกราฟิกทั่วไป ความละเอียดของโหมดจะมีถึง 640 จุดในแนวนอน และ 350 จุดในแนวตั้ง โหมดนี้จะใช้กับจอภาพชนิด CD ไม่ได้ต้องใช้กับจอภาพ ECD, VGA หรือ จอหลายความถี่ การควบคุมสีใช้เพลนสีถึง 4 เพลน ดังนั้นจึงสามารถแสดงผลได้ 16 สี ในเวลาเดียวกันยกเว้นการ์ด EGA ของไอบีเอ็มที่มีหน่วยความจำเพียง 64 กิโลไบต์ ซึ่งจะใช้เพลนสีได้แค่ 2 เพลน

ในโหมด 10H จะแสดงผลได้ 2 เพลจ (สำหรับ EGA ที่มี 64 กิโลไบต์จะมี 1 เพลจ) โดยหน่วยความจำแต่ละเพลจจะเป็นดังนี้

เพลจ 0 ตำแหน่ง A000:0000

เพลจ 1 ตำแหน่ง A000:8000

สีที่ใช้แสดงในโหมด 10 H จะอยู่ในตารางที่ 3.9

Plane	Full(128K+)	Partial(64 K+)
3 2 1 0	Colors	Colors
0 0 0 0	Black	Black
0 0 0 1	Blue	Blue
0 0 1 0	Green	Black
0 0 1 1	Cyan	Blue
0 1 0 0	Red	Red
0 1 0 1	Magenta	White
0 1 1 0	Brown	Red
0 1 1 1	Write	Write
1 0 0 0	Dark Gray	Black
1 0 0 1	Light Blue	Blue
1 0 1 0	Light Green	Black
1 0 1 1	Light Cyan	Blue

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Plane	Full(128K+)	Partial(64 K+)
1 1 0 0	Light Red	Red
1 1 0 1	Light Magenta	White
1 1 1 0	Yellow	Red
1 1 1 1	Intens White	White

ตารางที่ 3.8 สีมาตรฐานในการพิมพ์ 16 สี

3.4 การเขียนโปรแกรมการผลภาษาไทยด้วยเทอร์โบซี

รหัสที่ใช้แสดงผลภาษาไทยที่นิยมใช้กันโดยทั่วไปก็มีรหัสของ สมอง และ เกษตร แต่ในโครงการนี้ได้ใช้รูปแบบตัวอักษร (Font) ของวารสารไมโครคอมพิวเตอร์เป็นรหัสของ สมอง ที่มีขนาด 8x14 จุด อยู่ในไฟล์ชื่อ 8x14 fon โดยในรูปแบบตัวอักษรที่ใช้จะแสดงในรูปที่ 3.11

8x14 fon

	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								

จะเห็นได้ว่า จากตารางรูปแบบตัวอักษรดังรูป เมื่อเปรียบเทียบกันระหว่างรูปแบบตัวอักษรของผู้เขียน (8x14 fon) ดังนั้นจึงต้องเขียนโปรแกรมเปลี่ยนรหัสแอสกีให้ตรงกับรูปแบบตัวอักษรของผู้เขียนโดยใช้โปรแกรมดังนี้

```
switch (*ThaiStr) {
    case 143: *ThaiStr = 153; break;
    case 144: *ThaiStr = 156; break;
    case 145: *ThaiStr = 150; break;
    case 146: *ThaiStr = 154; break;
    case 147: *ThaiStr = 152; break;
    case 149: *ThaiStr = 158; break;
    case 150: *ThaiStr = 159; break;
    case 152: *ThaiStr = 149; break;
    case 153: *ThaiStr = 151; break;
    case 154: *ThaiStr = 155; break;
    case 155: *ThaiStr = 157; break; }
```

จากการที่กล่าวข้างต้นว่า 8x14 fon มีขนาด 8x14 จุด ต่อ 1 ตัวอักษรจะนั้นข้อมูลแต่ละตัวจะมีขนาด 14 ไบต์โดยรหัสแอสกีมีตั้งแต่ 0-255 คือ 256 ตัว ดังนั้น 8x14 fon จึงมีขนาดที่ $14 \times 256 = 3584$ ไบต์ หากเราต้องการแสดงผลตัวหนังสือขนาด 8x14 จุด โดยมีข้อมูล 1 ไบต์ แสดงผล 8 จุด ซึ่งเป็นจำนวน 14 เส้น (line) ค่าในข้อมูลตัวหนังสือเรียงลำดับกันจนครบ 14 ไบต์ และตำแหน่งของหน่วยความจำแต่ละเส้นของจอภาพในโหมด 10H สามารถหาได้จากสูตร

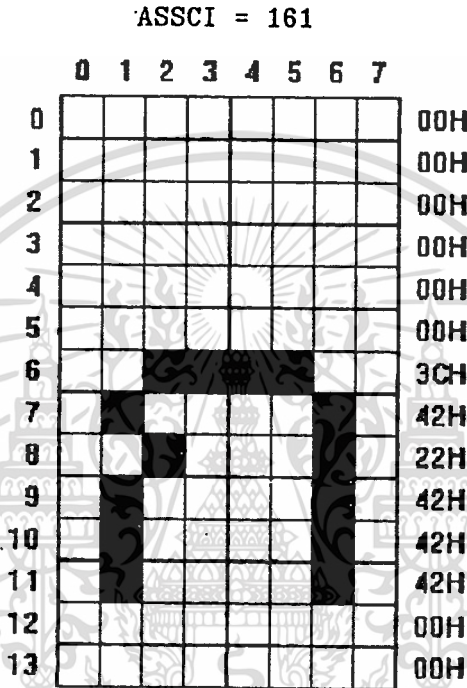
$$\text{ออฟเซตของเส้นที่ } N = N * 80$$

เพื่อความสะดวกและความรวดเร็วในการทำงานเราจะคำนวณค่าออฟเซตนี้ไว้ก่อน แล้วนำไปเก็บค่าไว้ในตัวแปรอะเรย์โดยสามารถเขียนโปรแกรมได้ดังนี้

```
int xcount
for (xcount = 0 ; xcount < 350 ; xcount++)
    off_line [xcount] = xcount*80;
```

เอกสารนี้เป็นเอกสารที่กล่าวมาข้างต้นจะได้ว่าตัวอักษร 1 ตัวอักษร จะเก็บข้อมูลไว้ 14 ไบต์ ยกตัวอย่าง
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังรูปที่ 3.12 เป็นตัวอักษร "ก" ซึ่งมีรหัสแอสกีเป็น 161 ดังนั้นข้อมูลตัวแรกของตัวอักษร "ก" จะอยู่ที่ไบต์ที่ $161 \div 8 = 20$ ของไฟล์ 8x14 font โดยจะมีข้อมูลเรียงตามลำดับต่อไปนี้ 00H, 00H, 00H, 00H, 00H, 3CH, 42H, 22H, 42H, 42H, 42H, 00H, 00H ตัวอักษรตัวอื่น ๆ ก็จะทำในลักษณะเดียวกัน



รูปที่ 3.12 การเก็บข้อมูลตัวอักษร 1 ตัวอักษร

3.4.1 การจัดระดับตัวอักษร

การจัดระดับของตัวอักษร แบ่งออกได้เป็น 4 ระดับคือ พยัญชนะ, สระบน, สระล่าง และวรรณยุกต์ โดยพยัญชนะจะอยู่ในช่วงเส้นสแกนที่ 5-11 สระบนอยู่ในช่วงเส้นสแกนที่ 2-4 สระล่างอยู่ในช่วงเส้นสแกนที่ 12-13 และวรรณยุกต์อยู่ในช่วงเส้นสแกนที่ 0-2 ดังรูปที่ 3.12

	0	1	2	3	4	5	6	7	
0									00H
1									00H
2							■		01H
3	■	■	■	■	■	■	■		7EH
4	■	■	■	■	■	■	■		FEH
5									00H
6									00H
7									00H
8									00H
9									00H
10									00H
11									00H
12									00H
13									00H

รูปที่ 3.13(ก) การเก็บตัวอักษร สระบน

	0	1	2	3	4	5	6	7	
0						■			วรัรรมุกค์
1									
2									
3	■	■	■	■	■	■	■		สระบน
4	■	■	■	■	■	■	■		
5									พยัญชนะ
6									
7		■					■		
8		■					■		
9									สระล่าง
10									
11		■					■		
12				■	■	■			
13									

รูปที่ 3.13(ข) การจัดระดับของตัวอักษร

ส่วนการนำข้อมูลในไฟล์ 8x14 fon มาแสดงผลที่จอภาพจะใช้คำสั่งเขียนเขียนหน่วยความจำ เอกสารนี้เป็นเอกสารที่โรงเรียนวิศวกรรมเพื่อการศึกษาเท่านั้น ไม่นับว่าตีพิมพ์ไปใช้ประโยชน์ด้านการค้า โดยตรงโดยใช้คำสั่ง pokeb(); ของเทอร์โบซี คำสั่งนี้จะเป็นการเขียนข้อมูล 1 ไบต์ (8 บิต) ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามเผยแพร่ต่อผู้อื่น และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลงสู่หน่วยความจำดังนั้นในการเขียนตัวอักษร 1 ตัวอักษร ลงบนจอภาพจะต้องส่งข้อมูลที่อยู่ในไฟล์ 8x14 .fon ทีละ 1 ไบต์จนครบ 14 ไบต์ โดยฟังก์ชันการแสดงผลภาษาไทยประกอบด้วยฟังก์ชันต่าง ๆ ดังนี้คือ

```
- OpenFont ();
-Thai_Write();
-Disp_Char();
```

ในการเรียกใช้ จะเรียกใช้ฟังก์ชัน G_write() และฟังก์ชัน G_Write() เพื่อจะทำการตรวจสอบว่าเป็นสระบน สระล่าง, พยัญชนะ หรือวรรณยุกต์ เพื่อการแสดงผลบนจอภาพเป็นไปได้อย่างเหมาะสมที่สุด

3.5 ชุดคำสั่ง OpenFont()

```
void OpenFont(char*fi,byte *ThaiFont)
{
FILE *Fp;
if((Fp=fopen(fi,"rb"))==NULL)
{
printf ("\n *** File %s not found ***",fi);
exit(0);
}
fread(& ThaiFont[0],sizeof(Gen-Map),1,Fp);
fclose(Fp);
return 0;
}
```

3.6 ชุดคำสั่ง G_Write()

```
void G_write(byte *ThaiStr,byte Colend)
{
byte ST,SP;
while(*ThaiStr != NULL)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(*ThaiStr==126)ThaiStr+=2;
if((*ThaiStr>=128)&&(*ThaiStr<=148)) returns;
if(*ThaiStr==5)
    switch(Line){
        case 0:Line=1;
        break;
        case 1:Line=0;
        break;
        default:break;
    }
if(*ThaiStr==2)
    switch(Blod){
        case 0:Blod=1;
        break;
        case 1:Blod=0;
        break;
        default:break;
    }
if(*ThaiStr==3)
    switch(Tab){
        case 0:Tab=1;Blod=1;
        break;
        case 1:Tab=0;Blod=0;
        break;
        default:break;
    }
switch(*ThaiStr){
    case 143: *ThaiStr=153; break;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    case 145: *ThaiStr=150; break;
    case 146: *ThaiStr=154; break;
    case 147: *ThaiStr=152; break;
    case 149: *ThaiStr=158; break;
    case 150: *ThaiStr=159; break;
    case 152: *ThaiStr=149; break;
    case 153: *ThaiStr=151; break;
    case 154: *ThaiStr=155; break;
    case 155: *ThaiStr=157; break;
    default:break;
}
if(*ThaiStr>31)
switch(*ThaiStr)
{
    case 216:case 217:case 218:{
        ST=12;SP=13;}break;
    case 209:case 212:case 213:case 214:case 215:case 231:
    { ST=2;SP=4;}break;
    case 232:case 233:case 234:case 235:case 236:case 237:
    {ST=0;SP=2;}break;
    default:{
        ST=0;SP=13;G_Col++;}
}
if(G_Col>Colend) return;
if((Blod&&Tab)&&(*ThaiStr>31))DispCharRev(ST,SP,Bit_Normal[*ThaiStr]);
else if(*ThaiStr>31)DispChar(ST,SP,Bit_Normal[*ThaiStr]);

*ThaiStr++;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่วางกรรมสิทธิ์ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 ชุดคำสั่ง DispChar()

```

void DispChar(byte ST,byte SP,byte *One_Char)
{
    byte Count,line;
    if(Line)
    {
        Set_Color(2);
        for(Count=ST;Count<=SP;Count++)
        {
            if((Count==12)&&(ST!=12)) line=0xff;
else line=0x00;
            if(pagedisplay==0)pokeb(0xA000,Off_Line[G_Row+Count]+G_Col,
                (One_Char[Count]);line);
            else pokeb(0xA800,Off_Line[G_Row+Count]+G_Col,
                (One_Char[Count]);line);
        }
    }
    if(Line)return;
    for(Count=ST;Count<=SP;Count++)
    {
        if(Blod)
        {
            Set_Color(7);
            if(pagedisplay==0)pokeb(0xA000,Off_Line[G_Row+Count]+G_Col,
                (One_Char[Count]);(One_Char[Count]>>1));
            else pokeb(0xA800,Off_Line[G_Row+Count]+G_Col,
                (One_Char[Count]);(One_Char[Count]>>1));
        }
    }
else

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Set_Color(14);
if (pagedisplay==0) pokeb(0xA000,Off_Line[G_Row+Count]+G_Col,
    One_Char[Count]);
else pokeb(0xA800,Off_Line[G_Row+Count]+G_Col,One_Char[Count]);
}
}
}

```

ฟังก์ชัน OpenFont จะทำหน้าที่เปิดไฟล์ที่ตัวแปร pointer fi ชื่ออยู่ โดยจะเปิดไฟล์แล้ว
โหลดลงหน่วยความจำในตัวแปรที่ ThaiFont ชื่ออยู่

ฟังก์ชัน G_Writ จะทำการตรวจสอบว่า รหัสแอสกีเป็นพยัญชนะ, สระบน, สระล่าง หรือ
วรรณยุกต์ เพื่อจัดระดับการแสดงผลให้เหมาะสมโดยภายในฟังก์ชันจะเรียกฟังก์ชัน DispChar()

ฟังก์ชัน DispChar() รับค่ารหัสแอสกี ที่ส่งมาจาก ฟังก์ชัน G_Write มาทำการแสดงผล
ตามตำแหน่งที่ฟังก์ชัน G_Write จัดไว้

เพื่อให้ง่ายต่อการใช้งานผู้เขียนได้ร่างฟังก์ชันขึ้นมาใหม่ เพื่อให้ง่ายต่อการนำฟังก์ชันไปใช้
งานจึงเขียนใหม่ให้เป็นฟังก์ชัน outthaixy() ดังโปรแกรมข้างล่าง

```

void outthaixy (int column,int row,int colend,char*str*)
{
    G_gotoxy (column,row);
    Thai_write(str,colend);
}

void G_gotoxy (int x, int y)
{
    if(x>80) x = 80
    if(y>25) y = 25
    x = x-1
    y = (y-1)*14;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดสอบ

ในการทดสอบการแสดงผลภาษาไทยนั้น ถ้าเราต้องการให้แสดงตารางจะเกิดข้อผิดพลาดขึ้น เนื่องจากที่สลับแบบตัวอักษรของ cw นั้นไม่ตรงกับรูปแบบตัวอักษรของไฟล์ 8x14.fon แต่ก็สามารถแก้ไขได้โดยการเขียนโปรแกรมเพิ่มเติมดังนี้

```
void G_write(byte *ThaiStr,byte Colend)
{
    byte ST,SP;

    while(*ThaiStr != NULL)
    {
        if(*ThaiStr==126)ThaiStr+=2;
        if((*ThaiStr>=128)&&(*ThaiStr<=148)) return;
        if(*ThaiStr==5)
            switch(Line){
                case 0:Line=1;
                break;
                case 1:Line=0;
                break;
                default:break;
            }

        if(*ThaiStr==2)
            switch(Blod){
                case 0:Blod=1;
                break;
                case 1:Blod=0;
                break;
                default:break;
            }
    }
}
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(*ThaiStr==3)
    switch(Tab){
        case 0:Tab=1;Blod=1;
        break;
        case 1:Tab=0;Blod=0;
        break;
        default:break;
    }
switch(*ThaiStr){
    case 143: *ThaiStr=153; break;
    case 144: *ThaiStr=156; break;
    case 145: *ThaiStr=150; break;
    case 146: *ThaiStr=154; break;
    case 147: *ThaiStr=152; break;
    case 149: *ThaiStr=158; break;
    case 150: *ThaiStr=159; break;
    case 152: *ThaiStr=149; break;
    case 153: *ThaiStr=151; break;
    case 154: *ThaiStr=155; break;
    case 155: *ThaiStr=157; break;
    default:break;
}
if(*ThaiStr>31)
switch(*ThaiStr)
{
    case 216:case 217:case 218: {
        ST=12;SP=13;}break;
    case 209:case 212:case 213:case 214:case 215:case 231:
        { ST=2;SP=4;}break;

```

```

{ ST=0;SP=2;}break;

default:{
    ST=0;SP=13;G_Col++;}
}

if(G_Col>Colend) return;

if((Blod&&Tab)&&(*ThaiStr>31))DispCharRev(ST,SP,Bit_Normal[*ThaiStr]);
else if(*ThaiStr>31)DispChar(ST,SP,Bit_Normal[*ThaiStr]);

*ThaiStr++;
}
}

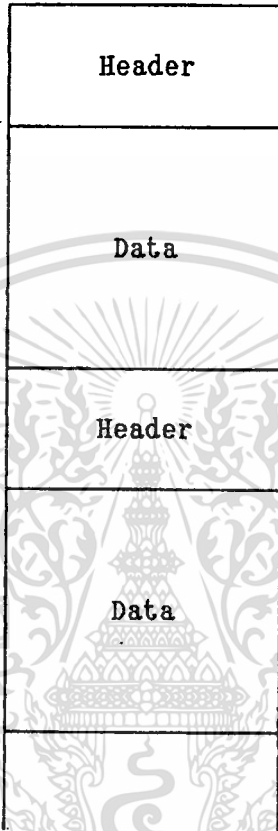
```

และในการแสดงตั้งอักษรของ (!) ไม่สามารถแสดงได้เนื่องจากรูปแบบของรหัสแอสกีตัวนี้ ไม่มีในไฟล์ 8x14 fon ดังนั้นจึงต้องให้โปรแกรมสำเร็จรูปช่วยโดยใช้ข้อมูล 14 ไบต์ ดังต่อไปนี้ 00H, 00H, 00H, 00H, 00H, 00H, 10H, 10H, 00H, 10H, 10H, 10H, 00H, 00H

	0	1	2	3	4	5	6	7	
0									00H
1									00H
2									00H
3									00H
4									00H
5									00H
6			█						10H
7			█						10H
8									00H
9			█						10H
10			█						10H
11			█						10H
12									00H
13									00H

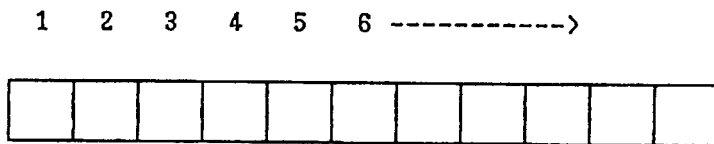
3.8 เขียนโปรแกรมแสดงส่วนให้ความช่วยเหลือ

หลักการแสดงข้อความช่วยเหลือสามารถอธิบายได้ดังรูปที่ 3.15



รูปที่ 3.15 หลักการแสดงข้อความช่วยเหลือ

จะเห็นว่าในแต่ละหน้า (Page) จะประกอบด้วยส่วนหัว (Header) และส่วนข้อมูล (data) โดยในส่วนหัวจะแสดงรายละเอียดของหน้า ดังนั้นรายละเอียดของส่วนหัวสามารถอธิบายได้ดังนี้



2 ไบต์แรกจะเป็นหมายเลขหน้า (page number) ของแต่ละหน้า ไบต์ที่3 จะเป็นจำนวนเอกสารนี้เป็นเอกสารที่วางไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไปอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าของหนานัน ๆ ไบต์ที่ 4 จะบอกว่าสามารถกด Page Up ได้หรือไม่ ไบต์ที่ 5 ระบุว่าสามารถไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กด page Down ได้หรือไม่ ไบต์ที่ 6 หมายความว่าในหน้านั้นมีตัวเลือกอยู่ที่ตัว ถ้าไบต์ที่ 6 นี้ไม่เป็นศูนย์ ไบต์ต่อไปจะเป็นหมายเลขหน้าของแต่ละตัวเลือก โดยแต่ละตัวเลือกจะมีขนาด 2 ไบต์ และ 2 ไบต์สุดท้ายของส่วนหัวจะเป็นค่าของจำนวนไบต์ นับจากส่วนของส่วนหัวไปจนจบหน้า

แต่ในการสร้างไฟล์ข้อมูลจะใช้เวิร์ดจุกัวในการเขียนดังนั้นจึงต้องเขียนโปรแกรมเพื่อแปลงไฟล์ข้อมูลให้อยู่ในรูปแบบที่ต้องการ โดยมีโปรแกรมดังนี้

```
#include<stdio.h>
#include<alloc.h>
#include<conio.h>
#include<stdlib.h>
#include<io.h>
#include<fcntl.h>
unsigned int swap(unsigned int value);
main(int argc,char *argv[])
{
    FILE *fp;
    int fi,page,byte,page_end,i,dot;
    char row,pup,pdown,inten,col,ch,rowfile;
    char str[5],*ptr,*ptr1,fname[15];
    int *pointer,*pointer1;
    unsigned int countbyte,pageindex[50];
    unsigned long size;
    if(argc!=3)exit(0);
    strcpy(fname,argv[1]);
    page_end=atoi(argv[2]);
    if((fi=open(fname,O_RDONLY|O_TEXT))<0)
    {
        printf("Cannot open file %s",fname);
        exit(0);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        if((size=filelength(fi))==-1)
    {
        printf("errno of handle of file");
        exit(0);
    }

    close(fi);

    ptr=farmalloc(size);
    ptr1=ptr;

    if((fp=fopen(fname,"rt"))==NULL)
    {
        printf("Cannot open file %s",fname);
        exit(0);
    }

    fgets(str,4,fp); /*read header file page1*/
    page=atoi(str); /*keep number page*/

do
    {
        getc(fp);
        fgets(str,3,fp);
        row=atoi(str); /*keep row*/

        getc(fp);
        pup=getc(fp);
        pup=atoi(&pup); /*keep pup*/

        getc(fp);
        pdown=getc(fp);
        pdown=atoi(&pdown); /*keep pdown*/

        getc(fp);
        fgets(str,3,fp);

        inten=atoi(str); /*keep inten*/

```

```

fgets(str,3,fp);
col=atoi(str);      /*keep col*/
for(i=0;i<inten;i++)
{
ch=getc(fp);
fgets(str,4,fp);
pageindex[i]=(unsigned int)atoi(str);
}
ch=getc(fp);
pointer=ptr;
*pointer=swap(page);
ptr+=2;
*ptr=row;
ptr++;
*ptr=pup;
ptr++;
*ptr=pdown;
ptr++;
*ptr=inten;
ptr++;
*ptr=col;
ptr++;
pointer1=ptr;
for(i=0;i<inten;i++)
{
*pointer1=(unsigned int)swap(pageindex[i]);
pointer1++;
}

ptr=pointer1;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ptr+=2;
countbyte=0;
rowfile=0;
do
{
ch=getc(fp);
countbyte++;
*ptr=ch;ptr++;
if(ch=='\n')rowfile++;
    } while(rowfile!=row);
*pointer=swap(countbyte);
fgets(str,4,fp); /*next page*/
page=atoi(str);
} while((page<=page_end)&&(page>0));
fclose(fp);
i=0;
do
{
dot=fname[i];
i++;
} while(dot!='.');
fname[i++]='h';
fname[i++]='l';
fname[i++]='p';
if((fp=fopen(fname,"wb"))==NULL)
{
printf("Cannot open file %s",fname);
exit(0);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
do {
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    putc(*ptr1,fp);
    ptr1++;
} while(ptr1!=ptr);
putc(0x1A,fp);
fclose(fp);
}
unsigned int swap(unsigned int value)
{
    int temp,tansmit;
    temp=value>>8;
    tansmit=(value<<8)!temp;
    return(tansmit);
}

```

รูปแบบของการเขียนไฟล์ข้อมูล โดยใช้ cu-writer ต้องเป็นไปตามข้อกำหนดคือ ในส่วนของส่วนหัวจะต้องมีการกำหนดหมายเลขหน้าจำนวนบรรทัดและอื่น ๆ ตามที่ได้กล่าวมาแล้วโดยค่าแต่ละค่าจะให้เครื่องหมายคอมม่า (,) เป็นตัวแยก ดังเช่นตัวอย่างข้างล่าง

001,14,0,1,02,75,005,011

จากตัวอย่างข้างบน หมายถึงว่าเป็นหน้าที่ 1 มีทั้งหมด 14 บรรทัด ไม่สามารถ Page Up เพื่อดูหน้าก่อนหน้านี้ได้ (หน้าปัจจุบันคือหน้าที่ 1) ได้ สามารถกด Page Down เพื่อดูหน้าถัดได้ในหน้านี้มีตัวเลือกอยู่ 2 ตัวเลือก โดยถ้าเลือกตัวเลือกแรกจะไปแสดงที่หน้า 5 ถ้าเลือกตัวที่ 2 จะไปแสดงที่หน้า 11 และในหน้านี้มีควมกว้าง 75 คอลัมน์

การทดสอบ

ในการทดสอบได้ลองสร้างไฟล์ซึ่งมีการกำหนดส่วนหัวของแต่ละหน้าตามข้อกำหนดที่ตั้งไว้แล้วนำมาใช้แสดงผล แต่ในการสร้างไฟล์โดยการใช้ cu word ไม่สามารถสร้างไฟล์มีความยาวมาก ๆ ตามที่ต้องการได้ ดังนั้นจึงต้องมีการเก็บเป็นหลาย ๆ ไฟล์ และเวลาแปลงไฟล์ให้อยู่ในรูปแบบที่โปรแกรมต้องการจะต้องนำไฟล์เหล่านั้นมาต่อกันก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้เฉพาะเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.9 การสร้างเท็กซ์ไฟล์

ในการสร้างเท็กซ์ไฟล์นั้น ผู้เขียนได้แยกออกเป็น 2 ส่วน คือ ส่วนที่ให้ความช่วยเหลือหลัก (Main Help) ในการเขียนเท็กซ์ไฟล์จะต้องมีส่วนหัว (Header) เป็นไปตามข้อกำหนดโดยบางส่วนของเท็กซ์ไฟล์แสดงดังนี้ เป็นตัวอย่างของเท็กซ์ไฟล์

```
#include<stdio.h>
#include<string.h>
#include<conio.h>
#include<io.h>

main(int argc,char *argv[])
{
FILE *fp,*fi,*fd;
char fname1[15],fname2[15],fname3[15];
unsigned char ch;
if(argc!=4) exit(0);
strcpy(fname1,argv[1]);
strcpy(fname2,argv[2]);
strcpy(fname3,argv[3]);
if((fd=fopen(fname3,"wb"))==NULL)
{
printf("Cannot open file %s",fname3);
exit(0);
}
if((fp=fopen(fname1,"rb"))==NULL)
{
printf("Cannot open file %s",fname3);
exit(0);
}
}
```

```

ch=getc(fp);
if(ch==0x1A) break;
putc(ch,fd);
}

fclose(fp);
if((fp=fopen(fname2,"rb"))==NULL)
{
printf("Cannot open file %s",fname3);
exit(0);
}

while(!(feof(fp)))
{
ch=getc(fp);
putc(ch,fd);
if(ch==0x1A)break;
}

fclose(fp);
fclose(fd);
printf("Program complete");
}

```

ในการเขียนใช้ cu-word connect p1.cw ,p2.cw ,p3.cw ,p4.cw ในการใช้ไฟล์ connect จะต้อง มี syntex ดังนี้

```
connect [file.cw][file.cw] output.file
```

ดังนั้นจะต้องทำตามขั้นตอนต่อไปนี้

```
c:\>connect p1.cw p2.cw dat.dat
```

```
c:\>connect p3.cw p4.cw dat.dat
```

```
c:\>connect dat.dat dat1.dat date2.dat
```

เสร็จแล้วนำไฟล์ dat2.dat ไป compli เป็น hlp โดยมีรูปแบบคำสั่งดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 complif [file.dat][page.end]

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นจะต้องทำดังนี้ (สมมติหน้าสกดท้ายเป็น 100)

c:\>complif.dat2.dat 100

การทดสอบ

นำเท็กซ์ไฟล์ที่ได้ทำเสร็จแล้วไปคอมไพล์เป็น .hlp แล้วนำไปใช้แสดงผล แต่เนื่องจากเท็กซ์ไฟล์มีขนาดใหญ่มากในการแสดงผลโปรแกรมบางครั้งจะเกิดปัญหาคือ หน่วยความจำไม่พอจึงเขียนโปรแกรมใหม่ โดยแทนที่จะโหลดลงในหน่วยความจำทั้งหมด ก็จะโหลดทีละ 30 หน้าเท่านั้น โดยค่าในการแสดงผลเลขหน้าไม่มีใน 30 หน้ามันก็จะโหลดไฟล์ .hlp ใหม่

3.10 โปรแกรมเช็คคีย์

ในส่วนของโปรแกรมเช็คคีย์นี้จะมีการตรวจสอบข้อความที่ต้องการให้ผู้เรียนได้พิมพ์ตามโดยจะมีการตรวจสอบคีย์ฟังก์ชันพิเศษซึ่งเป็นคีย์ช่วยเหลือนำได้แก่ Ar-row Key, Home, End, Ins, Del ทำให้สามารถเขียนข้อความได้สะดวกและรวดเร็วขึ้นสำหรับคีย์ปกติก็จะมีการตรวจสอบทุกตัวเพื่อใช้ในการแสดงผล เมื่อกดคีย์นั้น ๆ ลงไปโดยเฉพาะคีย์ Space Bar จะต้องมีการตรวจสอบเป็นพิเศษเพราะจะมีผลต่อคีย์ฟังก์ชันพิเศษด้วย สำหรับการเขียนโปรแกรมใช้งานจริง เราก็กึ่งเพียงแต่กำหนดตัวแปรที่สำคัญ 3 ตัว เพื่อใช้ตรวจสอบตำแหน่งเคอร์เซอร์, ตำแหน่งตัวอักษร และจำนวนตัวอักษรที่มีการพิมพ์ไปแล้ว โดยในที่นี้เราจะกำหนดตัวแปรต่าง ๆ ไว้ดังนี้

ตำแหน่งเคอร์เซอร์	Ad_Cu
ตำแหน่งตัวอักษร	Ad_St
จำนวนตัวอักษร	Num_St

ตัวแปรทั้ง 3 จะเป็นค่าที่ทำให้โปรแกรม ทราบตำแหน่งและจำนวนตัวอักษรได้ ดังนั้นโปรแกรมก็จะทำการตรวจสอบคีย์ต่าง ๆ ที่ถูกกด ถ้าเป็นคีย์ฟังก์ชันพิเศษก็จะมีหน้าที่ต่าง ๆ กัน เช่น Del ก็จะไปลบตัวอักษรตำแหน่งที่เคอร์เซอร์ปรากฏอยู่, Home จะทำให้เคอร์เซอร์ไปปรากฏอยู่ที่ตัวอักษรตัวแรกที่พิมพ์ไว้ เป็นต้น ซึ่งจะมีลักษณะเหมือนกับคอสปกติ สำหรับเคอร์เซอร์ที่ปรากฏอยู่บนจอภาพจะมีลักษณะเป็นเส้นบาร์อยู่ใต้ล่างตัวอักษร และกระพริบตลอดเวลา แต่เมื่อกดปุ่ม Ins ตัวเคอร์เซอร์ก็จะใหญ่ขึ้นเป็น 2 เท่าของเคอร์เซอร์เดิม เพื่อบอกให้รู้ว่าเป็นการแทรกข้อความ ในโปรแกรมส่วนนี้จะมีประโยชน์ในการแสดงผลออกทางหน้าจอภาพว่า ได้มีการกดคีย์ใด ๆ ซึ่งจะจำลองการทำงานเหมือนการกดคีย์อยู่ในคอสจริง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการเรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
โปรแกรมตรวจสอบคีย์จะมีการเรียกใช้ฟังก์ชัน 2 ฟังก์ชัน คือ
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังมีเหตุเปลี่ยนแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟังก์ชันแสดงเคอร์เซอร์จะเป็นการแสดงเคอร์เซอร์ได้ 2 ลักษณะ คือแสดงเคอร์เซอร์แบบพิมพ์ทับ และแสดงเคอร์เซอร์แบบพิมพ์แทรกซึ่งลักษณะของเคอร์เซอร์แบบพิมพ์แทรกนี้จะมีขนาดใหญ่กว่าแบบพิมพ์ทับประมาณ 1 เท่า ที่ทำเช่นนี้ก็เพื่อที่จะให้ผู้เขียนได้เห็นความแตกต่างของการกดปุ่ม Ins (พิมพ์แทรก) และไม่กดปุ่ม Ins (พิมพ์ทับ) ซึ่งในการเขียนโปรแกรมก็จะใช้หลักการที่ว่า "แสดงหน่วง ลบหน่วง" นั่นก็คือแสดงรูปเคอร์เซอร์แล้วก็หน่วงเวลา จากนั้นก็ลบเคอร์เซอร์แล้วหน่วงเวลาวนอยู่อย่างนี้ตลอดก็ทำให้ผู้เขียนเห็นว่าเคอร์เซอร์กระพริบที่หน้าจอภาพสำหรับโปรแกรมการใช้งานจะเป็นดังนี้

```
Cursor(int colum,int row,int ck,int dl){
    char *Inverse1,*Inverse2;
    Inverse1 = malloc(imagesize(colum,row,colum+8,row+14));
    getimage(colum,row,colum+8,row+14,Inverse1);
    do
    {
        putimage(colum,row,Inverse1,0);
        delay(dl);
        Inverse2 = malloc(imagesize(colum,row,colum+8,ck+row));
        getimage(colum,row,colum+8,ck+row,Inverse2);
        putimage(colum,row,Inverse2,4);
        delay(dl);
        free(Inverse2);
    }
    while(kbhit()==0);
    putimage(colum,row,Inverse1,0);
    free(Inverse1);
}
```

จากโปรแกรมจะใช้คำสั่ง `getimage` ในการเก็บตัวอักษรที่แสดงอยู่ ไว้เพื่อใช้ในการการเคลื่อน เคอร์เซอร์ (cursor) จากนั้นก็ต้องเก็บพื้นที่ไว้เท่ากับขนาดของเคอร์เซอร์ ใช้คำสั่ง `putimage` แสดงผลแบบ `inverse` เพื่อให้เห็นเป็นรูปเคอร์เซอร์ ซึ่งทำการหน่วงเวลาสลับไปไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สลับมาจนกว่า จะกดคีย์ใด ๆ ก็จะออกจากโปรแกรมนี้ สำหรับโปรแกรมนี้อาจรับค่า 4 ตัว คือ ตำแหน่งที่แสดง หลักที่ แถวที่ ส่วนค่า ck จะเป็นการบอกว่าแสดงแบบทับ หรือแบบพิมพ์แทรกและ ตัวสุดท้ายคือ dl จะบอกเวลาในการหน่วงของ cursor ว่าให้กระพริบเร็วหรือช้า

- ฟังก์ชันลบตัวอักษรจะใช้ในการลบตัวอักษรเมื่อกดปุ่ม Delete หรือปุ่ม Backspace ซึ่งเป็นการลบตำแหน่งเดิมหรือตำแหน่งก่อนหน้านั้น โดยมีการตรวจสอบตำแหน่งจากตัวแปร Ba_De ที่โปรแกรมตรวจสอบคีย์ส่งค่ามาให้ ลักษณะของโปรแกรมจะเป็นดังนี้

```
Del(int colum,int row,int Ba_De)
```

```
{
    char *pt;
    pt = malloc(imagesize(0,0,8,12));
    getimage(0,0,8,12,pt);
    putimage(colum-Ba_De,row,pt,3);
    free(pt);
}
```

จากโปรแกรมเก็บช่องว่างที่มีขนาดเท่ากับตัวอักษรโดยใช้คำสั่ง getimage จากนั้นก็แสดงผลแบบ and จะได้เป็นการลบข้อความโปรแกรมนี้จะรับค่าตำแหน่งการลบและมีตัวตรวจสอบอีก 1 ตัวว่าเป็นการกดปุ่ม Delete หรือปุ่ม Ins

โปรแกรมตรวจสอบคีย์จะทำการตรวจสอบว่าปุ่มที่ผู้เขียนกดนั้นเป็นคีย์ปกติ หรือคีย์ฟังก์ชันพิเศษโดยถ้าเป็นคีย์ปกติก็จะแสดงผลที่หน้าจอภาพทันที แต่ถ้าเป็นคีย์ฟังก์ชันพิเศษก็จะทำงานตามคีย์นั้น ๆ เช่น คีย์ Home จะทำให้เคอร์เซอร์กลับไปแสดงที่ตัวอักษรตัวแรก, คีย์ Delete ก็จะทำให้การลบตัวอักษรตำแหน่งเคอร์เซอร์แสดงอยู่, คีย์ Right จะเลื่อนตำแหน่งเคอร์เซอร์ไปทางขวา 1 ตำแหน่ง โปรแกรมตรวจสอบคีย์จะเป็นฟังก์ชันที่มีการรับค่าเข้ามาเพื่อบอกจำนวนตัวอักษรตำแหน่งแถว และหลักของจุดเริ่มต้นที่จะตรวจสอบการกดคีย์ การออกจากโปรแกรมทำได้ 2 วิธีคือการกดปุ่ม Enter และการพิมพ์ข้อความเท่ากับจำนวนตัวอักษรที่กำหนดไว้สำหรับตัวอักษรโปรแกรมตรวจสอบคีย์นี้ จะเป็นการตรวจสอบคีย์อักษรต่าง ๆ และคีย์ Backspace ส่วนคีย์ฟังก์ชันพิเศษก็มี คีย์ Home, Left Arrow, Right Arrow ซึ่งคีย์ที่เหลือก็จะมีลักษณะการเขียนโปรแกรมคล้าย ๆ กัน

```

{
char Swap[2],ch[1];
int loop;
do{
    ch[0]=getch();
    switch(ch[0]){
        case 8 : if(Ad_St>=1){
                    Ad_Cu--=8;
                    Ad_St--;
                    Num_St--;
                    gt[Ad_St]=0;
                    Del(column,row,8);
                    for(loop=Ad_St+1;loop<=num;loop++){
                        if(gt[loop]!=0){
                            gt[loop-1]=gt[loop];
                            gt[0]=gt[1];
                            gt[loop]=0;
                            Del(column,row,loop-Ad_St));
                            thaixy(column,row,num,&gt,0);
                        }
                    }
                break;

        case 13 : break;
        default : Del(column,row,8);
                    if(Ad_St<num)
                        thaixy(column,row,2,&ch[0],0);
                    Swap[0]=ch[0];
                    if(Ad_St<num){
                        if(ck==1){
                            for(loop=Ad_St;loop<=num;loop++){

```

```

Del(column,row,loop+1-Ad_St));
if(loop<num-1){
    Swap[1]=gt[loop+1];
    gt[loop+1]=Swap[0];
    Swap[0]=Swap[1];}
}

thaixy(column,row,num,&gt;,0);
if(Num_St!=num) Num_St++;
}}

Ad_Cu+=8;
if(Ad_St<=num) {Ad_St++; Num_St++;}
else exit(0);
break;
}
}

if(ch[0]==0){
    ch[0]=getch();
    switch(ch[0]){
/*HOME*/  case 71 : Ad_Cu=0;
                Ad_St=0;
                break;

/*LEFT*/  case 75 : if(Ad_St==0) break;
                Ad_Cu-=8;
                Ad_St--;
                break;

/*RIGHT*/ case 77 : if(Ad_St==Num_St) break;
                Swap[1]=0;
                for(loop=Ad_St;loop<=num;loop++)
                if((gt[loop]==32) || (gt[loop]==0))
                    Swap[1]++;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{Num_St=Ad_St;
```

```
break;}
```

```
Ad_Cu+=8;
```

```
Ad_St++;
```

```
break;
```

```
}
```

```
}
```

```
}
```

```
while(ch[0]!='13');
```

```
}
```

เมื่อได้ทำการเขียนส่วนต่าง ๆ เรียบร้อยแล้ว (ฟังก์ชันแสดงเคอร์เซอร์, ฟังก์ชันลบตัวอักษร และโปรแกรมตรวจสอบคีย์) ก็สามารถทำการทดสอบโปรแกรมตรวจสอบคีย์นี้ได้ โดยการเขียนโปรแกรมนี้จะเขียนในรูปลักษณะของฟังก์ชันรับค่าได้ดังนั้น เรา ก็สามารถกำหนดตำแหน่งเริ่มต้นของการแสดงผล และจำนวนตัวอักษรที่ต้องการรับ แล้วทำการ RUN โปรแกรมทดสอบ โดยการกดคีย์ต่าง ๆ บนแป้นพิมพ์ว่ามีการทำงานได้ครบตามที่กำหนดไว้ หรือไม่ ทดสอบหลาย ๆ ครั้งจนแน่ใจว่าไม่มีข้อผิดพลาดเกิดขึ้นแน่นอน เรา ก็สามารถนำโปรแกรมนี้ไปใช้งานร่วมกับส่วนอื่น ๆ ได้ต่อไป

3.11 โปรแกรมแสดงผลคำสั่ง

โปรแกรมแสดงผลคำสั่งนี้จะต้องมีการเรียกใช้โปรแกรมตรวจสอบคีย์บ่อยมาก เพราะมีการตรวจสอบคำสั่งที่พิมพ์เข้ามานั้นว่าถูกต้องหรือไม่ ถ้าไม่ถูกต้องจะต้องกลับไปพิมพ์ใหม่อีกครั้ง แต่ถ้าพิมพ์คำสั่งได้ถูกต้องก็จะมี การเรียกใช้ข้อมูล เพื่อแสดงผลจากการทำคำสั่งนั้น ๆ ให้ดูทางหน้าจอภาพ ซึ่งข้อมูลในส่วนนี้จะถูกเก็บไว้เป็นฟังก์ชัน เพื่อสะดวกในการเรียกใช้

คำสั่งที่มีการเรียกใช้จะมีจำนวนมาก ดังนั้นจึงแบ่งเป็นหัวข้อ ๆ เพื่อง่ายต่อการเรียนรู้โดยเมื่อผู้เรียนพิมพ์คำสั่งในเรื่องที่เรียนนั้น ๆ แล้ว คำสั่งที่พิมพ์นั้นผิดพลาดไม่ถูกต้อง ซึ่งจะตั้งใจหรือไม่ตั้งใจก็แล้วแต่โปรแกรมจะทำการเปรียบเทียบคำสั่งที่พิมพ์กับคำสั่งที่เก็บไว้ว่าตรงกันหรือไม่ ถ้าคำสั่งไม่ตรงกันจะมีข้อความบอกว่า พิมพ์ผิดให้พิมพ์ใหม่ ซึ่งผู้เรียนก็สามารถใช้คีย์พิเศษช่วยในการแก้ไขให้รวดเร็วขึ้นได้ โปรแกรมจะทำงานเช่นนั้นจนกว่าจะพิมพ์ได้ถูกต้อง จากนั้นจะทำการเรียกใช้ฟังก์ชันแสดงผลของการใช้คำสั่งนั้น ๆ โดยที่โปรแกรมนี้จะมีการตรวจสอบคำสั่งที่ผู้เรียนพิมพ์

เอกสารนี้เป็นเอกสารที่วางไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านธุรกิจ การพาณิชย์ หรือการค้ากำไรโดยไม่ได้รับอนุญาต หากมีข้อผิดพลาดประการใดขออภัยเป็นอย่างสูง และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Enter เพื่อจบคำสั่ง ผลของการทำงานที่แสดงจะคล้ายการทำงานของคอสจริง ๗

โปรแกรมใช้งานจริงของโปรแกรมจะมีส่วนสำคัญอยู่ 3 ส่วน คือ

3.11.1 ส่วนตรวจสอบคำสั่ง

จะมีการตรวจสอบว่าคำสั่งนั้น พิมพ์ถูกต้องตามรูปแบบหรือไม่ ถ้าพิมพ์ผิด หรือไม่ยอมพิมพ์ แต่กด Enter ก็จะทำให้พิมพ์ถูกต้อง แต่ถ้าถูกต้องก็จะเก็บข้อความที่ผู้เขียนพิมพ์ไว้เพื่อใช้แสดงผล เวลาเลื่อนข้อความบนหน้าจอขึ้น แล้วจึงทำการเรียกใช้ฟังก์ชันข้อมูลแสดงผล

3.11.2 ส่วนการจัดรูปแบบการแสดงผล

โดยจะมีการแสดงผลเลื่อนจากล่างขึ้นไปเรื่อย ๆ มีการใช้คำสั่งเกี่ยวกับ Loop มาใช้ประโยชน์มาก การแสดงผล ก็จะเป็นฟังก์ชันแสดงผล และข้อความเดิมที่พิมพ์ไว้

ตัวอย่างโปรแกรมทั้งส่วนตรวจสอบคำสั่ง และส่วนที่จัดรูปแบบแสดงผลในการเรียกใช้คำสั่ง DIR สามารถเขียนได้ดังโปรแกรมข้างล่างนี้ เริ่มต้นที่การตรวจสอบคำสั่งว่าถูกต้องหรือไม่ จากนั้นก็เก็บคำสั่งนั้นไว้ในตัวแปร Array แล้วเรียกใช้ฟังก์ชันการแสดงผล โดยมีการใช้คำสั่ง for ในการเลื่อนข้อมูลที่แสดงบนหน้าจอภาพขึ้นทีละบรรทัด

```

Check_Comm(void)
{
    int loop,nst=0;
    char st[]="DIR";
    char mix[3];
    do{
        Check_Key(3,0,1);
        for(loop=0;loop<=2;loop++){
            if(toupper(gt[loop])==st[loop])
                {mix[loop]=gt[loop]; nst++;}
            if(nst==3){
                cktp=0;
                for(loop=15;loop>=1;loop--){

```

```

        topic2_1(loop);}
    }

    while(gt==st);
}
mix2(int y,char *dt)
{
    thaixy(1,y,10,"C:\\\>",0);
    thaixy(5,y,23,dt,0);
    thaixy(1,y+1,40,"",0);
}

```

3.11.3. ส่วนที่เป็นข้อความ

และแสดงผลที่เกิดจากการเรียกใช้คำสั่งนั้น ๆ ในโปรแกรมส่วนนี้จะสร้างในลักษณะเป็น ฟังก์ชันเล็ก ๆ มากมายเพื่อความสะดวกเร็วในการเรียกใช้ และเกิดความสะอาดสบายในการเขียน โปรแกรม สำหรับตัวอย่างโปรแกรมนี้จะมีการรับค่าตัวแปรเข้ามา 1 ค่า เพื่อให้มีการเลื่อนข้อมูลที่ แสดงทางหน้าจอขึ้นไปข้างบนเรื่อย ๆ เมื่อมีการใช้คำสั่งเกี่ยวกับการวน loop

```

topic2_1(int y)
{
    Set_Color(15);
    thaixy(1,2+y,79," Volume in drive A has no label",0);
    thaixy(1,3+y,79," Directory of A:\",0);
    thaixy(1,4+y,79,"",0);
    Set_Color(2);
    thaixy(1,4+y,79," 1      2      3      4      5      6",0);
    Set_Color(15);
    thaixy(1,5+y,79,"",0);
    thaixy(1,5+y,79,"ALPHA   TST   113  3-10-91  4:16p",0);
    thaixy(1,6+y,79,"LETTER   560 10-09-91  3:51p",0);
}

```

```

thaixy(1,7+y,79,"PEOPLE DOC 560 10-09-92 3:51p",0);
thaixy(1,8+y,79,"MENU BAT 85 3-16-90 12:36p",0);
thaixy(1,9+y,79,"TEST DOC 624 1-01-92 12:06a",0);
thaixy(1,10+y,79,"PEOPLE REP 560 10-09-92 3:51p",0);
thaixy(1,11+y,79,"PEOPLE MEM 560 10-09-90 3:51p",0);
thaixy(1,12+y,79,"WORD <DIR> 9-17-91 7:20p",0);
thaixy(1,13+y,79," 8 File(s) 289792 bytes free",0);
thaixy(1,14+y,79," ",0);
}

```

เมื่อได้ทำการเขียนส่วนต่าง ๆ ครบแล้วส่วนของการจัดรูปแบบการแสดงผล และส่วนที่เป็นข้อความแสดงผลเราก็สามารถทำการทดสอบโปรแกรมแสดงผลคำสั่งนี้ได้ โดยการเขียนโปรแกรมนี้จะเขียนอยู่ในรูปลักษณะของโปรแกรมที่สมบูรณ์แล้วทำการ RUN ได้เลย จากนั้นตรวจสอบผลของโปรแกรมว่าได้ตามที่ต้องการหรือไม่ ทำการทดสอบหลาย ๆ ครั้งจนแน่ใจว่า ไม่มีข้อผิดพลาดเกิดขึ้นแน่นอน เราก็สามารถนำโปรแกรมนี้ ไปเป็นตัวอย่างในการเขียนโปรแกรมแสดงผลคำสั่งอื่น ๆ ที่มีมากมายอีกต่อไป

3.12 การเขียนโปรแกรมหลักในการให้บทเรียน

ในการเขียนโปรแกรมหลักจะเขียนโปรแกรมทำงานในลักษณะ HOT KEY โดยแต่ละคีย์จะแสดงไว้ในส่วนล่างของหน้าจอ โดยในการเขียนโปรแกรมส่วนนี้จะใช้คำสั่งในการ Control Flow ของโปรแกรม เช่น switch case, if case, do while ดังตัวอย่างต่อไปนี้

```

do{
    key=getch();
    if(key==32) key=13;
    if(key==0) {
        key=getch();
        switch(key){
            /*F1*/ case 59 : help(1);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/*ALT-D*/case 32 : osdos();
break;
/*ALT-X*/case 45 : closegraph();
exit(0);
}}
}while((key==59)!!(key==32));

```

ในการเขียนโปรแกรมในส่วนนี้ จะประสบปัญหาในการควบคุมการทำงานตามลำดับของโปรแกรม ที่เมื่อกลับมาโปรแกรมหลักแล้วค่าบางค่าที่กำหนดไว้ จะมีการสูญหายไประหว่างการที่มีการเรียก ฟังก์ชันอื่น ๆ ใช้งานอื่นเนื่องมาจากโปรแกรมเป็น โหมดกราฟฟิก จึงมักจะประสบปัญหา เช่น ปัญหาในการ แสดงผลผิดเพี้ยนไป ซึ่งฟังก์ชันทุก ๆ ฟังก์ชัน ต้องมีการปรับปรุงใหม่เมื่อมีการนำมาเขียนรวมกัน

การทดสอบ

ทดสอบโดยการทดลองเรียกใช้เมนูต่าง ๆ หลาย ๆ รอบ และ หลาย ๆ วิธีโดยพิจารณาที่จุดจุดการแสดงผล สีการแสดงผล สีต่าง ๆ ที่เลือกระหว่างการใช้นิวว่าตรงตามค่าที่กำหนดไว้หรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การใช้ และการทดสอบการแสดงผลของโปรแกรมช่วยสอนคอมพิวเตอร์

การเข้าสู่โปรแกรมช่วยสอนคอมพิวเตอร์จะมีขั้นตอนในการทำดังนี้

1. ทำการเปิดเครื่อง
2. สังเกตจอภาพจะปรากฏ
C:> ในกรณีที่ไม่มีฮาร์ดดิสก์
A:> หรือ B:> ในกรณีที่ไม่มีฮาร์ดดิสก์
3. ทำการโหลดโปรแกรมช่วยสอนคอมพิวเตอร์โดยการพิมพ์
c:> DOSTRAN

แล้วกด Enter จะแสดงเมนูหลักดังรูป



รูปที่ 4.1 แสดงเมนูหลัก

ซึ่งในส่วนของเมนูหลักนี้จะประกอบด้วยส่วนต่าง ๆ ดังนี้

1. ส่วนของคำอธิบายการใช้งานโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

2. ส่วนของบทเรียนซึ่งมีทั้งหมด 7 บท ประกอบด้วย

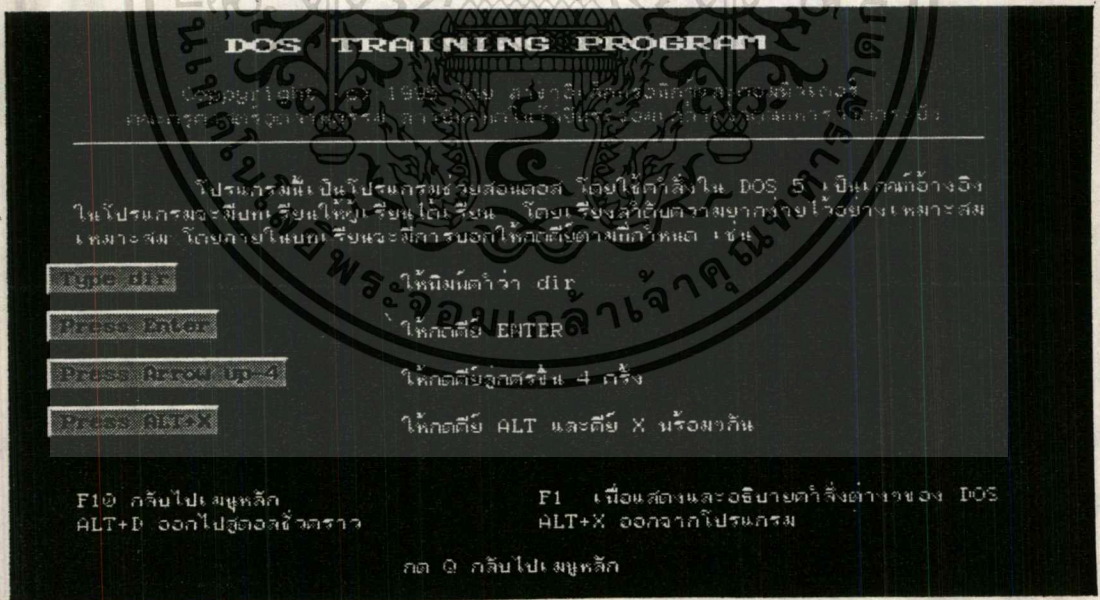
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามเผยแพร่เนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- บทที่ 1. คอมพิวเตอร์ และดอส
- บทที่ 2. ดิสก์ และไดเรกทอรี
- บทที่ 3. การจัดการเกี่ยวกับไฟล์
- บทที่ 4. 5 คำสั่งพิเศษสำหรับการจัดการไฟล์
- บทที่ 5. การจัดการเกี่ยวกับดิสก์
- บทที่ 6. การป้องกันไฟล์ และข้อมูล
- บทที่ 7. ดอสคีย์ และคำสั่ง

3. ส่วนการกลับสู่ดอส

4.1 ส่วนของคำอธิบายการใช้งานโปรแกรม

จากรูปที่ 4.1 ให้เลื่อนแถบสว่างมาที่ การใช้งานโปรแกรม แล้วกด Enter ซึ่งจะเห็นว่า เป็นส่วนของคำอธิบายต่าง ๆ ที่ใช้ในโปรแกรมโดยจะอธิบายถึงหน้าที่ของคีย์แต่ละคีย์ และอธิบาย สัญลักษณ์ต่าง ๆ ในการใช้คีย์ แสดงดังรูป



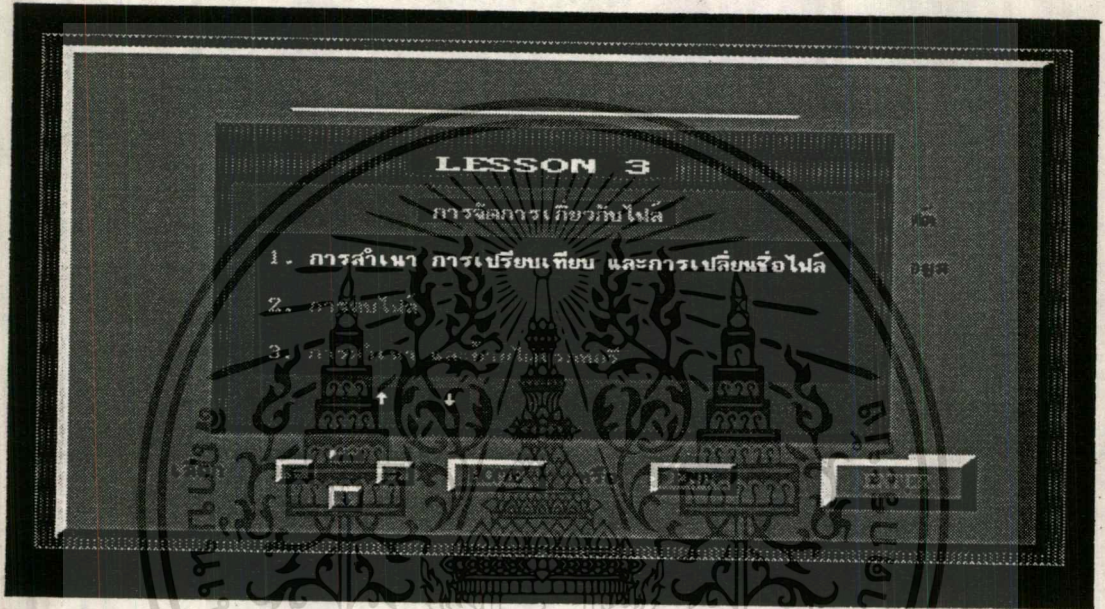
รูปที่ 4.2 ส่วนของคำอธิบายการใช้งานโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ส่วนของบทเรียน

ในแต่ละบทเรียนจะมีลักษณะการใช้งานที่เหมือนกัน ซึ่งในที่นี้จะขอยกตัวอย่างการเรียนของบทเรียนที่ 3 ซึ่งมีขั้นตอนดังนี้ ส่วนบทเรียนอื่น ๆ จะมีขั้นตอนเหมือนกัน

4.2.1 จากรูปที่ 4.1 เลื่อนแถบสว่างมาที่บทที่ 3 แล้วกด Enter ซึ่งจะแสดงดังรูป



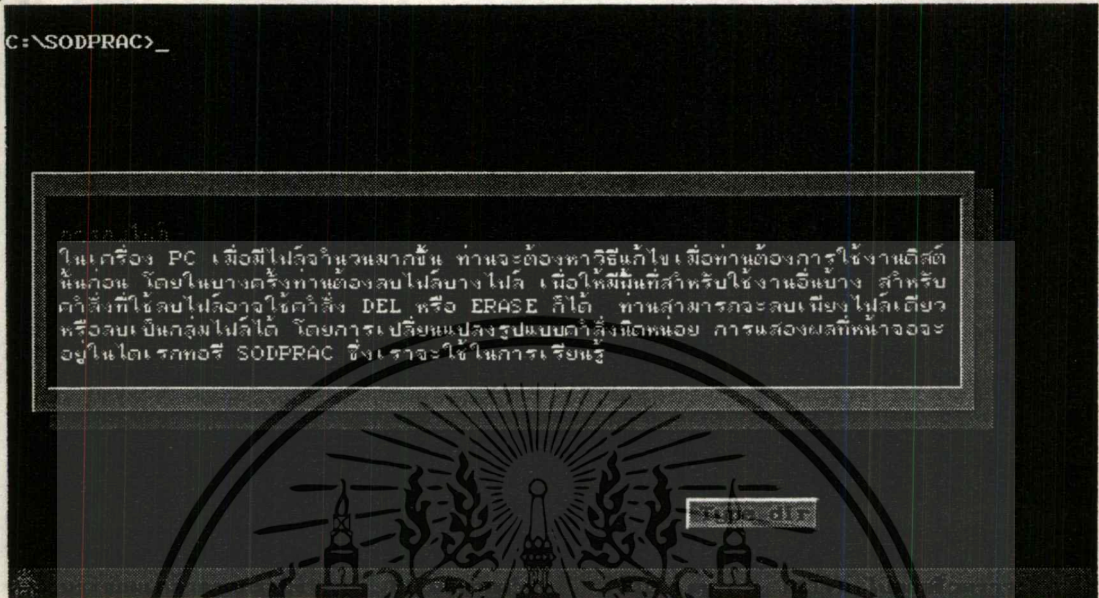
รูปที่ 4.3 ส่วนเมนูของบทเรียนที่ 3

จากรูปที่ 4.3 จะแสดงหัวข้อที่ต้องเรียนต่าง ๆ ในบทที่ 3 ซึ่งสามารถเลือกเรียนแต่ละเรื่องได้

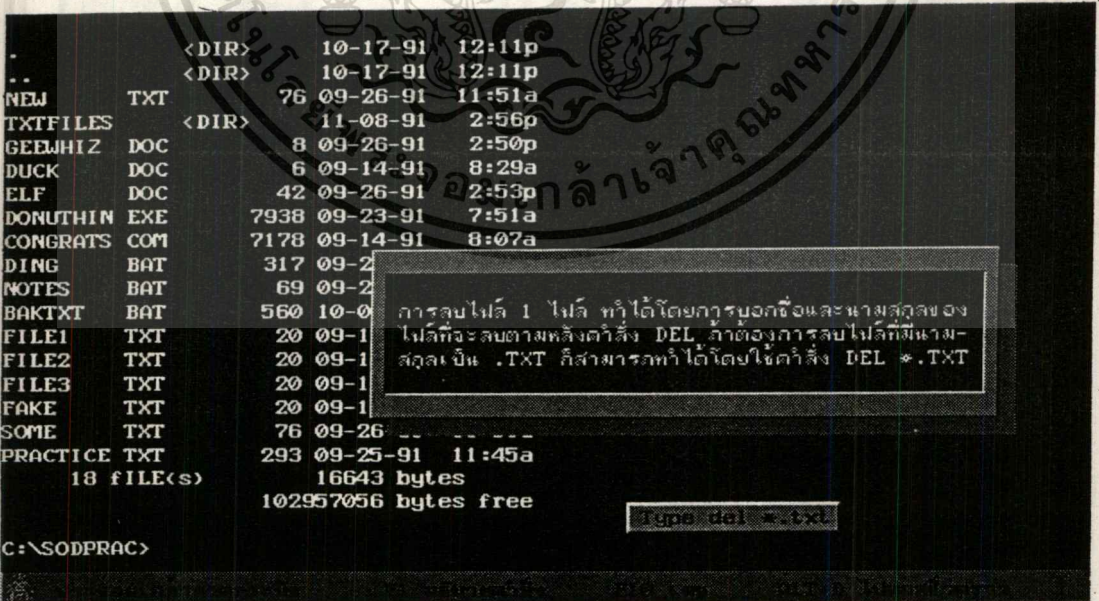
4.2.1.1 เลือกเรียนหัวข้อที่ 2 ซึ่งเป็นเรื่องของการลบไฟล์ ทำได้โดยการเลื่อนแถบสว่างมาที่ การลบไฟล์ แล้วกด Enter จะแสดงดังรูป

4.2.1.2 จากรูปที่ 4.4 ให้พิมพ์คำว่า DIR แล้วกด Enter เพื่อเข้าสู่ขั้นตอนการทำงานของคำสั่ง DIR แล้วจะบอกถึงคำสั่งต่อไปที่ต้องพิมพ์ คือคำสั่ง Del *.txt และยังมีคำอธิบายผลของคำสั่ง Del *.txt ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 แสดงคำอธิบายของหัวข้อที่เลือกเรียน



รูปที่ 4.5 ผลของการใช้คำสั่ง DIR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1.3 จากผลของคำสั่ง `del *.txt` จะแสดงดังรูปที่ 4.6 แล้วคำสั่งต่อไปให้พิมพ์

`tree /f`

```
C:\SODPRAC>del *.txt
C:\SODPRAC>dir

Volume in drive C is MS-DOS_5
Volume Serial Number is 16D8-76FC
Directory of C:\SODPRAC

. <DIR>      10-17-91  12:11p
.. <DIR>      10-1
TXTFILES <DIR>  11-0
GEELWHIZ  DOC       8 09-2
DUCK      DOC       6 09-1
ELF       DOC      42 09-2
DONUTHIN  EXE     7938 09-2
CONGRATS  COM     7178 09-1
DING      BAT      317 09-2
NOTES     BAT       69 09-2
BAKTXT    BAT      560 10-02
          11 FILE(S)      16118 bytes
          102985728 bytes free

C:\SODPRAC>
```

ภาพรวมไฟล์ทั้งหมดที่มีนามสกุลเป็น .TXT จากจับไตเรคทอรีชื่อ SODPRAC แต่ท่านยังไม่ได้ลบในจับไตเรคทอรี TXT-FILES สำหรับคำสั่ง TREE จะแสดงรายชื่อไฟล์ทั้งหมดในไตเรคทอรีนี้ แต่ถ้าใช้คำสั่ง TREE ร่วมกับ /F จะแสดงรายชื่อของไฟล์ทั้งหมดที่อยู่ในจับไตเรคทอรี

รูปที่ 4.6 ผลของการใช้คำสั่ง `Del *.txt`

```
C:\SODPRAC>tree /f
Directory PATH listing for Volume MS-DOS_5
Volume Serial Number is 16D8-76FC
C:..
  GEELWHIZ.DOC
  BIRD.DOC
  ELF.DOC
  DONUTHIN.EXE
  CONGRATS.COM
  DING.BAT
  NOTES.BAT
  BAKTXT.BAT
  TXTFILES
    NEW.TXT
    FILE1.TXT
    FILE2.TXT
    FILE3.TXT
    FAKE.TXT
    SOME.TXT
    PRACTICE.TXT

C:\SODPRAC>
```

ถ้าท่านดูที่จับไตเรคทอรีชื่อ TXTFILES ที่ท่านสร้างขึ้นในตอนแรก ท่านจะเห็นไฟล์ที่มีนามสกุล .TXT ต่อไปใช้คำสั่ง DEL ลบไฟล์ในเรคทอรีไฟล์ที่มีนามสกุลเป็น .TXT ในจับไตเรคทอรี TXTFILES

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในงานเพื่อการศึกษานับ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

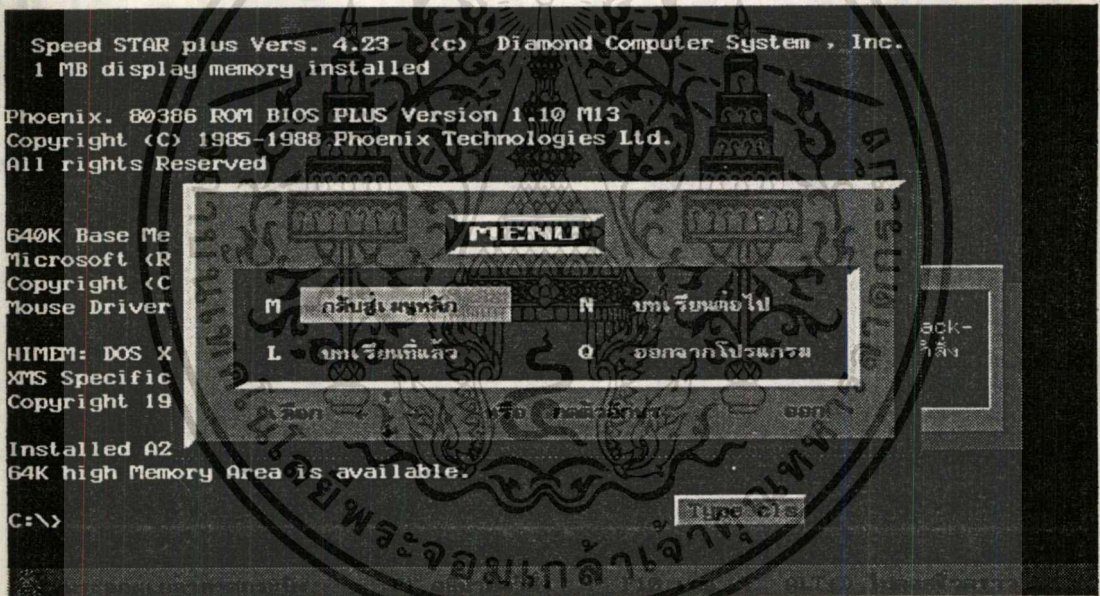
รูปที่ 4.7 ผลของการใช้คำสั่ง `tree /f`

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1.4 ผลของคำสั่ง tree /f แสดงดังรูปที่ 4.7 และจากรูปที่ 4.7 จะบอกถึงคำสั่งต่อไป คือคำสั่ง c:\txtfiles \new.txt ซึ่งจะอธิบายผลของคำสั่ง c:\txtfiles \new

4.2.1.5 ถ้าต้องการเรียนคำสั่ง ๆ ภายใบบทเรียนนี้ต่อไปก็สามารถทำได้โดยวิธีเดียวกับข้อ 4.2.1.1 - 4.2.1.4

4.2.2 ในกรณีที่ผู้เรียนต้องการที่จะข้ามไปยังบทเรียนบทต่อไป โดยยังเรียนบทเรียนนั้น ๆ ยังไม่จบ หรือต้องการกลับไปยังบทเรียนที่ผ่านมา หรือต้องการที่จะออกจากโปรแกรมในกรณีที่เรียนจบแล้ว ทำได้โดยการกด F10 ซึ่งแสดงดังรูป



รูปที่ 4.8 เมนูจากการกด F10

จากรูปที่ 4.8 จะแสดงเมนูทั้งหมด 4 หัวข้อดังนี้

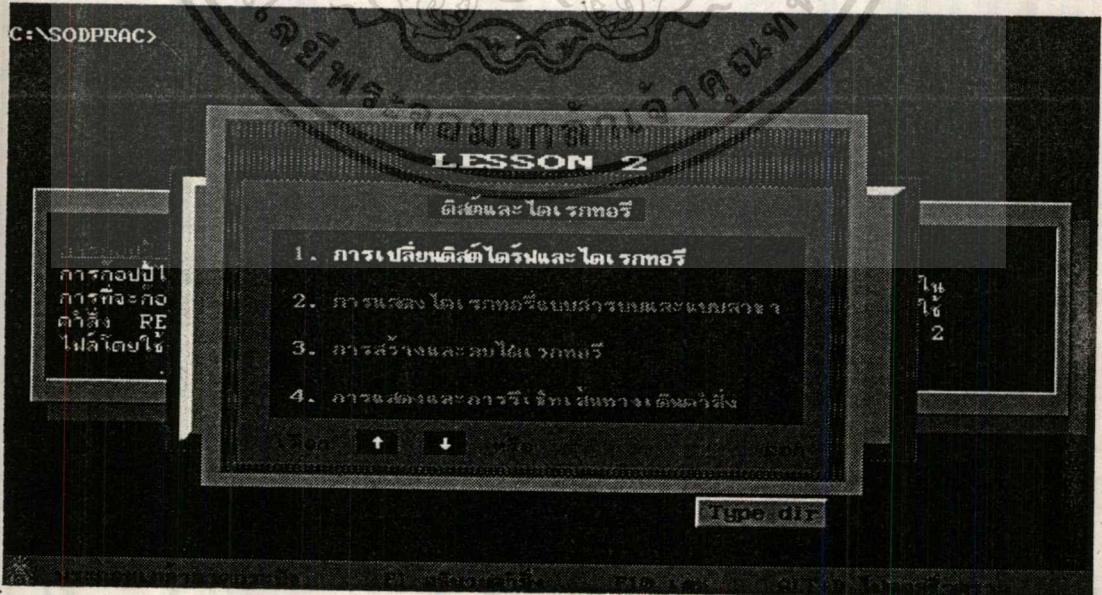
- 1) M กลับสู่เมนูหลัก
- 2) L บทเรียนที่แล้ว
- 3) N บทเรียนต่อไป
- 4) Q ออกจากโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2.1 ถ้าต้องการกลับไปยังเมนูหลักทำได้โดยการเลื่อนแถบสว่างมาที่กลับสู่เมนูหลัก แล้วกด Enter หรือกด M จะแสดงเมนูดังรูป



รูปที่ 4.9 ผลของการกลับสู่เมนูหลัก

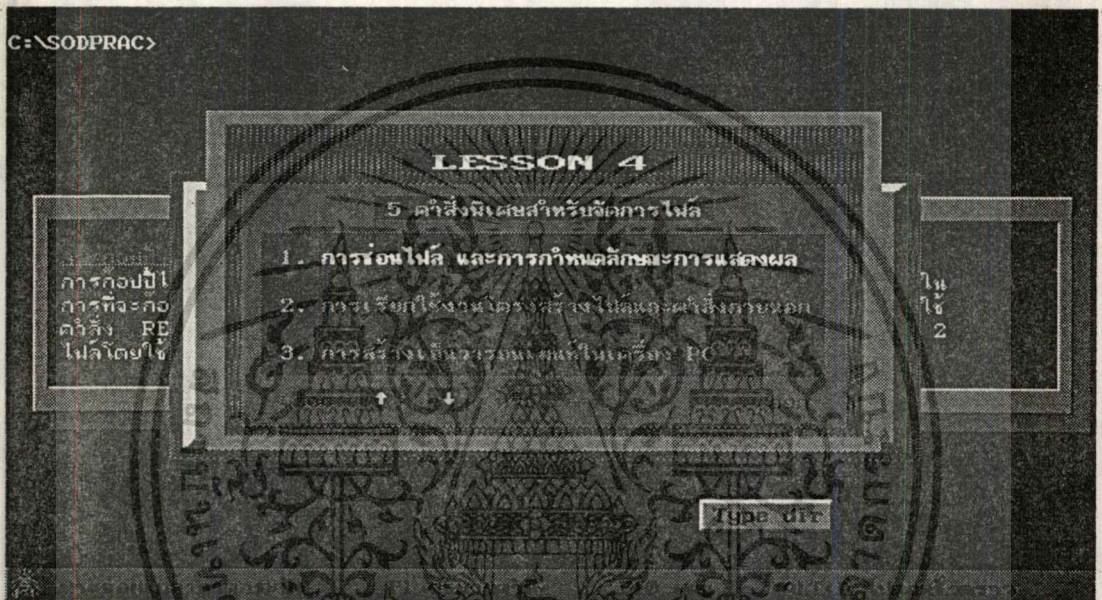


รูปที่ 4.10 ผลของการกลับไปยังบทเรียนที่แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

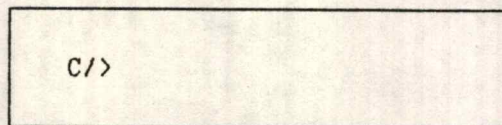
4.2.4.2 ถ้าหากต้องการกลับไปยังบทเรียนที่แล้วให้เลื่อนแถบสว่างไปที่ บทเรียนที่แล้ว แล้วกด Enter หรือกด L จะแสดงเมนูดังรูป ซึ่งจะเป็นบทเรียนที่ 2

4.2.2.3 ถ้าต้องการข้ามไปยังบทเรียนต่อไปให้เลื่อนแถบสว่างไปที่ บทเรียนต่อไปแล้ว กด Enter หรือกด N จะแสดงดังรูป ซึ่งจะเป็นบทที่ 4



รูปที่ 4.11 ผลของการข้ามไปยังบทเรียนบทต่อไป

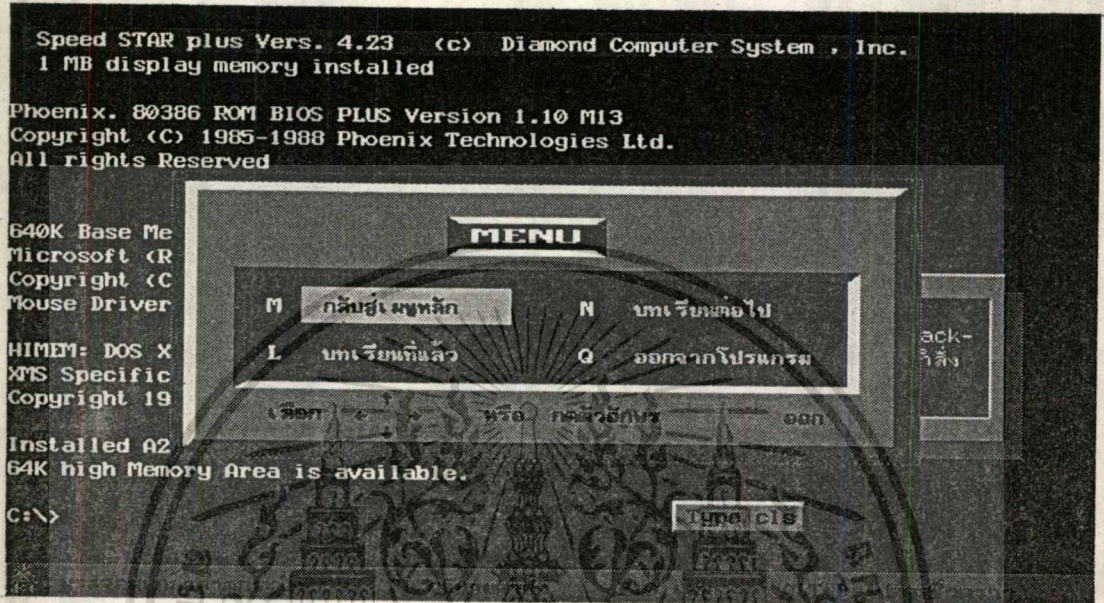
4.2.2.4 ถ้าต้องการออกจากโปรแกรมให้เลื่อนแถบสว่างไปที่ ออกจากโปรแกรม แล้ว กด Enter หรือกด Q จะแสดงดังรูป



รูปที่ 4.12 ผลของการออกจากโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

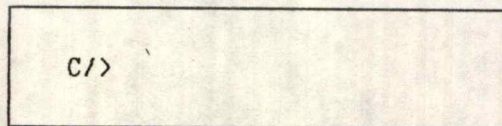
จากรูปที่ 4.8 จะเห็นว่ามีปุ่ม ESC เพื่อต้องการที่จะออกจากเมนู แล้วกลับไปยังคำสั่งที่ทำ
คำสั่งสุดท้าย จะแสดงดังรูป



รูปที่ 4.13 ผลของการออกจากเมนูโดยใช้ ESC

4.3 การกลับสู่ ดอส

เมื่อผู้เรียนเรียนจบแล้ว หรือไม่ต้องการที่เรียนต่อในครั้งนั้นแล้ว ให้ผู้เรียนเลื่อนแถบสว่าง
มาที่ กลับสู่ดอส ในเมนูหลักแล้วกด Enter จะแสดงดังรูป

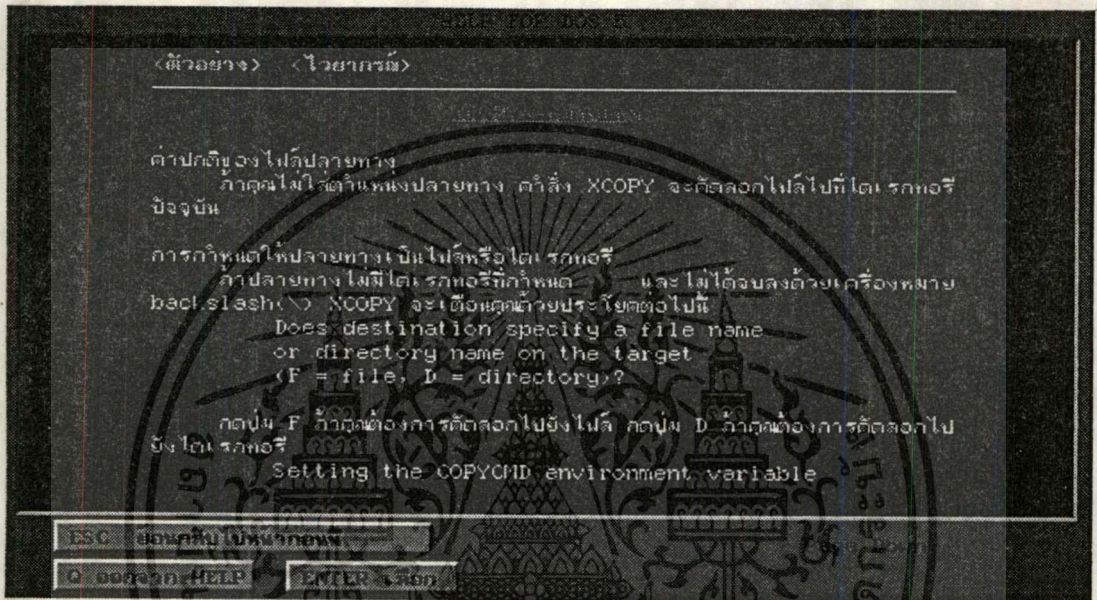


รูปที่ 4.14 ผลของการกลับสู่ ดอส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 การใช้ส่วนความช่วยเหลือ

ในกรณีที่ผู้เรียนกำลังเรียนอยู่ แต่ผู้เรียนต้องการทราบรายละเอียดของคำสั่งนั้น ๆ อย่างละเอียด ผู้เรียนสามารถกด F1 เพื่อต้องการที่จะแสดงคำสั่งนั้น ๆ เช่นดังรูป เป็นรายละเอียดของคำสั่ง Copy



รูปที่ 4.15 รายละเอียดของส่วนขอความช่วยเหลือ

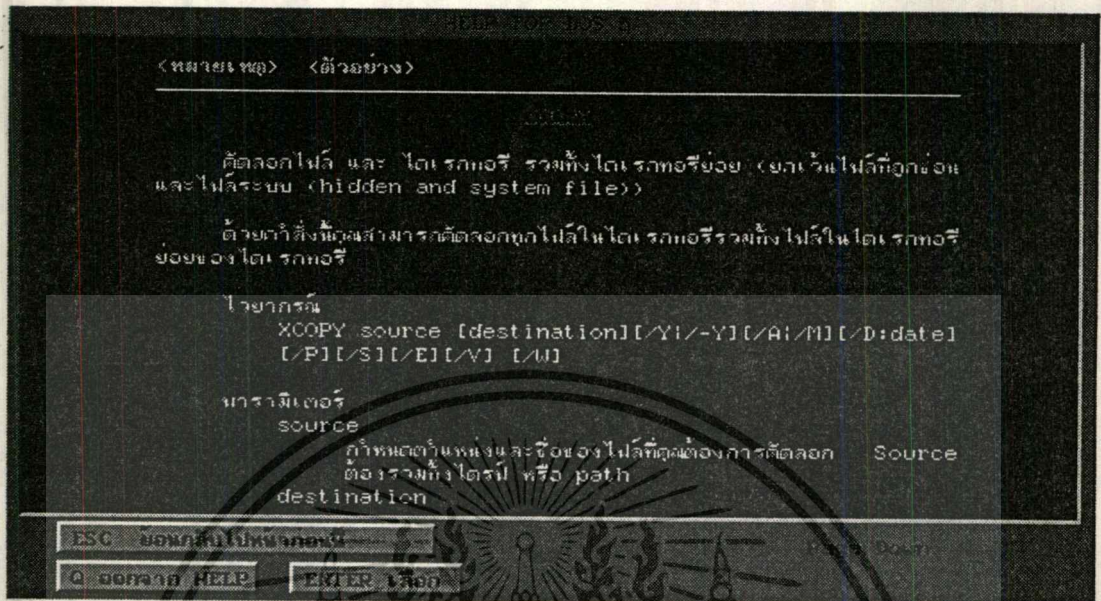
จากรูปที่ 4.15 จะแสดงส่วนประกอบ 3 ส่วน คือ

- 1) ไวยากรณ์
- 2) หมายเหตุ
- 3) ตัวอย่าง

4.4.1 ส่วนแสดงไวยากรณ์

จะกล่าวถึงประโยชน์ของคำสั่งนั้น ๆ รูปแบบการใช้งาน และอธิบายพารามิเตอร์ต่าง ๆ ที่ใช้ในคำสั่งนั้น ๆ ดังรูป

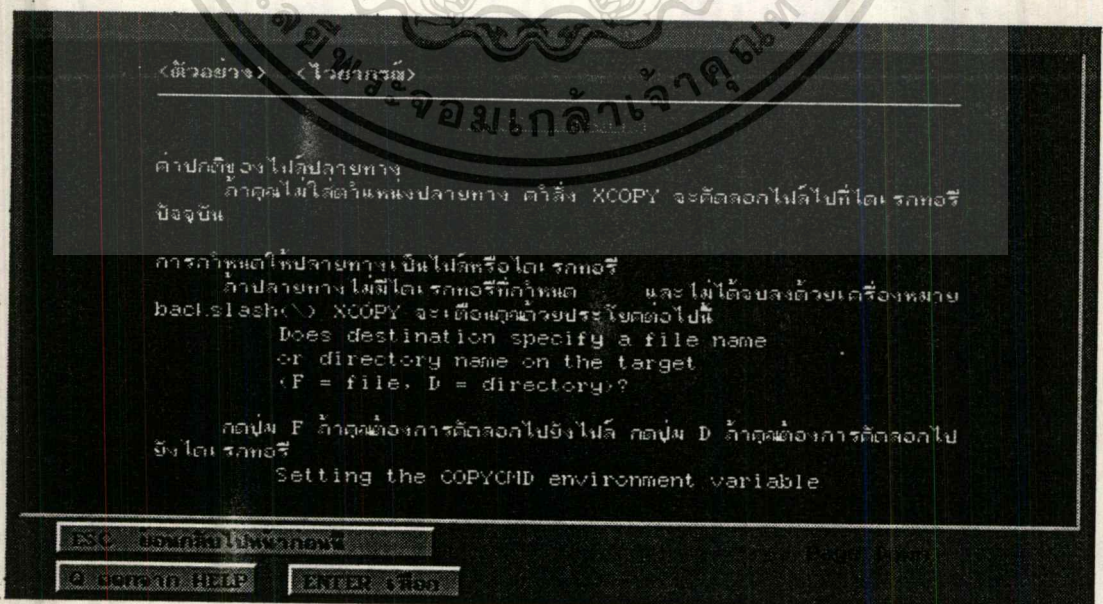
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.16 รูปแบบของส่วนการใช้งานไวยากรณ์

4.4.2 ส่วนแสดงหมายเหตุ

กล่าวถึงการใช้งานของคำสั่งอย่างกว้าง ๆ ดังรูป



รูปที่ 4.17 รูปแบบของส่วนการให้หมายเหตุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามแก้ไขตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทสรุปและวิจารณ์

5.1 สรุป

ปริศยานิพนธ์ฉบับนี้เป็นการนำเสนอโปรแกรมช่วยสอนคอส โปรแกรมนี้เหมาะกับกลุ่มบุคคลที่เริ่มต้นศึกษาเกี่ยวกับระบบปฏิบัติการคอส หรือผู้ที่มีความรู้อยู่บ้างแล้วให้มีความเข้าใจลึกซึ้งมากยิ่งขึ้น การศึกษาเกี่ยวกับระบบปฏิบัติการคอสด้วยตัวเองนั้นอาจจะเกิดปัญหาต่าง ๆ ได้แต่ปัญหาเหล่านั้นสามารถแก้ไขได้ด้วย DOS TRAINING PROGRAM

การเขียนโปรแกรมช่วยสอนคอสนี้จะใช้ภาษา C ในโหมดกราฟิกเขียนทั้งหมด ซึ่งจะจำลองหลักการทำงานคำสั่งของคอสและจะแสดงหลักการทำงานของคำสั่งเหล่านั้นเมื่อมีการเรียกใช้หรือสามารถเรียกดูวิธีการใช้คำสั่งต่าง ๆ ทั้งหมดของโปรแกรม โดยจะมีการแสดงผลเป็นภาษาไทย

ประสิทธิภาพในการทำงานของโปรแกรมช่วยสอนคอสนี้ สามารถทำงานได้ง่าย ทั้งยังมีส่วนของโปรแกรมการช่วยเหลือต่าง ๆ ให้ผู้ใช้ได้ใช้งานได้อย่างสะดวก แต่ในโครงการนี้ก็ยังมีข้อบกพร่องและปัญหาที่เกิดขึ้นหลายประการทางกลุ่มผู้จัดทำโครงการนี้ได้เขียนข้อเสนอแนะ และวิธีการแก้ไข เพื่อที่จะเป็นประโยชน์ให้กับผู้ที่จะนำโปรแกรมช่วยสอนคอสนี้ไปพัฒนาต่อไป ดังมีรายละเอียดต่อไปนี้

5.2 ปัญหาที่พบในการทำโครงการ

5.2.1 การเขียนโปรแกรมโดยใช้ภาษาซี ต้องเสียเวลาเป็นอย่างมาก ในการเขียนภาษาไทยกราฟฟิกออกหน้าจอ

5.2.2 การแสดงผลภาษาไทยให้ออกทางจอภาพเป็นการเขียนหน่วยความจำโดยตรงดังนั้นการจัดระดับของการแสดงผลจึงทำได้ยาก

5.2.3 เนื่องจากโปรแกรมการทำงานในกราฟิกโหมด ดังนั้นการแสดงผลทางจอภาพจะช้ากว่าการแสดงผลตามปกติ

5.2.4 ในการแสดงผลเคอร์เซอร์ ทำให้การแสดงผลการกดคีย์ช้า เพราะมีการใช้คำสั่งดีเลย์

5.2.5 เนื่องจากข้อมูลและโปรแกรมอยู่รวมกันจึงทำให้การจัดการหน่วยความจำทำได้ยาก เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 การแก้ปัญหา

5.3.1 ต้องพยายามเขียนโปรแกรมอย่างต่อเนื่องเพื่อความชำนาญ

5.3.2 ศึกษาจากโปรแกรมที่มีอยู่แล้ว แล้วมาพัฒนาต่อ

5.3.3 ในการแสดงผลใช้วิธีการเขียนหน่วยความจำโดยตรงซึ่งเป็นวิธีที่ทำให้การแสดงผลเร็วที่สุด

5.3.4 ลดค่าดีเลย์ลง หรือใช้เครื่องคอมพิวเตอร์ที่มีความเร็วมาก ๆ

5.3.5 ต้องพยายามแยกข้อมูลออกจากตัวโปรแกรม โดยตัวโปรแกรมจะเรียกใช้ข้อมูลจากภายนอกทำให้การจัดการหน่วยความจำ เป็นไปอย่างมีระเบียบ

5.4 ประโยชน์ที่ได้รับจากการทำโครงการ

5.4.1 สามารถใช้คำสั่งต่าง ๆ ของ DOS ได้อย่างมีประสิทธิภาพ

5.4.2 ได้รับความรู้ความเข้าใจในการเขียนโปรแกรมกราฟิคของภาษา C

5.4.3 ได้โปรแกรมสื่อการเรียนการสอนได้

5.4.4 นำโปรแกรมที่เขียนขึ้นไปใช้ในการเรียนการสอนจริง ๆ ได้

5.4.5 ประหยัดงบประมาณในการทำสื่อการเรียนการสอน

5.5 แนวทางการพัฒนาต่อ

5.5.1 เพิ่มความเร็วในการแสดงผลของโปรแกรม

5.5.2 เพิ่มบทเรียนให้ครอบคลุมทุกฟังก์ชันของดอส

5.5.3 แยกข้อมูลกับตัวโปรแกรมออกจากกันอย่างเด็ดขาด

5.5.4 เพิ่มลูกเล่นให้สวยงามขึ้นเพื่อดึงดูดผู้เรียน

5.5.5 เขียนโปรแกรมที่มีโครงที่แน่นอน เพื่อสะดวกต่อการแก้ไขและการพัฒนาต่อไป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมนี้เป็นโปรแกรมช่วยสอนคอส โดยใช้คำสั่งในคอส 5 เป็นเกณฑ์ในการอ้างอิงซึ่ง เป็นคำสั่งบางส่วนเท่านั้น ในโปรแกรมช่วยสอนนี้มีบทเรียนทั้งหมด 7 บท โดยสามารถเลือกเรียน บทใดก่อนหลังก็ได้ แต่ถ้าเรียนบทที่ 1 ก็จะต้องเรียนต่อเนื่องไปจนจบทั้งหมด 7 บท

การเรียนเราจะกำหนดความยากง่ายไว้เป็นขั้นตอน เริ่มจากง่าย ๆ ก่อน

ถ้าหากต้องการออกจากบทเรียนในระหว่างที่กำลังเรียนอยู่นั้น ก็สามารถทำได้โดยใช้คีย์ พิเศษ ที่เป็นคีย์ช่วยเหลือดังจะได้อธิบายในเรื่องของคีย์พิเศษต่อไป

การเริ่มใช้งานก็เริ่มจากผู้เรียนได้เรียกโปรแกรม และทำการพิมพ์คำสั่ง หรือกดคีย์ที่หน้าจอ กำหนดมาให้ เช่น

ให้พิมพ์คำว่า dir ที่หน้าจอก็จะกำหนดมาว่า Type dir

ให้กดคีย์ Enter ที่หน้าจอก็จะกำหนดมาว่า Press Enter

ให้กดคีย์ลูกศรขึ้น 4 ครั้ง ที่หน้าจอก็จะกำหนดมาว่า Arrow Up-u

ให้กดคีย์ Alt และคีย์ x พร้อม ๆ กัน ที่หน้าจอก็จะกำหนดมาว่า Press Alt+x

โดยเมื่อเราทำการพิมพ์ หรือกดไปแล้วจะมีคำอธิบายขึ้นมาเพื่อที่จะอธิบายคำสั่งนั้น ๆ ให้ เข้าใจยิ่งขึ้น แต่ถ้าหากไม่เข้าใจก็สามารถกด F1 เพื่อแสดงคำอธิบายคำสั่งที่กำลังเรียนอยู่นั้น โดยละเอียด

ในขณะที่กำลังเรียนอยู่นั้น เมื่อต้องการทดลองคำสั่งคอส 5 จริง ๆ ก็สามารถทดลองได้ โดยการออกจากโปรแกรมชั่วคราว โดยใช้คีย์พิเศษ Alt+x ซึ่งทำให้เกิดความคล่องตัวในการ เรียนรู้เมื่อเข้าใจแล้ว ก็สามารถกลับไปเรียนบทต่อไปได้โดยพิมพ์คำว่า Exit

สำหรับคีย์พิเศษซึ่งจะใช้ช่วยเหลือ เพื่อเพิ่มสมรรถภาพในการทำงานของโปรแกรม ซึ่งจะมี อยู่ 4 คีย์ โดยจะแสดงไว้ที่บรรทัดสุดท้ายของหน้าจอ และสามารถเรียกใช้ได้ทันทีโดยใช้คีย์ต่างๆ ดังนี้

F1 เพื่อแสดง และอธิบายคำสั่งต่าง ๆ ของคอสในคำสั่งที่กำลังเรียนอยู่

F10 แสดง เมนู สามารถเลือกการทำงานดังนี้

M กลับสู่เมนูหลัก

N ไปบทเรียนต่อไป

L ไปบทเรียนที่แล้ว

Q ออกจากโปรแกรม

Alt+D ออกไปสู่คอสชั่วคราว

Alt+x ออกจากโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุเปลี่ยนแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้คีย์ F10 นั้นเราสามารถเลือกได้ว่า จะไปเรียนบทเรียนต่อไปเมื่อไม่ต้องการเรียน
ในบทเรียนที่กำลังเรียนต่อ และสามารถกลับไปบทเรียนที่แล้วได้เมื่อยังไม่เข้าใจคีย์
คีย์ต่าง ๆ จะมีคำอธิบายไว้เป็นภาษาไทย เพื่อให้เข้าใจในการใช้ได้ง่าย



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการเขียนโปรแกรมเนื่องจากเป็นโปรแกรมใหญ่และซับซ้อน จึงได้ทำเป็นโปรเจกต์ไฟล์ชื่อ DOSTRAIN.PRJ โดยในโปรเจกต์ไฟล์จะประกอบไปด้วย ไฟล์ต่างๆ ต่อไปนี้

MAIN.C	TOPIC2.C
MAINMENU.C	TOPIC3.C
MENU.C	TOPIC4.C
MENUTOP.C	TOPIC5.C
THAIDRI.C	TOPIC6.C
HELP.C	TOPIC7.C
TOPIC1.C	

ในการคอมไพล์โปรแกรมจะต้องมีไฟล์เหล่านี้ประกอบ

BX14.FON	TOPIC2.HLP
HELP1.HLP	TOPIC3.HLP
HELP2.HLP	TOPIC4.HLP
HELP3.HLP	TOPIC5.HLP
HELP4.HLP	TOPIC6.HLP
HELP5.HLP	TOPIC7.HLP
TOPIC1.HLP	

หลังจากการคอมไพล์จะได้ DOSTRAIN.EXE ซึ่งในการรันโปรแกรมจะต้องมีไฟล์เหล่านี้ อยู่ในชั้นเดียวกัน

DOSTRAIN.EXE	EGAVGA.BGI
BX14.FON	TOPIC2.HLP
HELP1.HLP	TOPIC3.HLP
HELP2.HLP	TOPIC4.HLP
HELP3.HLP	TOPIC5.HLP
HELP4.HLP	TOPIC6.HLP
HELPS.HLP	TOPIC7.HLP

```
*****
***** PROGRAM MAIN.C *****
*****
```

```
#include<dos.h>
#include<graphics.h>
#include<stdio.h>
#include<alloc.h>
#include<io.h>
#include<fcntl.h>
#include<stdarg.h>
#include<stdlib.h>
#include<conio.h>

extern char scan,exten;
extern char Blod;
extern char Tab;
static int presstab=0;
unsigned int on;
extern unsigned statuskey;
extern int count;
extern char OldMode;
extern unsigned int Off_Line[350];
extern unsigned int G_Row;
extern unsigned int G_Col;
extern void barband();
extern void help(int page);
extern void error(int status);
extern void topic_3();
extern void Check_3(char *g,int topic);
```

```

extern void Check_2(char *g,int topic);
extern void topic_2();
extern void topic_1();
extern void Check_1(char *g,int topic);

int Num_St,Ad_Cu,Ad_St,cktp,Select;
char gt[30],st[30];
int to=1,res,con,bear=1,inter,gint=1;
void far *gscreen;

void Check_Key(int num,int colum,int row,int topic,int ok,int tp);
typedef unsigned char byte;
void thaixypix(int colum,int row,int colend,char *content,int status,
               int color,int level);
void hot_key(int colum,int row,int delay,int ok,char key);
void interrupt int8(void);
void interrupt (*oldfunc)();
void hotkey(char key);
void Gkey(void);
void osdos();
void Cint(int select);
void interrupt int8(void);
void interrupt (*oldfunc)();

/* ฟังก์ชัน main เป็นจุดเริ่มต้นทำงานของโปรแกรม โดยจะเรียกใช้การแสดงผลเมนูเพื่อเลือก
   การเรียนในบทเรียนต่าง ๆ หรือต้องการคำอธิบายวิธีใช้งานโปรแกรม */
void main()
{
    int index;
    opendir();

```

```

Mainmenu();
closegraph();
}

/* ฟังก์ชัน Check_Key เป็นการตรวจสอบคีย์ที่ถูกกดว่าเป็นคีย์อะไร โดยมีการแสดงผลการกด
คีย์ทางจอภาพ ความคมการทำงานของคีย์พิเศษ เช่น HOME, END, DELETE, INSERT */
void Check_Key(int num,int colum,int row,int topic,int ok,int tp)
{
    char Swap[2],ch[11];
    int loop1,loop2,gnum,ck,clearB;
    unsigned long size1,size2;
    char far *buf1,*buf2;

    Num_St=Ad_Cu=Ad_St=0;Swap[2]=0;
    for(gnum=0;gnum<=30;gnum++)
    gt[gnum]=0;
    for(gnum=0;gnum<=30;gnum++)
    st[gnum]=0;
    if(ok==0) thaixy(colum+1,row,79,"C:\\\\",0);
    if(ok==2) thaixy(colum+1,row,79,"A:\\\\",0);
    ck=0;cktp=1;clearB=0;
do{
    ch[0]=0;ch[11]=0;
    clearB++;
    if(clearB==1) clearb();
    Cursor(colum,row,Ad_Cu,ck,100);
    ch[0]=getch();
    switch(ch[0]){
        case 8 : if(Ad_St>=1) {

```

```

Ad_St--; Num_St--;
gt[Ad_St]=0;
Del(colum,row,Ad_Cu,0);
for(loop1=Ad_St+1;loop1<=num;loop1++){
    if(gt[loop1]!=0){
        gt[loop1-1]=gt[loop1];
        gt[0]=gt[1];
        gt[loop1]=0;
        thaixy(5+(Ad_St+(1*(loop1-1-Ad_St))+colum),
            row,79,&gt[loop1-1],0);
        Del(colum,row,Ad_Cu,8*(loop1-Ad_St));
    }
}
break;
case 13 : Check_t(&gt[0],topic,tp);
    if(cktp==1) {
        printf("\a");
        error(2);
        Ad_Cu=0;Ad_St=0;
    }
}
break;
case 0 : break;
case 27 : break;
default : Del(colum,row,Ad_Cu,0);
    if(Ad_St<num)
        thaixy(5+(1*Ad_St)+colum,row,79,&ch[0],0);
    Swap[0]=ch[0];
    if(Ad_St<num){

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุใดเปลี่ยนแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Swap[1]=gt[Ad_St];
gt[Ad_St]=Swap[0];
Swap[0]=Swap[1];
for(loop1=Ad_St;loop1<=num;loop1++) {
    Del(column,row,Ad_Cu,8*(loop1+1-Ad_St));
    if(loop1<num-1)
        thaixy(6+(Ad_St+(1*(loop1-Ad_St))+column),
            row,79,&Swap[1],0);
    if(loop1!=num) {
        Swap[1]=gt[loop1+1];
        gt[loop1+1]=Swap[0];
        Swap[0]=Swap[1];
        gt[num]=0;
    }
    if(Num_St!=num) Num_St++;
}
Ad_Cu+=8;
if(ck==0) gt[Ad_St]=ch[0];
if(Ad_St<=num) Ad_St++;
if(Num_St!=num) Num_St++;}
if(Ad_St>=num){ Check_t(&gt[0],topic,tp);
if(cktp==1)
    {printf("\a");error(2);Ad_Cu=0;Ad_St=0;}}
break;
}
if(ch[0]==0) {
    ch[0]=getch();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/*F10*/ case 68 : Menu();
                break;

/*F1*/ case 59 : if(gint==0) {restore1();Cint(inter);}
                help(on);
                break;

/*HOME*/ case 71 : Ad_Cu=0;Ad_St=0;
                break;

/*ALT-D*/ case 32 : osdos();
                break;

/*ALT-X*/ case 45 : closegraph();
                exit(0);

/*LEFT*/ case 75 : if(Ad_St==0) break;
                Ad_Cu--=8;
                Ad_St--;
                break;

/*RIGHT*/ case 77 : if(Ad_St==Num_St) break;
                Swap[1]=0;
                for(loop1=Ad_St;loop1<=num;loop1++)
                if((gt[loop1]==32)|| (gt[loop1]==0)) Swap[1]++;
                if(Swap[1]==(num-Ad_St+1)) {Num_St=Ad_St; break;}
                Ad_Cu+=8;
                Ad_St++;
                break;

/*END*/ case 79 : for(loop2=0;loop2<=num;loop2++)
                if((gt[loop2]==32)|| (gt[loop2]==0))
                {
                    Swap[1]=0;
                    for(loop1=loop2;loop1<=num;loop1++){
                        if((gt[loop1]==32)|| (gt[loop1]==0))
                            Swap[1]++;
                    }
                }

```

```

        if (Swap[1]==(num-loop2+1)){
            Num_St=loop2;
            break;
        }
    }
    Ad_Cu=8*Num_St; Ad_St=Num_St;
    break;

/*INS*/ case 82 : if(ck==0) ck=1;
                else ck=0;
                break;

/*DEL*/ case 83 : Del(column,row,Ad_Cu,0);
                if(Num_St!=Ad_St) Num_St--;
                gt[Ad_St]=0;
                for(loop1=Ad_St+1;loop1<=num;loop1++){
                    if(gt[loop1]!=0){
                        gt[loop1-1]=gt[loop1];
                        gt[loop1]=0;
                        thaixy(5+(Ad_St+(1*(loop1-1-Ad_St))+column),
                                row,79,&gt[loop1-1],0);
                        Del(column,row,Ad_Cu,8*(loop1-Ad_St));
                    }
                }
                break;

    default : break;
    }
}

while(cktp!=0);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/* ฟังก์ชัน Del เป็นการลบตัวอักษรตำแหน่งที่เรากำหนด ทำงานร่วมกับฟังก์ชัน Check_Key
   ในกรณีที่เกิดคีย์ DELETE และ BACKSPACE */
```

```
Del(int colum,int row,int Ad_Cu,int Ba_De)
```

```
{
    void far *bp;
    Ad_Cu=Ad_Cu-8;
    bp = farmalloc(imagesize(628,309,636,321));
    getimage(628,309,636,321,bp);
    putimage(40+Ad_Cu+Ba_De+(colum*8),2+((row-1)*14),bp,3);
    farfree(bp);
}
```

```
/* ฟังก์ชัน Cursor เป็นการแสดงผลเคอร์เซอร์เหมือนคอส โดยสามารถกำหนดตำแหน่งการ
   แสดงได้ที่หน้าจอภาพ */
```

```
Cursor(int colum,int row,int Ad_Cu,int ck,int dl)
```

```
{
    void *Inverse1,*Inverse2;
    dl=dl+200;
    Inverse1 = farmalloc(imagesize(32+Ad_Cu+(colum*8),(2+(row-1)*14),
        39+Ad_Cu+(colum*8),14+((row-1)*14)));
    getimage(32+Ad_Cu+(colum*8),(2+(row-1)*14),39+Ad_Cu+(colum*8),
        14+((row-1)*14),Inverse1);
    do{
        putimage(32+Ad_Cu+(colum*8),(2+(row-1)*14),Inverse1,0);
        delay(dl);
        Inverse2 = farmalloc(imagesize(32+Ad_Cu+(colum*8),13+((row-1)*14),
            39+Ad_Cu+(colum*8),13+ck+((row-1)*14)));
        getimage(32+Ad_Cu+(colum*8),13+((row-1)*14),39+Ad_Cu+(colum*8),
            13+ck+((row-1)*14),Inverse2);
```

```
putimage(32+Ad_Cu+(colum*8),13+((row-1)*14),Inverse2,1);
```

```

farfree(Inverse2);
Inverse2 = farmalloc(imagesize(32+Ad_Cu+(column*8),13+((row-1)*14),
                               39+Ad_Cu+(column*8),13+ck+((row-1)*14)));
getimage(32+Ad_Cu+(column*8),13+((row-1)*14),39+Ad_Cu+(column*8),
         13+ck+((row-1)*14),Inverse2);
putimage(32+Ad_Cu+(column*8),13+((row-1)*14),Inverse2,4);
delay(dl);
farfree(Inverse2);

```

```

}
while(kbhit()==0);
putimage(32+Ad_Cu+(column*8),(2+(row-1)*14),Inverse1,0);
farfree(Inverse1);
}

```

/* ฟังก์ชัน hotkey เป็นการตรวจสอบคีย์ที่ต้องการเพียงคีย์เดียว ถ้าไม่ใช้ก็จะรอรับจนกว่าจะ
มีการกดคีย์ที่ต้องการ */

```

void hotkey(char key)
{
    char temp;
    do
    {
        temp=tolower(getch());
        if(temp==0)
            temp=getch();
        if(temp!=key){printf("\a");error(2);}
    }
    while(temp!=key);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

/* ฟังก์ชัน Gkey มีลักษณะเหมือนกับฟังก์ชัน hotkey ต่างกันที่ ฟังก์ชัน Gkey จะเลือกรับคีย์ที่ต้องการ 1 คีย์แล้ว จะรับคีย์ F1, F10, ALT-D, ALT-X ด้วย */

```
void Gkey(void)
{
    char key;
    do{
        key=getch();
        if(key==32) key=13;
        if(key==0) {
            key=getch();
            switch(key){
                /*F1*/ case 59 : help(1);
                        break;
                /*F10*/ case 68 : Menu();
                        break;
                /*ALT-D*/case 32 : osdos();
                        break;
                /*ALT-X*/ case 45 : closegraph();
                        exit(0);
            }
        }
    }
    while((key==59)!!(key==32)!!(key==68));
}
```

/* ฟังก์ชัน hot_key เป็นการตรวจสอบคีย์ที่ต้องการเพียงคีย์เดียว ถ้าไม่ใช่ก็จะรอรับจนกว่าจะมีการกดคีย์ที่ต้องการ และสามารถแสดงเคอร์เซอร์ได้ด้วย */

```
void hot_key(int col,int row,int dy,int ok,char key)
```

```
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
do{
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(ok==0) Cursor(col,row,0,0,dy);
st[0]=tolower(getch());
if(st[0]==0)
    st[0]=getch();
if(st[0]!=key){ printf("\a");error(2);}
}
while(st[0]!=key);
}

/* ฟังก์ชัน Check_t เป็นการเลือกว่าทำงานในบทเรียนใดซึ่งมีบทที่ 1-7 */
Check_t(char *st,int t,int tp)
{
switch(tp){
case 1 : Check_1(&st[0],t);
        break;
case 2 : Check_2(&st[0],t);
        break;
case 3 : Check_3(&st[0],t);
        break;
case 4 : Check_4(&st[0],t);
        break;
case 5 : Check_5(&st[0],t);
        break;
case 6 : Check_6(&st[0],t);
        break;
case 7 : Check_7(&st[0],t);
        break;
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

/* ฟังก์ชัน clearb เป็นการเคลียร์บิตเฟลอร์ก่อนที่จะทำการรับคีย์ใด ๆ เข้ามาเพื่อป้องกัน
ความผิดพลาด */

```
clearb(void)
```

```
{
```

```
    int Cb;
```

```
    Cb=peek(0x40,0x1a);
```

```
    poke(0x40,0x1c,Cb);
```

```
}
```

/* ฟังก์ชัน setint เป็นการเรียกใช้ int 8 เพื่อให้มีการเน้นข้อความตามจังหวะของสัญญาณ
นาฬิกา */

```
setint(int Sel)
```

```
{
```

```
    Select=Sel;
```

```
    oldfunc=getvect(8);
```

```
    setvect(8,int8);
```

```
}
```

/* ฟังก์ชัน restore1 เป็นการยกเลิกการ int 8 */

```
restore1()
```

```
{
```

```
    setvect(8,oldfunc);
```

```
}
```

/* ฟังก์ชัน interrupt int8 เป็นฟังก์ชันที่ถูกฟังก์ชัน setint เรียกใช้ เพื่อให้เห็นผลสรุป
กระพริบ ตามตำแหน่งที่กำหนดไว้ */

```
void interrupt int8(void)
```

```
{
```

```
    static int c=0;
```

```
    char asd1[1],asd2[1],asd3[1],asd4[1];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

asd1[1]=asd2[1]=asd3[1]=asd4[1]=0;
asd1[0]=30;asd2[0]=31;asd3[0]=17;asd4[0]=16;
if (c<3) {
    setcolor(12);
    switch(Select) {
        case 1 : outtextxy (8*21,17*14+5,&asd3[0]);
                outtextxy (8*21,18*14+5,&asd3[0]);
                break;

        case 2 : outtextxy (8*19,20*14+5,&asd3[0]);
                break;

        case 3 : setcolor(13);
                outtextxy (8*77,20*14+5,&asd3[0]);
                outtextxy (0,20*14+5,&asd4[0]);
                break;

        case 4 : outtextxy (4,21*14+5,&asd2[0]);
                outtextxy (15,21*14+5,&asd2[0]);
                break;

        case 5 : outtextxy (0,21*14+5,&asd2[0]);
                outtextxy (8,21*14+5,&asd2[0]);
                outtextxy (16,21*14+5,&asd2[0]);
                setcolor(14);
                outtextxy (0,7*14+5,&asd4[0]);
                outtextxy (0,10*14+5,&asd4[0]);
                outtextxy (8*33,7*14+5,&asd3[0]);
                outtextxy (8*33,10*14+5,&asd3[0]);
                break;
    }
}

```

else

```
switch(select){
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    case 1 : clearblock(21,18,24,21);
              break;

    case 2 : clearblock(20,21,21,22);
              break;

    case 3 : clearblock(1,21,2,22);
              clearblock(77,21,79,22);
              break;

    case 4 : clearblock(1,22,4,23);
              break;

    case 5 : clearblock(1,22,5,23);
              clearblock(1,8,2,12);
              clearblock(33,8,36,12);
              break;
        }
    c++;
    if (c>6)
        c=0;
    (*oldfunc)();
}

```

/* ฟังก์ชัน osdos เป็นการออกสู่ดอสชั่วคราว เมื่อต้องการใช้คำสั่งเกี่ยวกับดอส ถ้าต้องการ
ใช้งานโปรแกรมต่อก็พิมพ์ EXIT */

```
void osdos()
```

```
{
```

```
    int grdriver=9, grmode=1;
```

```
    unsigned long size1,size2;
```

```
    char far *buf1,*buf2;
```

```
    size1=imagesize(0,0,639,175);
```

```
    size2=imagesize(0,176,639,349);
```

```
    if((buf1=farmalloc(size1))==NULL)
```

```
        {error(1);return;}

```

```

if((buf2=farmalloc(size2))==NULL)
    {error(1);return;}
getimage(0,0,639,175,buf1);
getimage(0,176,639,349,buf2);
if(gint==0) {restore1();Cint(inter);}
    closegraph();
printf("type EXIT to return to dos training program");
system("command");
initgraph(&grdriver,&grmode,"");
putimage(0,0,buf1,0);
putimage(0,176,buf2,0);
if(gint==0) setint(inter);
farfree(buf1);
farfree(buf2);
}

```



```
*****
***** PROGRAM MAINMENU.C *****
*****
```

```
#include<alloc.h>
#include<graphics.h>
void Mainmenu(void);
void howuse(void);
extern int to,res,con,bear,pagedisplay;

/* ฟังก์ชัน Mainmenu เป็นการแสดงผลเมนูเพื่อเลือกการเรียนในบทเรียนที่ต้องการ หรือต้อง
การคำอธิบายเกี่ยวกับการใช้งานโปรแกรม โดยสามารถใช้ Arrow คีย์และ คีย์ HOME,
คีย์ END ได้ */
void Mainmenu(void)
{
    void far *gcolor;
    char gkey;
    setcolor(7);
    rectangle(18,19,621,321);
    rectangle(30,26,610,314);
    setfillstyle(9,7);
    floodfill(25,25,7);
    setcolor(7);
    rectangle(30,26,610,314);
    setfillstyle(1,7);
    floodfill(35,35,7);
    setcolor(15);
    rectangle(30,26,610,314);
    line(35,29,605,29);
    line(35,29,35,310);
```

```

line(30,314,35,310);
line(610,26,605,29);
setfillstyle(1,15);
floodfill(32,27,15);
setcolor(8);
rectangle(30,26,610,314);
line(35,310,605,310);
line(605,29,605,310);
line(30,314,35,310);
line(610,26,605,29);
setfillstyle(1,8);
floodfill(607,311,8);
setcolor(15);
line(30,26,610,26);
line(30,26,30,314);
Bottom_A(1,31,-15);Bottom_A(1,31,15);Bottom_A(1,0,0);
Bottom_A(1,59,0);Arr();Bottom(1,120,0);Bottom(1,0,0);
setcolor(0);
rectangle(172,90,455,65);
setfillstyle(1,0);
floodfill(296,78,0);
thaixypix(24,6,79,"โปรแกรมชาวสอนคอส",2,5,0);
setcolor(15);
rectangle(302,70,305,72);
setfillstyle(1,15);
floodfill(303,71,15);
setcolor(8);
rectangle(172,65,455,90);
rectangle(164,60,463,95);
setfillstyle(1,8);

```

```

floodfill(170,64,8);
setcolor(15);
line(463,60,455,65);
line(172,65,164,60);
line(164,60,463,60);
line(172,65,455,65);
line(455,90,463,95);
line(172,90,164,95);
line(164,95,463,95);
line(172,90,455,90);
setfillstyle(1,15);
floodfill(170,93,15);
floodfill(170,63,15);
thaixypix(63,20,79," ENTER ",1,11,0);
thaixypix(12,20,79," เลือ ก ",1,4,0);
thaixypix(34,20,79," HOME ",1,8,0);
thaixypix(43,20,79," ทวี ร ",1,4,0);
thaixypix(49,20,79," END ",1,8,0);
thaixypix(15,9,79," คำอธิบายการใช้งานโปรแกรม ",1,4,0) ;
thaixypix(15,11,79," คอมพิวเตอร์และดอส ",1,1,0) ;
thaixypix(15,13,79," ดิสค์และไดเรกทอรี ",1,1,0) ;
thaixypix(15,15,79," การจัดการเกี่ยวกับไฟล์ ",1,1,0) ;
thaixypix(15,17,79," 5 คำสั่งพิเศษสำหรับจัดการไฟล์ ",1,1,0) ;
thaixypix(51,9,79," การจัดการเกี่ยวกับดิสค์ ",1,1,0) ;
thaixypix(51,11,79," การป้องกันไฟล์และข้อมูล ",1,1,0) ;
thaixypix(51,13,79," ดอสคีย์และคำสั่ง ",1,1,0) ;
thaixypix(51,15,79," กลับสู่ดอส ",1,0,0) ;
Set_Color(10);
thaixypix(10,9,79," * ",1,10,0);
thaixypix(10,11,79," 1 ",1,10,0);

```

```

thaixypix(10,13,79," 2 ",1,10,0);
thaixypix(10,15,79," 3 ",1,10,0);
thaixypix(10,17,79," 4 ",1,10,0);
thaixypix(46,9,79," 5 ",1,10,0);
thaixypix(46,11,79," 6 ",1,10,0);
thaixypix(46,13,79," 7 ",1,10,0) ;
thaixypix(45,15,79," ** ",1,10,0) ;
to=1;
do{
    res=0;con=27;bear=1;
    gcolor = farmalloc(imagesize(60,100,580,250));
    getimage(60,100,580,250,gcolor);
    do{
        putimage(60,100,gcolor,0);
        switch(to){
            case 1 : thaixypix(10,9,79," * ",1,4,0);
                    thaixy(15,9,79," คำอธิบายการใช้งานโปรแกรม ",0) ;
                    break;
            case 2 : thaixypix(10,11,79," 1 ",1,4,0);
                    thaixy(15,11,79," คอมพิวเตอร์และคอส ",0) ;
                    break;
            case 3 : thaixypix(10,13,79," 2 ",1,4,0);
                    thaixy(15,13,79," ดิสต์และไดเรกทอรี ",0) ;
                    break;
            case 4 : thaixypix(10,15,79," 3 ",1,4,0);
                    thaixy(15,15,79," การจัดการเกี่ยวกับไฟล์ ",0) ;
                    break;
            case 5 : thaixypix(10,17,79," 4 ",1,4,0);
                    thaixy(15,17,79," 5 คำสั่งพิเศษสำหรับจัดการไฟล์ ",0) ;
                    break;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

case 6 : thaixypix(46,9,79," 5 ",1,4,0);
        thaixy(51,9,79," การจัดการเกี่ยวกับคิสต์ ",0) ;
        break;

case 7 : thaixypix(46,11,79," 6 ",1,4,0);
        thaixy(51,11,79," การป้องกันไฟล์และข้อมูล ",0) ;
        break;

case 8 : thaixypix(46,13,79," 7 ",1,4,0) ;
        thaixy(51,13,79," คอส์คีย์และคำสั่ง ",0) ;
        break;

case 9 : thaixypix(45,15,79," ** ",1,4,0) ;
        thaixy(51,15,79," กลับสู่คอส ",0) ;
        break;
}
do{
gkey=getch();
if(gkey==0)
gkey=getch();
switch(gkey){
case 71 : Bottom(0,0,0);delay(400);Bottom(1,0,0);
        if(to==1) gkey=1;
        else to=1;
        break;

case 79 : Bottom(0,120,0);delay(400);Bottom(1,120,0);
        if(to==9) gkey=1;
        else to=9;
        break;

case 75 : Bottom_A(0,0,0);delay(400);Bottom_A(1,0,0);
        if(to==5) gkey=1;
        if(to>5) to-=5;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        break;
    case 77 : Bottom_A(0,59,0);delay(400);Bottom_A(1,59,0);
        if(to==5) gkey=1;
        if(to<5) to+=5;
            else gkey=1;
        break;
    }
}

while((gkey!=72)&&(gkey!=80)&&(gkey!=13)&&(gkey!=75)&&
    (gkey!=77)&&(gkey!=71)&&(gkey!=79));
switch(gkey){
    case 72 : to--;
        Bottom_A(0,31,-15);delay(400);
        Bottom_A(1,31,-15);
        if(to==0) to=9;
        break;
    case 80 : to++;
        if(to==10) to=1;
        Bottom_A(0,31,15);delay(400);
        Bottom_A(1,31,15);
        break;
    }
}

while(gkey!=13);
putimage(60,100,gcolor,0);
farfree(gcolor);
if(to==1) howuse();
if((to==2)&&(res!=1)) menuto1();
if((to==3)&&(res!=1)) menuto2();
if((to==4)&&(res!=1)) menuto3();

```

```

    if((to==5)&&(res!=1)) menuto4();
    if((to==6)&&(res!=1)) menuto5();
    if((to==7)&&(res!=1)) menuto6();
    if((to==8)&&(res!=1)) menuto7();
    if(to==9) {closegraph();exit(0);}
}
while(to!=9);
}

```

/* ฟังก์ชัน Bottom เป็นการแสดงปุ่ม HOME และ END ในลักษณะ 3 มิติ */

```

Bottom(int pum,int x,int y)
{
    int p1,p2;
    if(pum==1) { p1=15;p2=8;}
    else { p1=8;p2=15;}
    setcolor(p1);
    rectangle(260+x,265+y,317+x,283+y);
    rectangle(263+x,267+y,314+x,281+y);
    setfillstyle(1,p1);
    floodfill(261+x,266+y,p1);
    setcolor(p2);
    rectangle(260+x,265+y,317+x,283+y);
    rectangle(263+x,267+y,314+x,281+y);
    line(317+x,265+y,314+x,267+y);
    line(263+x,281+y,260+x,283+y);
    setfillstyle(1,p2);
    floodfill(316+x,282+y,p2);
    setcolor(p1);
    line(260+x,265+y,317+x,265+y);
    line(263+x,267+y,314+x,267+y);

```

```

line(263+x,267+y,263+x,281+y);
line(260+x,265+y,260+x,283+y);
line(317+x,265+y,314+x,267+y);
line(263+x,281+y,260+x,283+y);
}

/* ฟังก์ชัน Bottom_A เป็นการแสดงปุ่ม Arrow ในลักษณะ 3 มิติ */
Bottom_A(int pum,int x,int y)
{
    int p1,p2;
    if(pum==1) {p1=15,p2=8;}
        else {p1=8;p2=15;}
    setcolor(p1);
    rectangle(159+x,267+y,180+x,280+y);
    rectangle(161+x,269+y,178+x,278+y);
    setfillstyle(1,p1);
    floodfill(160+x,268+y,p1);
    setcolor(p2);
    line(180+x,267+y,178+x,269+y);
    line(161+x,278+y,159+x,280+y);
    line(180+x,267+y,180+x,280+y);
    line(178+x,278+y,178+x,269+y);
    line(159+x,280+y,180+x,280+y);
    line(178+x,278+y,161+x,278+y);
    setfillstyle(1,p2);
    floodfill(179+x,279+y,p2);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/* ฟังก์ชัน Arr เป็นการแสดงลูกศรทั้ง 4 ทิศทางในตำแหน่งที่กำหนด */
```

```
Arr(void)
```

```
{
```

```
char gkey,Asc1[1],Asc2[1],Asc3[1],Asc4[1];
```

```
Asc1[1]=Asc2[1]=Asc3[1]=Asc4[1]=0;
```

```
Asc1[0]=27;Asc2[0]=26,Asc3[0]=24;Asc4[0]=25;
```

```
setcolor(0);
```

```
outtextxy(166,271,&Asc1[0]);
```

```
outtextxy(196,256,&Asc3[0]);
```

```
outtextxy(196,286,&Asc4[0]);
```

```
outtextxy(226,271,&Asc2[0]);
```

```
}
```

```
/* ฟังก์ชัน howuse เป็นการแสดงคำอธิบายการใช้งานโปรแกรม */
```

```
void howuse(void)
```

```
{
```

```
char ch;
```

```
pagedisplay=1;
```

```
setactivepage(1);
```

```
setviewport(0,0,639,349,1);
```

```
clearviewport();
```

```
thaixypix(10,2,79,"DOS TRAINING PROGRAM",2,14,0);
```

```
thaixypix(14,4,79,"Copyright (c) 1995 โดย สาขาอิเล็กทรอนิกส์และคอมพิวเตอร์",0,13,0);
```

```
thaixypix(10,5,79,"คณะครูสาส์ตรู้ดสาทรกรรม สถาบันเทคโนโลยีพระจอมเกล้าเจ้า/  
คุณทหารลาดกระบัง",0,13,0);
```

```
thaixypix(6,6,79,"_____/"
```

```
_____",0,14,0);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

thaixypix(15,8,79,"โปรแกรมนี้เป็นโปรแกรมช่วยสอนดอส โดยใช้คำสั่งใน DOS 5 /
เป็นเกมที่อ้างอิง",0,14,0);
thaixypix(6,9,79,"ในโปรแกรมจะมีบทเรียนให้ผู้เรียนได้เรียน โดยเรียงลำดับความ/
ยากง่ายไว้อย่างเหมาะสม",0,14,0);
thaixypix(7,10,79,"เหมาะสม โดยภาษาในบทเรียนจะมีการบอกให้กดคีย์ตามที่กำหนด/
เช่น",0,14,0);
type1(1,12,"Type"," dir");
thaixypix(20,12,79,"ให้พิมพ์คำว่า dir",0,14,0);
type1(1,14,"Press"," Enter");
thaixypix(20,14,79,"ให้กดคีย์ ENTER",0,14,0);
type1(1,16,"Press"," Enter");
thaixypix(20,16,79,"ให้กดคีย์ลูกศรขึ้น 4 ครั้ง",0,14,0);
type1(1,18,"Press"," ALT+X");
thaixypix(20,18,79,"ให้กดคีย์ ALT และคีย์ X พร้อมๆกัน",0,14,0);
type1(1,20,"Press"," ALT+X");
thaixypix(20,20,79,"ในการใช้งานของโปรแกรมนี้นี้มีคีย์ ต่างๆที่ใช้โดยมีรายละเอียด/
ดังนี้",0,14,0);
thaixypix(40,22,79,"F1 เพื่อแสดงและอธิบายคำสั่งต่างๆของ DOS",0,14,0);
thaixypix(6,22,79,"F10 กลับไปเมนูหลัก",0,14,0);
thaixypix(6,23,79,"ALT+D ออกไปสู่คอสี่ตัวคราา",0,14,0);
thaixypix(40,23,79,"ALT+X ออกจากโปรแกรม",0,14,0);
rectangle(200,330,400,349);
setfillstyle(1,4);
floodfill(201,331,4);
thaixypix(30,25,79,"กด Q กลับไปเมนูหลัก",0,14,2);
setvisualpage(1);
do{
    ch=getch();
    ch=tolower(ch);

```

```
pagedisplay=0;  
setactivepage(0);  
setvisualpage(0);
```

```
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

*****
***** PROGRAM MAIN.C *****
*****

```

```

#include<dos.h>
#include<graphics.h>
#include<stdio.h>
#include<alloc.h>
#include<io.h>
#include<fcntl.h>
#include<stdarg.h>
#include<stdlib.h>
#include<conio.h>

extern char scan,exten;
extern char Blod;
extern char Tab;
static int presstab=0;
unsigned int on;
extern unsigned statuskey;
extern int count;
extern char OldMode;
extern unsigned int Off_Line[350];
extern unsigned int G_Row;
extern unsigned int G_Col;
extern void barband();
extern void help(int page);
extern void error(int status);
extern void topic_3();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ฌ่แม้ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

extern void Check_2(char *g,int topic);
extern void topic_2();
extern void topic_1();
extern void Check_1(char *g,int topic);

int Num_St,Ad_Cu,Ad_St,cktp,Select;
char gt[30],st[30];
int to=1,res,con,bear=1,inter,gint=1;
void far *gscreen;

void Check_Key(int num,int colum,int row,int topic,int ok,int tp);
typedef unsigned char byte;
void thaixypix(int colum,int row,int colend,char *content,int status,
               int color,int level);
void hot_key(int colum,int row,int delay,int ok,char key);
void interrupt int8(void);
void interrupt (*oldfunc)();
void hotkey(char key);
void Gkey(void);
void osdos();
void Cint(int select);
void interrupt int8(void);
void interrupt (*oldfunc)();

/* ฟังก์ชัน main เป็นจุดเริ่มต้นทำงานของโปรแกรม โดยจะเรียกใช้การแสดงผลเมนูเพื่อเลือก
   การเรียนในบทเรียนต่าง ๆ หรือต้องการคำอธิบายวิธีใช้งานโปรแกรม */
void main()
{
    int index;
    opengraph();

```

```

Mainmenu();
closegraph();
}

/* ฟังก์ชัน Check_Key เป็นการตรวจสอบคีย์ที่ถูกต้องว่าเป็นคีย์อะไร โดยมีการแสดงผลการกด
คีย์ทางจอภาพ ความคมการทำงานของคีย์พิเศษ เช่น HOME, END, DELETE, INSERT */
void Check_Key(int num,int colum,int row,int topic,int ok,int tp)
{
char Swap[2],ch[1];
int loop1,loop2,gnum,ck,clearB;
unsigned long size1,size2;
char far *buf1,*buf2;

Num_St=Ad_Cu=Ad_St=0;Swap[2]=0;
for(gnum=0;gnum<=30;gnum++)
gt[gnum]=0;
for(gnum=0;gnum<=30;gnum++)
st[gnum]=0;
if(ok==0) thaixy(colum+1,row,79,"C:\\>",0);
if(ok==2) thaixy(colum+1,row,79,"A:\\>",0);
ck=0;cktp=1;clearB=0;
do{
ch[0]=0;ch[1]=0;
clearB++;
if(clearB==1) clearb();
Cursor(colum,row,Ad_Cu,ck,100);
ch[0]=getch();
switch(ch[0]){

```

```

Ad_St--; Num_St--;
gt[Ad_St]=0;
Del(column,row,Ad_Cu,0);
for(loop1=Ad_St+1;loop1<=num;loop1++){
    if(gt[loop1]!=0){
        gt[loop1-1]=gt[loop1];
        gt[0]=gt[1];
        gt[loop1]=0;
        thaixy(5+(Ad_St+(1*(loop1-1-Ad_St))+column),
            row,79,&gt[loop1-1],0);
        Del(column,row,Ad_Cu,8*(loop1-Ad_St));
    }
}
break;
case 13 : Check_t(&gt[0],topic,tp);
    if(cktp==1) {
        printf("\a");
        error(2);
        Ad_Cu=0;Ad_St=0;
    }
}
break;
case 0 : break;
case 27 : break;
default : Del(column,row,Ad_Cu,0);
    if(Ad_St<num)
        thaixy(5+(1*Ad_St)+column,row,79,&ch[0],0);
    Swap[0]=ch[0];
    if(Ad_St<num){

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปดเปลี่ยนแปลง และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Swap[1]=gt[Ad_St];
gt[Ad_St]=Swap[0];
Swap[0]=Swap[1];
for(loop1=Ad_St;loop1<=num;loop1++) {
    Del(colum,row,Ad_Cu,8*(loop1+1-Ad_St));
    if(loop1<num-1)
        thaixy(6+(Ad_St+(1*(loop1-Ad_St))+colum),
            row,79,&Swap[1],0);
    if(loop1!=num) {
        Swap[1]=gt[loop1+1];
        gt[loop1+1]=Swap[0];
        Swap[0]=Swap[1];
        gt[num]=0;
    }
    if(Num_St!=num) Num_St++;
}
Ad_Cu+=8;
if(ck==0) gt[Ad_St]=ch[0];
if(Ad_St<=num) {Ad_St++;
if(Num_St!=num) Num_St++;}
if(Ad_St>=num){ Check_t(&gt[0],topic,tp);
if(cktp==1)
    {printf("\a");error(2);Ad_Cu=0;Ad_St=0;}}
break;
}
if(ch[0]==0) {

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น switch(ch[0]){ปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/*F10*/ case 68 : Menu();
            break;

/*F1*/ case 59 : if(gint==0) {restore1();Cint(inter);}
            help(on);
            break;

/*HOME*/ case 71 : Ad_Cu=0;Ad_St=0;
            break;

/*ALT-D*/ case 32 : osdos();
            break;

/*ALT-X*/ case 45 : closegraph();
            exit(0);

/*LEFT*/ case 75 : if(Ad_St==0) break;
            Ad_Cu-=8;
            Ad_St--;
            break;

/*RIGHT*/ case 77 : if(Ad_St==Num_St) break;
            Swap[1]=0;
            for(loop1=Ad_St;loop1<=num;loop1++)
            if((gt[loop1]==32)|| (gt[loop1]==0)) Swap[1]++;
            if(Swap[1]==(num-Ad_St+1)) {Num_St=Ad_St; break;}
            Ad_Cu+=8;
            Ad_St++;
            break;

/*END*/ case 79 : for(loop2=0;loop2<=num;loop2++)
            if((gt[loop2]==32)|| (gt[loop2]==0))
            {
                Swap[1]=0;
                for(loop1=loop2;loop1<=num;loop1++){
                    if((gt[loop1]==32)|| (gt[loop1]==0))

```

```

        if(Swap[1]==(num-loop2+1)){
            Num_St=loop2;
            break;
        }
    }
    Ad_Cu=8*Num_St; Ad_St=Num_St;
    break;
/*INS*/ case 82 : if(ck==0) ck=1;
                else ck=0;
                break;
/*DEL*/ case 83 : Del(column,row,Ad_Cu,0);
                if(Num_St!=Ad_St) Num_St--;
                gt[Ad_St]=0;
                for(loop1=Ad_St+1;loop1<=num;loop1++){
                    if(gt[loop1]!=0){
                        gt[loop1-1]=gt[loop1];
                        gt[loop1]=0;
                        thaixy(5+(Ad_St+(1*(loop1-1-Ad_St))+column),
                                row,79,&gt[loop1-1],0);
                        Del(column,row,Ad_Cu,8*(loop1-Ad_St));
                    }
                }
                break;
    default : break;
    }
}
}
while(cktp!=0);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/* ฟังก์ชัน Del เป็นการลบตัวอักษรตำแหน่งที่เรากำหนด ทำงานร่วมกับฟังก์ชัน Check_Key
   ในกรณีที่กดคีย์ DELETE และ BACKSPACE */
```

```
Del(int colum,int row,int Ad_Cu,int Ba_De)
```

```
{
    void far *bp;
    Ad_Cu=Ad_Cu-8;
    bp = farmalloc(imagesize(628,309,636,321));
    getimage(628,309,636,321,bp);
    putimage(40+Ad_Cu+Ba_De+(colum*8),2+((row-1)*14),bp,3);
    farfree(bp);
}
```

```
/* ฟังก์ชัน Cursor เป็นการแสดงผลเคอร์เซอร์เหมือนคอส โดยสามารถกำหนดตำแหน่งการ
   แสดงได้ที่หน้าจอกภาพ */
```

```
Cursor(int colum,int row,int Ad_Cu,int ck,int dl)
```

```
{
    void *Inverse1,*Inverse2;
    dl=dl+200;
    Inverse1 = farmalloc(imagesize(32+Ad_Cu+(colum*8),(2+(row-1)*14),
        39+Ad_Cu+(colum*8),14+((row-1)*14)));
    getimage(32+Ad_Cu+(colum*8),(2+(row-1)*14),39+Ad_Cu+(colum*8),
        14+((row-1)*14),Inverse1);
    do{
        putimage(32+Ad_Cu+(colum*8),(2+(row-1)*14),Inverse1,0);
        delay(dl);
        Inverse2 = farmalloc(imagesize(32+Ad_Cu+(colum*8),13+((row-1)*14),
            39+Ad_Cu+(colum*8),13+ck+((row-1)*14)));
        getimage(32+Ad_Cu+(colum*8),13+((row-1)*14),39+Ad_Cu+(colum*8),
            13+ck+((row-1)*14),Inverse2);
    }
```

```
putimage(32+Ad_Cu+(colum*8),13+((row-1)*14),Inverse2,1);
```

```

farfree(Inverse2);
Inverse2 = farmalloc(imagesize(32+Ad_Cu+(column*8),13+((row-1)*14),
                               39+Ad_Cu+(column*8),13+ck+((row-1)*14)));
getImage(32+Ad_Cu+(column*8),13+((row-1)*14),39+Ad_Cu+(column*8),
         13+ck+((row-1)*14),Inverse2);
putimage(32+Ad_Cu+(column*8),13+((row-1)*14),Inverse2,4);
delay(dl);
farfree(Inverse2);
}
while(kbhit()==0);
putimage(32+Ad_Cu+(column*8),(2+(row-1)*14),Inverse1,0);
farfree(Inverse1);
}
/* ฟังก์ชัน hotkey เป็นการตรวจสอบคีย์ที่ต้องการเพียงคีย์เดียว ถ้าไม่ใช่ก็จะรอรับจนกว่าจะ
มีการกดคีย์ที่ต้องการ */
void hotkey(char key)
{
char temp;
do
{
temp=tolower(getch());
if(temp==0)
temp=getch();
if(temp!=key){printf("\a");error(2);}
}
while(temp!=key);
}

```

/* ฟังก์ชัน Gkey มีลักษณะเหมือนกับฟังก์ชัน hotkey ต่างกันที่ ฟังก์ชัน Gkey จะเลือกรับคีย์ที่ต้องการ 1 คีย์แล้ว จะรับคีย์ F1, F10, ALT-D, ALT-X ด้วย */

```
void Gkey(void)
```

```
{
```

```
    char key;
```

```
    do{
```

```
        key=getch();
```

```
        if(key==32) key=13;
```

```
        if(key==0) {
```

```
            key=getch();
```

```
            switch(key){
```

```
                /*F1*/ case 59 : help(1);
```

```
                    break;
```

```
                /*F10*/ case 68 : Menu();
```

```
                    break;
```

```
                /*ALT-D*/case 32 : osdos();
```

```
                    break;
```

```
                /*ALT-X*/ case 45 : closegraph();
```

```
                    exit(0);
```

```
            }
```

```
        }
```

```
    }while((key==59)!!(key==32)!!(key==68));
```

```
}
```

/* ฟังก์ชัน hot_key เป็นการตรวจสอบคีย์ที่ต้องการเพียงคีย์เดียว ถ้าไม่ใช่ก็จะรอรับจนกว่าจะมีการกดคีย์ที่ต้องการ และสามารถแสดงเคอร์เซอร์ได้ด้วย */

```
void hot_key(int col,int row,int dy,int ok,char key)
```

```
{
```

```
    do{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

asd1[1]=asd2[1]=asd3[1]=asd4[1]=0;
asd1[0]=30;asd2[0]=31;asd3[0]=17;asd4[0]=16;
if (c<3) {
    setcolor(12);
    switch(Select) {
        case 1 : outtextxy (8*21,17*14+5,&asd3[0]);
                outtextxy (8*21,18*14+5,&asd3[0]);
                break;
        case 2 : outtextxy (8*19,20*14+5,&asd3[0]);
                break;
        case 3 : setcolor(13);
                outtextxy (8*77,20*14+5,&asd3[0]);
                outtextxy (0,20*14+5,&asd4[0]);
                break;
        case 4 : outtextxy (4,21*14+5,&asd2[0]);
                outtextxy (15,21*14+5,&asd2[0]);
                break;
        case 5 : outtextxy (0,21*14+5,&asd2[0]);
                outtextxy (8,21*14+5,&asd2[0]);
                outtextxy (16,21*14+5,&asd2[0]);
                setcolor(14);
                outtextxy (0,7*14+5,&asd4[0]);
                outtextxy (0,10*14+5,&asd4[0]);
                outtextxy (8*33,7*14+5,&asd3[0]);
                outtextxy (8*33,10*14+5,&asd3[0]);
                break;
    }
}
else
switch(select){

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

case 1 : clearblock(21,18,24,21);
        break;
case 2 : clearblock(20,21,21,22);
        break;
case 3 : clearblock(1,21,2,22);
        clearblock(77,21,79,22);
        break;
case 4 : clearblock(1,22,4,23);
        break;
case 5 : clearblock(1,22,5,23);
        clearblock(1,8,2,12);
        clearblock(33,8,36,12);
        break;
}
c++;
if (c>6)
c=0;
(*oldfunc)();
}

```

/* ฟังก์ชัน osdos เป็นการออกสู่คอสมอสชั่วคราว เมื่อต้องการใช้คำสั่งเกี่ยวกับคอสมอส ถ้าต้องการ
ทำงานโปรแกรมต่อก็คพิมพ์ EXIT */

```
void osdos()
```

```
{
```

```
int grdriver=9, grmode=1;
```

```
unsigned long size1,size2;
```

```
char far *buf1,*buf2;
```

```
size1=imagesize(0,0,639,175);
```

```
size2=imagesize(0,176,639,349);
```

```
if((buf1=farmalloc(size1))==NULL)
```

```
{error(1);return;}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if((buf2=farmalloc(size2))==NULL)
    {error(1);return;}
getimage(0,0,639,175,buf1);
getimage(0,176,639,349,buf2);
if(gint==0) {restore1();Cint(inter);}
    closegraph();
printf("type EXIT to return to dos training program");
system("command");
initgraph(&grdriver,&grmode,"");
putimage(0,0,buf1,0);
putimage(0,176,buf2,0);
if(gint==0) setint(inter);
farfree(buf1);
farfree(buf2);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
*****
***** MENUTOP.C *****
*****
```

```
#include<alloc.h>
```

```
#include<graphics.h>
```

```
void menutop(void);
```

```
extern int res,con,to,bear;
```

```
extern void far *gscreen;
```

```
/* ฟังก์ชัน menutop เป็นการแสดงผลเมนูเพื่อเลือกหัวข้อการเรียนคอส ในแต่ละบทเรียนว่า
ต้องการที่จะเรียนในหัวข้อใดก่อน โดยในการเลือกสามารถใช้ Arrow คีย์หรือ กดตัวเลข
ของหัวข้อที่ต้องการได้ */
```

```
void menutop(void)
```

```
{
```

```
void far *gscr, far *gcolor, far *gp;
```

```
char gkey, Asc1[11], Asc2[11];
```

```
int color=1;
```

```
Asc1[11]=Asc2[11]=0;
```

```
Asc1[0]=24; Asc2[0]=25;
```

```
gscr = farmalloc(imagesize(125,62,513,260));
```

```
getimage(125,62,513,260,gscr);
```

```
clearblock(18,6,64,19);
```

```
Set_Color(10);
```

```
thaixy(22,11,24,"1.",0);
```

```
thaixy(22,13,24,"2.",0);
```

```
thaixy(22,15,24,"3.",0);
```

```
Set_Color(12);
```

```
thaixy(25,11,60,"เริ่มต้นการทำงานคอสและการทำหน้าจอให้ว่าง",0);
```

```
thaixy(25,13,60,"ตรวจสอบส่วนช่วยเหลืองและเวอร์ชัน",0);
```

```

thaixy(25,15,60,"ติดตั้ง วัน, เวลา, และพร้อมท์",0);
gp = farmalloc(imagesize(145,76,470,220));
getimage(145,76,470,220,gp);
setcolor(10);
rectangle(125,62,513,260);
setfillstyle(1,10);
floodfill(180,185,10);
setcolor(12);
rectangle(136,70,502,252);
rectangle(146,105,492,246);
setfillstyle(9,12);
floodfill(153,100,12);
setcolor(15);
rectangle(158,134,479,216);
setfillstyle(1,15);
floodfill(160,136,15);
Set_Color(10);
thaixy(32,9,79," คอมพิวเตอร์และคอส ",0) ;
Set_Color(7);
thaixypix(30,7,60," LESSON 1 ",2,1,1);
putimage(145,76,gp,1);
farfree(gp);
Set_Color(9);
thaixypix(30,7,60," LESSON 1 ",2,1,1);
thaixypix(21,17,60,"เลือก ",0,0,0);
thaixypix(36,17,60," หรือ ",0,0,0);
thaixypix(41,17,60," กดตัวเลข",0,9,0);
Set_Color(10);
thaixy(27,17,60," ",0);
Set_Color(14);

```

```

outtextxy(216,227,&Asc1[0]);
Set_Color(10);
thaixy(32,17,60," ",0);
Set_Color(14);
outtextxy(256,227,&Asc2[0]);
thaixypix(51,17,60," Esc ",0,12,0);
thaixypix(56,17,60," ออก ",0,0,0);

gcolor = farmalloc(imagesize(158,134,479,216));
getimage(158,134,479,216,gcolor);
do{
    putimage(158,134,gcolor,4);
    Set_Color(3);
    switch(color){
        case 1 :
            thaixy(25,11,60," เริ่มต้นการทำงานตอนสและการทำงานหน้าจอให้ว่าง",0);
            break;
        case 2 :
            thaixy(25,13,60," ตรวจสอบส่วนซ้ายเหลือและเวอร์ชัน",0);
            break;
        case 3 :
            thaixy(25,15,60," ติดตั้ง วัน, เวลา, และพร้อมท์",0);
            break;
    }
}
do{
    gkey=getch();
    if(gkey==49)
        {color=1; gkey=13;}
    if(gkey==50)
        {color=2; gkey=13;}
}

```

```

    if(gkey==51)
        {color=3; gkey=13;}
    if(gkey==0)
        gkey=getch();
}

while((gkey!=72)&&(gkey!=80)&&(gkey!=con)&&(gkey!=13));
switch(gkey){
case 72 : color--;
        if(color==0) color=3;
        break;
case 80 : color++;
        if(color==4) color=1;
        break;
case 27 : break;
case 13 : farfree(gcolor);
        farfree(gscr);
        if(bear==0)
            { clearblock(1,1,81,26);barband();
              farfree(gscreen);bear++;}
        switch(color){
            case 1 : topic_1(1);
                    break;
            case 2 : topic_1(2);
                    break;
            case 3 : topic_1(3);
                    break;
        }
        break;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
while((gkey!=con)&&(gkey!=13));  
if(gkey==27){  
    farfree(gcolor);  
    res=1;  
    putimage(125,62,gscr,0);  
    farfree(gscr);}  
else  
    {con=13;to++;}  
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/*****
***** THAIDRI.C *****
*****/

```

```

#include<graphics.h>
#include<stdio.h>
#include<alloc.h>
#include<io.h>
#include<fcntl.h>
#include<stdarg.h>
#include<conio.h>
#define Tiltkey 41
#define Esc 1

typedef unsigned char byte;
typedef byte Gen_Map[256][14];
byte scan,exten;
char Blod=0,Tab=0,Line=0;
int pagedisplay=0;
int count;

Gen_Map Bit_Normal;

unsigned int Off_Line[350],G_Row=0,G_Col=0;

void Set_Color(byte Colour);
void restore();
void opengraph();
void G_gotoxy(byte X,byte Y);
void G_write(byte *ThaiStr,byte Colend);
void G_writerow(byte *ThaiStr,byte Colend,int status);

```

```

void DispChar(byte ST,byte SP,byte *One_Char);
void DispCharRev(byte ST,byte SP,byte *One_Char);
void page(int col,int row,int rowstop,int colend);
void outthaixy(int colum,int row,int colend,char *content);
void clearblock(int x,int y,int endx,int endy);
unsigned char complement(unsigned char asc);
void WidthChar(byte Count,byte One_Byte);
void thaixy(int colum,int row,int colend,char *content,int status);
void DispCharrow(byte ST,byte SP,byte *One_Char,int status);
void G_writerow1(byte *ThaiStr, byte Colend,int status,int color, int
level);
unsigned char keyboard( unsigned char mode, unsigned int *statkey,
unsigned char *scan,unsigned char *exten);
void DispCharrow1(byte ST,byte SP,byte *One_Char,int status,int color
,int level);
void thaixypix(int colum,int row,int colend,char *content,int status,
int color,int level);

/* ฟังก์ชัน OpenFont() ทำหน้าที่เปิดไฟล์ฟอนต์ 8x14 จุด */
OpenFont(char *fi,byte *ThaiFont)
{
FILE *Fp;
if((Fp=fopen(fi,"rb"))==NULL)
{
printf("\n*****File %s not found *****",fi);
return -1;
}
fread(&ThaiFont[0],sizeof(Gen_Map),1,Fp);fclose(Fp);
return 0;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/* ทำหน้าที่เข้าสู่โหมดกราฟฟิกโหมด 10 ความละเอียด 640x350 จุด 16 สี*/
void opengraph()
{
    int Count;

    int grdriver=9,grmode=1;

    initgraph(&grdriver,&grmode,"");
    for(Count=0;Count<350;Count++)
    Off_Line[Count]=Count*80;
    OpenFont("8X14.FON",*Bit_Normal);
}

/* ฟังก์ชัน Set_Color() เป็นฟังก์ชันตั้งค่าสีโดยสามารถตั้งค่าได้ตามสีมาตรฐานของ EGA
ได้ 16 สี */
void Set_Color(byte Colour)
{
    outportb(0x03C4,2);
    outportb(0x03C5,Colour);
}

/* ฟังก์ชัน DispChar() ทำหน้าที่แสดงตัวอักษร 1 ตัวอักษร โดยถ้า Line=1 เป็นตัวขีดเส้น
ใต้ Blod=1 เป็นตัวหนา โดยใช้งานร่วมกับฟังก์ชัน outthaixy()*/
void DispChar(byte ST,byte SP,byte *One_Char)
{
    byte Count,line;

    if(Line)
    {
        Set_Color(14);

        for(Count=ST;Count<=SP;Count++)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if((Count==12)&&(ST!=12)) line=0xff;
else line=0x00;

if(pagedisplay==0) pokeb( 0xA000, Off_Line[G_Row+Count]+G_Col,
                           (One_Char[Count]);line);
else pokeb(0xA800,Off_Line[G_Row+Count]+G_Col,(One_Char[Count])
           ;line);
}
}
if(Line)returns;
for(Count=ST;Count<=SP;Count++)
{
if(Blod)
{
Set_Color(7);
if(pagedisplay==0) pokeb(0xA000,Off_Line[G_Row+Count]+G_Col,
                           (One_Char[Count]);!(One_Char[Count]>>1));
else pokeb(0xA800,Off_Line[G_Row+Count]+G_Col,(One_Char[Count])
           ;!(One_Char[Count]>>1));
}
else
{
Set_Color(14);
if(pagedisplay==0) pokeb(0xA000,Off_Line[G_Row+Count]+G_Col,
                           One_Char[Count]);
else pokeb(0xA800,Off_Line[G_Row+Count]+G_Col,One_Char[Count]);
}
}
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

/* ฟังก์ชัน DispCharRev() ทำหน้าที่แสดงตัวอักษร 1 ตัวอักษรโดยแสดงเป็นแบบ reverse*/

```
void DispCharRev(byte ST,byte SP,byte *One_Char)
```

```
{
```

```
    byte Count;
```

```
    for(Count=ST;Count<=SP;Count++)
```

```
    {
```

```
        if(Blod)
```

```
        {
```

```
            Set_Color(3);
```

```
            if(pagedisplay==0) pokeb(0xA000,Off_Line[G_Row+Count]+G_Col,
                complement((One_Char[Count])!(One_Char[Count]>>1)));
```

```
            else pokeb(0xA800, Off_Line[G_Row+Count]+G_Col, complement
                ((One_Char[Count])!(One_Char[Count]>>1)));
```

```
        }
```

```
    else
```

```
    {
```

```
        if(pagedisplay==0) pokeb(0xA000,Off_Line[G_Row+Count]+G_Col,
            complement(One_Char[Count]));
```

```
        else pokeb(0xA800,Off_Line[G_Row+Count]+G_Col, complement
            (One_Char[Count]));
```

```
    }
```

```
}
```

```
}
```

/* ฟังก์ชัน G_gotoxy ทำหน้าที่ย้ายตำแหน่งเคอร์เซอร์ไปตามตำแหน่งที่กำหนด */

```
void G_gotoxy(byte X,byte Y)
```

```
{
```

```
    if(X>80) X=80;
```

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยราชภัฏวไลยอลงกรณ์ จังหวัดปทุมธานี ใ้บริการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

G_Col=X-2;
G_Row=(Y-1)*14;
}

/* ฟังก์ชัน G_write() ทำหน้าที่จัดระดับตัวอักษรโดยมีอยู่ 3 ระดับคือ พัดธงนะ สระและ
วรรณยุกต์ */
void G_write(byte *ThaiStr,byte Colend)
{
    byte ST,SP;

    while(*ThaiStr != NULL)
    {
        if(*ThaiStr==126)ThaiStr+=2;
        if((*ThaiStr>=128)&&(*ThaiStr<=148)) return;
        if(*ThaiStr==5)
        switch(Line)
        {
            case 0:Line=1;
                break;
            case 1:Line=0;
                break;
            default:break;
        }
        if(*ThaiStr==2)
        switch(Blod)
        {
            case 0:Blod=1;
                break;
            case 1:Blod=0;
                break;
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับครูผู้สอนเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        default:break;
    }

if(*ThaiStr==3)
    switch(Tab){
        case 0:Tab=1;Blod=1;
            break;
        case 1:Tab=0;Blod=0;
            break;
        default:break;
    }
switch(*ThaiStr){
    case 143: *ThaiStr=153; break;
    case 144: *ThaiStr=156; break;
    case 145: *ThaiStr=150; break;
    case 146: *ThaiStr=154; break;
    case 147: *ThaiStr=152; break;
    case 149: *ThaiStr=158; break;
    case 150: *ThaiStr=159; break;
    case 152: *ThaiStr=149; break;
    case 153: *ThaiStr=151; break;
    case 154: *ThaiStr=155; break;
    case 155: *ThaiStr=157; break;
    default:break;
}

if(*ThaiStr>31)
    switch(*ThaiStr)
    {

```

```

        case 216:case 217:case 218:

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 { ST=12;SP=13;}break;
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    case 209:case 212:case 213:case 214:case 215:case 231:
        { ST=2;SP=4;}break;

    case 232:case 233:case 234:case 235:case 236:case 237:
        {ST=0;SP=2;}break;

    default:
        {ST=0;SP=13;G_Col++;}
}

if(G_Col>Colend) return;

if((Blod&&Tab)&&(*ThaiStr>31)) DispCharRev(ST,SP, Bit_Normal
                                     [*ThaiStr]);
else if(*ThaiStr>31)DispChar(ST,SP,Bit_Normal[*ThaiStr]);
*ThaiStr++;
}
}

/* ฟังก์ชัน outthaixy() เป็นฟังก์ชันแสดงผลตัวอักษร 1 บรรทัด ตามตำแหน่งที่กำหนด */
void outthaixy(int colum,int row,int colend,char *content)
{
    G_gotoxy(colum,row);
    G_write(content,colend);
}

/*ฟังก์ชัน G_writerow() ทำหน้าเหมือนกับฟังก์ชัน G_write() แต่จะใช้ในการแสดงผลของ
ฟังก์ชัน thaixy() */
void G_writerow(byte *ThaiStr,byte Colend,int status)
{
    byte ST,SP;
    while(*ThaiStr != NULL)

```

```

switch(*ThaiStr){
    case 143: *ThaiStr=153; break;
    case 144: *ThaiStr=156; break;
    case 145: *ThaiStr=150; break;
    case 146: *ThaiStr=154; break;
    case 147: *ThaiStr=152; break;
    case 149: *ThaiStr=158; break;
    case 150: *ThaiStr=159; break;
    case 152: *ThaiStr=149; break;
    case 153: *ThaiStr=151; break;
    case 154: *ThaiStr=155; break;
    case 155: *ThaiStr=157; break;
    default:break;
}
if(*ThaiStr>31)
switch(*ThaiStr)
{
    case 216:case 217:case 218:
        {ST=12;SP=13;}break;
    case 209:case 212:case 213:case 214:case 215:case 231:
        {ST=2;SP=4;}break;
    case 232:case 233:case 234:case 235:case 236:case 237:
        {ST=0;SP=2;}break;
    default:
        {ST=0;SP=13;G_Col++;}
}
if(G_Col>Colend) return;

```

```

if(*ThaiStr>31)DispCharrow(ST,SP,Bit_Normal[*ThaiStr],status);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

switch(*ThaiStr){
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

case 153: *ThaiStr=143; break;
case 156: *ThaiStr=144; break;
case 150: *ThaiStr=145; break;
case 154: *ThaiStr=146; break;
case 152: *ThaiStr=147; break;
case 158: *ThaiStr=149; break;
case 159: *ThaiStr=150; break;
case 149: *ThaiStr=152; break;
case 151: *ThaiStr=153; break;
case 155: *ThaiStr=154; break;
case 157: *ThaiStr=155; break;
default:break;
}

```

```
*ThaiStr++;
```

```
}
```

```
}
```

```

/* ฟังก์ชัน thaixy() เป็นฟังก์ชันแสดงผลตัวอักษร 1 บรรทัด ตามตำแหน่งที่กำหนด */
void thaixy(int colum,int row,int colend,char *content,int status)

```

```
{
```

```
    if((row<1)|| (row>23)) return;
```

```
    G_gotoxy(colum,row);
```

```
    G_writerow(content,colend,status);
```

```
}
```

```

/* ฟังก์ชัน DispCharrow() ทำหน้าที่แสดงตัวอักษร 1 ตัวอักษร โดยถ้า status=0 เป็นตัว
   หนา โดยทำงานร่วมกับฟังก์ชัน G_writerow()*/

```

```
void DispCharrow(byte ST,byte SP,byte *One_Char,int status)
```

```
{
```

```
    byte Count;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for(Count=ST;Count<=SP;Count++)
{
    if(status==0)
    {
        pokeb(0xA000,Off_Line[G_Row+Count]+G_Col,(One_Char[Count]
            !(One_Char[Count]>>1)));
    }
    else
    {
        pokeb(0xA000,Off_Line[G_Row+Count]+G_Col,One_Char[Count]);
    }
}
}

/* ฟังก์ชัน thaixypix() เป็นฟังก์ชันแสดงผลตัวอักษร 1 บรรทัดตามตำแหน่งที่กำหนดโดยไม่
ทำให้ background เปลี่ยนแปลง */
void thaixypix(int colum,int row,int colend,char *content,int status,
                int color,int level)
{
    G_gotoxy(colum,row);
    G_writelow1(content,colend,status,color,level);
}

```

```

/* ฟังก์ชัน DispCharrow1() ทำหน้าที่แสดงตัวอักษร 1 ตัวอักษร โดยถ้า status=0 เป็นตัว
หนา โดยใช้งานร่วมกับฟังก์ชัน G_writelow1() โดยในการแสดงผลจะไม่ทำให้ back-
ground เปลี่ยนแปลง*/

```

```

void DispCharrow1(byte ST,byte SP,byte *One_Char,int status,int color,
                  int level)
{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 byte Count,Xcount,mask;
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

unsigned int Seg_ment;

for(Count=ST;Count<=SP;Count++)
{
    if(status==1)
    {
        if(pagedisplay!=1)Seg_ment=0xA000;
        else Seg_ment=0xA800;
        mask=(One_Char[Count])!((One_Char[Count]>>1);
        outportb(0x3CE,0x08);
        outportb(0x3CF,mask);
        outportb(0x3C4,0x02);
        outportb(0x3C5,0x0F);
        peekb(Seg_ment,Off_Line[G_Row+Count-level]+G_Col);
        pokeb(Seg_ment,Off_Line[G_Row+Count-level]+G_Col,0x00);
        outportb(0x3C5,color);
        pokeb(Seg_ment,Off_Line[G_Row+Count-level]+G_Col,0xFF);
        restore();
    }
    else if(status==2)
    {
        WidthChar(Count,One_Char[Count]);
    }
    else {
        if(pagedisplay!=1)Seg_ment=0xA000;
        else Seg_ment=0xA800;
        mask=One_Char[Count];
        outportb(0x3CE,0x08);
        outportb(0x3CF,mask);
        outportb(0x3C4,0x02);

```

```

    outportb(0x3C5,0x0F);
    peekb(Seg_ment,Off_Line[G_Row+Count-level]+G_Col);
    pokeb(Seg_ment,Off_Line[G_Row+Count-level]+G_Col,0x00);
    outportb(0x3C5,color);
    pokeb(Seg_ment,Off_Line[G_Row+Count-level]+G_Col,0xFF);
    restore();
}

```

/* ฟังก์ชัน complement() ทำหน้าที่ทำการอินเวอร์สค่าใน asc และส่งค่ากลับเป็น unsigned

char */

```

unsigned char complement(unsigned char asc)

```

```

{
    unsigned char ascrev;

    ascrev=0;
    if(!(asc&1)) ascrev|=1;
    if(!(asc&2)) ascrev|=2;
    if(!(asc&4)) ascrev|=4;
    if(!(asc&8)) ascrev|=8;
    if(!(asc&16)) ascrev|=16;
    if(!(asc&32)) ascrev|=32;
    if(!(asc&64)) ascrev|=64;
    if(!(asc&128)) ascrev|=128;

    return(ascrev);
}

```

/* ฟังก์ชัน clearblock() ทำหน้าที่เคลียร์หน้าจอตามตำแหน่งที่ต้องการ */

```

void clearblock(int x,int y,int endx,int endy)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

unsigned int Count,Xcount;

if(endy>24)endy=24;
Set_Color(15);
x--;
endx--;
y=(y-1)*14;
endy=(endy-1)*14;
for (Xcount=y;Xcount<endy;Xcount++)
for(Count=x;Count<endx;Count++)
{
if(pagedisplay==0)pokeb(0xA000,Off_Line[Xcount]+Count,0x00);
else pokeb(0xA800,Off_Line[Xcount]+Count,0x00);
}
}

/* ฟังก์ชัน WidthChar() ทำหน้าที่แสดงผลตัวอักษรเป็นตัวใหญ่ */
void WidthChar(byte Count,byte One_Byte)
{
byte XCount,One_Byte1;
unsigned int ReNum=3,DWNum=0;

One_Byte1=One_Byte|(One_Byte>>1);
for(XCount=0;XCount<8;XCount++)
{
if(One_Byte1 & 1)DWNum=DWNum+ReNum;
One_Byte1 >>= 1;
ReNum <<=2 ;
}

```

```

DWNum=(DWNum<<8)!ReNum;
if(pagedisplay==0) poke(0xA000,Off_Line[G_Row+Count]+G_Col-1,
                                                                    DWNum);
else poke(0xA800,Off_Line[G_Row+Count]+G_Col-1,DWNum);
}

/* ฟังก์ชัน restore() ทำหน้าที่คืนค่าแก่รีจิสเตอร์สี */
void restore()
{
    outportb(0x3CE,0x08);
    outportb(0x3CF,0xFF);
    outportb(0x3C4,0x02);
    outportb(0x3C5,0x0F);
}

/* ฟังก์ชัน G_writerow1() ทำหน้าที่จัดระดับตัวอักษรโดยมีอยู่ 3 ระดับคือ พยัญชนะ สระ
และ วรรณยุกต์ โดยใช้งานร่วมกับฟังก์ชัน thaixypix()*/
void G_writerow1(byte *ThaiStr,byte Colend,int status,int color,int level)
{
    byte ST,SP;

    while(*ThaiStr != NULL)
    {
        switch(*ThaiStr){
            case 143: *ThaiStr=153; break;
            case 144: *ThaiStr=156; break;
            case 145: *ThaiStr=150; break;
            case 146: *ThaiStr=154; break;
            case 147: *ThaiStr=152; break;
            case 149: *ThaiStr=158; break;
            case 150: *ThaiStr=159; break;

```

```

    case 152: *ThaiStr=149; break;
    case 153: *ThaiStr=151; break;
    case 154: *ThaiStr=155; break;
    case 155: *ThaiStr=157; break;
    default:break;
}
if(*ThaiStr>15)
switch(*ThaiStr)
{
    case 216:case 217:case 218:
        {ST=12;SP=13;}break;
    case 209:case 212:case 213:case 214:case 215:case 231:
        {ST=2;SP=4;}break;
    case 232:case 233:case 234:case 235:case 236:case 237:
        {ST=0;SP=2;}break;
    default:
        {ST=0;SP=13;G_Col++;}
}
if(status==2)G_Col++;
if(G_Col>Colend) return;

if(*ThaiStr>15)DispCharrow1(ST,SP,Bit_Normal[*ThaiStr],status,
                                color,level);

switch(*ThaiStr){
    case 153: *ThaiStr=143; break;
    case 156: *ThaiStr=144; break;
    case 150: *ThaiStr=145; break;
    case 154: *ThaiStr=146; break;
    case 152: *ThaiStr=147; break;
    case 158: *ThaiStr=149; break;

```

```
case 159: *ThaiStr=150; break;
case 149: *ThaiStr=152; break;
case 151: *ThaiStr=153; break;
case 155: *ThaiStr=154; break;
case 157: *ThaiStr=155; break;
default:break;
```

```
}
```

```
*ThaiStr++;
```

```
}
```

```
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/*****
***** HELP.C *****
*****/

```

```

#include<dos.h>
#include<graphics.h>
#include<stdio.h>
#include<alloc.h>
#include<io.h>
#include<fcntl.h>
#include<stdarg.h>
#include<conio.h>
#include<stdlib.h>
#include "keyboard.c"

extern char scan,exten;
extern char Blod;
extern char Tab;
extern int inter,gint;
static int presstab=0;
unsigned statuskey;
int xbufold,ybufold;
char far *bufold;
extern int count;
unsigned char str[110],cha;
char far *pointerindex,*ptrpage;
char far *pri,*buftype1,*buftype2;
int coltype1,rowtype1,coltype2,rowtype2;

```

```

extern unsigned char complement(unsigned char asc);
extern void clearblock(int x,int y,int endx,int endy);
extern void thaixypix(int colum,int row,int colend,char *content, int
                        status,int color,int pixel);
extern void outthaixy(int colum,int row,int colend,char *content);
extern void thaixy( int colum, int row, int colend, char *content,int
                        status);

void type1(int col,int row,char *str1,char *str2);
void helpsub(int col,int rowstart,int page1);
void type(int col,int row,char *str1,char *str2);
void openfile(char *fname);
unsigned int swap(unsigned int value);
void help(int page);
void background();
void prajoam(void);
void barband();
void error(int status);

/*ฟังก์ชัน barband() ทำหน้าที่สร้างแถบสีเหลี่ยมด้านล่างและภายในแถบสีจะบอกหน้าที่ต่างๆ
ของปุ่มที่ใช้ในโปรแกรม*/
void barband()
{
    setcolor(11);
    rectangle(0,330,639,349);
    setfillstyle(1,3);
    floodfill(1,331,11);
    setcolor(8);
    line(1,349,639,349);
    line(639,330,639,349);
    prajoam();
}

```

```

thaixypix(5,25,79,"พระจอมเกล้าลาดกระบัง",1,5,3);
thaixypix(29,25,79,"F1",1,4,3);
thaixypix(32,25,79,"อธิบายคำสั่ง",1,1,3);
thaixypix(46,25,79,"F10",1,4,3);
thaixypix(50,25,79,"เมนู",1,1,3);
thaixypix(58,25,79,"ALT+D",1,4,3);
thaixypix(64,25,79,"ไปคอลี่วคราว",1,1,3);
}

```

```

/* ฟังก์ชัน prajoam() ทำหน้าที่สร้างตราพระจอมเกล้า */
void prajoam(void)
{
    unsigned int pra[17]={ 0x0100,0x0920,0x0540,0x3398,0x0C60,0x1FF0,
                           0x6BAC,0x97D2,0x2448,0x4FE4,0x9012,0x1050,
                           0x3398,0x3418,0xB49A,0xB35A,0x701C };
    int i;
    setcolor(10);
    for(i=0;i<17;i++)
    {
        setlinestyle(4,pra[i],1);
        line(19,i+331,3,i+331);
    }
    setlinestyle(4,0xffff,1);
}

/* ฟังก์ชัน error() ทำหน้าที่แสดงว่าเกิดข้อบกพร่องเกิดขึ้น */
void error(int status)
{

```

```

    char far *pt;

```

```

    int x,y,x1,y1,lenght;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
unsigned long size;
```

```
if(status==1)lenght=0;
```

```
else lenght=30;
```

```
x=random(59);
```

```
y=random(21);
```

```
if(x==0)x=1;
```

```
if(y==0)y=1;
```

```
x1=x*8;
```

```
y1=y*14;
```

```
size=imagesize(x1,y1,x1+150+lenght,y1+30);
```

```
pt=farmalloc(size);
```

```
getimage(x1,y1,x1+150+lenght,y1+30,pt);
```

```
setviewport(x1,y1,x1+150+lenght,y1+30,1);
```

```
clearviewport();
```

```
setviewport(0,0,639,349,1);
```

```
setcolor(15);
```

```
rectangle(x1,y1,x1+150+lenght,y1+30);
```

```
setfillstyle(1,4);
```

```
floodfill(x1+1,y1+1,15);
```

```
if(status==1)
```

```
{
```

```
    thaixypix(x+3,y+1,79,"หน่วยความจำไม่พอ ",0,14,-1);
```

```
    thaixypix(x+2,y+2,79,"กดปุ่มใดเพื่อทำงานต่อ",0,14,-1);
```

```
}
```

```
else
```

```
{
```

```
    thaixypix(x+2,y+1,79,"ท่านกดคีย์ไม่ถูกต้อง พยายาม ",0,14,-1);
```

```
    thaixypix(x+2,y+2,79,"ใหม่ กดคีย์ใดๆเพื่อทำงานต่อ",0,14,-1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

clearb();
getch();
putimage(x1,y1,pt,0);
farfree(pt);
return;
}

```

```

/* ฟังก์ชัน swap() จะทำหน้าที่สลับค่าใน value และส่งค่ากลับเป็น unsigned int */
unsigned int swap(unsigned int value)

```

```

{
    unsigned int temp,tansmit;

    temp=value>>8;
    tansmit=(value<<8)!temp;
    return(tansmit);
}

```

```

/* ฟังก์ชัน help() จะแสดงข้อมูลข่าวสารตามหมายเลขหน้าของ page */

```

```

void help(int page)
{
    FILE *fp;
    int fi;
    unsigned long size;
    unsigned int countbyte;
    extern unsigned char str[110];
    char far *ptr1,*ptr2,*ptr3,*ptr4,*ptr5,*ptr6,*ptr7,*ptr,*buf;
    char row,Pup,Pdown,inten,col,rowstart=3,ch,loop,find,row1,row2=0,no;
    char fname[15],fnameold[15];
    int far *pointer1;
    int i,temp,pageindex[50],select,enter=0,oldpage[20],pageold=0;
    extern int pagedisplay;

```

```

if((fp=fopen(fname,"rb"))==NULL) /*open for load to memory*/
{
    printf("Cannot open file %s",fname);
    exit(0);
}
strcpy(fnameold,fname);
ptr=farmalloc(size); /*ptr start of file*/
if(ptr==NULL)
{
    printf("\a");
    exit(0);
}
pointer1=ptr2=ptr3=ptr1=ptr;
count=0;
do
{
    *ptr2=getc(fp);
    ptr2++;
    count++;
} while(count!=size);
ptr2--; /*end of file*/
} /*end if*/

if(pageold==page)goto getkey;
pointer1=ptr3=ptr1=ptr; /*pointer1 is index page*/
while(page!=(swap(*pointer1)))
{
    ptr1+=5;
    if(*ptr1!=0)
    {
        temp=*ptr1;

```

```

        for(i=0;i<temp;i++)
        ptr1+=2;
    }

    ptr1+=2;
    pointer1=ptr1;
    countbyte=swap(*pointer1);
    ptr1+=2;
    ptr1=ptr1+(unsigned int)countbyte;
    pointer1=ptr1;
}
/*end of while*/
ptr1+=2;
row=*ptr1;
ptr1++;
Pup=*ptr1;
ptr1++;
Pdown=*ptr1;
ptr1++;
inten=*ptr1;
ptr1++;
col=*ptr1;
ptr1++;
pointer1=ptr1;
if(inten!=0)
{
    for(i=0;i<inten;i++)
    {
        pageindex[i]=swap((unsigned int)*pointer1);
        pointer1++;
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

countbyte=swap((unsigned int)*pointer1);
ptr1=pointer1;
ptr1+=2;          /*ptr1 is start of page*/
ptr4=ptr1;
ptr3=ptr1+countbyte;    /*ptr3 is end of page*/
rowstart=3;
do {
    i=0;
    do {
        str[i]=*ptr4;
        ptr4++;
        i++;
    } while((str[i-1]!='\n')&&(ptr4!=ptr3));
    str[i]='\0';
    Set_Color(15);    /*clear next row*/
    clearblock(12,rowstart,75,rowstart+1);
    outthaixy(12,rowstart,79,str);
    if(ptr4==ptr3)
    while(rowstart!=21)
    {
        clearblock(12,rowstart+1,75,rowstart+2);
        rowstart++;
    }
    rowstart++;
} while((rowstart<=(row+3))&&(ptr4!=ptr3));
setvisualpage(1);
if((Pdown==1)&&(Pup==1))
{
    putimage(450,310,buf,0);

```

```

else if((Pup==1)&&(Pdown==0))
{
    putimage(450,310,buf,0);
    thaixypix(70,23,79,"Page Up",1,4,0);
}
else if((Pdown==1)&&(Pup==0))
{
    putimage(450,310,buf,0);
    thaixypix(60,23,79,"Page Down ",1,4,0);
}
else putimage(450,310,buf,0);
select=0;
pageold=page;
do
{
do
{
    getkey: cha=keyboard(0,&statuskey,&scan,&exten);
    switch(scan){
    case 75:
    case 77:
    case 15: if(inten==0)break;
            select++;
            if(select==1){row1=0;no=0;}
            if(select>inten){select=1;row1=0;no=1;}
            if((select!=1)!!(no!=0))
            {
                loop=0;ptr4--;
            }
        }
    }
}
ptr4--;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

while((*ptr4!='\n')&&(ptr4!=ptr1));
if(ptr4!=ptr1)ptr4++;
i=0;
do{
    str[i]=*ptr4;
    ptr4++;i++;
    } while(str[i-1]!='\n');
str[i]='\0';
outthaixy(12,rowstart+row2,79,str);
Set_Color(15);
}
rowstart=3;
find=(select-1)*2;
if(find==0) find=1;
ptr4=ptr1;
loop=0;
while(loop<find)
{
    if(*ptr4==0x02)loop++;
    if((*ptr4=='\n')&&(inten!=select))row1++;
    if((*ptr4=='\n')&&(inten==1))row1++;
    ptr4++;
}
if(find==1)ptr4-- ;
loop=0;
while(loop<2)
{
    if(*ptr4==0x02)
    {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 *ptr4=0x03;
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

loop++;
if(loop==1)ptr5=ptr6=ptr4;
}
else
{
    if(*ptr4=='\n')row1++;
    ptr4++;
}
} /*end of while*/
row2=row1;
do
    ptr4--;
while((*ptr4!='\n')&&(ptr4!=ptr1));
if(ptr4!=ptr1)ptr4++;
i=0;
do{
    str[i]=*ptr4;
    i++;ptr4++;
} while(str[i-1]!='\n');
str[i]='\0';
outthaixy(12,rowstart+row1,79,str);
Set_Color(15);
loop=0;
while(loop<2)
{
    if(*ptr6==0x03)
    {
        loop++;
        *ptr6=0x02;
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        ptr6++;
    }
    break;

    default:break;
} /*end of switch(scan)*/

} while((scan!=81)&&(scan!=73)&&(scan!=16)&&(scan!=28)&&(scan!=1));
switch(scan) {
    case 81:if(Pdown==1)page++;
        break;
    case 73:if(Pup==1)page--;
        break;
    case 28:type(22,25,"ENTER","เลือก");
        if(inten==0)break;
        if(select==0)select++;
        oldpage[enter]=page;
        enter++;
        if(enter==20)
        {
            for(i=1;i==19;i++)
            {
                oldpage[i-1]=oldpage[i];
            }
            enter=19;
        }
        page=pageindex[select-1];
        break;

```

```

        if(enter<=0)break;

        enter--;

        page=oldpage[enter];

        break;

    default:break;
}    /*end of switch*/

} while(((scan!=81)!!(Pdown==0))&&((scan!=73)!!(Pup==0))&&(scan!=16)
        &&(scan!=28)&&(scan!=1));

if(!(strcmp(fname,"paper1.hlp")))
{
    if(!(page>=1)&&(page<=26))farfree(ptr);
}
else if(!(strcmp(fname,"paper11.hlp")))
{
    if(!(page>=27)&&(page<=58))farfree(ptr);
}
else if(!(strcmp(fname,"paper2.hlp")))
{
    if(!(page>=59)&&(page<=111))farfree(ptr);
}
else if(!(strcmp(fname,"paper3.hlp")))
{
    if(!(page>=112)&&(page<=161))farfree(ptr);
}
else if(!(strcmp(fname,"paper4.hlp")))
{
    if(!(page>=162)&&(page<=205))farfree(ptr);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    cha=toupper(cha);
    if(cha=='Q') type(5,25,"Q","  ออกจาก HELP");
} while(cha!='Q');
farfree(ptr);
farfree(buf);
setactivepage(0);
pagedisplay=0;
setvisualpage(0);
if(gint==0) setint(inter);
return ;
}

/* ฟังก์ชัน openfile() ทำหน้าที่เปิดเท็กซ์ไฟล์และโหลดลงหน่วยความจำโดยใช้พอยน์เตอร์
pointerindex เป็นตัวชี้โดยเท็กซ์ไฟล์ที่โหลดลงหน่วยความจำนี้ จะนำไปใช้ร่วมกับฟังก์ชัน
helpsub() */
void openfile(char *fname)
{
    FILE *fp1;
    int fi,count;
    unsigned long size;
    extern char far *pointerindex, *ptrpage;

    if((fi=open(fname,O_RDONLY;O_BINARY))<0)
    {
        printf("Cannot open file test11.hlp");
        exit(0);
    }
    size=filelength(fi);
    close(fi);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if((fp1=fopen(fname,"rb"))==NULL)
{
    printf("Cannot open file %s",fname);
    exit(0);
}

pointerindex=farmalloc(size);
ptrpage=pointerindex;
count=0;
do
{
    /*ptr2 check end of file*/
    *ptrpage=getc(fp1);
    ptrpage++;
    count++;
} while(count!=size);
ptrpage--; /*end of file*/
fclose(fp1);
} /*end of function*/

```

/* ฟังก์ชัน helpsub() ทำหน้าที่แสดงข้อมูลและส่วนช่วยเหลือในขณะ เรียบโดยการชี้ฟังก์ชัน จะต้องกำหนดตำแหน่งที่ต้องการนำไปวางกับหมายเลขหน้าให้กับฟังก์ชัน ภายในฟังก์ชันจะทำการตีกรอบให้เสร็จและเก็บภาพหน้าจอไว้ให้ด้วย */

```

void helpsub(int col,int rowstart,int page1)
{
    int far *pointerint;
    int i,temp,c,r;
    char far *ptr1, far *ptr2, far *ptr3, far *ptr4;
    char Pup,Pdown,inten,colend,row,pageindex[50];
    unsigned int countbyte,size,length;
    extern int xbufold,ybufold;
    extern char far *bufold;

```

```

pointerint=ptr3=ptr1=pointerindex; /*pointerint is index page*/
while(page1!=(swap(*pointerint)))
{
    ptr1+=5;
    if(*ptr1!=0)
    {
        temp=*ptr1;
        for(i=0;i<temp;i++)
            ptr1+=2;
    }
    ptr1+=2;
    pointerint=ptr1;
    countbyte=swap(*pointerint);
    ptr1+=2;
    ptr1=ptr1+(unsigned int)countbyte;
    pointerint=ptr1;
} /*end of while*/

ptr1+=2;
row=*ptr1;
ptr1++;
Pup=*ptr1;
ptr1++;
Pdown=*ptr1;
ptr1++;
inten=*ptr1;
ptr1++;
colend=*ptr1;
ptr1++;

pointerint=ptr1;

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์สงวนลิขสิทธิ์การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(inten!=0)
{
    for(i=0;i<inten;i++)
    {
        pageindex[i]=swap((unsigned int)*pointerint);
        pointerint++;
    }
}

countbyte=swap((unsigned int)*pointerint);
ptr1=pointerint;
ptr1+=2;          /*ptr1 is start of page*/
ptr4=ptr1;
ptr3=ptr1+countbyte; /*ptr3 is end of page*/
size=imagesize((col-3)*8,(rowstart-3)*14,(col+colend+1)*8+10,
               (row+rowstart+1)*14+10);
if((c=(col+colend+1)*8+10)>640)c=639;
if((bufold=farmalloc(size))!=NULL)
{
    getimage((col-3)*8,(rowstart-3)*14,c,(row+rowstart+1)*14+10,
            bufold);
}

else printf("\a");

setviewport((col-3)*8,(rowstart-3)*14,c-10,(row+rowstart+1)
            *14,1);

clearviewport();

setviewport(0,0,getmaxx(),getmaxy(),1);

setcolor(3);

rectangle( (col-3)*8+10,(rowstart-3)*14+10, (col+colend+1)*8
            +10,(row+rowstart+1)*14+10);

setfillstyle(1,3);

```

```

floodfill((col-3)*8+11,(rowstart-3)*14+11,3);
setcolor(15);
rectangle( (col-3)*8, (rowstart-3)*14, (col+colend+1)*8,(row
+rowstart+1)*14);
rectangle( (col-2)*8, (rowstart-2)*14 , (col+colend)*8, (row
+rowstart)*14);

setfillstyle(1,7);
floodfill((col-3)*8+1,(rowstart-3)*14+1,15);
setfillstyle(1,1);
floodfill((col-2)*8+1,(rowstart-2)*14+1,15);
setcolor(8);
line( (col-3)*8, (row+rowstart+1)*14, (col+colend+1)*8, (row
+rowstart+1)*14);
line( (col+colend+1)*8,(rowstart-3)*14,(col+colend+1)*8,(row
+rowstart+1)*14);
line((col-2)*8,(rowstart-2)*14,(col-2)*8,(row+rowstart)*14);
line((col-2)*8,(rowstart-2)*14, (col+colend)*8, (rowstart-2)
*14);
do{
    i=0;
    do{
        str[i]=*ptr4;
        ptr4++;
        i++;
    } while((str[i-1]!='\n')&&(ptr4!=ptr3));
    str[i]='\0';
    outthaixy(col,rowstart,79,str);
    rowstart++;
} while((rowstart<=(row+rowstart))&&(ptr4!=ptr3));

```

```

ybufold=(rowstart-3-row)*14;
Set_Color(15);
}

/* ฟังก์ชัน type() ทำหน้าที่แสดงรูปปุ่มที่ใช้ในการกดคีย์และในขณะที่กดจะทำให้ปุ่มในฮุบลงไป
ด้วย */
void type(int col,int row,char *str1,char *str2)
{
    unsigned int length;
    int x,y,offset,delaytime,cou,i;
    char far *str;

    switch(scan){
        case 16:
        case 28:
        case 1 :delaytime=500;break;
        default:delaytime=0;break;
    }

    length=strlen(str2);
    cou=0;
    str=str2;

    for(i=0;i<=length;i++,str2++)
    switch(*str2)
    {
        case 216:case 217:case 218:
        case 209:case 212:case 213:case 214:case 215:case 231:
        case 232:case 233:case 234:case 235:case 236:
        case 237:cou++;
        default:break;
    }
}

```

```

length-=cou;
str2=str;
length+=strlen(str1);
length*=8;
x=(col-1)*8;
y=(row-1)*14;
if(row==25) y=y-5;
setcolor(8);
rectangle(x,y-2-2,x+length+5,y+16-2);
rectangle(x+1,y+1-2-2,x+length+4,y+15-2);
setfillstyle(1,7);
floodfill(x+2,y+2-2,8);
setcolor(15);
line(x,y+16-2,x+length+5,y+16-2); /* -- */
line(x+1,y+15-2,x+length+5-1,y+15-2); /* -- */
line(x+length+5,y-2-2,x+length+5,y+16-2); /* ! */
line(x+length+5-1,y+1-2-2,x+length+5-1,y+15-2); /* ! */
if(row==25)offset=7;
else offset=2;
thaixypix(col+1,row,79,str1,1,8,offset);
thaixypix(col+strlen(str1),row,79,str2,1,4,offset);
delay(delaytime);
setcolor(15);
rectangle(x,y-2-2,x+length+5,y+16-2);
rectangle(x+1,y+1-2-2,x+length+4,y+15-2);

setcolor(8);
line(x,y+16-2,x+length+5,y+16-2); /* -- */

```

```

line(x+length+5-1,y+1-2-2,x+length+5-1,y+15-2); /* ! */
}

/* เป็นฟังก์ชันที่มีลักษณะเหมือนกับฟังก์ชัน type แต่จะใช้ในการแสดงว่าต้องการให้ผู้เรียน
พิมพ์อะไร */

void type1(int col,int row,char *str1,char *str2)
{
    unsigned int length;
    int x,y,cou,i;
    char far *str;
    extern char far *buftype1;
    unsigned long sizeborder;
    extern coltype1,rowtype1;

    length=strlen(str2);
    cou=0;
    str=str2;
    for(i=0;i<=length;i++,str2++)
    switch(*str2)
    {
        case 216:case 217:case 218:
        case 209:case 212:case 213:case 214:case 215:case 231:
        case 232:case 233:case 234:case 235:case 236:
        case 237:cou++;
        default:break;
    }

    length-=cou;
    str2=str;

    length+=strlen(str1);
    length*=8;

```

```

x=(col-1)*8;
y=(row-1)*14;
if(row==25) y=y-5;
setcolor(15);
sizeborder=imagesize(x,y-2-3,x+length+5,y+16-3);
if((buftype1=farmalloc(sizeborder))!=NULL) printf("\a");
getimage(x,y-2-3,x+length+5,y+16-3,buftype1);
setviewport(x,y-2-3,x+length+5,y+16-3,1);
clearviewport();
setviewport(0,0,639,349,1);
rectangle(x,y-2-3,x+length+5,y+16-3);
rectangle(x+1,y+1-2-3,x+length+4,y+15-3);
setfillstyle(1,7);
floodfill(x+2,y-2+2,15);
setcolor(8);
line(x,y+16-3,x+length+5,y+16-3); /* -- */
line(x+1,y+15-3,x+length+5-1,y+15-3); /* -- */
line(x+length+5,y-2-3,x+length+5,y+16-3); /* ! */
line(x+length+5-1,y+1-2-3,x+length+5-1,y+15-3); /* ! */
thaixypix(col+1,row,79,str1,1,8,4);
thaixypix(col+strlen(str1),row,79,str2,1,4,4);
coltype1=x;
rowtype1=y-2-3;

```

}

```

/*****
***** PROGRAM TOPIC1.C *****/
*****/

#include<alloc.h>
#include<graphics.h>
extern int cktp,xbufold,ybufold,coltype1,rowtype1,coltype2,rowtype2,
        inter,gint;
extern char far *pointerindex;
extern char gt[30];
extern char far *bufold;
extern unsigned int on;
extern char far *buftype1,*buftype2;
void topic_1(int cas);
void Check_1(char *g,int topic);

/* ฟังก์ชัน topic_1 เป็นการเรียนานบทที่ 1 หัวข้อที่ 1-3 */
void topic_1(int cas)
{
    barband();
    gint=1;
    openfile("topic1.hlp");
    if(cas==1){
        clearblock(1,1,81,24);
        Check_1(&gt[0],1);
        on=26;
        helpsub(30,14,22);
        type1(50,22,"Type"," cls");
        Check_Key(3,0,22,2,0,1);}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(cas<=2){
clearblock(1,1,81,24);
on=109;
helpsub(5,10,2);
type1(50,22,"Type"," Help");
Check_Key(4,0,1,3,0,1);
helpsub(30,12,3);
type1(50,22,"Type"," Help");
Check_Key(4,0,23,4,0,1);
helpsub(30,14,22);
type1(50,22,"Type"," Help cls");
Check_Key(8,0,23,5,0,1);
helpsub(30,14,4);
inter=1;
setint(inter);gint=0;
type1(50,22,"Type"," cls /?");
Check_Key(6,0,23,6,8,1);
helpsub(30,14,5);
setint(inter);gint=0;
type1(50,22,"Type"," Help");
Check_Key(4,0,23,7,0,1);
on=26;
helpsub(30,14,7);
inter=2;
setint(inter);gint=0;
type1(50,22,"Type"," cls");
Check_Key(3,0,23,2,0,1);
restore1();Cint(inter);gint=1;
on=191;
helpsub(30,14,8);

```

```

type1(56,22,"Type"," ver /?");
Check_Key(6,0,2,8,0,1);
helpsub(35,14,9);
type1(56,22,"Type"," ver");
Check_Key(3,0,7,9,0,1);
thaixy(1,11,8,"C:\\\>" ,0);
helpsub(35,15,10);
type1(50,22,"Press"," Page down");
hot_key(0,0,0,1,81);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(bufold);
farfree(buftype1);}

if(cas<=3){
clearblock(1,1,81,24);
on=39;
helpsub(5,10,11);
type1(50,22,"Type"," date");
Check_Key(4,0,2,10,0,1);
on=163;
helpsub(30,14,13);
type1(50,22,"Type"," time");
Check_Key(4,0,6,11,0,1);
helpsub(30,12,14);
type1(50,22,"Type"," 10:00");
Check_Key(5,12,8,12,1,1);
helpsub(30,14,15);
type1(50,22,"Type"," time");
Check_Key(4,0,10,13,0,1);

```

```

on=134;
helpsub(30,14,17);
type1(50,22,"Type"," prompt /?");
Check_Key(9,0,14,14,0,1);
helpsub(30,12,18);
inter=3;
setint(inter);gint=0;
type1(50,22,"Type"," prompt");
Check_Key(6,0,23,15,0,1);
helpsub(30,13,19);
inter=4;
setint(inter);gint=0;
type1(50,22,"Type"," prompt Hi!");
thaixy(1,23,5,"C>",0);
Check_Key(10,-2,23,16,1,1);
helpsub(30,14,20);
inter=5;
setint(inter);gint=0;
type1(35,22,"Type"," prompt $p$g");
Check_Key(11,-1,23,17,1,1);
thaixy(1,23,8,"C:\\>" ,0);
type1(47,22,"Press"," Page Down");
hot_key(0,23,250,0,81);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
farfree(buftype1);
}
farfree(pointerindex);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

/* ฟังก์ชัน Check_1 เป็นการตรวจสอบตัวอักษรที่พิมพ์ว่า พิมพ์ตามที่กำหนดไว้หรือไม่ ถ้าไม่ใช่
ก็กลับไปพิมพ์ใหม่ หรือรอรับคีย์ที่ต้องการให้กดก็ได้ */

```
void Check_1(char *g,int topic)
{
    int too,loop,nst=0;
    char st_2[]="CLS",st_3[]="HELP",st_5[]="HELP CLS",st_6[]="CLS /?",
        .st_8[]="VER /?",st_9[]="VER",st_10[]="DATE",st_11[]="TIME",
        st_12[]="10:00",st_14[]="PROMPT /?",st_15[]="PROMPT",
        st_16[]="PROMPT HI!",st_17[]="PROMPT $P$G";
    static char mix_4[4],mix_5[8],mix_6[6],mix_7[4],mix_10[4],mix_11[4],
        mix_13[4],mix_14[9],mix_15[6],mix_16[10],mix_17[11];
    switch(topic) {
        case 1 : helpsub(5,10,1);
            type1(50,23,"กด", " ENTER ");
            thaixypix(60,23,79,"จำลองการเปิดเครื่อง",1,0,4);
            hotkey(13);
            if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
            if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
            farfree(bufold);
            farfree(buftype1);
            topic1_1(0,1);
            break;
        case 2 : for(loop=0;loop<=2;loop++){
            if(toupper(g[loop])==st_2[loop]) nst++;}
            if(nst==3) {
                keep();
                cktp=0;
                clearblock(1,1,79,24);
            }
    }
```

เอกสารนี้เป็นเอกสาร **break**; สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

case 3 : for(loop=0;loop<=3;loop++){
    if(toupper(g[loop])==st_3[loop]) nst++;}
if(nst==4) {
    keep();
    cktp=0;
    topic1_22(1,1);
    topic1_2();
    topic1_21(1,1);
    for(loop=0;loop<19;loop++)
    topic1_21(-loop,0);
    }
break;
case 4 : for(loop=0;loop<=3;loop++){
    if(toupper(g[loop])==st_3[loop])
        {mix_4[loop]=g[loop]; nst++;}}
if(nst==4) {
    cktp=0;
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    farfree(buftype1);
    type1(65,22,"Press"," Esc");
    hot_key(0,0,0,1,27);
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(bufold);
    farfree(buftype1);
    for(too=0;too<=1;too++){
        topic1_21(-too-18,0);
        mix(-too+23,&mix_4,4);}

```

```

type1(65,22,"Press"," Enter");
hot_key(0,23,250,0,13);
if(buftype1!=NULL)
    putimage(coltype1,rowtype1,buftype1,0);
farfree(buftype1);
topic1_21(-20,0);
mix(21,&mix_4,4);
    }
break;
case 5 : for(loop=0;loop<=7;loop++){
    if(toupper(g[loop])==st_5[loop])
        {mix_5[loop]=g[loop]; nst++;}
if(nst==8) {
    keep();
    cktp=0;
    for(too=1;too<=5;too++){
        topic1_21(-too-20,0);
        mix(21-too,&mix_4,4);
        mix(23-too,&mix_5,5);}
    }
break;
case 6 : for(loop=0;loop<=5;loop++){
    if(toupper(g[loop])==st_6[loop])
        {mix_6[loop]=g[loop];nst++;}
if(nst==6) {
    restore1();
    Cint(inter);
    gint=1;
    keep();
    cktp=0;

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for(too=1;too<=5;too++){
    topic1_21(-too-25,0);
    mix(16-too,&mix_4,4);
    mix(18-too,&mix_5,5);
    mix(23-too,&mix_6,6);}
}

```

```
break;
```

```

case 7 : for(loop=0;loop<=3;loop++){
    if(toupper(g[loop])==st_3[loop])
        {mix_7[loop]=g[loop];nst++;}}
if(nst==4) {
    restore1();
    Cint(inter);
    gint=1;
    keep();
    cktp=0;
    for(too=1;too<=22;too++){
        topic1_21(-too-28,0);
        mix(11-too,&mix_4,4);
        mix(13-too,&mix_5,5);
        mix(18-too,&mix_6,6);
        mix(-too+23,&mix_7,7);
        topic1_22(-too+23,1);}
    helpsub(30,14,6);
    type1(65,22,"Type"," CTRL+C");
    hot_key(3);
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    farfree(buftype1);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        farfree(bufold);
        for(loop=0;loop<2;loop++){
            topic1_22(-loop,0);}
        }
    break;
case 8 : for(loop=0;loop<=5;loop++){
        if(toupper(g[loop])==st_8[loop]) nst++;}
    if(nst==6) {
        keep();
        cktp=0;
        thaixy(1,3,30,"Display the DOS version.",0);
        thaixy(1,5,7,"VER",0);
        }
    break;
case 9 : for(loop=0;loop<=2;loop++){
        if(toupper(g[loop])==st_9[loop]) nst++;}
    if(nst==3) {
        cktp=0;
        keep();
        thaixy(1,9,30,"DOS Version 5 ",0);
        }
    break;
case 10 : for(loop=0;loop<=3;loop++){
        if(toupper(g[loop])==st_10[loop])
            {mix_10[loop]=g[loop];nst++;}
        if(nst==4) {
            cktp=0;
            keep();
            mix(2,&mix_10,10);
            helpsub(30,14,12);

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ซึ่งห้ามเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

type1(65,22,"Press"," Enter");
hot_key(23,4,250,0,13);
if(buftype1!=NULL)
    putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(bufold);
farfree(buftype1);
    }

```

```
break;
```

```

case 11 : for(loop=0;loop<=3;loop++){
    if(toupper(g[loop])==st_11[loop])
        {mix_11[loop]=g[loop];nst++;}
if(nst==4) {
    cktp=0;
    keep();
    mix(6,&mix_11,11);
    }

```

```
break;
```

```

case 12 : for(loop=0;loop<=4;loop++){
    if(toupper(g[loop])==st_12[loop]) nst++;}
if(nst==5) {
    keep();
    cktp=0; }

```

```
break;
```

```

case 13 : for(loop=0;loop<=3;loop++){
    if(toupper(g[loop])==st_11[loop])
        {mix_13[loop]=g[loop];nst++;}
if(nst==4) {
    cktp=0;

```

```
keep();
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mix(10,&mix_13,13);
helpsub(30,14,16);
type1(65,22,"Press"," Enter");
hot_key(12,12,250,0,13);
if(buftype1!=NULL)
    putimage(coltype1,rowtype1,buftype1,0);
farfree(buftype1);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(bufold);
    }
break;
case 14 : for(loop=0;loop<=8;loop++){
    if(toupper(g[loop])==st_14[loop])
        {mix_14[loop]=g[loop];nst++;}
if(nst==9) {
    cktp=0;
    keep();
    for(too=0;too<14;too++){
        mix(-too+1,&mix_10,10);
        mix(-too+5,&mix_11,12);
        mix(-too+9,&mix_13,13);
        mix(-too+13,&mix_14,14);
        topic1_3(-too+14); }
    }
break;
case 15 : for(loop=0;loop<=5;loop++){
    if(toupper(g[loop])==st_15[loop])
        {mix_15[loop]=g[loop]; nst++;;}
if(nst==6) {

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        restore1();
        Cint(inter);
        gint=1;
        keep();
        for(too=0;too<+2;too++){
            topic1_3(-too);
            mix(-too+22,&mix_15,15);}
        }
    break;

case 16 : for(loop=0;loop<=9;loop++){
    if(toupper(g[loop])==st_16[loop])
        {mix_16[loop]=g[loop]; nst++;}
    if(nst==10){
        cktp=0;
        restore1();
        Cint(inter);
        gint=1;
        keep();
        for(too=3;too<+5;too++){
            topic1_3(-too+1);
            mix(-too+23,&mix_15,15);
            mix(2-too+23,&mix_16,16);}
        thaixy(1,23,3,"Hi!",0);
        }
    break;

case 17 : for(loop=0;loop<=10;loop++){
    if(toupper(g[loop])==st_17[loop])
        {mix_17[loop]=g[loop]; nst++;}
    if(nst==11) {
        cktp=0;

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        restore1();
        Cint(inter);
        gint=1;
        keep();
        for(too=0;too<+2;too++){
            topic1_3(-too-4);
            mix(-too-5+23,&mix_15,15);
            mix(-too-3+23,&mix_16,16);
            mix(-too-1+23,&mix_17,17);}
        }
    break;
}
}

```

/* ฟังก์ชัน mix เป็นการนำข้อความที่เก็บเข้ามาแสดงที่หน้าจอ บนลักษณะที่เลื่อนขึ้นๆเรื่อยๆ โดยจะรับค่าบรรทัด ข้อความและหัวข้อที่ต้องการให้แสดง */

```

mix(int y,char *dt,int cas)
{
    if(cas!=16) thaixy(1,y,4,"C:\\\>",0);
    if((cas==14)|| (cas==15)) thaixy(5,y,9,dt,0);
        else thaixy(5,y,7,dt,0);
    if(cas==16) {
        thaixy(1,y,3,"C>",0);
        thaixy(3,y,10,dt,0);}
    thaixy(1,y+1,79,"

    if((cas==5)|| (cas==6)) {
        if(cas==5) thaixy(5,y,11,dt,0);
        if(cas==6) thaixy(5,y,10,dt,0);

```

เอกสารนี้เป็น `thaixy(1,y+1,20,"Clear the screen",0);` ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณี `thaixy(1,y+3,20,"CLSปลงเนื้อหา และต้อง",0);` ถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

thaixy(1,y+2,79,"          ",0);
thaixy(1,y+4,79,"          ",0);
    }

else {
    if(cas==4) thaixy(9,y,20,"\\          ",0);
    if((cas==10)!!(cas==11)!!(cas==12)!!(cas==13)){
        if(cas==10) {
            thaixy(1,y+1,50,"Current date is Wed 01 01 1992",0);
            thaixy(1,y+2,50,"Enter new date (mm-dd-yy):          ",0);}
        if((cas==11)!!(cas==12)) {
            thaixy(1,y+1,50,"Current time is 4:10:39.58p ",0);
            if(cas==11) thaixy(1,y+2,50,"Enter new time:          ",0);
            if(cas==12) thaixy(1,y+2,50,"Enter new time: 10:00 ",0);
            }
        if(cas==13) {
            thaixy(1,y+1,50,"Current time is 10:00:18.79a ",0);
            thaixy(1,y+2,50,"Enter new time:          ",0);}
        thaixy(1,y+3,50,"          ",0);
        }
    if(cas==14) thaixy(12,y,12,"/?",0);
    if(cas==16) thaixy(10,y,11,&dt[7],0);
    if(cas==17) {
        thaixy(1,y,79,"Hi!          ",0);
        thaixy(4,y,10,dt,0);
        thaixy(11,y,13,&dt[7],0); }
    if(cas==7) thaixy(1,y+1,79,"
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/*****
***** PROGRAM TOPIC2.C *****/
*****/

#include<alloc.h>
#include<graphics.h>
extern int on;
extern char gt[30];
extern char far *pointerindex;
extern int cktp,xbufold,ybufold,coltype1,rowtype1,coltype2,rowtype2,
        gint,inter;
extern char far *bufold;
extern char far *buftype1,*buftype2;
void topic_2(int cas);
void Check_2(char *g,int topic);

/* ฟังก์ชัน topic_2 เป็นการเรียนานบทที่ 2 หัวข้อที่ 1-4 */
void topic_2(int cas)
{
    barband();
    gint=1;
    openfile("topic2.hlp");
    if(cas==1){
        clearblock(1,1,81,24);
        on=1;
        helpsub(5,10,1);
        type1(52,22,"Type"," A:");
        Check_Key(2,0,2,0,0,2);
        helpsub(30,14,2);
        type1(52,22,"Type"," R");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Check_Key(1,15,4,1,1,2);
on=47;
helpsub(30,14,28);
type1(50,22,"Type"," dir");
Check_Key(3,0,6,2,2,2);
delay(3000);
helpsub(36,3,3);
type1(65,22,"Press"," Enter");
hotkey(13);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
on=10;
helpsub(30,13,4);
type1(47,22,"Type"," cd word");
Check_Key(7,0,17,3,2,2);
thaixy(1,19,8,"A:\\WORD>",0);
helpsub(30,14,29);
inter=6;
setint(inter);gint=0;
type1(55,22,"Type"," cd\\");
Check_Key(3,4,19,4,1,2);}

if(cas<=2){
clearblock(1,1,81,24);
on=47;
helpsub(5,10,5);
type1(51,22,"Type"," dir");

```

เอกส Check_Key(3,0,2,5,3,2);การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

helpsub(30,12,6);
type1(50,22,"Type"," dir /p");
Check_Key(6,0,23,6,3,2);
helpsub(30,10,7);
type1(40,22,"Type"," dir c:\\sodprac /w");
Check_Key(17,0,23,7,0,2);
helpsub(30,13,8);
type1(38,22,"Type"," dir c:\\sodprac\\*.bat");
Check_Key(20,0,23,8,0,2);
on=168;
helpsub(30,11,9);
type1(50,22,"Type"," tree ! more");
Check_Key(11,0,23,9,0,2);
thaixy(1,23,79,"C:\\>",0);
helpsub(35,12,10);
type1(47,22,"Press"," Page Down");
hot_key(0,23,250,0,81);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(bufold);
farfree(buftype1);}

```

```

if(cas<=3){
clearblock(1,1,81,24);
on=112;
helpsub(5,10,11);
type1(51,22,"Type"," md pracdir");
Check_Key(10,0,2,10,0,2);
on=10;
helpsub(30,13,12);

```

```

type1(51,22,"Type"," cd praddir");
Check_Key(10,0,4,11,0,2);
on=139;
helpsub(30,13,13);
type1(51,22,"Type"," rd praddir");
Check_Key(10,0,6,12,4,2);
helpsub(30,13,14);
type1(54,22,"Type"," cd\\");
Check_Key(3,0,10,13,4,2);
on=47;
helpsub(30,14,15);
type1(51,22,"Type"," dir /p");
Check_Key(6,0,12,14,0,2);
on=139;
helpsub(30,13,17);
type1(51,22,"Type"," rd praddir");
Check_Key(10,0,23,15,0,2);
on=47;
helpsub(30,13,18);
type1(52,22,"Type"," dir /p");
Check_Key(6,0,23,16,0,2);}

```

```

if(cas<=4){
clearblock(1,1,81,24);
on=126;
helpsub(5,10,20);
type1(50,22,"Type"," path");
Check_Key(4,0,2,17,0,2);
helpsub(30,10,21);
type1(50,22,"Type"," path;");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

st_10[]="MD PRACDIR",st_11[]="CD PRACDIR",st_12[]="RD PRACDIR",
st_13[]="CD\\",st_17[]="PATH",st_18[]="PATH;",st_20[]="HELP",
st_21[]="PATH C:\\DOS";

static char mix_2[3],mix_5[3],mix_6[6],mix_7[20],mix_8[20],
        mix_9[20],mix_10[20],mix_11[20],mix_12[10],mix_13[3],
        mix_14[20],mix_15[20],mix_16[20],mix_17[20],mix_18[20],
        mix_19[20],mix_20[20],mix_21[20],mix_22[20],mix_23[20];

switch(topic) {
    case 0 : for(loop=0;loop<=1;loop++){
        if(toupper(g[loop])==st_0[loop]) nst++;}
    if(nst==2) {
        cktp=0;
        if(buftype1!=NULL)
            putimage(coltype1,rowtype1,buftype1,0);
        type1(65,22,"Type"," Enter");
        hotkey(13);
        if(buftype1!=NULL)
            putimage(coltype1,rowtype1,buftype1,0);
        if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
        farfree(buftype1);
        farfree(bufold);
        thaixy(1,3,79,"Not read reading drive A",0);
        thaixy(1,4,79,"Abort, Retry, Fail?",0);
    }
    break;
    case 1 : if(toupper(g[0])==st_1[0])
        cktp=0;
        if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
        farfree(buftype1);
        if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
}

```

```

farfree(bufold);

break;

case 2 : for(loop=0;loop<=2;loop++){
    if(toupper(g[loop])==st_2[loop])
        {mix_2[loop]=g[loop]; nst++;}
if(nst==3) {
    cktp=0;
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    farfree(buftype1);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(bufold);
    clearblock(1,1,79,24);
    for(too=15;too>=1;too--){
        mix2(too+1,&mix_2,2);
        topic2_1(too+1);}
    }
break;

case 3 : for(loop=0;loop<=6;loop++){
    if(toupper(g[loop])==st_3[loop]) nst++;}
if(nst==7) {
    cktp=0;
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    farfree(buftype1);
    type1(65,22,"Press"," Enter");
    hotkey(13);
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
}

```

```

    farfree(buftype1);
    farfree(bufold);
}

break;

case 4 : for(loop=0;loop<=2;loop++){
    if(toupper(g[loop])==st_4[loop]) nst++;}
if(nst==3) {
    cktp=0;
    restore1();
    Cint(inter);
    gint=1;
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    farfree(buftype1);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(bufold);
    type1(65,22,"Press"," Enter");
    hotkey(13);
    restore1();
    Cint(inter);
    gint=1;
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    farfree(buftype1);
    thaixy(1,21,79,"A:\\>",0);
    type1(47,22,"Press"," Page Down ");
    hot_key(0,21,250,0,81);
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    farfree(buftype1);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        clearblock(1,1,79,24);
    }

    break;

case 5 : for(loop=0;loop<=2;loop++){
        if(toupper(g[loop])==st_2[loop])
            {mix_5[loop]=g[loop]; nst++;}
    if(nst==3) {
        cktp=0;
        if(buftype1!=NULL)
            putimage(coltype1,rowtype1,buftype1,0);
        farfree(buftype1);
        type1(65,22,"Press"," Enter");
        hotkey(13);
        if(buftype1!=NULL)
            putimage(coltype1,rowtype1,buftype1,0);
        if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
        farfree(buftype1);
        farfree(bufold);
        for(loop=0;loop<=14;loop++){
            mix2(-loop+2,&mix_5,5);
            topic2_2(-loop+2);        }
        delay(1000);
    }

    break;

case 6 : for(loop=0;loop<=5;loop++){
        if(toupper(g[loop])==st_6[loop])
            {mix_6[loop]=g[loop]; nst++;}
    if(nst==6) {
        cktp=0;
        if(buftype1!=NULL)

```

```

        putimage(coltype1,rowtype1,buftype1,0);
        farfree(buftype1);
        type1(65,22,"Press"," Enter");
        hotkey(13);
        if(buftype1!=NULL)
            putimage(coltype1,rowtype1,buftype1,0);
        if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
        farfree(buftype1);
        farfree(bufold);
        for(too=0;too<=22;too++){
            topic2_2(-too-12,1);
            mix2(-too+23,&mix_6,6);
            topic2_2(-too+23,0);}
        type1(60,22,""," Press any key");
        Gkey();
        putimage(coltype1,rowtype1,buftype1,0);
        for(too=2;too<=17;too++){
            mix2(-too+2,&mix_6,6);
            topic2_2(-too+2,0);}
    }

```

break;

```

case 7 : for(loop=0;loop<=16;loop++){
        if(toupper(g[loop])==st_7[loop])
            {mix_7[loop]=g[loop]; nst++;}
        if(nst==17) {
            cktp=0;
            if(buftype1!=NULL)
                putimage(coltype1,rowtype1,buftype1,0);
            farfree(buftype1);

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของโรงเรียนอัสสัมชัญกรุงเทพฯ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

hotkey(13);
if(buftype1!=NULL)
    putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
for(too=0;too<=14;too++){
    topic2_2(-too-15,0);
    mix2(-too+23,&mix_7,7);
    topic2_21(-too+23);}
delay(1000);
}
break;
case 8 : for(loop=0;loop<=19;loop++){
    if(toupper(g[loop])==st_8[loop])
        {mix_8[loop]=g[loop]; nst++;}
if(nst==20) {
    cktp=0;
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    farfree(buftype1);
    type1(65,22,"Press"," Enter");
    hotkey(13);
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(buftype1);
    farfree(bufold);
    for(too=14;too<=26;too++){

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    mix2(-too+23,&mix_7,7);
    topic2_21(-too+23);
    mix2(37-too,&mix_8,8);
    topic2_22(37-too);}
    delay(1000);
}

```

```
break;
```

```

case 9 : for(loop=0;loop<=10;loop++){
    if(toupper(g[loop])==st_9[loop])
        {mix_9[loop]=g[loop]; nst++;}
    if(nst==11) {
        cktp=0;
        if(buftype1!=NULL)
            putimage(coltype1,rowtype1,buftype1,0);
        farfree(buftype1);
        type1(65,22,"Press"," Enter");
        hotkey(13);
        if(buftype1!=NULL)
            putimage(coltype1,rowtype1,buftype1,0);
        if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
        farfree(buftype1);
        farfree(bufold);
        for(too=0;too<=24;too++){
            topic2_21(-too-3);
            mix2(11-too,&mix_8,8);
            topic2_22(11-too);
            mix2(-too+23,&mix_9,9);
            topic2_23(-too+24);}
        for(loop=0;loop<=1;loop++){
            type1(56,22,""," Press any key");

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของงานที่ออกโดยสำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ ให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Gkey();
putimage(coltype1,rowtype1,buftype1,0);
for(too=0;too<=23;too++)
    topic2_23(-too-(loop*24)-1);
    }

type1(56,22,""," Press any key");
Gkey();
putimage(coltype1,rowtype1,buftype1,0);
for(too=0;too<=16;too++)
    topic2_23(-too-51);
    }
break;
case 10 : for(loop=0;loop<=9;loop++){
    if(toupper(g[loop])==st_10[loop])
        {mix_10[loop]=g[loop]; nst++;}
    if(nst==10) {
        keep();
        mix2(2,&mix_10,10);
        cktp=0; }
    break;
case 11 : for(loop=0;loop<=9;loop++){
    if(toupper(g[loop])==st_11[loop])
        {mix_11[loop]=g[loop]; nst++;}
    if(nst==10) {
        keep();
        mix2(4,&mix_11,11);
        cktp=0; }
    break;
case 12 : for(loop=0;loop<=9;loop++){

```

เอกสารนี้เป็นเอกสารที่ if(toupper(g[loop])==st_12[loop]) อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        {mix_12[loop]=g[loop]; nst++;}}
if(nst==10) {
    keep();
    mix2(6,&mix_12,12);
    cktp=0; }
break;
case 13 : for(loop=0;loop<=2;loop++){
    if(toupper(g[loop])==st_13[loop])
        {mix_13[loop]=g[loop]; nst++;}}
if(nst==3) {
    keep();
    mix2(10,&mix_13,13);
    cktp=0; }
break;
case 14 : for(loop=0;loop<=5;loop++){
    if(toupper(g[loop])==st_6[loop])
        {mix_14[loop]=g[loop]; nst++;}}
if(nst==6) {
    cktp=0;
    keep();
    for(too=0;too<=10;too++){
        mix2(-too+1,&mix_10,10);
        mix2(-too+3,&mix_11,11);
        mix2(-too+5,&mix_12,12);
        mix2(-too+9,&mix_13,13);
        mix2(-too+11,&mix_14,14);
        topic2_3(-too+12,1);}
    helpsub(35,13,15);
    type1(65,22,"Press", " CTRL+C");

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

setint(7);gint=0;
hot_key(0,0,0,1,3);
restore1();
Cint(inter);
gint=1;
if(buftype1!=NULL)
    putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
for(too=1;too<=3;too++){
    topic2_3(-too+2,1);
    thaixy(1,24-too,20,"^C",0);
    thaixy(1,25-too,20," ",0);}
break;
case 15 : for(loop=0;loop<=9;loop++){
    if(toupper(g[loop])==st_12[loop])
        {mix_15[loop]=g[loop]; nst++;}
if(nst==10) {
    cktp=0;
    keep();
    for(too=1;too<=2;too++){
        topic2_3(-too-1,1);
        thaixy(1,-too+21,20,"^C",0);
        thaixy(1,-too+22,20," ",0);
        mix2(-too+23,&mix_15,15);}
    }
break;

```

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(toupper(g[loop])==st_6[loop])
    {mix_16[loop]=g[loop]; nst++;}
if(nst==6) {
    cktp=0;
    keep();
    for(too=1;too<=22;too++){
        topic2_3(-too-3,1);
        thaixy(1,19-too,20,"^C",0);
        thaixy(1,20-too,20,"  ",0);
        mix2(-too+21,&mix_15,15);
        mix2(-too+23,&mix_16,16);
        topic2_3(-too+24,0);}
    helpsub(30,13,19);
    type(60,22,"Press"," Page Down");
    hot_key(0,0,0,1,81);
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(buftype1);
    farfree(bufold);
    }
break;
case 17 : for(loop=0;loop<=3;loop++){
    if(toupper(g[loop])==st_17[loop])
        {mix_17[loop]=g[loop]; nst++;}
    if(nst==4) {
        keep();
        mix2(2,&mix_17,17);
        thaixy(1,3,60,"PATH=C:\\WINDOWS;C:\\DOS;/"

```

```

        thairy(1,4,60,"
        cktp=0; }
break;
case 18 : for(loop=0;loop<=4;loop++){
        if(toupper(g[loop])==st_18[loop])
            {mix_18[loop]=g[loop]; nst++;}
if(nst==5) {
        if(buftype1!=NULL)
            putimage(coltype1,rowtype1,buftype1,0);
        type1(65,22,"Type"," Enter");
        hotkey(13);
        if(buftype1!=NULL)
            putimage(coltype1,rowtype1,buftype1,0);
        if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
        farfree(buftype1);
        farfree(bufold);
        mix2(5,&mix_18,18);
        cktp=0;
        }
break;
case 19 : for(loop=0;loop<=3;loop++){
        if(toupper(g[loop])==st_17[loop])
            {mix_19[loop]=g[loop]; nst++;}
if(nst==4) {
        if(buftype1!=NULL)
            putimage(coltype1,rowtype1,buftype1,0);
        type1(65,22,"Type"," Enter");
        hotkey(13);
        if(buftype1!=NULL)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
        farfree(buftype1);
        farfree(bufold);
        mix2(7,&mix_19,19);
        thaixy(1,8,60,"No path",0);
        thaixy(1,9,60,"",0);
        cktp=0;
    }

    break;

case 20 : for(loop=0;loop<=3;loop++){
        if(toupper(g[loop])==st_20[loop])
            {mix_20[loop]=g[loop]; nst++;}
        if(nst==4) {
            if(buftype1!=NULL)
                putimage(coltype1,rowtype1,buftype1,0);
            type1(65,22,"Type"," Enter");
            hotkey(13);
            if(buftype1!=NULL)
                putimage(coltype1,rowtype1,buftype1,0);
            if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
            farfree(buftype1);
            farfree(bufold);
            mix2(10,&mix_20,20);
            thaixy(1,11,60,"Bad command or file name",0);
            thaixy(1,12,60,"",0);
            cktp=0;
        }

        break;

case 21 : for(loop=0;loop<=10;loop++){

```

เอกสารนี้เป็นเอกสารที่ `if(toupper(g[loop])==st_21[loop])` อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        { mix_21[loop]=g[loop]; nst++;}}
if(nst==11) {
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    type1(65,22,"Type"," Enter");
    hotkey(13);
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(buftype1);
    farfree(bufold);
    mix2(13,&mix_21,21);
    cktp=0;
}
break;
case 22 : for(loop=0;loop<=3;loop++){
    if(toupper(g[loop])==st_17[loop])
        {mix_22[loop]=g[loop]; nst++;}}
if(nst==4) {
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    type1(65,22,"Type"," Enter");
    hotkey(13);
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(buftype1);
    farfree(bufold);
    mix2(15,&mix_22,22);

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของ ThaiXy (1,16,60,"PATH=C:\DOS" นั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์); การค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    thaixy(1,17,60,"",0);
    cktp=0;
    }
break;
case 23 : for(loop=0;loop<=3;loop++){
    if(toupper(g[loop])==st_20[loop])
        {mix_23[loop]=g[loop]; nst++;}
if(nst==4) {
    cktp=0;
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    type1(65,22,"Type"," Enter");
    hotkey(13);
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(buftype1);
    farfree(bufold);
    for(too=0;too<=16;too++){
        mix2(-too+1,&mix_17,17);
        thaixy(1,-too+2,60,"PATH=C:\\WINDOWS;C:\\DOS;C:\\\
            QPRO;C:\\UTIL;C:\\WP51\\ ",0);
        thaixy(1,-too+3,60,"",0);
        mix2(-too+4,&mix_18,18);
        mix2(-too+6,&mix_19,19);
        thaixy(1,-too+7,60,"No path",0);
        thaixy(1,-too+8,60,"",0);
        mix2(-too+9,&mix_20,20);
        thaixy(1,-too+10,60,"Bad command or file name ",0);
        thaixy(1,-too+11,60,"",0);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์อื่นใด การคัดลอกหรือการเผยแพร่โดยไม่ได้รับอนุญาตถือว่าผิดกฎหมาย และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mix2(-too+12,&mix_21,21);
mix2(-too+14,&mix_22,22);
thaixy(1,-too+15,60,"PATH=C:\DOS",0);
thaixy(1,-too+16,60,"",0);
mix2(-too+17,&mix_23,23);
topic1_22(-too+17,1);
}

helpsub(30,12,27);
type1(65,22,"Type"," CTRL+C");
hot_key(0,0,0,1,3);
if(buftype1!=NULL)
    putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
}
break;
}
}

```

/* ฟังก์ชัน mix เป็นการนำข้อความที่เก็บไว้มาแสดงที่หน้าจอ ในลักษณะที่เลื่อนขึ้นๆเรื่อยๆ โดยจะรับค่าบรรทัด ข้อความและหัวข้อที่ต้องการให้แสดง */

```

mix2(int y,char *dt,int cas)
{
    if((cas!=2)|| (cas!=5)|| (cas!=6)|| (cas!=12)|| (cas!=13)){
        thaixy(1,y,10,"C:\\>",0);
        thaixy(5,y,23,dt,0);}
    if(cas==2){
        thaixy(1,y,4,"A:\\>",0);
        thaixy(5,y,7,dt,0);}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if((cas==5)||!(cas==6)){
    thaixy(1,y,10,"C:\\SODPRAC>",0);
    thaixy(12,y,18,dt,0);
    if(cas==6) thaixy(16,y,50,&dt[4],0);
    }
if((cas==12)||!(cas==13)) {
    thaixy(1,y,10,"C:\\PRACDIR>",0);
    thaixy(12,y,13,dt,0);
    if(cas==12){
        thaixy(15,y,20,&dt[3],0);
        thaixy(1,y+1,50,"Invalid path, not directory,    ",0);
        thaixy(1,y+2,50,"or directory not empty    ",0);
        thaixy(1,y+3,79,"    ",0);
    }
}
if(cas!=12) thaixy(1,y+1,79,"    ",0);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/*****
***** PROGRAM TOPIC3.C *****/
*****/

#include<alloc.h>
extern char gt[30];
extern unsigned int on;
extern int cktp,xbufold,ybufold,coltype1,rowtype1,coltype2,rowtype2,
        inter,gint;
extern char far *pointerindex;
extern char far *bufold;
extern char far *buftype1,*buftype2;
void topic_3(int cas);
void Check_3(char *g,int topic);
void keep(void);

/* ฟังก์ชัน topic_3 เป็นการเรียนในบทที่ 3 หัวข้อที่ 1-3 */
void topic_3(int cas)
{
    barband();
    gint=1;
    openfile("topic3.hlp");
    if(cas==1){
        clearblock(1,1,81,24);
        on=47;
        helpsub(5,11,1);
        type1(52,22,"Type"," dir");
        Check_Key(3,0,2,1,3,3);
        on=27;

```

เอกสาร helpsub(35,13,2); สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

type1(30,22,"Type"," copy some.txt some.txt");
Check_Key(22,0,23,2,3,3);
helpsub(35,13,3);
type1(30,22,"Type"," copy some.txt new.txt");
inter=8;
setint(inter);gint=0;
Check_Key(21,0,23,3,3,3);
helpsub(30,13,4);
inter=9;
setint(inter);gint=0;
type1(37,22,"Type"," fc some.txt new.txt");
Check_Key(19,0,23,4,3,3);
inter=10;
setint(inter);gint=0;
on=47;
helpsub(30,13,5);
type1(50,22,"Type"," dir /w");
Check_Key(6,0,23,5,3,3);
on=144;
helpsub(36,13,6);
inter=11;
setint(inter);gint=0;
type1(38,22,"Type"," ren duck.doc bird.doc");
Check_Key(21,0,23,6,3,3);
on=47;
helpsub(30,13,7);
type1(50,22,"Type"," dir /w");
Check_Key(6,0,23,7,3,3);
on=112;
helpsub(30,12,8);

```

เอกสารนี้เป็นลิขสิทธิ์ของมหาวิทยาลัยราชภัฏวไลยอลงกรณ์ จังหวัดปทุมธานี เพื่อการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

inter=12;
setint(inter);gint=0;
type1(48,22,"Type"," md txtfiles");
Check_Key(11,0,23,8,3,3);
on=10;
helpsub(30,13,9);
type1(46,22,"Type"," cd txtfiles");
Check_Key(11,0,23,9,3,3);
on=27;
helpsub(30,11,10);
type1(36,22,"Type"," copy c:\\sodprac\\*.txt");
Check_Key(21,0,23,10,5,3);
on=47;
helpsub(30,13,11);
type1(50,22,"Type"," dir");
Check_Key(3,0,23,11,5,3);
helpsub(30,11,12);
type1(40,22,"Type"," dir file?.txt");
Check_Key(13,0,23,12,5,3);
thaixy(1,23,23,"C:\\SODPRAC\\TXTFILES>",0);
helpsub(30,13,13);
type1(48,22,"Press"," Page Down");
hot_key(16,23,250,0,81);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);}

if(cas<=2){

```

```

helpsub(5,10,14);
type1(51,22,"Type"," dir");
Check_Key(3,0,2,13,3,3);
helpsub(30,13,15);
type1(47,22,"Type"," del *.txt");
Check_Key(9,0,23,14,3,3);
helpsub(30,13,16);
type1(50,22,"Type"," dir");
Check_Key(3,0,23,15,3,3);
on=168;
helpsub(30,12,17);
type1(46,22,"Type"," tree /f");
Check_Key(7,0,23,16,3,3);
on=42;
helpsub(30,12,18);
type1(34,22,"Type"," del c:\\txtfiles\\new.txt");
Check_Key(23,0,23,17,3,3);
on=47;
helpsub(30,12,19);
type1(40,22,"Type"," dir txtfiles");
Check_Key(12,0,23,18,3,3);
thaixy(1,23,23,"C:\\SODPRAC>",0);
helpsub(30,11,20);
type1(47,22,"Press"," Page Down");
hot_key(7,23,250,0,81);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(cas<=3){
clearblock(1,1,81,24);
on=112;
helpsub(5,9,21);
type1(44,22,"Type"," md newdir");
Check_Key(9,0,2,19,0,3);
on=10;
helpsub(30,13,22);
type1(44,22,"Type"," cd newdir");
Check_Key(9,0,4,20,0,3);
on=195;
helpsub(30,13,23);
type1(44,22,"Type"," xcopy /?");
Check_Key(8,0,6,21,6,3);
helpsub(30,11,24);
inter=13;
setint(inter);gint=0;
type1(36,22,"Type"," xcopy c:\\sodprac /s");
Check_Key(19,0,23,22,6,3);
on=47;
helpsub(30,7,25);
inter=14;
setint(inter);gint=0;
type1(50,22,"Type"," dir");
Check_Key(3,0,23,23,6,3);
on=42;
helpsub(30,13,26);
type1(48,22,"Type"," del *.*");
Check_Key(7,0,23,24,6,3);
on=47;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

helpsub(30,13,28);
type1(50,22,"Type"," dir");
Check_Key(3,0,23,25,6,3);
on=42;
helpsub(30,13,29);
type1(43,22,"Type"," del txtfiles");
inter=15;
setint(inter);gint=0;
Check_Key(12,0,23,26,6,3);
on=139;
helpsub(30,12,31);
type1(42,22,"Type"," rd txtfiles");
Check_Key(11,0,23,27,6,3);
on=47;
helpsub(30,13,32);
type1(50,22,"Type"," dir");
Check_Key(3,0,23,28,6,3);
thaixy(1,23,23,"C:\\NEWDIR>",0);
hot_key(6,23,250,0,81);}
farfree(pointerindex);
}

```

/* ฟังก์ชัน Check_3 เป็นการตรวจสอบตัวอักษรทั้งหมดว่า พิมพ์ตามที่กำหนดไว้หรือไม่ ถ้าไม่ช
ก็กลับไปพิมพ์ใหม่ หรือรอรับคีย์ที่ต้องการให้กดก็ได้ */

```

void Check_3(char *g,int topic)
{
int too,loop,nst=0;
char st_1[]="DIR",st_2[]="COPY SOME.TXT SOME.TXT",st_16[]="TREE /F",
st_3[]="COPY SOME.TXT NEW.TXT",st_4[]="FC SOME.TXT NEW.TXT",

```

st_5[]="DIR /W",st_6[]="REN DUCK.DOC BIRD.DOC",ให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

st_8[]="MD TXTFILES",st_9[]="CD TXTFILES",st_21[]="XCOPY /?",
st_10[]="COPY C:\\SODPRAC\\*.TXT",st_12[]="DIR FILE?.TXT",
st_14[]="DEL *.TXT",st_17[]="DEL C:\\TXTFILES\\NEW.TXT",
st_18[]="DIR TXTFILES",st_19[]="MD NEWDIR",st_20[]="CD NEWDIR",
st_22[]="XCOPY C:\\SODPRAC /S",st_24[]="DEL *.*",
st_26[]="DEL TXTFILES",st_27[]="RD TXTFILES";

static char mix_1[11],mix_2[22],mix_3[22],mix_4[22],mix_5[11],
mix_6[22],mix_7[11],mix_8[11],mix_9[11],mix_10[21],
mix_11[21],mix_12[21],mix_13[11],mix_14[11],mix_15[11],
mix_16[11],mix_17[23],mix_18[23],mix_19[12],mix_20[12],
mix_21[19],mix_22[19],mix_23[19],mix_24[19],mix_25[19],
mix_26[19],mix_27[19],mix_28[19];

switch(topic) {
case 1 : for(loop=0;loop<=2;loop++){
    if(toupper(g[loop])==st_1[loop])
        {mix_1[loop]=g[loop]; nst++;}
    if(nst==3) {
        cktp=0;
        keep();
        for(too=1;too<=3;too++){
            mix3(-too+2,&mix_1,1);
            topic3_1(-too+2,3); }
        }
        break;
case 2 : for(loop=0;loop<=21;loop++){
    if(toupper(g[loop])==st_2[loop])
        {mix_2[loop]=g[loop]; nst++;}
    if(nst==22) {
        cktp=0;

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        for(too=1;too<=4;too++){
            topic3_1(-too-1,3);
            mix3(-too+23,&mix_2,2); }
        }

    break;

case 3 : for(loop=0;loop<=20;loop++){
        if(toupper(g[loop])==st_3[loop])
            {mix_3[loop]=g[loop]; nst++;}
    if(nst==21) {
        cktp=0;
        keep();
        restore1();
        Cint(inter);
        gint=1;
        for(too=1;too<=3;too++){
            topic3_1(-too-5,3);
            mix3(-too+19,&mix_2,2);
            mix3(-too+23,&mix_3,3); }
        }

    break;

case 4 : for(loop=0;loop<=18;loop++){
        if(toupper(g[loop])==st_4[loop])
            {mix_4[loop]=g[loop]; nst++;}
    if(nst==19) {
        cktp=0;
        keep();
        restore1();
        Cint(inter);
        gint=1;

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของคณะวิศวกรรมศาสตร์เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

topic3_1(-too-8,3);
mix3(-too+16,&mix_2,2);
mix3(-too+20,&mix_3,3);
mix3(-too+23,&mix_4,4);}
}

```

```
break;
```

```

case 5 : for(loop=0;loop<=5;loop++){
    if(toupper(g[loop])==st_5[loop])
        {mix_5[loop]=g[loop]; nst++;}
if(nst==6) {
    cktp=0;
    keep();
    restore1();
    Cint(inter);
    gint=1;
    for(too=1;too<=13;too++){
        topic3_1(-too-12,3);
        mix3(-too+12,&mix_2,2);
        mix3(-too+16,&mix_3,3);
        mix3(-too+19,&mix_4,4);
        mix3(-too+23,&mix_5,5);
        topic3_12(-too+24,1); }
    }
}

```

```
break;
```

```

case 6 : for(loop=0;loop<=20;loop++){
    if(toupper(g[loop])==st_6[loop])
        {mix_6[loop]=g[loop]; nst++;}
if(nst==21) {
    cktp=0;
    keep();

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

restore1();
Cint(inter);
gint=1;
for(too=1;too<=2;too++){
    mix3(-too-1,&mix_2,2);
    mix3(-too+3,&mix_3,3);
    mix3(-too+6,&mix_4,4);
    mix3(-too+10,&mix_5,5);
    topic3_12(-too+11,1);
    mix3(-too+23,&mix_6,6);}
}
break;
case 7 : for(loop=0;loop<=5;loop++){
    if(toupper(g[loop])==st_5[loop])
        {mix_7[loop]=g[loop]; nst++;}
    if(nst==6) {
        cktp=0;
        keep();
        for(too=1;too<=13;too++){
            mix3(-too+1,&mix_3,3);
            mix3(-too+4,&mix_4,4);
            mix3(-too+8,&mix_5,5);
            topic3_12(-too+9,1);
            mix3(-too+21,&mix_6,6);
            mix3(-too+23,&mix_7,7);
            topic3_12(-too+24,0);}
        }
    }
break;
case 8 : for(loop=0;loop<=10;loop++){

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การเชิงพาณิชย์เพื่อการค้าเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        {mix_8[loop]=g[loop]; nst++;}}
if(nst==11) {
    cktp=0;
    keep();
    restore1();
    Cint(inter);
    gint=1;
    for(too=1;too<=2;too++){
        topic3_12(-too-4,1);
        mix3(-too+8,&mix_6,6);
        mix3(-too+10,&mix_7,7);
        topic3_12(-too+11,0);
        mix3(-too+23,&mix_8,8);}
    }
break;
case 9 : for(loop=0;loop<=10;loop++){
    if(toupper(g[loop])==st_9[loop])
        {mix_9[loop]=g[loop]; nst++;}}
if(nst==11) {
    cktp=0;
    keep();
    for(too=1;too<=2;too++){
        topic3_12(-too-6,1);
        mix3(-too+6,&mix_6,6);
        mix3(-too+8,&mix_7,7);
        topic3_12(-too+9,0);
        mix3(-too+21,&mix_8,8);
        mix3(-too+23,&mix_9,9);}
    }
}

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

case 10 : for(loop=0;loop<=20;loop++){
    if(toupper(g[loop])==st_10[loop])
        {mix_10[loop]=g[loop]; nst++;}
    if(nst==21) {
        cktp=0;
        keep();
        for(too=1;too<=10;too++){
            topic3_12(-too-8,1);
            mix3(-too+4,&mix_6,6);
            mix3(-too+6,&mix_7,7);
            topic3_12(-too+7,0);
            mix3(-too+19,&mix_8,8);
            mix3(-too+21,&mix_9,9);
            mix3(-too+23,&mix_10,10);
            topic3_13(-too+23,0);}
        }
        break;
case 11 : for(loop=0;loop<=2;loop++){
    if(toupper(g[loop])==st_1[loop])
        {mix_11[loop]=g[loop]; nst++;}
    if(nst==3) {
        cktp=0;
        keep();
        for(too=1;too<=18;too++){
            topic3_12(-too-3,0);
            mix3(-too+9,&mix_8,8);
            mix3(-too+11,&mix_9,9);
            mix3(-too+13,&mix_10,10);
            topic3_13(-too+13,1);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้ `mix3(-too+23,&mix_11,11);` มิ อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        topic3_1(-too+24,0);}
    }
    break;
case 12 : for(loop=0;loop<=12;loop++){
    if(toupper(g[loop])==st_12[loop])
        {mix_12[loop]=g[loop]; nst++;}
    if(nst==13) {
        cktp=0;
        keep();
        for(too=1;too<=12;too++){
            topic3_13(-too-5,1);
            mix3(-too+5,&mix_11,11);
            topic3_1(-too+6,0);
            mix3(-too+23,&mix_12,12);
            topic3_1(-too+24,1);}
        }
        break;
case 13 : for(loop=0;loop<=2;loop++){
    if(toupper(g[loop])==st_1[loop])
        {mix_13[loop]=g[loop]; nst++;}
    if(nst==3) {
        cktp=0;
        keep();
        for(too=1;too<=6;too++){
            mix3(-too+2,&mix_13,13);
            topic3_1(-too+3,2); }
        }
        break;
case 14 : for(loop=0;loop<=8;loop++){

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์และห้ามมิให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        {mix_14[loop]=g[loop]; nst++;}}
if(nst==9) {
    cktp=0;
    keep();
for(too=1;too<=2;too++){
    topic3_1(-too-3,2);
    mix3(-too+23,&mix_14,14);}
    }
break;
case 15 : for(loop=0;loop<=2;loop++){
if(toupper(g[loop])==st_1[loop])
    {mix_15[loop]=g[loop]; nst++;}}
if(nst==3) {
    cktp=0;
    keep();
for(too=1;too<=20;too++){
    topic3_1(-too-5,2);
    mix3(-too+21,&mix_14,14);
    mix3(-too+23,&mix_15,15);
    topic3_1(-too+24,4);}
    }
break;
case 16 : for(loop=0;loop<=6;loop++){
    if(toupper(g[loop])==st_16[loop])
        {mix_16[loop]=g[loop]; nst++;}}
if(nst==7) {
    cktp=0;
    keep();
for(too=1;too<=21;too++){

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ (mix3(-too+1,&mix_14,14);) นั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        mix3(-too+3,&mix_15,15);
        topic3_1(-too+4,4);
        mix3(-too+23,&mix_16,16);
        topic3_2(-too+23);}
    }

    break;

case 17 : for(loop=0;loop<=22;loop++){
        if(toupper(g[loop])==st_17[loop])
            {mix_17[loop]=g[loop]; nst++;}}
    if(nst==23) {
        cktp=0;
        keep();
        for(too=1;too<=2;too++){
            topic3_2(-too+2);
            mix3(-too+23,&mix_17,17);}
        }
    break;

case 18 : for(loop=0;loop<=11;loop++){
        if(toupper(g[loop])==st_18[loop])
            {mix_18[loop]=g[loop]; nst++;}}
    if(nst==12) {
        cktp=0;
        keep();
        for(too=1;too<=17;too++){
            topic3_2(-too);
            mix3(-too+21,&mix_17,17);
            mix3(-too+23,&mix_18,18);
            topic3_1(-too+24,5);}
        }
    break;

```

```

case 19 : for(loop=0;loop<=8;loop++){
    if(toupper(g[loop])==st_19[loop])
        {mix_19[loop]=g[loop]; nst++;}
    if(nst==9){
        cktp=0;
        keep(); }
    break;
case 20 : for(loop=0;loop<=8;loop++){
    if(toupper(g[loop])==st_20[loop])
        {mix_20[loop]=g[loop]; nst++;}
    if(nst==9) {cktp=0;keep();}
    break;
case 21 : for(loop=0;loop<=7;loop++){
    if(toupper(g[loop])==st_21[loop])
        {mix_21[loop]=g[loop]; nst++;}
    if(nst==8) {
        cktp=0;
        keep();
        for(too=1;too<=2;too++){
            mix3(-too+2,&mix_19,19);
            mix3(-too+4,&mix_20,20);
            mix3(-too+6,&mix_21,21);
            topic3_3(-too+7);}
        }
    break;
case 22 : for(loop=0;loop<=18;loop++){
    if(toupper(g[loop])==st_22[loop])
        {mix_22[loop]=g[loop]; nst++;}
    if(nst==19) {

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

keep();
for(too=1;too<=18;too++){
    restore1();
    Cint(inter);
    gint=1;
    mix3(-too+2,&mix_20,20);
    mix3(-too+4,&mix_21,21);
    topic3_3(-too+5);
    mix3(-too+23,&mix_22,22);
    topic3_31(-too+23,0);}
}
break;
case 23 : for(loop=0;loop<=2;loop++){
    if(toupper(g[loop])==st_1[loop])
        {mix_23[loop]=g[loop]; nst++;}
    if(nst==3) {
        cktp=0;
        keep();
        restore1();
        Cint(inter);
        gint=1;
        for(too=1;too<=20;too++){
            topic3_3(-too-13);
            mix3(-too+5,&mix_22,22);
            topic3_31(-too+5,1);
            mix3(-too+23,&mix_23,23);
            topic3_1(-too+24,6);}
        }
    break;

```

เอกสารนี้เป็นลิขสิทธิ์ของสำนักงานส่งเสริมการค้าในต่างประเทศ ณ นครเชียงใหม่
 case 24 : for(loop=0;loop<=6;loop++){ เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(toupper(g[loop])==st_24[loop])
    {mix_24[loop]=g[loop]; nst++;}
if(nst==7) {
    cktp=0;
    keep();
    for(too=1;too<=2;too++){
        topic3_31(-too-17,1);
        mix3(-too+3,&mix_23,23);
        topic3_1(-too+4,6);
        mix3(-too+23,&mix_24,24);
        topic3_32(-too+23,1);}
    helpsub(35,10,27);
    type1(52,22,"Press"," y");
    hot_key(14,23,100,0,121);
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(buftype1);
    farfree(bufold);
    for(too=1;too<=2;too++){
        topic3_1(-too+2,6);
        mix3(-too+21,&mix_24,24);
        topic3_32(-too+21,0);}
    }
break;
case 25 : for(loop=0;loop<=2;loop++){
    if(toupper(g[loop])==st_1[loop])
        {mix_25[loop]=g[loop]; nst++;}
    if(nst==3) {

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

keep();
for(too=1;too<=12;too++){
    topic3_1(-too,6);
    mix3(-too+19,&mix_24,24);
    topic3_32(-too+19,0);
    mix3(-too+23,&mix_25,25);
    topic3_1(-too+24,7);}
}

break;

case 26 : for(loop=0;loop<=11;loop++){
    if(toupper(g[loop])==st_26[loop])
        {mix_26[loop]=g[loop]; nst++;}
if(nst==12) {
    cktp=0;
    keep();
    restore1();
    Cint(inter);
    gint=1;
    for(too=1;too<=2;too++){
        topic3_1(-too-12,6);
        mix3(-too+7,&mix_24,24);
        topic3_32(-too+7,0);
        mix3(-too+11,&mix_25,25);
        topic3_1(-too+12,7);
        mix3(-too+23,&mix_26,26);
        topic3_32(-too+23,1);}
    helpsub(30,11,30);
    typeI(52,22,"Press"," y");
    hot_key(14,23,100,0,121);
    if(buftype1!=NULL)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(buftype1);
    farfree(bufold);
    for(too=1;too<=2;too++){
        topic3_1(-too-14,6);
        mix3(-too+5,&mix_24,24);
        topic3_32(-too+5,0);
        mix3(-too+9,&mix_25,25);
        topic3_1(-too+10,7);
        mix3(-too+21,&mix_26,26);
        topic3_32(-too+21,0);}
    break;
case 27 : for(loop=0;loop<=10;loop++){
    if(toupper(g[loop])==st_27[loop])
        {mix_27[loop]=g[loop]; nst++;}
    if(nst==11) {
        cktp=0;
        keep();
        for(too=1;too<=2;too++){
            topic3_1(-too-16,6);
            mix3(-too+3,&mix_24,24);
            topic3_32(-too+3,0);
            mix3(-too+7,&mix_25,25);
            topic3_1(-too+8,7);
            mix3(-too+19,&mix_26,26);
            topic3_32(-too+19,0);
            mix3(-too+23,&mix_27,27);}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบุคลากรในหน่วยงานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        break;
    case 28 : for(loop=0;loop<=2;loop++){
                if(toupper(g[loop])==st_1[loop])
                    {mix_28[loop]=g[loop]; nst++;}
            if(nst==3) {
                cktp=0;
                keep();
                for(too=1;too<=11;too++){
                    mix3(-too+1,&mix_24,24);
                    topic3_32(-too+1,0);
                    mix3(-too+5,&mix_25,25);
                    topic3_1(-too+6,7);
                    mix3(-too+17,&mix_26,26);
                    topic3_32(-too+17,0);
                    mix3(-too+21,&mix_27,27);
                    mix3(-too+23,&mix_28,28);
                    topic3_1(-too+24,8);}
                break;
            }
    }
}

```

/* ฟังก์ชัน mix เป็นการนำข้อความที่เก็บไว้มาแสดงที่หน้าจอ บนลักษณะที่เลื่อนขึ้นเบ เรือข ๆ
 ุคขจะรับค่าบรรทัด ข้อความและหัวข้อที่ต้องการให้แสดง */.

```

mix3(int y,char *dt,int cas)

```

```

{
    if(cas<19){
        thaixy(1,y,10,"C:\\SODPRAC>",0);
        thaixy(12,y,21,dt,0);}

```

เอกสารนี้ **else** { if((cas==19) || (cas==20)) { ศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

thaixy(1,y,10,"C:\\>",0);
thaixy(5,y,17,dt,0);}
else { thaixy(1,y,10,"C:\\NEWDIR>",0);
      thaixy(11,y,28,dt,0);}
}

if((cas==17)!!(cas==18)) thaixy(12,y,33,dt,0);
if((cas==10)!!(cas==11)!!(cas==12)) {
  thaixy(1,y,25,"C:\\SODPRAC\\TXTFILES>",0);
  thaixy(21,y,40,dt,0);}
if((cas!=2)&&(cas!=3)&&(cas!=4)) {
  thaixy(1,y+1,79,"",0);}
if((cas==2)!!(cas==3)!!(cas==4)!!(cas==6)) {
  thaixy(12,y,32,dt,0);
  if((cas==2)!!(cas==4)){
    if(cas==2) {
      thaixy(1,y+1,79,"File cannot be copied onto itself ",0);
      thaixy(1,y+2,79,"      0 file(s) copied      ",0);
    }
  }
  if(cas==4) {
    thaixy(1,y+1,79,"Comparing files SOME.TXT and NEW.TXT",0);
    thaixy(1,y+2,79,"FC: no differences encountered      ",0);}
    thaixy(1,y+3,79,"",0);
  }
  if(cas==3) {
    thaixy(1,y+1,79,"      1 file(s) copied      ",0);
    thaixy(1,y+2,79,"",0);
  }
}
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

/* ฟังก์ชัน keep เป็นการเก็บคำสั่งที่ต้องการใช้บ่อย ๆ เอาไว้เพื่อสะดวก และรวดเร็วในการเขียนโปรแกรม ซึ่งฟังก์ชันนี้จะมีการเรียกใช้มากกว่าแต่ละบทเรียน */

```
void keep(void)
{
    if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
    farfree(buftype1);
    type1(65,22,"Press"," Enter");
    hotkey(13);
    if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(buftype1);
    farfree(bufold);
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/*****
***** PROGRAM TOPIC4.C *****/
*****/

```

```

#include<alloc.h>

extern char gt[30];

extern unsigned int on;

extern int cktp,xbufold,ybufold,coltype1,rowtype1,coltype2,rowtype2,
        inter,gint;

extern char *pointerindex;

extern char far *bufold;

extern char far *buftype1,*buftype2;

void topic_4(int cas);

void Check_4(char *g,int topic);

/* ฟังก์ชัน topic_4 เป็นการเรียนานบทที่ 4 หัวข้อที่ 1-3 */

void topic_4(int cas)
{
    barband();
    gint=1;
    openfile("topic4.hlp");
    if(cas==1){
        clearblock(1,1,81,24);
        on=16;
        helpsub(5,8,1);
        type1(48,22,"Type","  chkdsk");
        Check_Key(6,0,2,1,0,4);
        on=47;
        type1(48,22,"Type","  dir /o");
        helpsub(30,13,2);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

inter=16;
setint(inter);gint=0;
Check_Key(6,0,20,2,0,4);
helpsub(30,13,3);
type1(46,22,"Type"," dir /o /a");
Check_Key(9,0,23,3,0,4);
on=1;
type1(48,22,"Type"," attrib");
helpsub(30,13,4);
inter=17;
setint(inter);gint=0;
Check_Key(6,0,23,4,0,4);
thaixy(1,23,23,"C:\\>",0);
helpsub(30,12,5);
type1(47,22,"Press"," Page Down");
hot_key(0,23,250,0,81);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);}

if(cas<=2){
clearblock(1,1,81,24);
on=47;
helpsub(5,9,6);
type1(40,22,"Type"," dir /o");
Check_Key(6,0,2,5,0,4);
helpsub(30,12,7);
inter=18;

```

```

type1(42,22,"Type"," dir c:\\dos /p");
Check_Key(13,0,23,6,0,4);
helpsub(30,12,8);
type1(46,22,"Type"," dir *.bat");
Check_Key(9,0,23,7,0,4);
on=172;
helpsub(30,11,9);
type1(44,22,"Type"," type autoexec.bat");
Check_Key(17,0,23,8,0,4);
type1(44,22,"Type"," type config.sys");
Check_Key(15,0,23,9,0,4);
helpsub(30,12,10);
type1(47,22,"Press"," Page Down");
thaixy(1,23,23,"C:\\>",0);
hot_key(0,23,250,0,81);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);}

if(cas<=3){
clearblock(1,1,81,24);
on=157;
helpsub(5,10,11);
type1(40,22,"Type"," set");
Check_Key(3,0,2,10,0,4);
on=135;
helpsub(30,13,12);
type1(40,22,"Type"," prompt $d");
Check_Key(9,0,10,11,0,4);

```

```

on=157;
helpsub(30,13,13);
type1(40,22,"Type"," set");
Check_Key(3,0,12,12,7,4);
thaixy(1,20,30,"Wed 01-01-1992",0);
helpsub(30,12,13);
type1(47,22,"Press"," Page Down");
inter=19;
setint(inter);gint=0;
hot_key(10,20,250,0,81);
restore1();
Cint(inter);
gint=1;
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);}
farfree(pointerindex);
}

```

/* ฟังก์ชัน Check_4 เป็นการตรวจสอบตัวอักษรที่พิมพ์ว่า พิมพ์ตามที่กำหนดไว้หรือไม่ ถ้าไม่ใช่
ก็กลับไปพิมพ์ใหม่ หรือรอรับคีย์ที่ต้องการให้กดก็ได้ */

```
void Check_4(char *g,int topic)
```

```
{
```

```
int too,loop,nst=0;
```

```
char st_1[]="CHKDSK",st_2[]="DIR /O",st_3[]="DIR /O /A",
```

```
st_4[]="ATTRIB",st_6[]="DIR C:\\DOS /P",st_7[]="DIR *.BAT",
```

```
st_8[]="TYPE AUTOEXEC.BAT",st_9[]="TYPE CONFIG.SYS",
```

```
st_10[]="SET",st_11[]="PROMPT $D";
```

```
static char mix_1[17],mix_2[17],mix_3[17],mix_4[17],mix_5[17],
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของสำนักงานส่งเสริมการค้าในต่างประเทศ ณ นครเชียงใหม่ โดยไม่อนุญาตให้เผยแพร่หรือใช้ซ้ำโดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mix_6[17],mix_7[17],mix_8[17],mix_9[17];
switch(topic) {
  case 1 : for(loop=0;loop<=5;loop++){
            if(toupper(g[loop])==st_1[loop])
              {mix_1[loop]=g[loop]; nst++;}}
            if(nst==6) {
              cktp=0;
              keep();
              topic4_1(3,0);
            }
            break;
  case 2 : for(loop=0;loop<=5;loop++){
            if(toupper(g[loop])==st_2[loop])
              {mix_2[loop]=g[loop]; nst++;}}
            if(nst==6) {
              cktp=0;
              keep();
              restore1();
              Cint(inter);
              gint=1;
              for(too=1;too<=61;too++){
                mix4(-too+2,&mix_1);
                topic4_1(-too+3,1);
                mix4(-too+20,&mix_2);
                topic4_11(-too+21,1);}
            }
            break;
  case 3 : for(loop=0;loop<=8;loop++){
            if(toupper(g[loop])==st_3[loop])
              {mix_3[loop]=g[loop]; nst++;}}

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์และสงวนสิทธิ์ในชื่อเอกสารทุกประการ ขออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(nst==9) {
    cktp=0;
    keep();
    for(too=1;too<=66;too++){
        topic4_11(-too-40,1);
        mix4(-too+23,&mix_3);
        topic4_11(-too+24,0);}
    }
break;
case 4 : for(loop=0;loop<=5;loop++){
    if(toupper(g[loop])==st_4[loop])
        {mix_4[loop]=g[loop]; nst++;}
if(nst==6) {
    cktp=0;
    keep();
    restore1();
    Cint(inter);
    gint=1;
    for(too=1;too<=23;too++){
        topic4_11(-too-42,0);
        mix4(-too+23,&mix_4);
        topic4_12(-too+23);}
    }
break;
case 5 : for(loop=0;loop<=5;loop++){
    if(toupper(g[loop])==st_2[loop])
        {mix_5[loop]=g[loop]; nst++;}
if(nst==6) {
    cktp=0;
    keep();

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for(too=1;too<=43;too++){
    mix4(-too+2,&mix_5);
    topic4_11(-too+3,1);}
    }
break;
case 6 : for(loop=0;loop<=12;loop++){
    if(toupper(g[loop])==st_6[loop])
        {mix_6[loop]=g[loop]; nst++;}
if(nst==13) {
    cktp=0;
    keep();
    restore1();
    Cint(inter);
    gint=1;
    for(too=1;too<=22;too++){
        topic4_11(-too-40,1);
        mix4(-too+23,&mix_6);
        topic4_2(-too+24);}
    getch();
    for(too=1;too<=24;too++){
        mix4(-too+1,&mix_6);
        topic4_2(-too+2);    }
    getch();
    for(loop=1;loop<=1;loop++){
        for(too=0;too<=24;too++)
            topic4_2(-too-22-(24*loop));
        getch();    }
    }
for(too=1;too<=25;too++)
    topic4_2(-too-70);

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของสำนักงานส่งเสริมการค้าในต่างประเทศ ณ นครเชียงใหม่ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

break;
case 7 : for(loop=0;loop<=8;loop++){
    if(toupper(g[loop])==st_7[loop])
        {mix_7[loop]=g[loop]; nst++;}
    if(nst==9) {
        .cktp=0;
        keep();
        for(too=1;too<=12;too++){
            topic4_2(-too-95);
            mix4(-too+23,&mix_7);
            topic4_11(-too+24,2);}
        }
    break;
case 8 : for(loop=0;loop<=16;loop++){
    if(toupper(g[loop])==st_8[loop])
        {mix_8[loop]=g[loop]; nst++;}
    if(nst==17) {
        cktp=0;
        keep();
        for(too=1;too<=13;too++){
            topic4_2(-too-107);
            mix4(-too+11,&mix_7);
            topic4_11(-too+12,2);
            mix4(-too+23,&mix_8);
            topic4_21(-too+23);}
        }
    break;
case 9 : for(loop=0;loop<=14;loop++){
    if(toupper(g[loop])==st_9[loop])

```

เอกสารนี้เป็นเอกสารที่สงวน {mix_9[loop]=g[loop]; nst++;} ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(nst==15) {
    cktp=0;
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    farfree(buftype1);
    type1(65,22,"Press"," Enter");
    hotkey(13);
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    farfree(buftype1);
    for(too=1;too<=9;too++){
        topic4_11(-too-1,2);
        mix4(-too+10,&mix_8);
        topic4_21(-too+10);
        mix4(-too+23,&mix_9);
        topic4_22(-too+23);}
    }
break;
case 10 : for(loop=0;loop<=2;loop++){
    if(toupper(g[loop])==st_10[loop]) nst++;}
if(nst==3) {
    cktp=0;
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    farfree(buftype1);
    type1(65,22,"Press"," Enter");
    hotkey(13);
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);

```

เอกสารนี้เป็นเอกสารที่สง if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        farfree(buftype1);
        farfree(bufold);
        topic4_3(2,1);}

    break;

case 11 : for(loop=0;loop<=8;loop++){
            if(toupper(g[loop])==st_11[loop]) nst++;}
            if(nst==9) {cktp=0; keep();}
            break;

case 12 : for(loop=0;loop<=2;loop++){
            if(toupper(g[loop])==st_10[loop]) nst++;}
            if(nst==3) {
                cktp=0;
                keep();
                topic4_3(12,0);}
            break;
        }
}

/* ฟังก์ชัน mix เป็นการนำข้อความที่เก็บไว้มาแสดงที่หน้าจอ ในลักษณะที่เลื่อนขึ้นไปเรื่อย ๆ
โดยจะรับค่าบรรทัด ข้อความและหัวข้อที่ต้องการให้แสดง */
mix4(int y,char *dt)
{
    thaixy(1,y,10,"C:\\>",0);
    thaixy(5,y,20,dt,0);
    thaixy(1,y+1,79,"",0);
}

```

```

/*****
***** PROGRAM TOPIC5.C *****/
*****/

```

```

#include<alloc.h>
extern char gt[30];
extern unsigned int on;
extern int cktp,xbufold,ybufold,coltype1,rowtype1,coltype2,rowtype2,
        gint,inter;
extern char far *pointerindex;
extern char far *bufold;
extern char far *buftype1,*buftype2;
void topic_5(int cas);
void Check_5(char *g,int topic);

/* ฟังก์ชัน topic_5 เป็นการเรียนานบทที่ 5 หัวข้อที่ 1-5 */
void topic_5(int cas)
{
    barband();
    gint=1;
    openfile("topic5.hlp");
    if(cas==1){
        clearblock(1,1,81,24);
        on=47;
        helpsub(5,9,1);
        type1(50,22,"Type"," dir a:");
        Check_Key(6,0,2,1,0,5);
        helpsub(30,13,2);
        type1(52,22,"Type"," a");
        Check_Key(1,15,6,2,1,5);

```

```

on=99;
helpsub(30,11,4);
type1(48,22,"Type"," format a:");
Check_Key(9,0,8,3,0,5);
helpsub(34,10,6);
type1(47,22,"Type"," practice 1");
Check_Key(10,42,16,4,1,5);
helpsub(30,13,7);
type1(52,22,"Type"," N");
Check_Key(1,17,23,5,1,5);
helpsub(30,13,8);
type1(50,22,"Type"," dir a:");
Check_Key(6,0,23,6,0,5);
helpsub(35,13,9);
type1(48,22,"Type"," label a:");
Check_Key(8,0,23,7,0,5);
helpsub(40,14,10);
type1(47,22,"Type"," tutorial");
Check_Key(8,42,23,8,1,5);
on=193;
helpsub(30,12,11);
type1(50,22,"Type"," vol a:");
Check_Key(6,0,23,9,0,5);
thaixy(1,23,23,"C:\\>",0);
helpsub(30,14,12);
type1(47,22,"Press"," Page Down");
hot_key(0,23,250,0,81);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);

```

```

farfree(bufold);}

if(cas<=2){
clearblock(1,1,81,24);
thaixy(1,2,20,"C:\\\>DIR A:",0);
topic5_2(3);
on=99;
helpsub(30,11,13);
type1(48,22,"Type"," format a:");
Check_Key(9,0,19,11,0,5);
helpsub(30,13,16);
type1(52,22,"Type"," N");
Check_Key(1,17,23,12,1,5);
on=47;
helpsub(30,13,17);
type1(50,22,"Type"," dir a:");
Check_Key(6,0,23,13,0,5);
on=186;
helpsub(30,13,18);
type1(48,22,"Type"," unformat a:");
Check_Key(11,0,23,14,0,5);
helpsub(40,15,19);
type1(52,22,"Type"," Y");
Check_Key(1,67,23,15,1,5);
on=47;
helpsub(30,13,20);
type1(50,22,"Type"," dir a:");
Check_Key(6,0,23,16,0,5);
helpsub(30,13,21);
type1(47,22,"Press"," Page Down");

```

```

thaixy(1,23,23,"C:\\>",0);
hot_key(0,23,250,0,81);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);}

```

```

if(cas<=3){
clearblock(1,1,81,24);
on=99;
helpsub(5,10,22);
type1(45,22,"Type"," format a: /s");
Check_Key(12,0,2,17,0,5);
helpsub(34,9,25);
type1(52,22,"Type"," N");
inter=22;
setint(inter);gint=0;
Check_Key(1,17,23,18,1,5);
on=47;
helpsub(30,15,26);
type1(50,22,"Type"," dir a:");
Check_Key(6,0,23,19,0,5);
helpsub(30,12,27);
type1(48,22,"Type"," dir a: /a");
Check_Key(9,0,23,20,0,5);
thaixy(1,23,23,"C:\\>",0);
helpsub(30,13,28);
type1(48,22,"Press"," Page Down");
hot_key(0,23,250,0,81);

```

```

if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);}

if(cas<=4){
clearblock(1,1,81,24);
on=16;
helpsub(5,9,29);
type1(49,22,"Type","chkdsk a:");
Check_Key(9,0,2,21,0,5);
on=66;
helpsub(30,13,30);
type1(47,22,"Type","diskcopy a: a:");
Check_Key(14,0,17,22,0,5);
helpsub(30,13,33);
type1(52,22,"Type","N");
Check_Key(1,24,23,23,1,5);
on=16;
helpsub(30,13,34);
type1(49,22,"Type","chkdsk a:");
Check_Key(9,0,23,24,0,5);
thaixy(1,23,23,"C:\\>",0);
helpsub(30,11,35);
type1(47,22,"Press","Page Down");
hot_key(0,23,250,0,81);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(cas<=5){
clearblock(1,1,81,24);
on=47;
helpsub(5,10,36);
type1(48,22,"Type"," dir a: /a");
Check_Key(9,0,2,25,0,5);
on=157;
helpsub(30,12,40);
type1(50,22,"Type"," set");
Check_Key(3,0,23,26,2,5);
thaixy(1,23,23,"A:\\>",0);
type1(47,22,"Press"," Page Down");
hot_key(0,23,250,0,81);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
farfree(buftype1);}
farfree(pointerindex);
}

/* ฟังก์ชัน Check_5 เป็นการตรวจสอบตัวอักษรที่พิมพ์ว่า พิมพ์ตามที่กำหนดไว้หรือไม่ ถ้าไม่เซ
ก็กลับมาพิมพ์ใหม่ หรือรอรับคีย์ที่ต้องการให้กดก็ได้ */
void Check_5(char *g,int topic)
{
int i,too,loop,nst=0;
char st_1[]="DIR A:",st_2[]="A",st_3[]="FORMAT A:",st_5[]="N",
st_4[]="PRACTICE 1",st_7[]="LABEL A:",st_8[]="TUTORIAL",
st_9[]="VOL A:",st_14[]="UNFORMAT A:",st_15[]="Y",
st_17[]="FORMAT A: /S",st_20[]="DIR A: /A",st_21[]="CHKDSK A:",
st_22[]="DISKCOPY A: A:",st_26[]="SET";
static char mix_1[14],mix_2[2],mix_3[14],mix_4[14],mix_5[2],
mix_6[14],mix_7[14],mix_8[14],mix_9[14],mix_11[14],
if(toupper(g[loop])==st_3[loop])
{mix_3[loop]=g[loop]; nst++;}
if(nst==9) {
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
cktp=0;
keep();
helpsub(40,15,5);
type1(65,22,"Type"," Enter");

```

```

topic5_11(8,0);
if(buftype1!=NULL)
    putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
topic5_12(15,1);
    }
break;
case 4 : for(loop=0;loop<=9;loop++){
    if(toupper(g[loop])==st_4[loop])
        {mix_4[loop]=g[loop]; nst++;}
if(nst==10) {
    cktp=0;
    keep();
    delay(1500);
    for(too=1;too<=3;too++){
        mix5(-too+2,&mix_1,1);
        topic5_1(-too+4,1);
        mix5(-too+6,&mix_2,2);
        mix5(-too+8,&mix_3,3);
        topic5_11(-too+8,1);
        topic5_12(-too+15,1);
        mix5(-too+16,&mix_4,4);
        topic5_12(-too+15,0);}
    }
break;
case 5 : if(toupper(g[0])==st_5[0]) {
    keep();

```

เอกสารนี้เป็นเอกสารที่ `mix_5[0]=g[0];` งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

cktp=0;
for(too=1;too<=3;too++){
    topic5_1(-too+1,1);
    mix5(-too+3,&mix_2,2);
    mix5(-too+5,&mix_3,3);
    topic5_11(-too+5,1);
    topic5_12(-too+12,1);
    mix5(-too+13,&mix_4,4);
    topic5_12(-too+12,0);
    mix5(-too+23,&mix_5,5);}
}
break;
case 6 : for(loop=0;loop<=5;loop++){
    if(toupper(g[loop])==st_1[loop])
        {mix_6[loop]=g[loop]; nst++;}
if(nst==6) {
    cktp=0;
    keep();
    for(too=1;too<=8;too++){
        mix5(-too+2,&mix_3,3);
        topic5_11(-too+2,1);
        topic5_12(-too+9,1);
        mix5(-too+10,&mix_4,4);
        topic5_12(-too+9,0);
        mix5(-too+20,&mix_5,5);
        mix5(-too+23,&mix_6,6);
        topic5_13(-too+24,0);}
    }
}

```

break;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 case 7 : for(loop=0;loop<=7;loop++){
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุดเบสลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(toupper(g[loop])==st_7[loop])
    {mix_7[loop]=g[loop]; nst++;}
if(nst==8) {
    cktp=0;
    keep();
    for(too=1;too<=3;too++){
        topic5_11(-too-6,1);
        topic5_12(-too+1,1);
        mix5(-too+2,&mix_4,4);
        topic5_12(-too+1,0);
        mix5(-too+12,&mix_5,5);
        mix5(-too+15,&mix_6,6);
        topic5_13(-too+16,0);
        mix5(-too+23,&mix_7,7);
        topic5_12(-too+23,2);}
    }
break;
case 8 : for(loop=0;loop<=7;loop++){
    if(toupper(g[loop])==st_8[loop])
        {mix_8[loop]=g[loop]; nst++;}
if(nst==8) {
    cktp=0;
    keep();
    for(too=1;too<=2;too++){
        topic5_12(-too-2,1);
        mix5(-too-1,&mix_4,4);
        topic5_12(-too-2,0);
        mix5(-too+9,&mix_5,5);
        mix5(-too+12,&mix_6,6);
        topic5_13(-too+13,0);

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของมหาวิทยาลัยราชภัฏวชิรเวศน์บุรีรัมย์ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        mix5(-too+20,&mix_7,7);
        topic5_12(-too+20,2);
        mix5(-too+23,&mix_8,8); }
    }

    break;

case 9 : for(loop=0;loop<=5;loop++){
    if(toupper(g[loop])==st_9[loop])
        {mix_9[loop]=g[loop]; nst++;}}
if(nst==6) {
    .cktp=0;
    keep();
    for(too=1;too<=5;too++){
        topic5_12(-too-4,1);
        mix5(-too-3,&mix_4,4);
        topic5_12(-too-4,0);
        mix5(-too+7,&mix_5,5);
        mix5(-too+10,&mix_6,6);
        topic5_13(-too+11,0);
        mix5(-too+18,&mix_7,7);
        topic5_12(-too+18,2);
        mix5(-too+21,&mix_8,8);
        mix5(-too+23,&mix_9,9);
        topic5_13(-too+24,1); }
    }

    break;

case 11 : for(loop=0;loop<=8;loop++){
    if(toupper(g[loop])==st_3[loop])
        {mix_11[loop]=g[loop]; nst++;}}

```

```

if(nst==9) {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ฅกทพ=0;
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

keep();
helpsub(40,15,19);
type1(65,22,"Press"," Enter");
topic5_21(19,0);
if(buftype1!=NULL)
    putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
thaixy(1,1,20,"C:\\>DIR A:",0);
topic5_2(2);
mix5(18,&mix_11,11);
topic5_21(18,2);
delay(1000);
for(too=1;too<=2;too++){
    topic5_2(-too);
    mix5(-too+16,&mix_11,11);
    topic5_21(-too+16,1);
    topic5_22(-too+21,1);
    topic5_12(-too+24,1);}
helpsub(37,12,15);
inter=20;gint=0;
setint(20);
type1(65,22,"Press"," Enter");
hot_key(42,23,100,0,13);
if(buftype1!=NULL)
    putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);

```

```

        restore1();
        Cint(inter);
        gint=1;
        delay(1500);
        for(too=1;too<=10;too++){
            topic5_2(-too-2);
            mix5(-too+14,&mix_11,11);
            topic5_21(-too+14,1);
            topic5_22(-too+19,1);
            topic5_12(-too+22,0);}
        }
    break;
case 12 : if(toupper(g[0])==st_5[0]) {
        keep();
        mix_12[0]=g[0];
        cktp=0;
        for(too=1;too<=3;too++){
            topic5_2(-too-12);
            mix5(-too+4,&mix_11,11);
            topic5_21(-too+4,1);
            topic5_22(-too+9,1);
            topic5_12(-too+12,0);
            mix5(-too+23,&mix_12,12);}
        }
    break;
case 13 : for(loop=0;loop<=5;loop++){
        if(toupper(g[loop])==st_1[loop])
            {mix_13[loop]=g[loop]; nst++;}
        if(nst==6) {

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

keep();
for(too=1;too<=8;too++){
    mix5(-too+1,&mix_11,11);
    topic5_21(-too+1,1);
    topic5_22(-too+6,1);
    topic5_12(-too+9,0);
    mix5(-too+20,&mix_12,12);
    mix5(-too+23,&mix_13,13);
    topic5_13(-too+24,0);}
}
break;
case 14 : for(loop=0;loop<=10;loop++){
    if(toupper(g[loop])==st_14[loop])
        {mix_14[loop]=g[loop]; nst++;}
if(nst==11) {
    cktp=0;
    keep();
    for(too=1;too<=3;too++){
        topic5_22(-too-2,1);
        topic5_12(-too+1,0);
        mix5(-too+12,&mix_12,12);
        mix5(-too+15,&mix_13,13);
        topic5_13(-too+16,0);
        mix5(-too+23,&mix_14,14);
        topic5_23(-too+24);}
    type1(65,22,"Press"," Enter");
    hot_key(0,0,0,1,13);
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);

```

```

for(too=1;too<=13;too++){
    topic5_12(-too-2,0);
    mix5(-too+9,&mix_12,12);
    mix5(-too+12,&mix_13,13);
    topic5_13(-too+13,0);
    mix5(-too+20,&mix_14,14);
    topic5_23(-too+21);}
delay(4000);
for(too=1;too<=6;too++){
    topic5_13(-too,0);
    mix5(-too+7,&mix_14,14);
    topic5_23(-too+8);}
}
break;
case 15 : if(toupper(g[0])==st_15[0]) {
    keep();
    mix_15[0]=g[0];
    cktp=0;
    delay(1500);
    for(too=1;too<=6;too++){
        mix5(-too+1,&mix_14,14);
        topic5_23(-too+2);
        mix5(-too+23,&mix_15,15);
        topic5_24(-too+24);}
    }
break;
case 16 : for(loop=0;loop<=5;loop++){
    if(toupper(g[loop])==st_1[loop])
        {mix_16[loop]=g[loop]; nst++;}

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของสำนักงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

cktp=0;
keep();
for(too=1;too<=17;too++){
    topic5_23(-too-4);
    mix5(-too+17,&mix_15,15);
    topic5_24(-too+18);
    mix5(-too+23,&mix_16,16);
    topic5_2(-too+24);}
}
break;
case 17 : for(loop=0;loop<=11;loop++){
    if(toupper(g[loop])==st_17[loop])
        {mix_17[loop]=g[loop]; nst++;}
if(nst==12) {
    cktp=0;
    keep();
    helpsub(40,15,23);
    type1(65,22,"Press"," Enter");
    topic5_21(2,0);
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(buftype1);
    farfree(bufold);
    delay(2000);
    topic5_3(7,0);
    helpsub(37,16,24);
    inter=21;
    setint(inter);gint=0;
    type1(65,22,"Press"," Enter");

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

hot_key(42,12,100,0,13);
if(buftype1!=NULL)
    putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
delay(1500);
topic5_3(7,2);
restore1();
Cint(inter);
gint=1;
}
break;
case 18 : if(toupper(g[0])==st_5[0]) {
    keep();
    mix_18[0]=g[0];
    cktp=0;
    restore1();
    Cint(inter);
    gint=1;
    for(too=1;too<=2;too++){
        mix5(-too+2,&mix_17,17);
        topic5_21(-too+2,1);
        topic5_3(-too+7,1);
        topic5_3(-too+7,2);
        mix5(-too+23,&mix_18,18);}
    }
break;
case 19 : for(loop=0;loop<=5;loop++){

```

เอกสารนี้เป็นเอกสารที่ if(toupper(g[loop])==st_1[loop]) ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        {mix_19[loop]=g[loop]; nst++;}}
if(nst==6) {
    cktp=0;
    keep();
    for(too=1;too<=10;too++){
        topic5_21(-too,1);
        topic5_3(-too+5,1);
        topic5_3(-too+5,2);
        mix5(-too+21,&mix_18,18);
        mix5(-too+23,&mix_19,19);
        topic5_31(-too+24,0);}
    }
break;
case 20 : for(loop=0;loop<=8;loop++){
    if(toupper(g[loop])==st_20[loop])
        {mix_20[loop]=g[loop]; nst++;}}
if(nst==9) {
    cktp=0;
    keep();
    for(too=1;too<=12;too++){
        topic5_3(-too-5,1);
        topic5_3(-too-5,2);
        mix5(-too+11,&mix_18,18);
        mix5(-too+13,&mix_19,19);
        topic5_31(-too+14,0);
        mix5(-too+23,&mix_20,20);
        topic5_31(-too+24,1);}
    }
break;

```

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นโดยศูนย์เทคโนโลยีสารสนเทศและการสื่อสาร สำนักงานปลัดกระทรวงศึกษาธิการ เพื่อใช้ในการเรียนการสอนและเผยแพร่ความรู้แก่ผู้สนใจทั่วไป ไม่สามารถนำเอกสารนี้ไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        if(toupper(g[loop])==st_21[loop])
            {mix_21[loop]=g[loop]; nst++;}
    if(nst==9) {
        cktp=0;
        keep();
        topic5_4(2,1);
        }
    break;
case 22 : for(loop=0;loop<=13;loop++){
        if(toupper(g[loop])==st_22[loop])
            {mix_22[loop]=g[loop]; nst++;}
    if(nst==14) {
        cktp=0;
        keep();
        topic5_41(18,0);
        helpsub(37,14,31);
        type1(60,22,""," Press Any key ");
        Gkey();
        if(buftype1!=NULL)
            putimage(coltype1,rowtype1,buftype1,0);
        if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
        farfree(buftype1);
        farfree(bufold);
        for(too=1;too<=1;too++){
            mix5(-too+2,&mix_21,21);
            topic5_4(-too+2,1);
            mix5(-too+17,&mix_22,22);
            topic5_41(-too+18,0);
            topic5_41(-too+18,1);}
        delay(4000);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ห้ามเผยแพร่โดยไม่ได้รับอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for(too=1;too<=4;too++){
    mix5(-too+1,&mix_21,21);
    topic5_4(-too+1,1);
    mix5(-too+16,&mix_22,22);
    topic5_41(-too+17,3);}
helpsub(37,14,31);
type1(60,22,""," Press Any key ");
Gkey();
if(buftype1!=NULL)
    putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
delay(4000);
for(too=1;too<=4;too++){
    topic5_4(-too-3,1);
    mix5(-too+12,&mix_22,22);
    topic5_41(-too+13,3);}
}
break;
case 23 : if(toupper(g[0])==st_5[0]) {
    keep();
    mix_23[0]=g[0];
    cktp=0;
    for(too=1;too<=2;too++){
        topic5_4(-too-7,1);
        mix5(-too+8,&mix_22,22);
        topic5_41(-too+9,3);
        mix5(-too+23,&mix_23,23);}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

break;

case 24 : for(loop=0;loop<=8;loop++){
    if(toupper(g[loop])==st_21[loop])
        {mix_24[loop]=g[loop]; nst++;}}
if(nst==9) {
    cktp=0;
    keep();
    for(too=1;too<=15;too++){
        topic5_4(-too-9,1);
        mix5(-too+6,&mix_22,22);
        topic5_41(-too+7,3);
        mix5(-too+21,&mix_23,23);
        mix5(-too+23,&mix_24,24);
        topic5_4(-too+23,0);}
    }
break;

case 25 : for(loop=0;loop<=8;loop++){
    if(toupper(g[loop])==st_20[loop])
        {mix_25[loop]=g[loop]; nst++;}}
if(nst==9) {
    cktp=0;
    keep();
    topic5_5(3);
    thaixy(1,22,79,"C:\\>",0);
    helpsub(30,11,37);
    type1(40,22,"Press", " ALT+F1");
    hot_key(0,0,0,1,104);
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);

```

```

farfree(buftype1);
farfree(bufold);
topic1_1(-1,0);
topic5_51(20);
topic1_1(-2,2);
topic5_51(19);
helpsub(30,11,37);
type1(65,22,"Press"," Enter");
hot_key(12,23,100,0,13);
if(buftype1!=NULL)
    putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
for(too=1;too<=6;too++){
    topic1_1(-too-2,2);
    topic5_51(-too+19);}
}
break;
case 26 : for(loop=0;loop<=2;loop++){
    if(toupper(g[loop])==st_26[loop])
        {mix_26[loop]=g[loop]; nst++;}
if(nst==3) {
    cktp=0;
    keep();
    for(too=1;too<=4;too++){
        topic1_1(-too-8,2);
        topic5_51(-too+13);
        mix5(-too+23,&mix_26,26);
        topic5_52(-too+23);}
}

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของศูนย์เทคโนโลยีสารสนเทศและการสื่อสาร สำนักงานปลัดกระทรวงศึกษาธิการ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        break;
    }
}

/* ฟังก์ชัน mix เป็นการนำข้อความที่เก็บไว้มาแสดงที่หน้าจอ ในลักษณะที่เลื่อนขึ้นไปเรื่อย ๆ
โดยจะรับค่าบรรทัด ข้อความและหัวข้อที่ต้องการให้แสดง */
mix5(int y,char *dt,int cas)
{
    if(cas==2) thaixy(20,y,22,dt,0);
    if(cas==4) thaixy(47,y,58,dt,0);
    if((cas==5)|| (cas==12)|| (cas==18)) thaixy(22,y,24,dt,0);
    if(cas==23) thaixy(29,y,31,dt,0);
    if(cas==8) thaixy(47,y,56,dt,0);
    if(cas==15) thaixy(72,y,74,dt,0);
    if((cas!=2)&&(cas!=4)&&(cas!=5)&&(cas!=12)&&(cas!=8)&&(cas!=15)&&
(cas!=18)&&(cas!=23)){
        if(cas==26) thaixy(1,y,10,"A:\\>",0);
        else thaixy(1,y,10,"C:\\>",0);
        thaixy(5,y,17,dt,0);}
    thaixy(1,y+1,79,"ระจ่อมเกล้าเจ้าคุณทหารลาดกระบัง",0);
}

```

```

/*****
***** PROGRAM TOPIC6.C *****/
*****/

```

```

#include<alloc.h>

extern char gt[30];
extern unsigned int on;
extern int cktp,xbufold,ybufold,coltype1,rowtype1,coltype2,rowtype2,
        inter,gint;
extern char far *pointerindex;
extern char far *bufold;
extern char far *buftype1,*buftype2;
void topic_6(int cas);
void Check_6(char *g,int topic);

/* ฟังก์ชัน topic_6 เป็นการเรียนรู้ในบทที่ 6 หัวข้อที่ 1-3 */
void topic_6(int cas)
{
    barband();
    gint=1;
    openfile("topic6.hlp");
    if(cas==1){
        clearblock(1,1,81,24);
        on=8;
        thaixy(1,2,30,"C:\\>DIR C:\\SODPRAC",0);
        topic3_1(3,4);
        helpsub(5,9,1);
        delay(1000);

        type1(38,22,"Type"," backup c:\\sodprac a: /s");

```

```

on=47;
helpsub(30,13,3);
type1(50,22,"Type"," dir a:");
Check_Key(6,0,23,3,0,6);
thaixy(1,23,23,"C:\\>",0);
helpsub(30,12,4);
type1(47,22,"Press"," Page Down");
hot_key(0,23,250,0,81);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);}

if(cas<=2){
clearblock(1,1,81,24);
on=148;
helpsub(5,10,5);
type1(65,22,"Press"," Enter");
hotkey(13);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
helpsub(30,12,6);
type1(32,22,"Type"," restore a: c:\\sodprac\\*.*/s");
Check_Key(28,0,2,4,0,6);
on=47;
helpsub(30,12,8);
type1(45,22,"Type"," dir c:\\sodprac");

```

```

thaixy(1,23,23,"C:\\\>",0);
helpsub(30,12,9);
type1(48,22,"Press"," Page down");
hot_key(0,23,250,0,81);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);}

```

```

if(cas<=3){
clearblock(1,1,81,24);
on=47;
helpsub(5,8,10);
type1(40,22,"Type"," dir *.doc");
Check_Key(9,0,2,6,3,6);
on=42;
helpsub(30,13,11);
type1(47,22,"Type"," del *.doc");
Check_Key(9,0,14,7,3,6);
on=47;
helpsub(30,14,12);
type1(50,22,"Type"," dir");
Check_Key(3,0,16,8,3,6);
on=175;
helpsub(30,12,13);
type1(46,22,"Type"," undelete *.doc");
Check_Key(14,0,23,9,3,6);
helpsub(30,13,14);
inter=23;

```

```

type1(52,22,"Press"," Y");
Check_Key(1,62,23,10,1,6);
helpsub(30,13,15);
type1(52,21,"Press"," G");
Check_Key(1,52,23,11,1,6);
helpsub(30,13,16);
type1(52,22,"Press"," Y");
Check_Key(1,62,23,12,1,6);
helpsub(30,13,17);
type1(52,21,"Press"," B");
Check_Key(1,52,23,13,1,6);
helpsub(30,13,18);
type1(52,22,"Press"," Y");
Check_Key(1,62,23,14,1,6);
helpsub(30,13,19);
type1(52,21,"Press"," E");
Check_Key(1,52,23,15,1,6);
on=47;
helpsub(30,13,20);
type1(52,22,"Type"," dir");
Check_Key(3,0,23,16,3,6);
helpsub(35,10,21);
inter=24;
setint(inter);gint=0;
thaixy(1,23,23,"C:\\>",0);
type1(47,22,"Press"," Page Down");
hot_key(0,23,250,0,81);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);

```

```

farfree(bufold);
restore1();
Cint(inter);
gint=1;}
farfree(pointerindex);
}

```

/* ฟังก์ชัน Check_6 เป็นการตรวจสอบตัวอักษรที่พิมพ์ว่า พิมพ์ตามที่กำหนดไว้หรือไม่ ถ้าพิมพ์
ก็กลับไปที่พิมพ์ใหม่ หรือรอรับคีย์ที่ต้องการให้กดก็ได้ */

```

void Check_6(char *g,int topic)
{
int too,loop,nst=0;
char st_1[]="DIR C:\\SODPRAC",st_2[]="BACKUP C:\\SODPRAC A: /S",
st_3[]="DIR A:",st_4[]="RESTORE A: C:\\SODPRAC\\*.* /S",
st_6[]="DIR *.DOC",st_7[]="DEL *.DOC",st_8[]="DIR",
st_9[]="UNDELETE *.DOC",st_10[]="Y",st_11[]="G",st_13[]="B",
st_15[]="E";
static char mix_2[28],mix_3[28],mix_4[28],mix_5[28],mix_6[14],
mix_7[14],mix_8[14],mix_9[14],mix_10[2],mix_11[2],
mix_12[2],mix_13[2],mix_14[2],mix_15[2],mix_16[14];
switch(topic) {
case 2 : for(loop=0;loop<=22;loop++){
if(toupper(g[loop])==st_2[loop])
{mix_2[loop]=g[loop]; nst++;}}
if(nst==23) {
cktp=0;
keep();
for(too=1;too<=5;too++){
thaixy(1,-too+2,30,"C:\\>DIR C:\\SODPRAC",0);
thaixy(1,-too+3,30,"ศึกษาเท่านั้น ไม่นอนุญาตให้แก้ไข",0);
}
}
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้แก้ไขหรือเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    topic3_1(-too+3,4);
    mix6(-too+22,&mix_2,2);
    topic6_1(-too+23);}

helpsub(30,12,2);
type1(60,22,""," Press any key ");
Gkey();
if(buftype1!=NULL)
    putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
for(too=1;too<=3;too++){
    topic3_1(-too-2,4);
    mix6(-too+17,&mix_2,2);
    topic6_1(-too+18);}
delay(1500);
for(too=1;too<=17;too++){
    topic3_1(-too-5,4);
    mix6(-too+14,&mix_2,2);
    topic6_1(-too+15);
    topic3_31(-too+23,2);
    delay(700);}
}

break;
case 3 : for(loop=0;loop<=5;loop++){
    if(toupper(g[loop])==st_3[loop])
        {mix_3[loop]=g[loop]; nst++;}}
if(nst==6) {
    cktp=0;

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for(too=1;too<=10;too++){
    topic6_1(-too-2);
    topic3_31(-too+6,2);
    mix6(-too+23,&mix_3,3);
    topic6_11(-too+24);}
    }
break;
case 4 : for(loop=0;loop<=27;loop++){
    if(toupper(g[loop])==st_4[loop])
        {mix_4[loop]=g[loop]; nst++;}
if(nst==28) {
    cktp=0;
    keep();
    topic6_2(3,0);
    helpsub(45,14,7);
    type1(60,22,""," Press any key ");
    Gkey();
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(buftype1);
    farfree(bufold);
    delay(500);
    topic6_2(3,1);
    topic6_2(3,2);
    topic3_31(7,3);
    mix6(1,&mix_4,4);
    topic6_2(2,1);
    topic6_2(2,2);

```

```

        topic3_31(6,2);
    }

    break;

case 5 : for(loop=0;loop<=13;loop++){
    if(toupper(g[loop])==st_1[loop])
        {mix_5[loop]=g[loop]; nst++;}
    if(nst==14) {
        cktp=0;
        keep();
        for(too=1;too<=20;too++){
            mix6(-too+1,&mix_4,4);
            topic6_2(-too+2,1);
            topic6_2(-too+2,2);
            topic3_31(-too+6,2);
            mix6(-too+23,&mix_5,5);
            topic3_1(-too+24,4);}
        }
        break;

case 6 : for(loop=0;loop<=8;loop++){
    if(toupper(g[loop])==st_6[loop])
        {mix_6[loop]=g[loop]; nst++;}
    if(nst==9) {
        cktp=0;
        keep();
        topic6_3(3,0);}
        break;

case 7 : for(loop=0;loop<=8;loop++){
    if(toupper(g[loop])==st_7[loop])
        {mix_7[loop]=g[loop]; nst++;}

```

```

break;
case 8 : for(loop=0;loop<=2;loop++){
    if(toupper(g[loop])==st_8[loop])
        {mix_8[loop]=g[loop]; nst++;}
if(nst==3) {
    cktp=0; keep();
    for(too=1;too<=10;too++){
        mix6(-too+2,&mix_6,6);
        topic6_3(-too+3,0);
        mix6(-too+14,&mix_7,7);
        mix6(-too+16,&mix_8,8);
        topic6_3(-too+17,1);}
    break;
case 9 : for(loop=0;loop<=13;loop++){
    if(toupper(g[loop])==st_9[loop])
        {mix_9[loop]=g[loop]; nst++;}
if(nst==14) {
    cktp=0;
    keep();
    for(too=1;too<=12;too++){
        topic6_3(-too-7,0);
        mix6(-too+4,&mix_7,7);
        mix6(-too+6,&mix_8,8);
        topic6_3(-too+7,1);
        mix6(-too+23,&mix_9,9);
        topic6_31(-too+24);}
    }
    break;

```

เอกสารนี้เป็นเอกสารที่... **case 10** : if(toupper(g[0])==st_10[0]) { นั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(buftype1!=NULL)
    putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
mix_10[0]=g[0];
cktp=0;
restore1();
Cint(inter);
gint=1;
topic6_3(-6,1);
mix6(10,&mix_9,9);
mix6(22,&mix_10,10);
topic6_31(11);}
break;
case 11 : if(toupper(g[0])==st_11[0]) {
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(buftype1);
    farfree(bufold);
    mix_11[0]=g[0];
    cktp=0;
    for(too=1;too<=4;too++){
        topic6_3(-too-6,1);
        mix6(-too+10,&mix_9,9);
        mix6(-too+22,&mix_10,10);
        topic6_31(-too+11);
        mix6(-too+23,&mix_11,11);}

```

```

break;

case 12 : if(toupper(g[0])==st_10[0]) {
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(buftype1);
    farfree(bufold);
    mix_12[0]=g[0];
    cktp=0;
    topic6_3(-11,1);
    mix6(5,&mix_9,9);
    mix6(17,&mix_10,10);
    topic6_31(6);
    mix6(18,&mix_11,11);
    mix6(22,&mix_12,12);
}
break;

case 13 : if(toupper(g[0])==st_13[0]) {
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(buftype1);
    farfree(bufold);
    mix_13[0]=g[0];
    cktp=0;
    for(too=1;too<=4;too++){
        topic6_3(-too-11,1);
        mix6(-too+5,&mix_9,9);
        mix6(-too+17,&mix_10,10);

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ (topic6_31(-too+6)); การศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        mix6(-too+18,&mix_11,11);
        mix6(-too+22,&mix_12,12);
        mix6(-too+23,&mix_13,13);}
    }

    break;

case 14 : if(toupper(g[0])==st_10[0]) {
        if(buftype1!=NULL)
            putimage(coltype1,rowtype1,buftype1,0);
        if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
        farfree(buftype1);
        farfree(bufold);
        mix_14[0]=g[0];
        cktp=0;
        topic6_3(-16,1);
        mix6(12,&mix_10,10);
        topic6_31(1);
        mix6(13,&mix_11,11);
        mix6(17,&mix_12,12);
        mix6(18,&mix_13,13);
        mix6(22,&mix_14,14);}

    break;

case 15 : if(toupper(g[0])==st_15[0]) {
        if(buftype1!=NULL)
            putimage(coltype1,rowtype1,buftype1,0);
        if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
        farfree(buftype1);
        farfree(bufold);
        mix_15[0]=g[0];
        cktp=0;

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์หรือการแจ้งให้ทราบก่อนการนำออกใช้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mix6(-too+12,&mix_10,10);
topic6_31(-too+1);
mix6(-too+13,&mix_11,11);
mix6(-too+17,&mix_12,12);
mix6(-too+18,&mix_13,13);
mix6(-too+22,&mix_14,14);
mix6(-too+23,&mix_15,15);}
}

```

```
break;
```

```

case 16 : for(loop=0;loop<=2;loop++){
    if(toupper(g[loop])==st_8[loop])
        {mix_16[loop]=g[loop]; nst++;}
    if(nst==3){
        cktp=0;
        keep();
        for(too=1;too<=20;too++){
            mix6(-too+7,&mix_10,10);
            topic6_31(-too-4);
            mix6(-too+8,&mix_11,11);
            mix6(-too+12,&mix_12,12);
            mix6(-too+13,&mix_13,13);
            mix6(-too+17,&mix_14,14);
            mix6(-too+18,&mix_15,15);
            mix6(-too+23,&mix_16,16);
            topic6_3(-too+24,2);}
        }

```

```
break;
```

```
}
```

```
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

/* ฟังก์ชัน mix เป็นการนำข้อความที่เก็บไว้มาแสดงที่หน้าจอ ในลักษณะที่เลื่อนขึ้นบนเรื่อย ๆ
โดยจะรับค่าบรรทัด ข้อความและหัวข้อที่ต้องการให้แสดง */

```

mix6(int y,char *dt,int cas)
{
    if(cas<6){
        thaixy(1,y,10,"C:\\>",0);
        thaixy(5,y,31,dt,0);}
    else{
        if((cas<10)||!(cas>15)){
            thaixy(1,y,10,"C:\\SODPRAC>",0);
            thaixy(12,y,24,dt,0);}
        else {
            if((cas==10)||!(cas==12)||!(cas==14)) thaixy(67,y,68,dt,0);
            if((cas==11)||!(cas==13)||!(cas==15)) thaixy(57,y,58,dt,0);
        }
    }
    if((cas!=10)&&(cas!=12)&&(cas!=14))
        thaixy(1,y+1,79,"",0);
}

```

```

/*****
***** PROGRAM TOPIC7.C *****/
*****/

```

```

#include<alloc.h>

extern int cktp,xbufold,ybufold,coltype1,rowtype1,coltype2,rowtype2,
        inter,gint;

extern char far *pointerindex;

extern unsigned int on;

extern char far *bufold;

extern char far *buftype1,*buftype2;

extern char gt[30];

void topic_7(int cas);

void Check_7(char *g,int topic);

static char mix_1[3],mix_2[7],mix_3[9],mix_5[19]="CLS",
        mix_6[19]="VER",mix_7[19]="\CD",mix_9[19]="CLS",
        mix_8[19]="CD\SODPRAC\TMPFILES",mix_11[12]="DIR \SODPRAC",
        mix_13[19]="CD\SODPRAC\TXTFILES",mix_14[3];

int cur=0;

char ch,ch1,gc1,gc2;

```

```

/* ฟังก์ชัน topic_7 เป็นการเขียนนามบทที่ 7 หัวข้อที่ 1-3 */

```

```

void topic_7(int cas)
{
    int ck,loop;
    barband();
    gint=1;
    openfile("topic7.hlp");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

clearblock(1,1,81,24);
on=10;
thaixy(1,2,10,"C:\\TESTDIR>",0);
helpsub(5,10,1);
type1(52,22,"Type"," cd\\");
Check_Key(3,7,2,1,1,7);
thaixy(1,4,4,"C:\\>",0);
on=73;
helpsub(30,11,2);
type1(52,22,"Press"," F3");
hot_key(0,4,250,0,61);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
thaixy(5,4,7,mix_1,0);
helpsub(30,13,3);
type1(48,22,"Type"," sodprac");
Check_Key(7,3,4,2,1,7);
thaixy(1,6,12,"C:\\SODPRAC>",0);
helpsub(35,13,4);
type1(52,22,"Press"," F3");
hot_key(7,6,250,0,61);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
thaixy(12,6,13,mix_1,0);
thaixy(15,6,20,mix_2,0);
helpsub(35,12,5);

```

```

type1(48,22,"Type"," \\txtfiles");
Check_Key(9,17,6,3,1,7);
thaixy(1,8,19,"C:\\SODPRAC\\TXTFILES>",0);
helpsub(30,12,6);
type1(52,22,"Press"," F3");
hot_key(16,8,250,0,61);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
thaixy(21,8,30,mix_1,0);
thaixy(24,8,30,mix_2,0);
thaixy(31,8,40,mix_3,0);
helpsub(35,14,7);
type1(46,22,"Press"," backspace-9");
for(ck=0;ck<=8;ck++){
hot_key(35-ck,8,250,0,8);
Del(35-ck,8,0,-8);}
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
farfree(buftype1);

type1(65,22,"Press"," Enter");
hotkey(13);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(bufold);
farfree(buftype1);
thaixy(1,10,23,"C:\\SODPRAC>",0);
helpsub(30,13,8);

```

```

hot_key(7,10,250,0,81);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(bufold);
farfree(buftype1);}

```

```

if(cas<=2){
clearblock(1,1,81,24);
on=73;
helpsub(5,10,9);
type1(49,22,"Type"," doskey");
Check_Key(6,0,2,4,3,7);
on=26;
helpsub(30,13,10);
type1(50,22,"Type"," cls");
Check_Key(3,0,5,5,3,7);
clearblock(1,1,79,23);
on=191;
helpsub(35,12,11);
type1(50,22,"Type"," ver");
Check_Key(3,0,2,6,3,7);
on=10;
helpsub(30,13,12);
type1(50,22,"Type"," cd\\");
Check_Key(3,0,7,7,3,7);
helpsub(30,13,13);
type1(36,22,"Type"," cd\\sodprac\\tmpfiles");
Check_Key(19,0,9,8,0,7);
helpsub(30,13,14);

```

เอกสาร type1(50,22,"Type","รับcls"); เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Check_Key(3,0,12,9,0,7);
clearblock(1,1,79,24);
thaixy(1,2,4,"C:\\>",0);
on=73;
helpsub(30,13,15);
type1(52,22,"Press"," F7");
hot_key(0,2,250,0,65);
    if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(buftype1);
    farfree(bufold);
helpsub(30,13,16);
type1(43,22,"Press"," Arrow up-4");
Check_7(&gt;,10);

helpsub(30,13,17);
type1(40,22,"Type"," dir \\sodprac");
Check_Key(12,0,13,11,0,7);
helpsub(30,13,18);
type1(52,22,"Type"," v");
Check_Key(1,0,23,12,0,7);

thaixy(1,23,23,"C:\\SODPRAC>",0);
helpsub(30,13,20);
type1(45,22,"Press"," Page Down");
hot_key(7,23,250,0,81);
    if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(bufold);
    farfree(buftype1);}

```

```

if(cas<=3){
clearblock(1,1,81,24);
on=73;
thaixy(1,2,4,"C:\\\>",0);
thaixy(5,2,22,&mix_8,0);
helpsub(5,10,21);
type1(49,22,"Press"," Home");
hot_key(19,2,250,0,71);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
helpsub(30,13,22);
type1(50,22,"Press"," End");
Kkey(0,&mix_8,15,79,0);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
helpsub(30,13,23);
type1(42,22,"Press"," Arrow left-7");
hot_key(19,2,250,0,75);
for(ck=0;ck<=5;ck++)
    Kkey(18-ck,&mix_8,15,75,18-ck);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
helpsub(30,13,24);

```

เอกสาร type1(50,22,"Press"," Ins"); เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Kkey(12,&mix_8,15,82,12);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
helpsub(30,13,25);
type1(50,22,"Type"," xt");
Kkey(12,&mix_8,11,120,12);
thaixy(17,2,18,&ch1,0);
gc1=ch1;
thaixy(18,2,23,&mix_8[12],0);
Kkey(13,&mix_8,11,116,12);

thaixy(18,2,19,&ch1,0);
gc2=ch1;
thaixy(19,2,24,&mix_8[12],0);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
helpsub(30,13,26);
type1(50,22,"Press"," Del");
for(loop=0;loop<=1;loop++){
    Kkey(14,&mix_8,11,83,12+loop);
    Del(14,2,0,0);
    thaixy(19,2,30,&mix_8[13+loop],0);
    thaixy(25-loop,2,30,"    ",0);}
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);

```

```

farfree(bufold);
helpsub(30,13,27);
type1(48,22,"Press"," Enter");
Kkey(14,&mix_8,11,13,14);

if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(buftype1);
farfree(bufold);
for(loop=0;loop<=19;loop++)
    mix_13[loop]=mix_8[loop];
mix_13[12]=gc1;
mix_13[13]=gc2;
Check_7(&gt,13);
on=47;
type1(50,22,"Press"," dir");
Check_Key(3,20,16,14,1,7);
on=73;
helpsub(35,13,35);
inter=25;
setint(inter);gint=0;
topic7_1(0,23,0);
type1(47,22,"Press"," Page Down");
hot_key(16,23,250,0,81);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(bufold);
farfree(buftype1);
restore1();

```

เอกสาร Cint(inter); รวบรวมไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

gint=1;
}
farfree(pointerindex);
}

/* ฟังก์ชัน Check_7 เป็นการตรวจสอบตัวอักษรที่พิมพ์ว่า พิมพ์ตามที่กำหนดไว้หรือไม่ ถ้าไม่เช
ก็กลับไปพิมพ์ใหม่ หรือรอรับคีย์ที่ต้องการให้กดก็ได้ */
void Check_7(char *g,int topic)
{
int too,loop,nst=0;
char st_1[]="CD\\",st_2[]="SODPRAC",st_3[]="\\TEXTFILES",st_5[]="CLS",
st_4[]="DOSKEY",st_6[]="VER",st_8[]="CD\\SODPRAC\\TMPFILES",
st_11[]="DIR \\SODPRAC",st_12[]="V",st_14[]="DIR";
switch(topic) {
case 1 : for(loop=0;loop<=2;loop++){
if(toupper(g[loop])==st_1[loop])
{mix_1[loop]=g[loop]; nst++;}}
if(nst==3) {cktp=0;keep();}
break;
case 2 : for(loop=0;loop<=6;loop++){
if(toupper(g[loop])==st_2[loop])
{mix_2[loop]=g[loop]; nst++;}}
if(nst==7) {cktp=0;keep();}
break;
case 3 : for(loop=0;loop<=8;loop++){
if(toupper(g[loop])==st_3[loop])
{mix_3[loop]=g[loop]; nst++;}}
if(nst==9) {cktp=0;keep();}
break;

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
 ไม่ว่ากรรมใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        if(toupper(g[loop])==st_4[loop]) nst++;}
    if(nst==6) {
        cktp=0;
        keep();
        thaixy(1,3,20,"DOSKEY installed.",0);}
    break;
case 5 : for(loop=0;loop<=2;loop++){
        if(toupper(g[loop])==st_5[loop])
            {mix_5[loop]=g[loop]; nst++;}}
    if(nst==3) {cktp=0;keep();}
    break;
case 6 : for(loop=0;loop<=2;loop++){
        if(toupper(g[loop])==st_6[loop])
            {mix_6[loop]=g[loop]; nst++;}}
    if(nst==3) {
        cktp=0;keep();
        thaixy(1,4,20,"MS-DOS Version 5",0);}
    break;
case 7 : for(loop=0;loop<=2;loop++){
        if(toupper(g[loop])==st_1[loop])
            {mix_7[loop]=g[loop]; nst++;}}
    if(nst==3) {cktp=0;keep();}
    break;
case 8 : for(loop=0;loop<=18;loop++){
        if(toupper(g[loop])==st_8[loop])
            {mix_8[loop]=g[loop]; nst++;}}
    if(nst==19) {
        cktp=0;keep();
        thaixy(1,10,20,"Invalid directory",0);}
    break;

```

```

case 9 : for(loop=0;loop<=2;loop++){
    if(toupper(g[loop])==st_5[loop])
        {mix_9[loop]=g[loop]; nst++;}
    if(nst==3) {cktp=0;keep();}
    break;
case 10 : for(loop=1;loop<=5;loop++)
    mix7(0,2+loop,loop,1);
    thaixy(1,8,10,"C:\\\>",0);
    for(loop=5;loop>=2;loop--){
        hot_key(0+cur,8,250,0,72);
        mix7(1,8,loop,0);}
    keep();
    thaixy(1,10,20,"MS-DOS Version 5",0);
    break;
case 11 : for(loop=0;loop<=11;loop++){
    if(toupper(g[loop])==st_11[loop])
        {mix_11[loop]=g[loop]; nst++;}
    if(nst==12) {
        cktp=0;
        keep();
        for(too=1;too<=11;too++){
            thaixy(1,-too+2,10,"C:\\\>",0);
            for(loop=1;loop<=5;loop++)
                mix7(0,-too+2+loop,loop,1);
            thaixy(1,-too+8,10,"C:\\\>",0);
            thaixy(5,-too+8,10,&mix_6,0);
            thaixy(1,-too+9,20,"",0);
            thaixy(1,-too+10,20,"MS-DOS Version 5",0);
            thaixy(1,-too+11,20,"",0);

```

```

        topic6_3(-too+14,3);  }}
break;
case 12 : if(toupper(g[0])==st_12[0])
    {cktp=0;
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    farfree(buftype1);
    type1(52,22,"Press"," F8");
    hot_key(1,23,250,0,66);
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(buftype1);
    farfree(bufold);
    thaixy(5,23,10,&mix_6,0);
    helpsub(36,13,19);
    type1(65,22,"Press"," Enter");
    hotkey(13);
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(bufold);
    farfree(buftype1);
    for(too=1;too<=5;too++){
        topic6_3(-too+3,3);
        mix77(-too+23,&mix_6);
        thaixy(1,-too+25,20,"MS-DOS Version 5",0);
        thaixy(1,-too+26,20,"
    }

```

เอกสารนี้เป็นเอกสารที่ break; สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

case 13 : topic7_1(0,4,0);
        helpsub(30,13,28);
        type1(52,22,"Press"," F7");
        hot_key(16,4,250,0,65);
        if(buftype1!=NULL)
            putimage(coltype1,rowtype1,buftype1,0);
        farfree(buftype1);
        if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
        farfree(bufold);
        mix7_1(5);
        topic7_1(0,14,0);
        helpsub(32,12,29);
        type1(52,22,"Press"," F9");
        hot_key(16,14,250,0,67);
        if(buftype1!=NULL)
            putimage(coltype1,rowtype1,buftype1,0);
        if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
        farfree(bufold);
        farfree(buftype1);
        topic7_1(20,14,1);
        helpsub(32,8,30);
        type1(53,22,"Press"," 3");
        hot_key(30,14,250,0,51);
        if(buftype1!=NULL)
            putimage(coltype1,rowtype1,buftype1,0);
        if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
        farfree(bufold);
        farfree(buftype1);
        topic7_1(20,14,2);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 thaixy(21,14,35," ",0);
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามเผยแพร่ต่อสาธารณะ และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

thaixy(21,14,35,mix_7,0);
helpsub(30,13,31);
type1(65,22,"Press"," Enter");
hot_key(19,14,250,0,13);
if(buftype1!=NULL)
    putimage(coltype1,rowtype1,buftype1,0);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(bufold);
farfree(buftype1);
thaixy(1,16,4,"C:\\\>",0);
helpsub(35,12,32);
type1(52,22,"Press"," F9");
hot_key(0,16,250,0,67);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
farfree(buftype1);
topic7_1(4,16,1);
type1(52,22,"Press"," 9");
hot_key(14,16,250,0,57);
if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
farfree(bufold);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
farfree(buftype1);
topic7_1(4,16,3);
thaixy(5,16,35,"",0);
thaixy(5,16,22,mix_13,0);
helpsub(30,9,33);
type1(48,22,"Press"," CTRL+T");
hot_key(19,16,250,0,20);
if(buftype1!=NULL)putimage(coltype1,rowtype1,buftype1,0);
farfree(buftype1);

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

thaixy(24,16,79,"๙",0);
break;
case 14 : for(loop=0;loop<=2;loop++){
    if(toupper(g[loop])==st_14[loop])
        {mix_14[loop]=g[loop]; nst++;}
if(nst==3) {
    cktp=0;
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(bufold);
    farfree(buftype1);
    helpsub(35,13,34);
    type(65,22,"Press"," Enter");
    hotkey(13);
    if(buftype1!=NULL)
        putimage(coltype1,rowtype1,buftype1,0);
    if(bufold!=NULL)putimage(xbufold,ybufold,bufold,0);
    farfree(bufold);
    farfree(buftype1);
    for(too=1;too<=13;too++){
        thaixy(1,-too+2,4,"C:\\\>",0);
        thaixy(5,-too+2,22,&mix_13,0);
        thaixy(1,-too+3,30,"",0);
        topic7_1(0,-too+4,0);
        mix7_1(-too+5);
        topic7_1(0,-too+14,0);
        thaixy(21,-too+14,35,mix_7,0);
        thaixy(1,-too+16,4,"C:\\\>",0);
        thaixy(5,-too+16,35,mix_13,0);

```

```

    thaixy(24,-too+16,79,"๙",0);
    thaixy(25,-too+16,28,mix_14,0);
    thaixy(1,-too+17,28,"",0);
    thaixy(1,-too+17,4,"C:\\\>",0);
    thaixy(5,-too+17,22,&mix_13,0);
    thaixy(1,-too+18,30,"",0);
    topic7_1(0,-too+19,0);
    thaixy(21,-too+19,22,&mix_14,0);
    topic3_1(-too+20,5);}
}
break;
}
}
mix7(int x,int y,int cas,int ok)
{
    if(cas==1){
        if(ok==1) thaixy(1,y,10,"1: ",0);thaixy(4+x,y,21,mix_5,0);}
    if(cas==2){
        if(ok==1) thaixy(1,y,10,"2: ",0);thaixy(4+x,y,21,mix_6,0);}
    if(cas==3){
        if(ok==1) thaixy(1,y,10,"3: ",0);thaixy(4+x,y,21,mix_7,0);
        if(ok==0){
            cur=3;thaixy(7+x,y,25,"",0);}}
    if(cas==4){
        if(ok==1) thaixy(1,y,10,"4: ",0);thaixy(4+x,y,21+x,mix_8,0);
        if(ok==0) cur=19;}
    if(cas==5){
        if(ok==1) thaixy(1,y,10,"5: ",0);thaixy(4+x,y,21,mix_9,0);
        if(ok==0) cur=3;}
    thaixy(1,y+1,79,"",0);}
}

```

/* ฟังก์ชัน mix เป็นการนำข้อความที่เก็บไว้มาแสดงที่หน้าจอ ในลักษณะที่เลื่อนขึ้นมาเรื่อย ๆ
 อยจะรับค่าบรรทัด ข้อความและหัวข้อที่ต้องการให้แสดง */

```
mix7_1(int y)
```

```
{
    thaixy(1,y,10,"1: ",0);thaixy(4,y,21,mix_5,0);
    thaixy(1,y+1,10,"2: ",0);thaixy(4,y+1,21,mix_6,0);
    thaixy(1,y+2,10,"3: ",0);thaixy(4,y+2,21,mix_7,0);
    thaixy(1,y+3,10,"4: ",0);thaixy(4,y+3,21,mix_8,0);
    thaixy(1,y+4,30,"",0);
    thaixy(1,y+4,10,"5: ",0);thaixy(4,y+4,21,mix_9,0);
    thaixy(1,y+5,10,"6: ",0);thaixy(4,y+5,21,mix_6,0);
    thaixy(1,y+6,10,"7: ",0);thaixy(4,y+6,14,mix_11,0);
    thaixy(1,y+7,30,"",0);
    thaixy(1,y+7,10,"8: ",0);thaixy(4,y+7,21,mix_6,0);
    thaixy(1,y+8,10,"9: ",0);thaixy(4,y+8,21,mix_13,0);
    thaixy(1,y+9,79,"",0);
}
```

```
mix77(int y,char *dt)
```

```
{
    thaixy(1,y,10,"C:\\>",0);
    thaixy(5,y,17,dt,0);
    thaixy(1,y+1,79,"",0);
}
```

/* ฟังก์ชัน Krapib เป็นการแสดงตัวอักษรกระหิบบเป็นสีที่เราต้องการ */

```
Krapib(int x,int y,int dl,char *pt,int color)
```

```
{
    do{
```

```
        Del(x,y,0,0);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

delay(dl);
Set_Color(color);
thaixy(x+5,y,22,pt,0);
delay(dl);
}
while(kbhit()==0);
Set_Color(15);
thaixy(x+5,y,22,pt,0);
}

```

/* ฟังก์ชัน Kkey เป็นการกระหิบบตัวอักษรเป็นสีที่ เราต้องการและรอการกดคีย์ที่เรากำหนดไว้ เพื่อตรวจสอบด้วย */

```
Kkey(int colum,char *st,int colour,int key,int x)
```

```

{
do{
Krapib(colum,2,250,&st[x],colour);
ch=getch();
ch1=ch;
ch=tolower(ch);
if(ch==0) ch=getch();
}
while(ch!=key);
}

```

/* ฟังก์ชัน topic7_1 เป็นการแสดงผลการกดคีย์ F9 */

```
topic7_1(int x,int y,int cas)
```

```

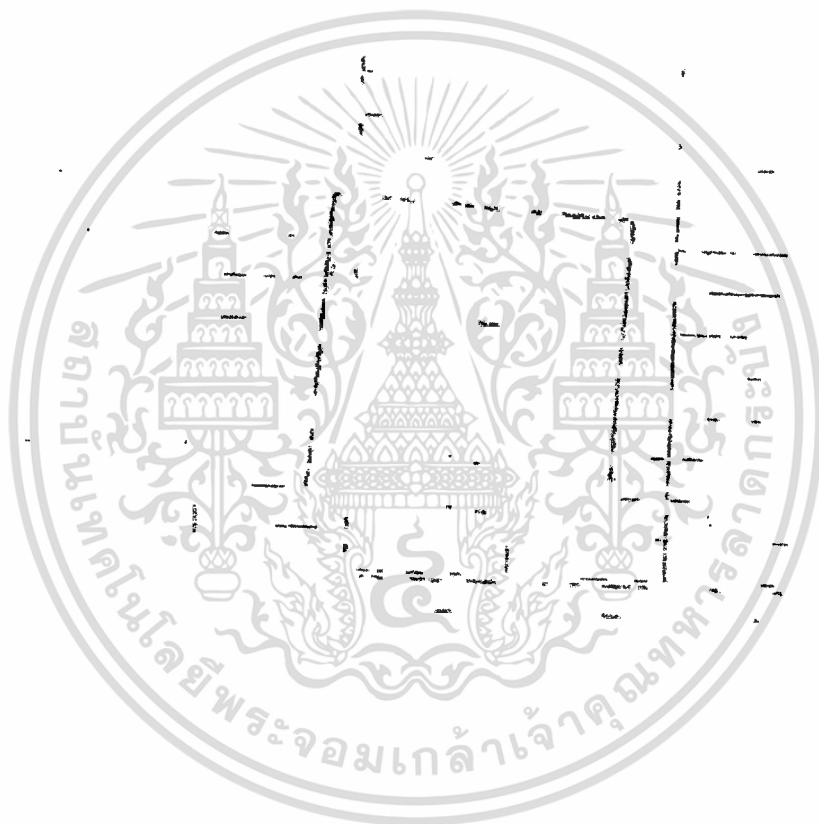
{
if(cas==0) thaixy(1+x,y,30,"C:\\SODPRAC\\TXTFILES>",0);
if(cas==1) thaixy(1+x,y,40,"Line number :          ",0);

```

if(cas==2) thaixy(1+x,y,40,"Line number : 3 ไม่นอนุญาต",0);

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของกรมส่งเสริมการค้าระหว่างประเทศ กระทรวงพาณิชย์ ไม่ใช่ว่ากรณีนี้อะไรทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if(cas==3) thaixy(1+x,y,40,"Line number : 9      ",0);  
thaixy(1,y+1,30,"                                ",0);  
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. เกียรติ ไกร เรืองอร่าม. "การฝึกทักษะการคิดข้อมูลแบบสัมพันธ์", ปรินซ์นิพนธ์ภาควิชา ครุศาสตร์อุตสาหกรรม คณะครุศาสตร์อุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ
2. กิตติ องค์คุณารักษ์. "แอลวานซ์ แอสเซมบลี", บริษัท ซีเอ็ดดูเคชั่น จำกัด, 2521
3. จเร เลิศสุดวิชัย. "การโปรแกรมใช้งานการ์ด VGA ตอนที่ 1", ไมโครคอมพิวเตอร์ 55, 2536
4. วีรวัฒน์ บัณฑิตวงศ์, สรวุฒิ เอ็งอุทัยวุฒิ. "การเขียนโปรแกรมการใช้งาน EGA/VGA" , บริษัท ซีเอ็ดดูเคชั่น จำกัด, 2521
5. ไพบูลย์ เอ็มพันธ์. "คู่มือการสรุปใช้งานของ MS-DOS 5", บริษัท ซีเอ็ดดูเคชั่น จำกัด, 2521
6. มัทนา ปราการสมุทร. "การเขียนชุดคำสั่งภาษา ซี", ไซเทคพริ้นติ้ง จำกัด, 2534
7. ถวัลย์ ตั้งจิตานนท์. "เปิดโลก VGA ฮาร์ดแวร์", ไมโครคอมพิวเตอร์ 55, 2536
8. ศิววัฒน์ ศิวบาร, พรชัย จักรธำรงค์, จิรศักดิ์ ชัยวิริยกุล. "ภาษาแอสเซมบลีขั้นสูงสำหรับ IBM PC", บริษัท ซีเอ็ดดูเคชั่น จำกัด, 2521
9. สมพันธ์ ช่างศิลป์. "ภาษา ซี (C Language)", หน่วยงานบรรณงานบริหารและธุรการ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยขอนแก่น, 2535
10. สมพัฒน์ รุ่งตะวันเรืองศรี. "เรารู้คอมพิวเตอร์กราฟิก 2 มิติ ด้วยภาษา ซี", บริษัท ซีเอ็ดดูเคชั่น จำกัด, 2521
11. Library Referance. " Borland C⁺⁺ V.2.0 ", Borland, 1991
12. Herbert Shildt. "Turbo C/C⁺⁺ : The Complete Referance", osbome McGraw-Hill, 1990
13. Michael Tischer. "Pc System Programming", Abacus, 1991