การจำลองภาพภูมิประเทศในลักษณะสามมิติ

3 DIMENSION OF CONTOUR

โดย

นายธีรพล ปวิตรปก

ปริญญานิพนธ์นี้เป็นการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตร์บัณฑิต

สาขา เทคโนโลยีการวัดคุมทางอุตสาหกรรม

สถาบัน เทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2535

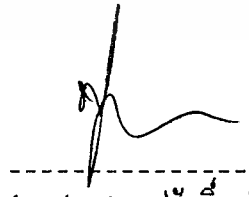
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

032663

ภาควิชา เทคโนโลยีการวัดคุมทางอุตสาหกรรม
สาขา เทคโนโลยีการวัดคุมทางอุตสาหกรรม
คณะ วิศวกรรมศาสตร์
สถาบัน เทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การจำลองภาพภูมิประเทศในลักษณะสามมิติ
[3 DIMENSION OF CONTOUR]

ผู้จัดทำ นาย ธีรพล ปวิตรปก 34161109



(รศ.ดร. ฟูศักดิ์ ชีวสุวิทย์) อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

032663

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้ เป็นการประยุกต์ใช้งานทางด้านคอมพิวเตอร์กราฟฟิกกับงานทางด้าน image processing กล่าวคือเป็นการนำเอาภาพระดับความสูงของภูมิประเทศ(contour) ผ่านการคำนวณเพื่อหาค่าเฉลี่ยความสูงทั้งภาพโดยวิธีการ interpolation จากนั้นก็นำมาพล็อตรูปในลักษณะ 3 มิติ

การพล็อตภาพระดับความสูงของภูมิประเทศในลักษณะ 3 มิตินี้สามารถจะนำมาหมุนภาพดังกล่าวเพื่อเปลี่ยนมุมมองในการพิจารณาความสูงในมุมต่าง ๆ ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Abstract

This thesis is an application of computer graphic to the task of image processing which is use image of contour that is calculated to determined an image everage value by interpolation mathod then taken the image to ploted three dimensional picture

ploting the image of contour in three dimensional picture can be rotated th change perspective on consideration of height in various sizes

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

บทนำ		1
บทที่1	โครงสร้างไฟล์ *.bmp และการใช้งาน -format file -การใช้งาน	2
บทที่2	การกำหนดความสูงให้กับ contour -ทฤษฎีสหสัมพันธ์ด้วยลูกโซ่(chain code correlation) -การพัฒนาเพื่อการใช้งาน	6
บทที่3	Interpolation of image -ทฤษฎี Linear interpolation -การใช้งาน	11
บทที่4	Three dimension -ทฤษฎี -การใช้งาน	15
บทที่5	การใช้งานโปรแกรม -โครงสร้างการใช้งานทั้งโปรแกรม -รายละเอียดในการใช้คำสั่งต่าง ๆ	30
สรุปผลการทดลอง		33
กิตติกรรมประกาศ		
หนังสืออ้างอิง		
ภาคผนวก	โปรแกรมใช้งานจริง	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทนำ

ปัจจุบันนี้ เทคโนโลยีทางด้านกราฟฟิคเข้ามามีบทบาทอย่างมากในการใช้งานทางด้านคอมพิวเตอร์ ซึ่งการใช้งานทางด้านคอมพิวเตอร์นั้นเข้าไปเกี่ยวข้องกับงานประเภทอื่นๆมากมาย ไม่ว่าจะเป็นงานด้านการแพทย์ วิทยาศาสตร์ วิศวกรรม ฯลฯ

ในงานทางด้านวิศวกรรมนั้น ส่วนใหญ่จะใช้ในลักษณะจำลองภาพขึ้นมา เพื่อสามารถที่จะแก้ไขปรับปรุงลักษณะต่าง ๆ เพื่อให้มีคุณสมบัติทางด้านวิศวกรรมที่ดีขึ้น หรืออาจจะใช้ในการจำลองภาพเพื่อให้เห็นลักษณะที่แท้จริงหรือใกล้เคียงกับความจริง

วัตถุประสงค์ของปริญญานิพนธ์

เป็นการศึกษา computer graphics เพื่อใช้ในงานทางด้าน image processing โดยเป็นการจำลองภาพภูมิประเทศ ผลของการจำลองภาพนี้สามารถจะนำไปใช้ประโยชน์ทางด้านการสำรวจพื้นที่ภูมิประเทศจากภาพถ่ายดาวเทียม

ขอบเขตของปริญญานิพนธ์

ในโครงงานนี้จะเป็นลักษณะที่ทำการจำลองภาพ 2 มิติซึ่งเป็นภาพของเส้นระดับความสูงของภูมิประเทศ(contour) ให้อยู่ในรูปของภาพ 3 มิติซึ่งระดับความ สูงต่าง ๆ ของภาพ 2 มิติ จะถูกยกขึ้นเพื่อให้ลักษณะของภาพใกล้ เคียงกับพื้นที่จริงมากที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

โครงสร้างของไฟล์ *.BMP และการใช้งาน

ไฟล์ที่มีนามสกุล *.BMP นั้น จะเป็นไฟล์ ซึ่งมีข้อมูลของภาพอยู่โดยที่ไฟล์ดังกล่าวนั้นจะได้ มาจาก การใช้ scanner (photo styler)

โครงสร้างของไฟล์ *.BMP

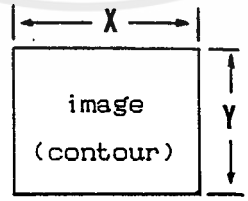
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	42	4D	CE	ED	02	00	00	00	00	00	36	04	00	00	28	00
10	00	00	8C	01	00	00	E2	01	00	00	01	00	08	00	00	00
20	00	00	98	E9	02	00	3A	0F	00	00	3A	0F	00	00	00	00
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.

8c Ø1 จะเป็นจำนวน pixel ของภาพทางด้าน x + header file = $(Ø1\ 8c)_{16}$

E2 Ø1 จะเป็นจำนวน pixel ของภาพทางด้าน y = $(Ø1\ E2)_{16}$

98 E9 Ø2 จะเป็นจำนวน pixel ทั้งหมดของภาพ = $(Ø2\ E9\ 98)_{16}$

ข้อมูลทั้ง 3 ชุดนี้จะอยู่ในส่วนของ header file

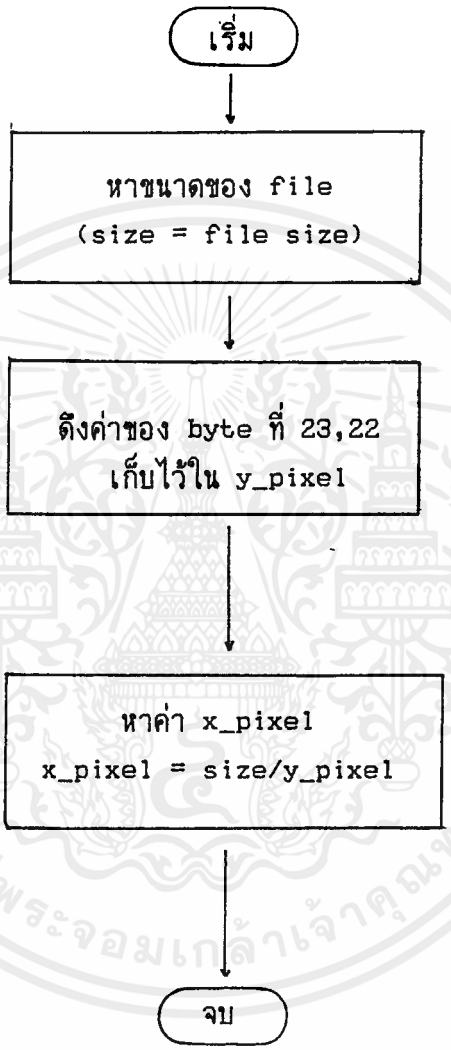


จะสังเกตได้ว่าที่จำนวน pixel ของภาพทางด้าน แกน x จะมี header file บวกอยู่ด้วยซึ่ง header file ก็คือ ข้อมูลในส่วนหัวของไฟล์ จะมีทั้งหมด 1Ø78 byte ดังนั้นข้อมูลของภาพจริงจะเริ่มขึ้น byte ที่ 1Ø78 ข้อมูลใน header file ที่จะนำไปใช้ทำงานคือ จำนวนของ pixel ของภาพทางด้าน y ซึ่งจะเริ่มที่ byte 22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

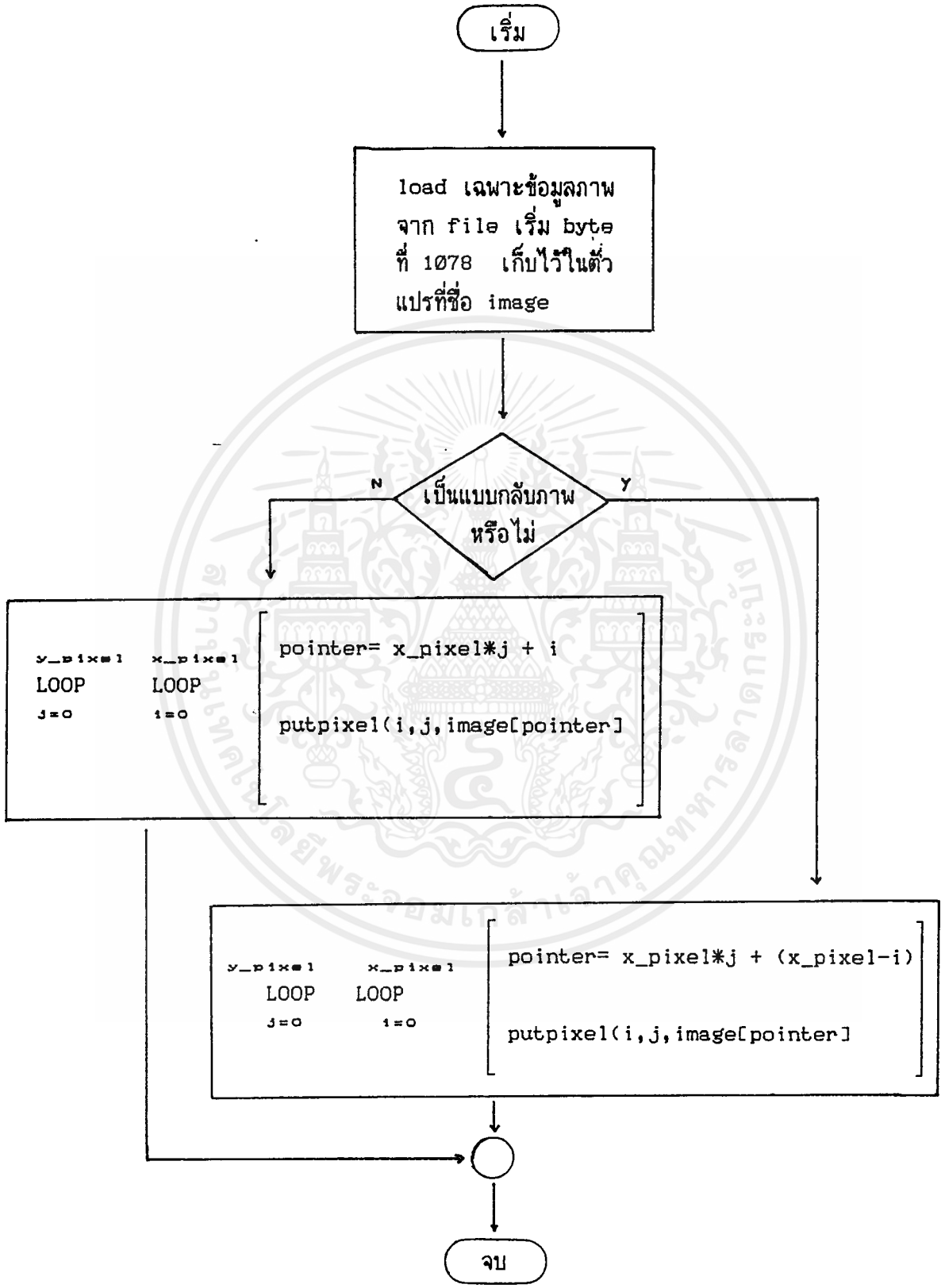
การใช้งาน

algorithm ในการหา ค่าจำนวนของ pixel ทาง ด้าน x (x_pixel)
ในการหา ค่าจำนวนของ pixel ทาง ด้าน y (y_pixel)



จาก algorithm นี้ สามารถหาค่า x_pixel,y_pixel ได้จากนั้นก็จะนำค่าดังกล่าวมาเป็นตัวแปรในการ plot ภาพ ลงบนจอ monitor ซึ่งการ plot แบ่งได้ 2 แบบ คือ แบบปรกติ คือ plot ตามข้อมูลของ file , แบบกลับภาพ(mirror) คือ ภาพที่ได้จากการ plot จะกลับกับภาพที่อยู่บน file ตาม algorithm ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก algorithm ที่ผ่านมา จะเห็นได้ว่า ความแตกต่างระหว่าง การ plot ภาพแบบปรกติกับแบบกลับภาพนั้นต่างกันเพียงค่า pointer ในการชี้ที่จะนำข้อมูลออกแสดงเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

การกำหนดความสูงให้กับ CONTOUR

"contour" ความหมาย ตามโครงงานนี้จะหมายถึง เส้นที่บอกถึงระดับความสูงของภูมิประเทศ โดย contour ที่นำมาใช้ในโครงงานนี้ เป็น contour ของพื้นที่จริง และพื้นที่ดังกล่าวคือ เกาะลันดาน้อย และ เกาะ สมุย โดย contour ดังกล่าว ได้มาจาก กรมแผนที่ทหาร

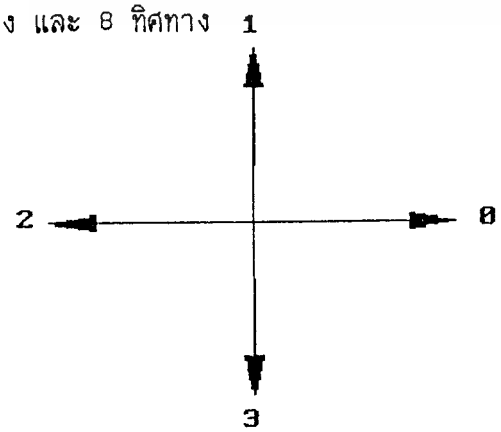
จากนั้นใช้ scanner เก็บภาพดังกล่าว(บทที่ 1) ดังนั้นเมื่อทำการ plot ภาพจาก file ชื่อ LANTANOI.BMP และ ISLET.BMP(สมุย) จึงได้ภาพของ contour ดังนี้

เกาะลันดาน้อย

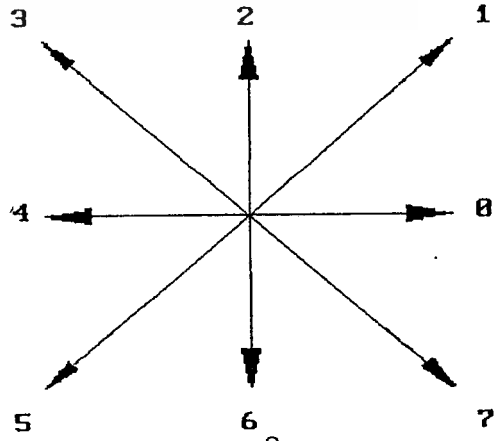
เกาะสมุย

ทฤษฎีสหสัมพันธ์ด้วยลูกโซ่ (chain code correlation)

วิธีหนึ่งจากหลายๆ วิธีในการตรวจสอบว่า ภาพวัตถุทั้งสองเป็นวัตถุเดียวกันหรือไม่นั้นทำได้โดยการตรวจสอบจากรหัสลูกโซ่ของภาพวัตถุทั้งสอง สำหรับรหัสลูกโซ่ที่นิยมใช้กันมี อยู่ 2ชนิด คือ แบบ 4 ทิศทาง และ 8 ทิศทาง

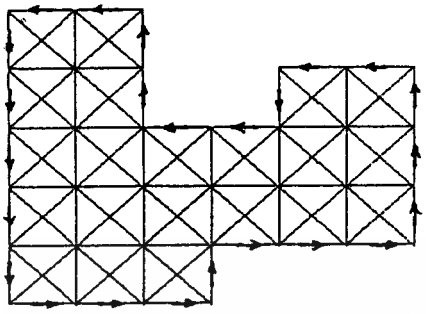


แบบ 4 ทิศทาง

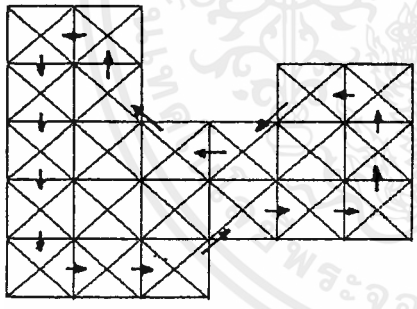


แบบ 8 ทิศทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รหัสลูกโซ่ แบบ 4 ทิศทาง
: 333330001000111223221122

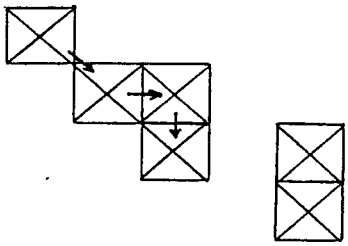


รหัสลูกโซ่ แบบ 8 ทิศทาง
: 66660010022454324

<u>ปัญหาที่เกิดขึ้น</u>

ถ้าภาพ contour ที่ทำการ scan ภาพเข้ามา มีข้อบกพร่องคือ เส้น control line จะไม่ต่อเนื่องกันตลอดทั้งเส้นจะทำให้รหัสลูกโซ่ที่ได้มีข้อผิดพลาดตามไปด้วยและถ้าเส้น control line ในภาพ contour ที่ scan เข้ามามีขนาดใหญ่(หนา) คือ ขนาดของเส้น มีค่ามากกว่า 1 pixel จะทำให้รหัสลูกโซ่ที่ได้มีข้อผิดพลาดคือจะหาได้เฉพาะของของเส้นเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รหัสลูกโซ่ แบบ 8 ทิศทาง
: 706__

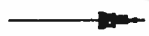
<u>การพัฒนาทฤษฎีเพื่อใช้งาน</u>

จากปัญหาที่เกิดขึ้นจึงแบ่งปัญหาออกเป็น 2ส่วนใหญ่ ๆ คือ

1. เส้น control line ในภาพ contour ขาด
2. เส้น control line ในภาพ contour มีขนาดเส้นหนากว่า 1 pixel

โดยปกติแล้วทฤษฎีสหสัมพันธ์ด้วยลูกโซ่นั้น จะใช้แบบ 8 ทิศทาง แต่มีวงรอบในการตรวจเช็คเพียงวงเดียวจึงทำให้ไม่สามารถเช็คการขาดของเส้นได้ ดังนั้นวิธีการแก้ก็คือการเพิ่มวงรอบในการเช็คให้มากขึ้นอีก 2 วง รวมเป็น 3 วงรอบในการเช็ค ดังรูป

3	2	1
4		0
5	6	7



66	56	46	06	16	26	36
65	55	45	05	15	25	35
64	54	44	04	14	24	34
60	50	40	00	10	20	30
61	51	41	01	11	21	31
62	52	42	02	12	22	32
63	53	43	03	13	23	33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

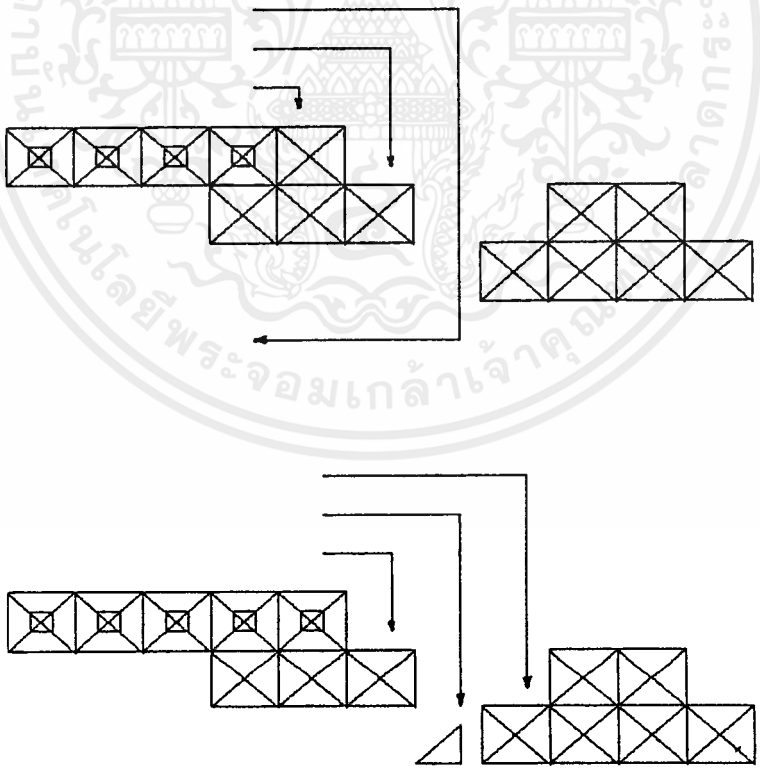
จากการเพิ่มวงรอบในการเช็คจึงต้องทำให้มีการใส่ code ใหม่แทน code เก่าซึ่งมีเพียง 8 ค่า เท่านั้น การใส่ code ใหม่เพื่อให้สะดวกแก่การเช็ค โดยจะต้องมีการถอด code ดังกล่าว

<u>format ของ code</u>

XX
- ค่าที่จะต้องนำไปบวกหรือลบกับตำแหน่ง y ที่จุด reference (จุดกลางของตาราง)
- ค่าที่จะต้องนำไปบวกหรือลบกับตำแหน่ง x ที่จุด reference (จุดกลางของตาราง)

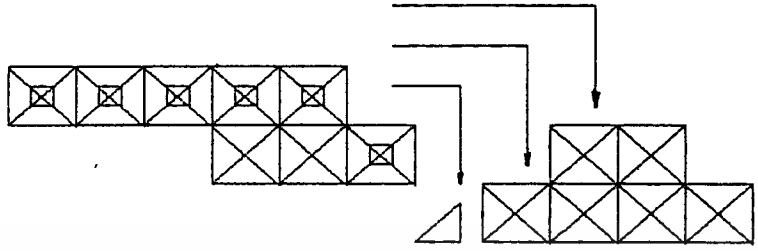
1: บวก 1	4: ลบ1
2: บวก 2	5: ลบ3
3: บวก 3	6: ลบ3

จากการพัฒนาทฤษฎีดังกล่าวนั้นสามารถทำให้แก้ไขข้อบกพร่องที่เกิดขึ้นได้ ดังตัวอย่าง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

032663



จากรูปในช่วงแรกของการเช็คจะได้ค่ามาลักษณะคล้าย ๆ กับทฤษฎีเดิมโดยจะได้ค่ามาเป็น code ซึ่งจะต่อเนื่องกันไป(coordinate จะต่อเนื่องกันไป)จากนั้นจะเริ่มมีปัญหาขึ้นที่ตำแหน่ง วงกลม(ดังรูป) จะเห็นได้ว่าเส้นนั้นขาด แต่เนื่องจากการเช็คนั้นเป็นการเช็คแบบ 3 วงรอบ และ การเช็คในแต่ละครั้งจะมีการเชื่อมต่อจุดระหว่างวงรอบที่เช็คได้ (ถึงแม้จะไม่ขาดก็จะมีการเชื่อมต่อจุดเช่นเดียวกันแต่จะทับลงบนจุดเดิมจึงทำให้เหมือนกับไม่มีการเชื่อมต่อจุด) แต่ถ้าวงรอบเช็คแล้ว ปรากฏว่าวงรอบนั้นไม่มีข้อมูลของเส้น control line เดิม(เนื่องจากมีเส้นบางเส้นถูกเช็คและทำการใส่ค่าระดับความสูงลงไปแล้ว) ในวงรอบนั้นจะไม่มีการเชื่อมต่อกับวงรอบอื่นจะเห็นได้ว่า เส้น control line ที่ขาดถูกเชื่อมต่อแล้ว

ส่วนปัญหาในเรื่องของเส้น control line หนาเกินกว่า 1pixel นั้น วิธีการแก้ก็คือ เมื่อทำการเช็คได้แล้ว จะได้ค่าวงรอบที่ 1-3 ที่มี coordinate อยู่ที่ตำแหน่งใดแล้ว ก็เก็บค่า ดังกล่าวไว้ จากนั้นก็ทำการลบค่า control line ที่ยังไม่ได้ใส่ระดับความสูงวงรอบที่ 1-3 ทับอยู่ จากนั้นจึงใส่ระดับความสูงลงไปใน coordinate ที่เก็บไว้ตอนแรก การทำเช่นนี้จะทำให้เส้น control line ที่ได้มีทั้งระดับความสูงที่มีค่าเป็น code สี และสามารถทำให้ control line ดังกล่าว มีความหนาเพียง 1 pixel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

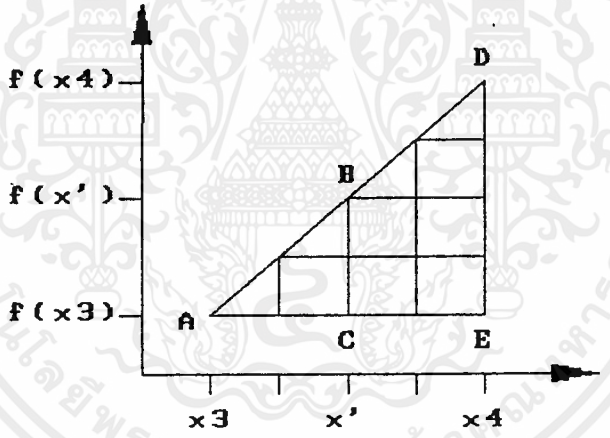
บทที่ 3

Interpolation of Image

การ Interpolate หมายถึง การเฉลี่ยค่าระหว่าค่า 2 ค่า หรือมากกว่า ที่มีความแตกต่างกัน วิธีการ Interpolation นั้นมีอยู่หลายวิธีด้วยกัน เช่น

1. Linear interpolation
2. Interpolating polynomial
3. Lagrangian interpolation
4. Aitker iterated Linear interpolation

นอกจาก 4 วิธีนี้แล้วยังมีวิธีอื่นอีกมาก แต่ในที่นี้มิได้กล่าวไว้ ส่วนโครงงานนี้จะใช้วิธี **"Linear interpolation"** ในการเฉลี่ยความสูงของภาพ (contour)



รูป 3.1

จากการลากเส้นตรงเชื่อมจุดระหว่างจุด $[x_3,f(x_3)]$ และ $[x_4,f(x_4)]$ จากรูปเราจะได้ความสัมพันธ์

$$bc = (ac/ae) * de$$

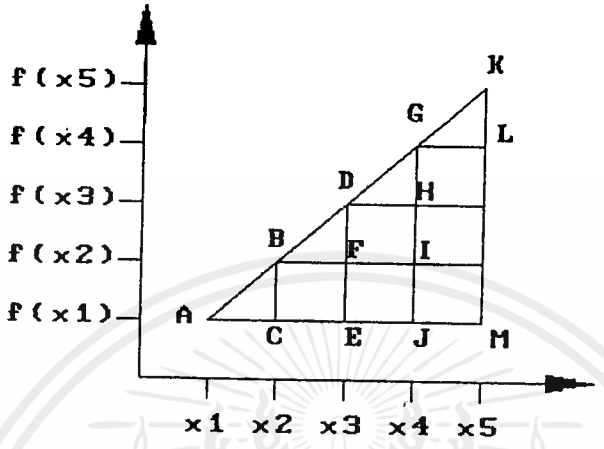
$$= ((x'-x_3) / (x_4-x_3)) * [f(x_4)-f(x_3)]$$

$$f(x') = f(x_3) + ((x'-x_3) / (x_4-x_3)) * [f(x_4)-f(x_3)] \quad ..(3.1)$$

จากสมการเราจะได้ค่า f(x') ซึ่งเป็นค่าที่เฉลี่ยจาก f(x3) และ f(x4) แต่สมการดังกล่าวเมื่อนำไปเขียนเป็นโปรแกรมแล้วจะมีความยากพอสมควร ดังนั้นจึงมีการดัดแปลงสมการดังกล่าวให้เหมาะแก่การเขียนโปรแกรมมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งาน



รูป 3.2

[f(x) ในการใช้งานจริงจะถูกแทนด้วยความสูงของเส้น control line ของภาพ contour]

จากรูปจะทำการ Interpolationระหว่างค่า x_1 ถึง x_5 และ มีการเฉลี่ยช่วงภายใน x_1 ถึง x_5 มี 3 ค่าคือ x_2, x_3, x_4 ดังนั้นถ้าใช้สมการ 3.1 ในการหาค่าจะต้องใช้สมการดังกล่าวในการหาถึง 3 ครั้ง ซึ่งในแต่ละครั้ง ค่า x ต่าง ๆ ในสมการจะต้องเปลี่ยนไปทำให้ยากแก่การเขียนโปรแกรม ดังนั้นจึงมองสมการเสียใหม่ จากรูป 3.2 จะเห็นได้ว่า $x_2-x_1 = x_3-x_2 = x_4-x_3 = x_5-x_4$ ซึ่งจะทำให้ bc = df = gh = kl

พิสูจน์โดยใช้ สามเหลี่ยมคล้าย เช่น bc=df

$$(de+fe)/bc = (ac+ce)/ac$$

fe = bc ; ac = ce

$$(de/bc) + 1 = 1 + 1$$

$$df = bc$$

ส่วนในค่าอื่น ๆ ก็มีลักษณะเช่นเดียวกัน

$$bc = (ac/am) * km$$

ให้ ac =1; $x_2-x_1 = 1$

$$bc = [1/(x_5-x_1)] * f(x_5)-f(x_1)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$f(x_1) = f(x_{1-1}) + bc \qquad ..(3.2)$$

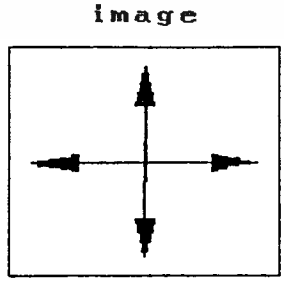
จากสมการจะเห็นได้ว่ามีความง่ายต่อการเขียนโปรแกรมเป็นอย่างมากเพราะค่า bc เป็นค่าคงที่และเป็นการบวกเท่านั้น เช่นถ้าต้องการจะหาค่า $f(x_2)$ ก็เพียงแต่แทนค่า $f(x_1)$ ลงในสูตรแล้วบวกด้วยค่า bc ก็จะได้ค่า $f(x_2)$

จากการ Interpolate ที่ได้จะสังเกตได้ว่าเป็นเพียงการเฉลี่ยค่า ในแนวเดียว หรือ 2 จุด (รูป 3.3) แต่ในความเป็นจริงนั้น จุดที่มีการเฉลี่ยออกมาจะต้องมีการเฉลี่ยของจุดรอบข้างอย่างน้อย 4 จุดหรือ 8 จุด ดังรูป 3.3

แบบ 2 จุด แบบ 4 จุด แบบ 8 จุด

รูป 3.3

ในโครงงานนี้มีการเฉลี่ยของจุดรอบข้าง 4 จุด ดังรูป 3.4



รูป 3.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นการ Interpolate จึงเกิดขึ้น 2 ครั้งคือ การ Interpolate แกน x และ แกน y จากนั้นจึงทำการเฉลี่ยค่าที่ได้จากการInterpolate แกน x กับ แกน y โดยเฉลี่ยแบบจุดต่อจุดก็คือ การนำเอาค่าของการ Interpolate ที่ตำแหน่งเดียวกันของทั้งแกน x และ แกน y มาบวกกันแล้วหารด้วย 2 แล้วนำค่าที่ได้เก็บไว้ในตำแหน่งเดิมของรูปภาพ (Image) ทำเช่นนี้ตลอดทั้งภาพเราก็จะได้ภาพที่มีการ Interpolate แบบ linear Interpolation ทั้งภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่4

3-DIMENSION

ภาพในรูปของ 3 มิติ

ในส่วนนี้เราจะพิจารณาถึงการจะทำอย่างไรให้รูป 3 มิติสามารถแสดงผลบนจอ Computer ได้ ซึ่งก็คือการแสดงผลที่อยู่ในรูป 2 มิตินั่นเอง ในการสร้างนั้นเราจะใช้วิธี ที่เรียกว่า "Perspective projection" ก็คือ การสร้างภาพ 2 มิติ โดยการจำลองการมองเห็นของภาพ 3 มิติ (การมองเห็นของมนุษย์ที่ประสาทรับสัญญาณภาพ(Ratina)เป็นแบบ 2 มิติ แต่เราสามารถมองเป็น 3 มิติ เพราะเราใช้ความรู้สึกร่วม กับการมองเห็น) เราจะพิจารณา Perspective Project ซึ่งมีอยู่ 2 แบบคือ "the central prejection" และ the arbitrary projection อนึ่ง ในการกระทำ Perspective Projection จาก 3-D ให้เป็น 2-D สิ่งซึ่งสำคัญคือ การเปลี่ยนระบบจาก the world coordinate system ไปเป็น the eye coordinate system ซึ่งมีความจำเป็นมากในโครงงานนี้ จุดประสงค์ของการเปลี่ยนระบบของภาพเพื่อการคำนวนตำแหน่งบนจอที่แสดงภาพ(ขนาดของจอมอนิเตอร์เอง มีความละเอียดไม่เท่ากัน) ซึ่งจะเป็นจุดต่าง ๆ บนภาพ 3 มิติ

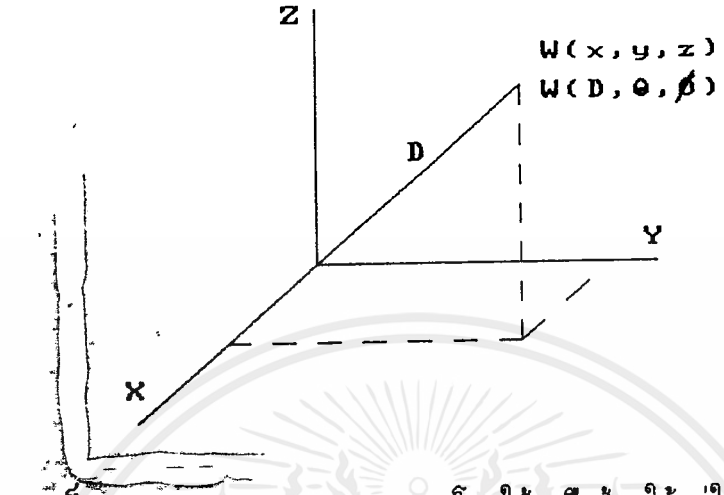
center projection

ในรูปที่ 4.9 จะใช้ภาพในการแสดงถึงการกระทำ center projection โดยการกำหนดตำแหน่งขึ้นมา 1 จุดบนแกนของ world coordinate (e.c.s) และจุดดังกล่าวก็จะเป็นที่ตั้งของระบบ eye coordinate(e.c.s) และเป็นจุดกึ่งกลางของการ project(จุดที่ Ø,Ø ของระบบ e.c.s) เราจะให้ จุดกึ่งกลางของการ project ตั้งอยู่บนแกน Z ในความหมายนี้คือการทำให้แกน Z ของทั้ง 2 ระบบ ทับกัน ดังนั้น display screen(X_s,Y_s) จะขนานกับxy-plane(ดังรูป 4.9.b)เราจะได้ Perspective display ซึ่งเกิดจากการ project จุดต่างของวัตถุที่เป็น 3 มิติในระบบ world coordinate มายัง display screen ซึ่งเป็น 2 มิติ ดังรูป 4.9

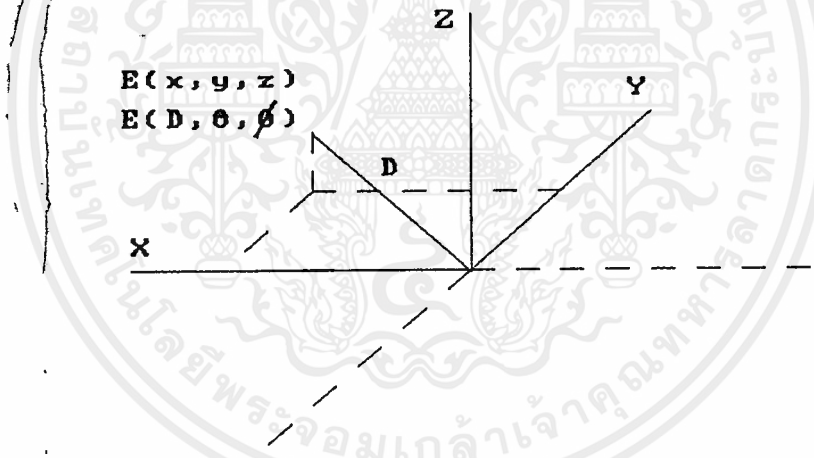
เมื่อเรามาพิจารณาความสัมพันธ์ของ e.c.s นั้น เราจะได้ว่า e.c.s. เป็นไปตามกฎมือซ้าย (left handed coordinate system (ระบบของ E.C.S คือ X_s Y_s, Z_s) โดยที่ Z_s จะเป็นแกนที่ต่อไปยังจุด origin(Ø,Ø) ของระบบ W.C.S. X_sจะเป็นแกนที่มีทิศไปทางขวา และ Y_sจะเป็นแกนที่มีทิศขึ้นด้านบน (ดังรูป 4.9.b) และเมื่อมาพิจารณาถึงW.C.S นั้นจะเป็นไปตามกฎมือขวา (right-handed systems)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากระบบ cartesian coordinate นั้น เราสามารถมองในระบบทรงกลม (spherical coordinates) ได้โดยจะเขียนอยู่ในรูปของเวกเตอร์ ได้เป็น W(D,Θ,Ø)



ดังนั้น เราก็สามารถจะมอง coordinate โดยใช้กฎมือซ้ายให้อยู่ในระบบทรงกลมได้เช่นกัน ดังนี้ e(D,270,Ø)

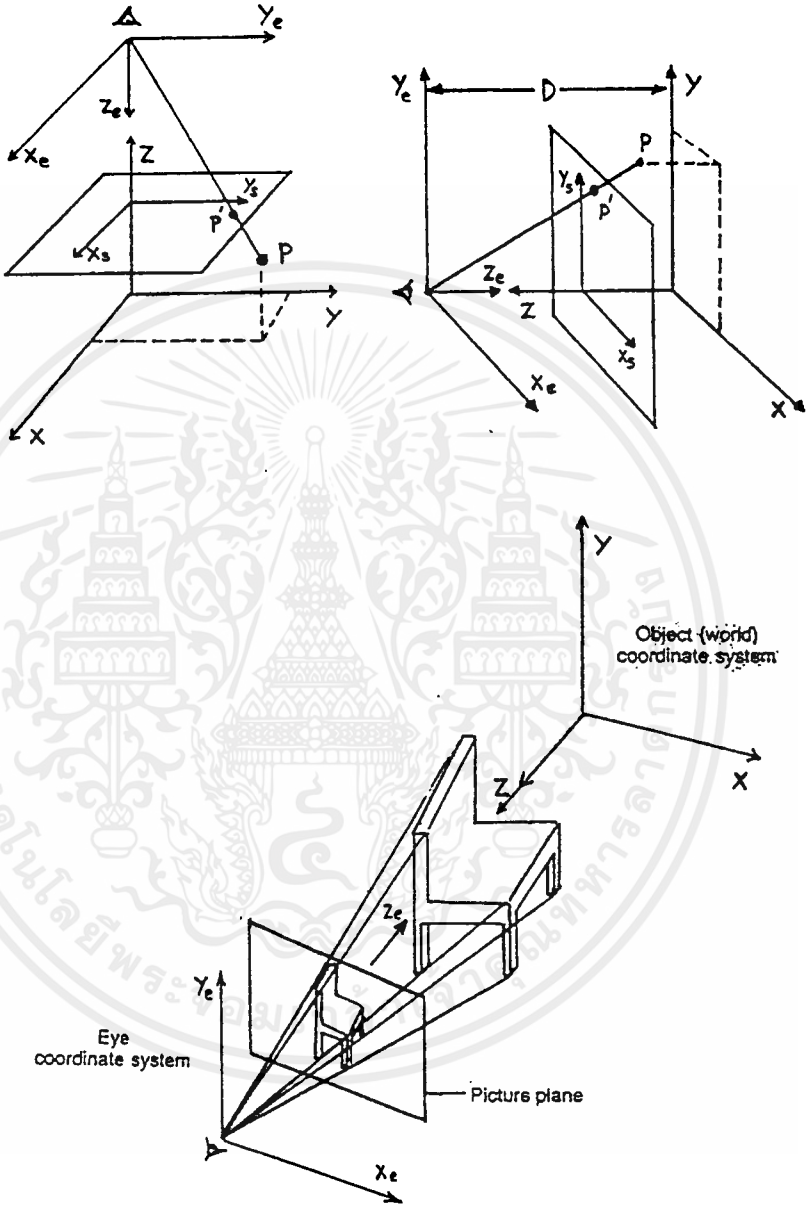


จากการแปลงระบบ cartesian (x,y,z) ให้เป็นระบบ spherical (ทรงกลม) (D,Θ,Ø) ทั้งระบบที่เป็นกฎมือขวา และกฎมือซ้ายมาแล้ว ต่อไปจะนำการแปลงดังกล่าวมาพิจารณารูป 4.9.b โดยจะพิจารณาที่ e.c.s.คือจะสามารถแปลงเป็นรูปเวกเตอร์ได้เป็น (D,270,0) (โดยที่ Ø = 0 เพราะระยะทาง D อยู่ในแกน Z)

<u>การแปลงของภาพ</u>

การคำนวณตำแหน่งบน display screen ของจุดต่าง ๆ ของรูปนั้น เราต้องทำการแปลงจุดต่าง ๆ ดังกล่าวที่อยู่ในระบบ W.C.S. มาอยู่ในรูปของ ระบบ e.c.s. เสียก่อน ซึ่งตำแหน่งของจุด origin ที่ e.c.s. จะอยู่ตำแหน่งในแนวเดียวกับจุด origin ของ w.c.s. ตามแนวของ Z และ Z_ ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูป a. จุด P(x,y,z) เป็นจุดที่อ้างอิงอยู่กับ W.C.S. เมื่อเราสังเกตจุด P ที่จุดของการมอง (o_e) จะเห็นได้ว่า ค่า x และ y จะไม่เปลี่ยนแปลง ดังนั้น จุดของการมองจะมีระยะห่าง D จากระนาบ xy ซึ่งจะเป็นผล ต่อตำแหน่งบนแกน Z_e เท่ากับ D-Z

การแปลงจุด P ซึ่งอ้างอิงอยู่กับ W.C.S. มาอ้างอิงอยู่กับ e.c.s. ได้ดังนี้

$$(X,\ Y,\ D-Z) \ = \ (X_e,\ Y_e,\ Z_e)$$

จากหลักการง่าย ๆ ดังกล่าว สามารถจะนำมาเขียนเป็นรูปของ MATRIX ได้ดังนี้

<u>ขั้นที่ 1</u> แปลงจุด ORIGIN ของ (X,Y,Z) มาเป็น ORIGIN ของ (X_e,Y_e,Z_e)

$$T_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -D & 1 \end{bmatrix}$$

<u>ขั้นที่ 2</u> กลับทิศทางของ Z_e

$$T_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

<u>ขั้นที่ 3</u> รวมคุณสมบัติของขั้นตอนที่ 1 และ 2

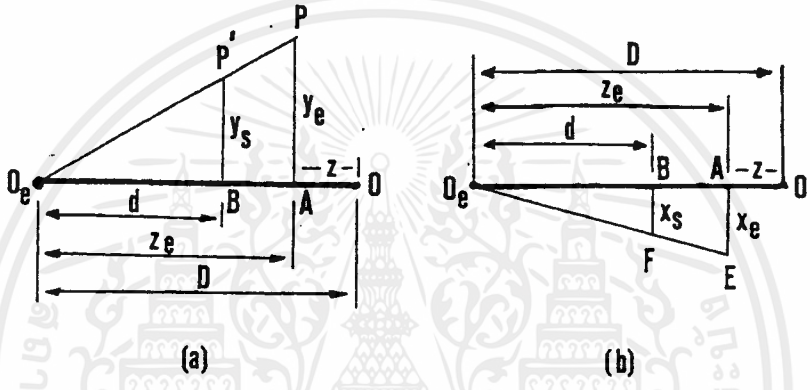
$$T = T_1.T_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & D & 1 \end{bmatrix}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นที่ 4 แปลงจาก (X,Y,Z) เป็น (X_e,Y_e,Z_e)

$$(X_e,Y_e,Z_e,1) = (X,Y,Z,1)\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & D & 1 \end{bmatrix}$$

$$= (X,Y,D-Z,1)$$



รูป 4.10

การแปลงเป็น screen coordinate

จุดP ในรูป4.10 ถูกอ้างอิงใน eye coordinate(X_e,Y_e,Z_e) ซึ่งจุดดังกล่าวจะถูก project ให้มาอยู่ใน screen coordinates ซึ่งการคำนวณนั้นในการ project นั้นเราพิจารณากับ ระนาบ Y_e,Z_e ดังรูป 4.10(a) โครงสร้างของระนาบ Y_e, Z_e นั้นจะถูกจัดให้เป็นฐานดังรูป 4.9(b) เราได้กำหนดค่าตัวแปร "d"ขึ้นมาเพื่อเป็นระยะทางระหว่าง display screen กับ View point(จุดมอง) จากนั้นเราก็อาศัยสามเหลี่ยมคล้าย A O_e P และ B O_e P' เพื่อทำการคำนวณในการ project จาก eye coordinate มาอยู่ใน screen coordinate (display screen) ได้ดังนี้

(ที่แกน y ดังรูป 4.10 a)

$$\frac{Y_s}{d} = \frac{Y_e}{Z_e}$$

$$\therefore \quad Y_s = \frac{d\,Y_e}{Z_e}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการหา x_s นั้น เราจะพิจารณารูป 4.10 (b) ซึ่งจะใช้ระนาบ X_s Z_s ในรูป 4.9 (b) โดยอาศัยสามเหลี่ยมคล้าย A O_s P และ B O_s F ในการหา X_s ดังรูป 4.10 (b)

$$\frac{X_s}{d} = \frac{X_s}{Z_s}$$

$$\therefore \quad X_s = \frac{d\, X_s}{Z_s}$$

<u>การ project ในลักษณะที่จุดมองอยู่ตำแหน่งใด ๆ</u> **(the arbitrary projection)**

เมื่อเราต้องการจุดมองให้อยู่ในตำแหน่งใด ๆ ใน 3 มิติ(แกน x, y, z) เราจะต้องกำหนดตำแหน่งของ display screen ห่างจากจุดมอง (view point) เป็นระยะทาง "d" โดย display screen จะตั้งฉากกับเส้นที่เชื่อมต่อกับจุดมองกับจุด ORIGIN ในระบบ eye coordinate ทิศทางของแกน Z_s จะตรงไปยัง ORIGIN ของ World coordinate และแกน x_s จะมีทิศทางไปทางขวา ส่วน Y_s จะมีทิศทางขึ้นบน ในรูป 4.11 แสดงให้เห็นถึงความแตกต่างระหว่าง การproject ที่จุดกึ่งกลาง (ตามแกน Z) กับการ project ที่จุดมองอยู่ในตำแหน่งใด ๆ

ในลักษณะเดียวกัน screen coordinate (x_s, y_s) มีความสัมพันธ์กับจุด P(x,y,z) เหมือนกับการคำนวณในลักษณะการ project ที่จุดกึ่งกลางซึ่งมีลักษณะ การกระทำ 2 แบบ ดังนี้

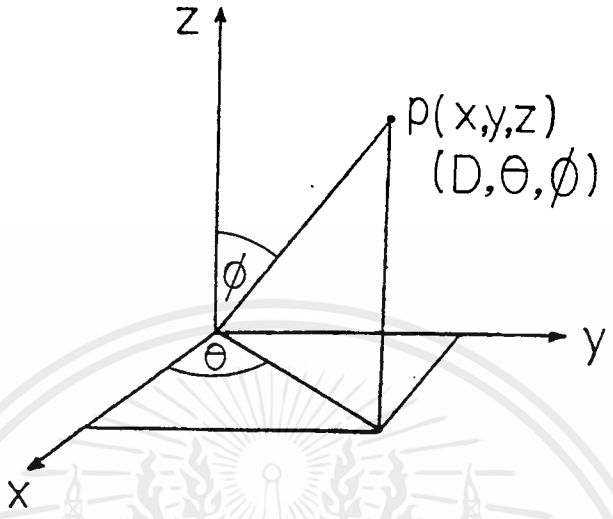
1. แปลงค่า world coordinates(x,y,z) ให้มาอยู่ในeye coordinates (x_s,y_s,z_s)
2. แปลงค่า eye coordinates (x_s,y_s,z_s)ให้มาอยู่ใน screen coordinates (x_s,y_s)

จากรูป 4.2 , 4.11 จะเป็น coordinate แบบทรงกลม (D,θ,ϕ) โดยมีสมการพื้นฐานดังนี้

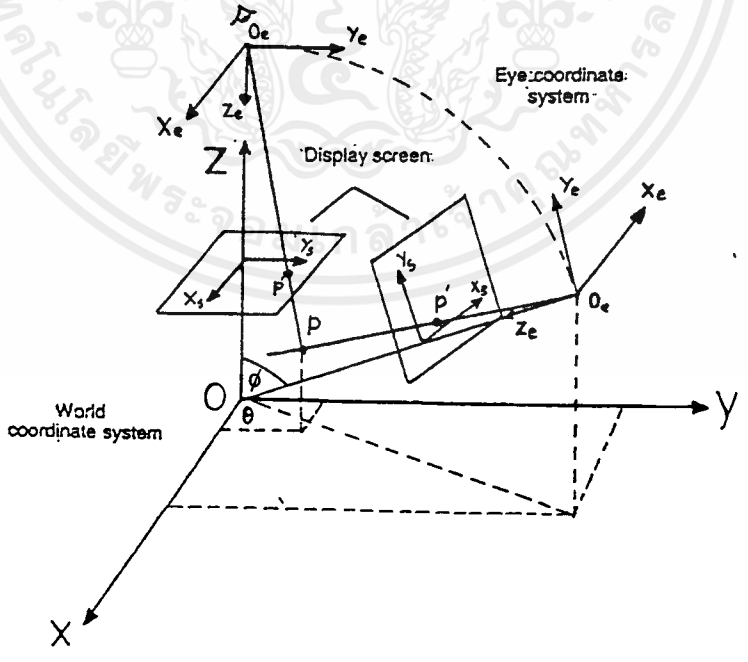
$$X = D.\sin\phi \cos\theta$$
$$Y = D.\sin\phi \sin\theta$$
$$Z = D \cos\phi$$

$$D = X^2 + Y^2 + Z^2$$
$$\theta = \tan^{-1}(Y/X)$$
$$\phi = \cos^{-1}(Z/D)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

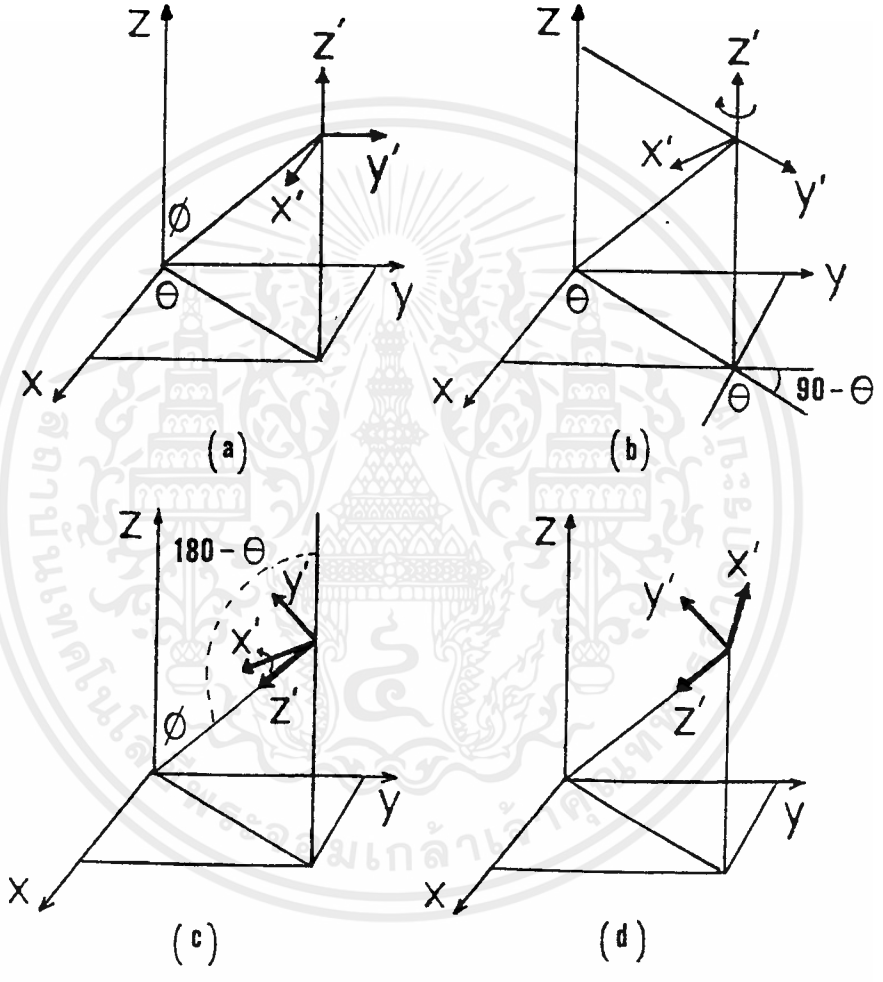


รูป 4.2



รูป 4.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสมการพื้นฐานเราสามารถใช้สมการดังกล่าวทำการแปลง coordinate ได้ดังนี้ (ดูรูป 4.12 ประกอบ)

1) แปลงจุด ORIGIN ไปอยู่ที่ O_e กล่าวคือ เปลี่ยนจุด ORIGIN ให้มาอยู่ใน coorginates ใหม่ ซึ่งก็คือ eye coordinate ซึ่งสามารถเปลี่ยนโดยอาศัย MATRIX คือ

$$T_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -D\sin\phi\cos\theta & -D\sin\phi\cos\theta & -D\cos\phi & 1 \end{bmatrix}$$

2) หมุนแกน Z' ของระบบ (eye coordinate) แบบตามเข็มนาฬิกาเท่ากับ $90-\theta$ องศา แต่เราทราบว่า $\sin(90-\theta) = \cos\theta$ และ $\cos(90-\theta) = \sin\theta$ ดังนั้น เราสามารถนำมาเขียน MATRIX ได้ดังนี้

$$T_2 = \begin{bmatrix} \sin\theta & \cos\theta & 0 & 0 \\ -\cos\theta & \sin\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3) หมุนแกน X' ของระบบ แบบทวนเข็มนาฬิกาเท่ากับ $180-\phi$ องศา แต่เราทราบว่า $\sin(180-\phi) = \sin\phi$ และ $\cos(180-\phi) = -\cos\phi$ จากผลดังกล่าว สามารถนำมาเขียนเป็น MATRIX ได้ดังนี้

$$T_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -\cos\phi & -\sin\phi & 0 \\ 0 & \sin\phi & -\cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

เมื่อกระทำ MATRIX นี้เสร็จแล้วจะทำให้แกน Z' มีทิศทางชี้ไปยัง ORIGIN ของ world coordinates

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) แปลงจากกฎมือซ้ายให้มาอยู่ในรูปของกฎมือขวา โดยการกลับทิศทางของแกน X' ซึ่งสามารถนำมาเขียนเป็น MATRIX ได้ดังนี้

$$T_4 = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

จากคุณสมบัติทั้ง 4 ข้อ ที่กล่าวมาจะต้องนำมาใช้ในการแปลง coordinate ดังนี้จึงมีการรวมคุณสมบัติทั้ง 4 ข้อ ไว้ดังนี้

$$T = T_1 \cdot T_2 \cdot T_3 \cdot T_4$$

∴ T คือ MATRIX ที่ใช้ในการแปลง coordinates ซึ่งรวมคุณสมบัติทั้ง 4 ข้อที่กล่าวมาแล้ว

$$T = \begin{bmatrix} -\sin\theta & -\cos\theta\cos\phi & -\cos\theta\sin\phi & 0 \\ \cos\theta & -\sin\theta\cos\phi & -\sin\theta\sin\theta & 0 \\ 0 & \sin\phi & -\cos\phi & 0 \\ 0 & 0 & D & 1 \end{bmatrix}$$

จาก MATRIX ดังกล่าวสามารถจะนำมาเขียนเป็นสมการ ได้ดังนี้

$$(X_e, Y_e, Z_e, 1) = (X, Y, Z, 1).T$$

หรือ

$$X_e = -X\sin\theta + Y\cos\theta$$
$$Y_e = -X\cos\theta\cos\phi - Y\sin\theta\cos\phi + Z\sin\phi$$
$$Z_e = -X\cos\theta\sin\phi - Y\sin\theta\sin\phi - Z\cos\phi + D$$

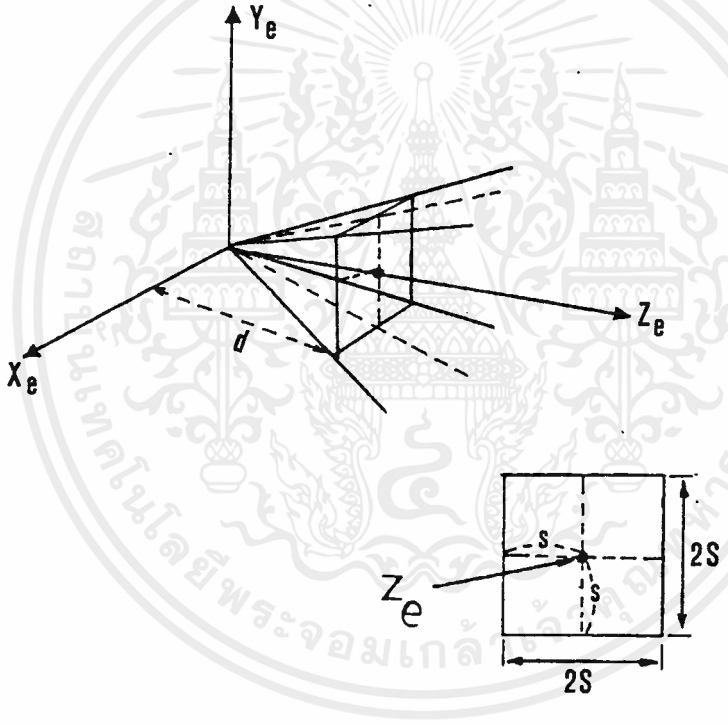
จากสมการข้างต้นจะเห็นได้ว่า เราได้ค่าของตำแหน่ง X_e, Y_e, Z_e ที่อ้างอิงกับ eye coordinates จากนั้นเราก็ใช้วิธีการเดิมที่กล่าวมาแล้ว คือ เปลี่ยนจาก eye coordinate เป็น screen-coordinate ซึ่งได้กล่าวมาแล้วข้างต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$X_{\blacksquare} = \frac{d\,X_{\blacksquare}}{Z_{\blacksquare}}$$

$$Y_{\blacksquare} = \frac{d\,Y_{\blacksquare}}{Z_{\blacksquare}}$$

จากสมการที่ผ่านมา $X_{\blacksquare}$ และ $Y_{\blacksquare}$ นั้น จะขึ้นอยู่กับค่า d แต่ในความเป็นจริง แล้วจะต้องขึ้นกับตัวแปรอีกหลายตัว ซึ่งจะอธิบายเป็นลำดับต่อไป



รูป 4.13

จากรูป จะเห็นได้ว่ามี ตัวแปร "s" เข้ามาเกี่ยวข้องเพราะตัวแปร "s" เป็นค่าที่ใช้ในการกำหนดขนาดของ screen ดังนั้น สมการจึงเขียนใหม่ได้เป็น

$$X_{\blacksquare} = \frac{d\,X_{\blacksquare}}{s\,Z_{\blacksquare}}$$

$$Y_{\blacksquare} = \frac{d\,Y_{\blacksquare}}{s\,Z_{\blacksquare}}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพบนจอ COMPUTER

หลังจากที่ทำการแปลง coordinate ต่าง ๆ มาจนกระทั่งได้เป็น screen coordinates แล้ว coordinates ดังกล่าวก็ยังไม่สามารถจะนำมา plot บนจอ computer ได้จะต้องมีตัวแปรอื่นๆ เข้ามาเกี่ยวข้อง ดังนั้นสมการที่สามารถจะนำไปใช้งานและนำค่าที่คำนวนได้จากสมการมา plotบนจอ computer ได้จริง มีดังนี้

$$X_s = SCF\ (d/s)\ (X_s/Z_s)\ V_x + L$$
$$Y_s = -(d/s)\ (Y_s/Z_s)\ V_y + M$$

scf(screen scaling adjustment factor) = v/h

$$v = \frac{\text{ความยาวของ screen ทางด้าน vertical}}{\text{จำนวน pixel สูงสุด ทางด้าน vertical}}$$

$$h = \frac{\text{ความยาวของ screen ทางด้าน horizontal}}{\text{จำนวน pixel สูงสุด ทางด้าน horizontal}}$$

d ระยะห่างจากจุดกึ่งกลางของ screen กับจุดศูนย์ของ eye coordinate (รูป3.2)
s ระยะห่างจากจุดกึ่งกลางของ screen กับขอบด้านใดด้านหนึ่งของ screen (รูป3.2)
l ตำแหน่งของจุดกึ่งกลางของ screen ในแนวนอน (x)
m ตำแหน่งของจุดกึ่งกลางของ screen ในแนวตั้ง (y)
v_x เป็น factor ในการคูณทาง horizontal
v_y เป็น factor ในการคูณทาง vertical

สรุปทฤษฎีเพื่อนำมาใช้งานจริง(program)

สิ่งแรกที่จะกล่าวคือ world coorinate ซึ่งจะเป็นระบบที่รูปภาพที่มี 3 มิติ(กว้าง,ยาว,ลึก) อ้างถึงอยู่กับระบบนี้ ส่วน eye coordinate จะเป็นระบบที่ตาของผู้มองอ้างอิงอยู่และ screen coodinate ก็เป็นระบบที่อ้างอิงอยู่กับ eye coordinate (รูป4.1) การมองภาพ 3 มิติ ให้ปรากฏบนจอคอมพิวเตอร์ (2มิติ) นั้น จะกล่าวได้อีกอย่างก็คือ การเปลี่ยนระบบ coordinate จาก world coordinate ให้อยู่ใน sereen coordinate

ขั้นตอนในการเปลี่ยนระบบจาก world coordinate ไปเป็น screen coordinate จะมีขั้นตอนใหญ่ๆ 2 ขั้นตอน คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 1 เปลี่ยนจาก world coordinate ไปอยู่ในระบบ eye coordinate
ขั้นตอนที่ 2 เปลี่ยนจาก eye coordinate ไปอยู่ในระบบ screen coordinate

ขั้นตอนที่ 1

การเปลี่ยนจาก world coodinate ไปอยู่ในระบบ eye coodinate จะมีสูตรทางด้านคณิตศาสตร์ เข้ามาเกี่ยวข้อง (ซึ่งเป็นผลมาจากการคูณกับของ matrix หลาย ๆ matrix) ดังนี้

$$x_e = -x\sin\theta + y\cos\theta$$
$$y_e = -x\cos\theta\cos\phi - y\sin\theta\cos\phi + z\sin\phi$$
$$z_e = -x\cos\theta\sin\phi - y\sin\theta\sin\phi - z\cos\phi + D$$

D คือระยะห่างระหว่างจุดศูนย์ของworld coordinate กับจุดศูนย์ของ eye coordinate

การใช้งานสมการดังกล่าวนั้นเมื่อเรามี ภาพที่เป็น 3 มิติ โดยภาพดังกล่าวเราจะให้อยู่ใน world coordinate เสมอ ดังนั้นเราก็จะได้ค่า x,y,z ออกมาด้วย จากนั้นเราก็นำเอาค่าดังกล่าวมาแทนในสมการเราก็จะได้ค่า x,y,z ใหม่ เป็น x_e, y_e, z_e คือ จะมีการเปลี่ยนจุดอ้างอิงของภาพดังกล่าวจาก world coordinate มาอยู่ในระบบ eye coordinate (รูป 3.3)

ขั้นตอนที่ 2

เมื่อทำการเปลี่ยน coordinate ตามขั้นตอนที่ 1 แล้ว จากนั้นก็ทำการเปลี่ยนระบบอีกอย่างหนึ่ง คือจาก eye coordinate มาอยู่ใน screen coordinate แต่การเปลี่ยนระบบในครั้งนี้อาจจะเรียกอีกอย่างหนึ่งว่า การ projection ก็ได้ โดยจะมีสมการในการ ดังนี้

$$x_s = scf*[\ (d/s)*(x_e/z_e)*v_x + 1\ ]$$
$$y_s = -(d/s)*(y_e/z_e) + m$$

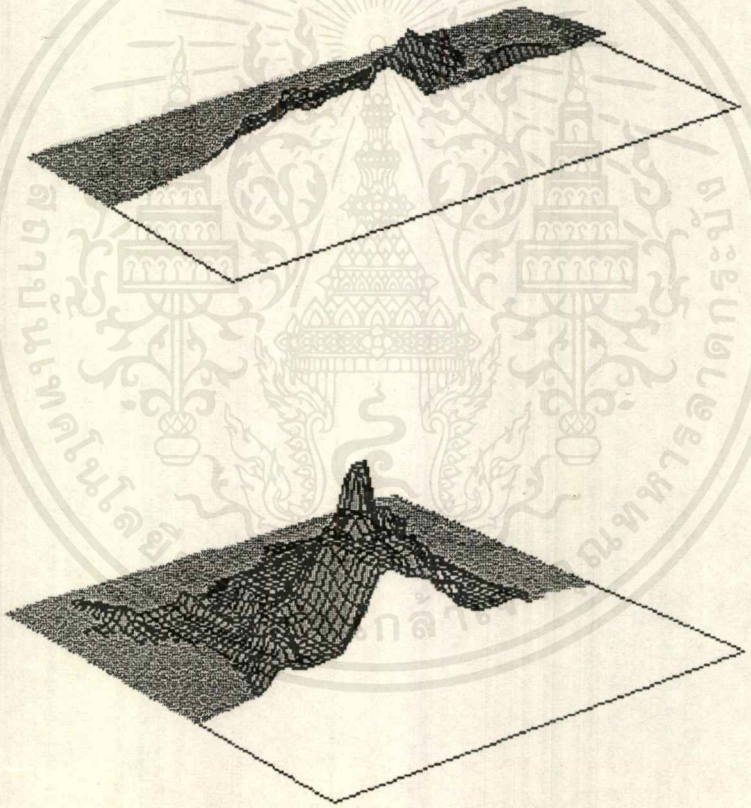
จากรูป 4.11 เราจะเห็นได้ว่า ถ้าต้องการจะหมุนภาพเพื่อจะมองในมุมต่างๆนั้นก็เพียงแต่เปลี่ยนค่าของ θ , ϕ เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำ HIDDEN LINE และ การแยกระดับความสูงด้วยสี

การทำ hidden line นี้จะอาศัยเทคนิคในการใช้ภาษา c โดย function ที่ใช้คือ fillpoly() การใช้ function นี้ จะต้องกำหนดตำแหน่ง 4 ตำแหน่งและจะใช้ร่วมกับ function setcolor() และ setfillstyle() โดยที่ setcolor() นั้นจะเป็นการ set สีที่ขอบของสี่เหลี่ยมที่กำหนดโดยตำแหน่งทั้ง4(input ของ fillpoly) และ setfillstyle() จะเป็นการ set สีภายในกรอบสี่เหลี่ยมนั้น

จากหลักการดังกล่าว เราจะนำภาพที่ทำการ interpolate แล้วมา plot โดยจะนำมา plot ครั้งละ 4 จุด แต่ละจุดห่างกัน 10 pixel และการ plot จะ plot จากด้านในของภาพ 3 มิติเสมอ จากการ plot ดังกล่าวจะเกิดการ hidden line กล่าวคือ block(10 x 10 pixel) จากด้านในจะถูก block จากด้านนอกทับเสมอเมื่อมุมมองของภาพทับกัน ดังรูป

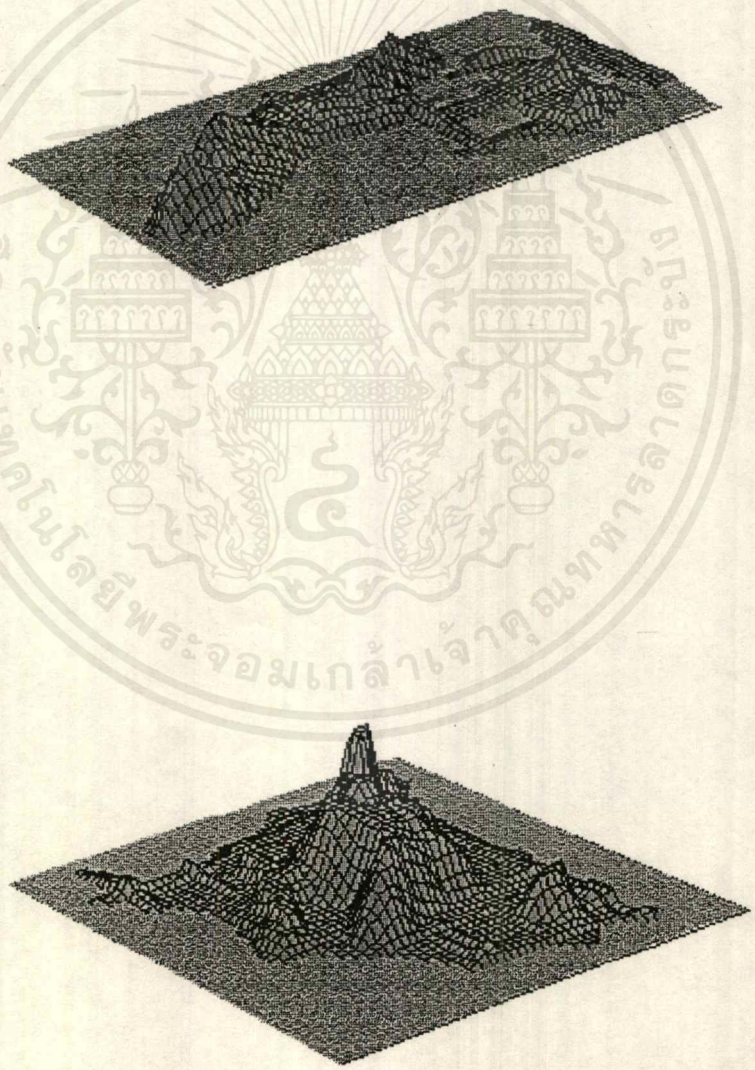


การแยกระดับความสูงด้วยสีนั้นเราอาศัยการเช็คความสูงของภาพแล้วทำการกำหนดสีที่function setcolor() จากการกำหนดสีดังกล่าวจะทำให้ความสูงของภาพ 3 มิติ ถูกแบ่งด้วยสีที่ขอบของ block เล็กๆเหล่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากทฤษฎีที่ผ่านมาถ้าเราทราบค่า x,y,z ของภาพใดๆ เราก็สามารถจะ plot ภาพนั้นออกมาทางจอคอมพิวเตอร์ได้(x_s,y_s)

จากบทที่ผ่านมาเราจะได้ภาพที่มีทั้งความกว้าง(x pixel), ความยาว(y pixel) และความสูง(linear interpolation)(บทที่3) ดังนั้นเราก็เพียงแต่นำค่าต่างๆ ที่กล่าวมาผ่านสูตรในการเปลี่ยนระบบจาก world coordinate มาเป็น screen coordinate แล้ว plot แบบ hidden line พร้อมกับเช็คระดับความสูงเพื่อกำหนดสีแสดงความแตกต่างของระดับความสูง เราก็จะได้ภาพ 3 มิติปรากฏบนจอคอมพิวเตอร์(ซึ่งเป็น 2 มิติ) ดังรูป



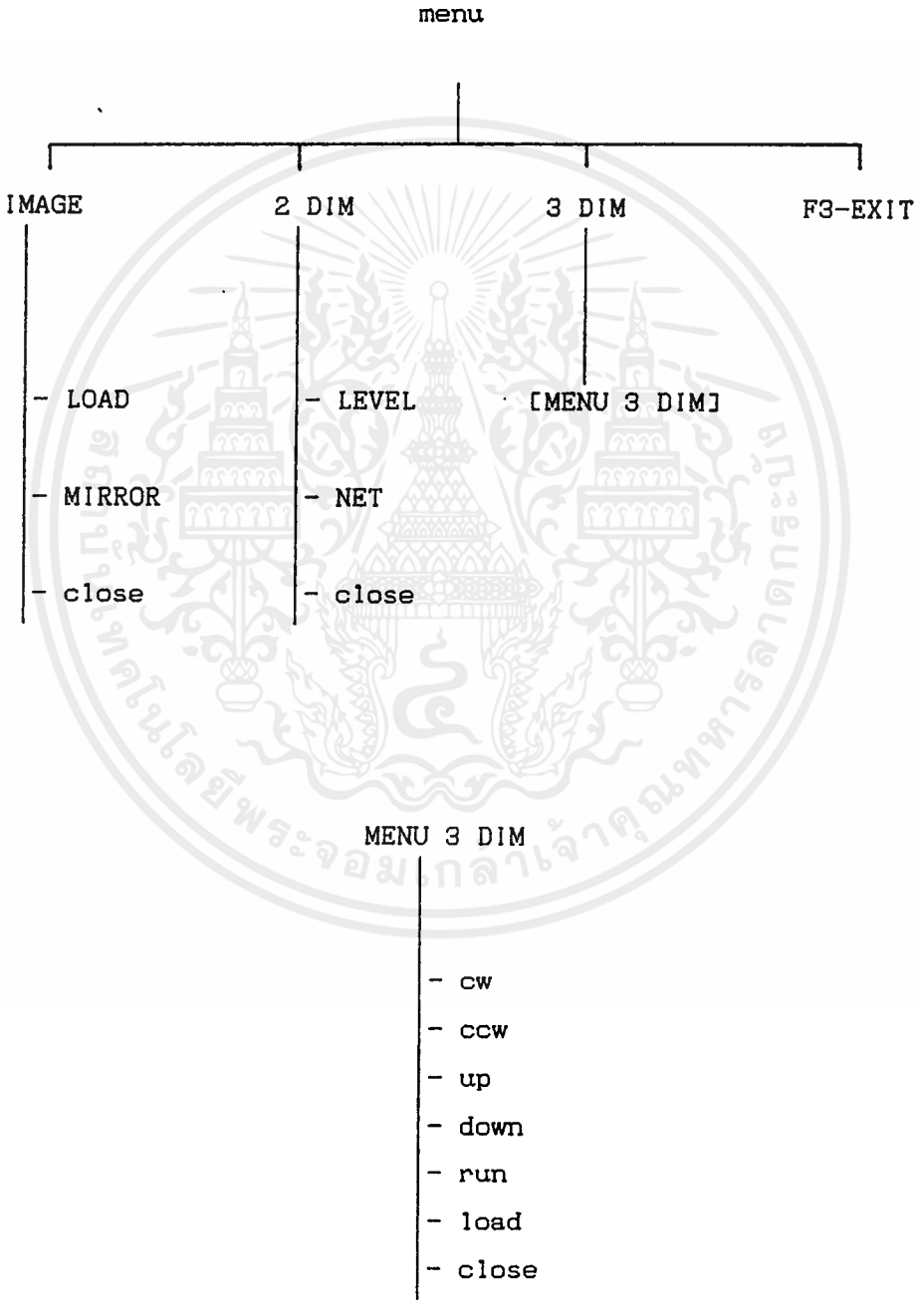
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การใช้งาน Program

ในโครงงานนี้จะมีลักษณะการใช้งานเป็น menu และใน menu จะมีคำสั่งต่างๆ ตามผัง ดังนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดในการใช้คำสั่งต่างๆ

LOAD	เป็นคำสั่งในการ load ข้อมูลจาก file *.bmp มาแสดงผลที่จอภาพ
MIRROR	เป็นคำสั่งในการ load ข้อมูลจาก file *.bmp มาแสดงผลที่จอ แต่มีลักษณะกลับด้านแบบกระจก
LEVEL	เป็นคำสั่งในการ set ระดับของเส้น control line โดยการ set ระดับนี้จะ set ค่าได้ตั้ง แต่ 1-255 ระดับ (ระดับ"o" ไม่ต้องทำการ set เครื่องจะ defualt ไว้ที่ ระดับ"o" ถ้ามีการป้อนค่าระดับเกิน 255 จะทำให้ภาพที่ได้ออกมาเกิด error(ไม่เหมือนกับความเป็นจริง)
NET	เป็นคำสั่งทีมีหน้าที่ 2 อย่างคือ 1. INTERPOLATE ภาพที่มีการ set ระดับความสูงแล้ว 2. SAVE ภาพที่มีการ INTERPOLATE (ควรให้นามสกุลเป็น *.NET)
cw	หมุนภาพ 3 มิติ แบบ ตามเข็มนาฬิกา
ccw	หมุนภาพ 3 มิติ แบบ ทวนเข็มนาฬิกา
up	มอง object(3มิติ) ในมุนบน
down	มอง object(3มิติ) ในมุนล่าง
run	plot ภาพ 3 มิติ
load	load ข้อมูลจาก file ที่มีการ interpolate (ควรที่จะเป็นนามสุกล *.net)
close	ปิด menu นั้นๆ
F3(EXIT)	ออกจากโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับขั้นการใช้งาน

1. ใช้ program photo styleและเครื่อง scanner ทำการ scan ภาพที่ต้องการ(ภาพ contour ของภูมิประเทศ) โดยเมื่อ scan เสร็จแล้วโปรแกรมจะให้เราเลือกว่าเป็น format file แบบไหน จากนั้นก็เลือกไปที่ BMP แล้วทำการ save file ดังกล่าว

2. ใช้ program three dimension of contour(project) ทำการ load file ที่ได้จากข้อ 1 โดยเลือก menu ไปที่ IMAGE จากนั้นก็จะได้ menu ใหม่ขึ้นมาอีก 3 menu คือ LOAD, MIRROR,close(ซึ่งแต่ละ menu ได้อธิบายไปแล้วในตอนต้น) จากนั้นก็เลือกการใช้ สมมุติว่า file ที่ได้มาจาก scanner เป็นภาพที่ตรงข้ามกับภาพจริง ดังนั้นจึงต้องใช้ MIRROR ในการ load ภาพซึ่งการ load แบบนี้ จะทำการกลับภาพอีกครั้งหนึ่ง จึงทำให้ภาพที่ปรากฏบนจอ monitor ตรงกับภาพพื้นที่จริง

3. เมื่อปิด menu IMAGE แล้ว program จะกลับมาที่menuหลักอีกครั้งหนึ่ง จากนั้นก็ให้เลือกไปที่ 2D ก็จะได้ menu ใหม่ขึ้นมาอีก 3 menu คือ LEVEL,NET,close(ซึ่งแต่ละ menu ได้อธิบายไปแล้วในตอนต้น)ให้เลือกไปที่ LEVEL ก่อน เพื่อทำการ set ค่า ความสูงให้แก่ control line แต่ละเส้น โดยค่าที่ set นั้นจะอยู่ระหว่าง 1 - 255 ส่วนเส้นใดมีค่าความสูงเป็นศูนย์ ก็ไม่ต้อง set ระดับความสูงกับเส้นนั้นเพราะ program ทำการ defualt ค่าไว้ที่ศูนย์เสมอ เมื่อใส่ระดับความสูงจนครบทุกเส้นแล้วให้เลือก menu ไปที่ NET เพื่อทำการ interpolate ภาพทั้งภาพแล้ว save ภาพดังกล่าว โดยก่อนการ interpolate นั้น program จะให้ใส่ชื่อของ file เมื่อทำการ interpolate เสร็จแล้วจะได้นำภาพดังกล่าวใส่ไว้ในชื่อที่กำหนดไว้ การใส่ชื่อควรจะใส่ให้ตรงกับชื่อของภาพเดิมที่ยังไม่ทำการ interpolate นามสกุลของ file ให้เปลี่ยนเป็น NET เพื่อความสะดวกในการจำ

4. ปิด menu 2D จากนั้นก็ให้เลือกไปที่ menu 3D เพื่อ plot ภาพ ที่ทำการ interpolate แล้ว(*.NET) จากข้อ 3

ขั้นตอนในการใช้ menu นี้

4.1 กด F6. เพื่อ load file นามสกุล NET

4.2 เลือก มุมมองของภาพ ซึ่งเป็นแบบ perspective โดยใช้ function key(F1.-F4.)

4.3 เมื่อเลือกมุมมองจนพอใจแล้วจากนั้นให้ กด F5. เพื่อทำการ plot(run) ภาพในลักษณะ3มิติ ที่มุมมองตามที่เลือกไว้ตามข้อ 4.2

4.4 กด F7. เมื่อต้องการออกจาก munu 3D

5. ถ้าต้องการออกจาก program เพื่อจะกลับสู่ DOS ให้กลับไปที่ MAIN MENU แล้ว กด F3. จากนั้น program จะถามอีกครั้งหนึ่ง ถ้าต้องการออกจาก program เพื่อจะกลับสู่ DOS จริง ให้กด ENTER program จะคืนระบบให้กับ DOS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการทดลอง

การเขียนโปรแกรม การจำลองภาพภูมิประเทศในลักษณะสามมิติ(3 dimension of contour) ในเทอมนี้นั้น สามารถที่จะจำลองภาพภูมิประเทศได้ กล่าวคือ สามารถที่จะนำเอาแผนที่ระดับความสูงที่มีระดับความสูงต่างๆ ซึ่งอยู่ในลักษณะ 2 มิติ นำมาผ่านกระบวนการทางคณิตศาสตร์เพื่อให้สามารถนำมา plot เป็นภาพสามมิติ(perspective)และสามารถหมุนภาพสามมิติดังกล่าวเพื่อดูลักษณะของภูมิประเทศในมุมต่างๆกันได้

โปรแกรมนี้จะต้องใช้งานร่วมกับโปรแกรม rixai8.exe rvga.com และเครื่องคอมพิวเตอร์ที่มีความสามารถดังนี้

cpu	80386 ขึ้นไป
ram	> 640 Kb
monitor	สี (1024*768,256สี)
card svga	memory 1 Mb

ปัญหาที่เกิดขึ้น

ปัญหาที่เกิดขึ้นในการเขียนโปรแกรมที่ผ่านมานั้นส่วนใหญ่จะเกิดขึ้นจากทฤษฎีที่นำมาใช้กับโปรแกรมนี้ไม่สามารถใช้ได้ทันที แต่จะต้องมีการแก้ไขปรับปรุง เพื่อให้ใช้งานได้และสะดวกในการเขียนโปรแกรม โดยปัญหาที่เกิดขึ้นสามารถแก้ไขลุล่วงไปได้ด้วยดี แต่ก็อาจเกิดข้อผิดพลาดขึ้นในอนาคตได้(bug) ซึ่งถ้าเกิดขึ้นก็จะต้องแก้ไขเป็นจุดๆไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอขอบพระคุณ รศ.ดร.ฟูศักดิ์ ชีวสุวิทย์ เป็นอย่างสูง ที่ได้ให้การประสิทธิ์ประสาทวิชา และให้คำแนะนำปรึกษาในเรื่องต่างๆ แก่ผู้เขียนและขอขอบคุณสักรียา ชิตวงศ์ และคุณอาโมทย์ สมบูรณ์แก้ว ที่ได้ให้ความช่วยเหลือและให้คำแนะนำต่างๆในการทำปริญญานิพนธ์นี้ ตั้งแต่ต้นตลอดมา จนสำเร็จลุล่วงไปได้ด้วยดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

CHAN S.PARK. INTERACTIVE MICROCOMPUTER GRAPHICS.

PRINCIPLE OF INTERACTIVE COMPUTER GRAPHICS. : MEMORY HALL OF LADKARBANG (T385 N4 C.3)

ROY A.PLASTOCK AND GORDON KALLEY. THEORY AND PROBLEMS OF COMPUTER GRAPHICS.

TURBO C. REFERENCE GUIDE.

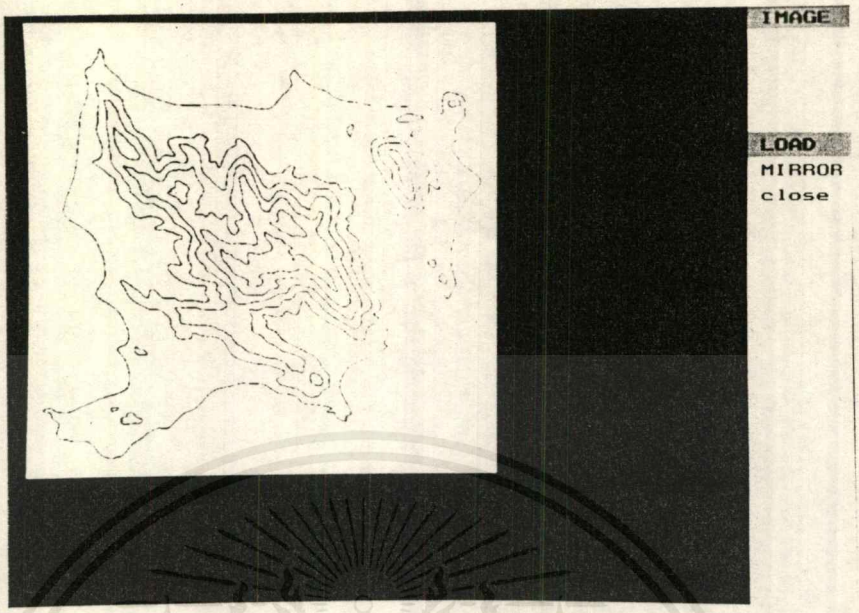
HERBERT SCHILDT. C POWER USER'S

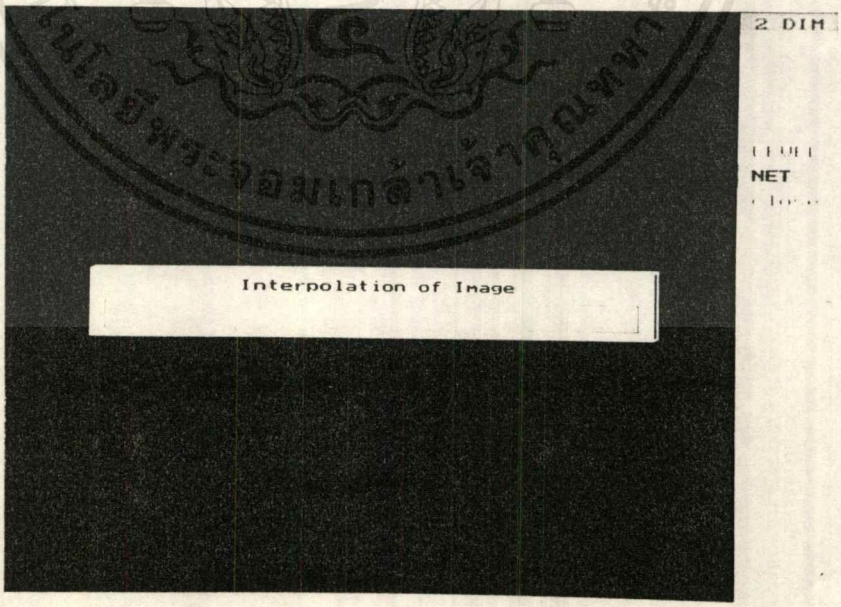
รศ.ดร. จเร สุรวัฒน์ปัญญา, ชีส วิชา COMPUTATIONAL METHODS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

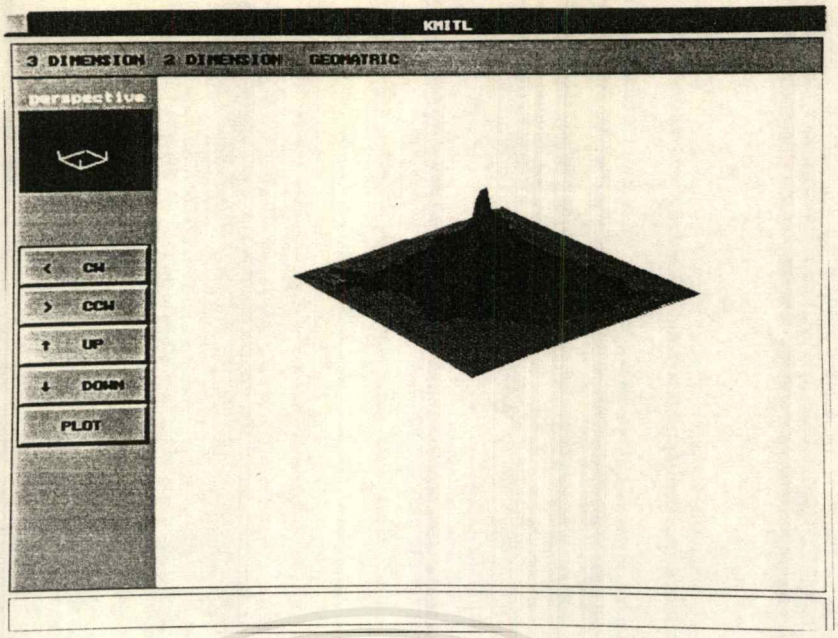
ภาคผนวก

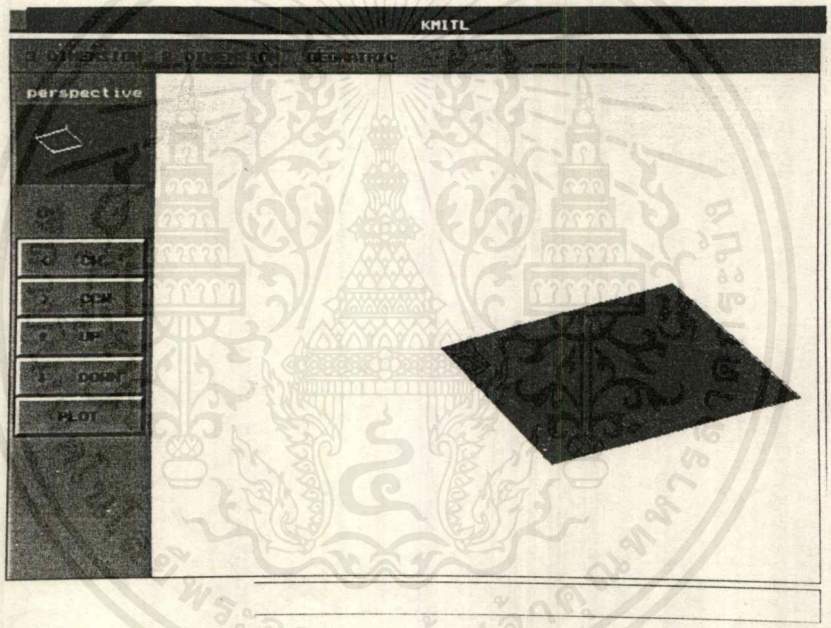
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

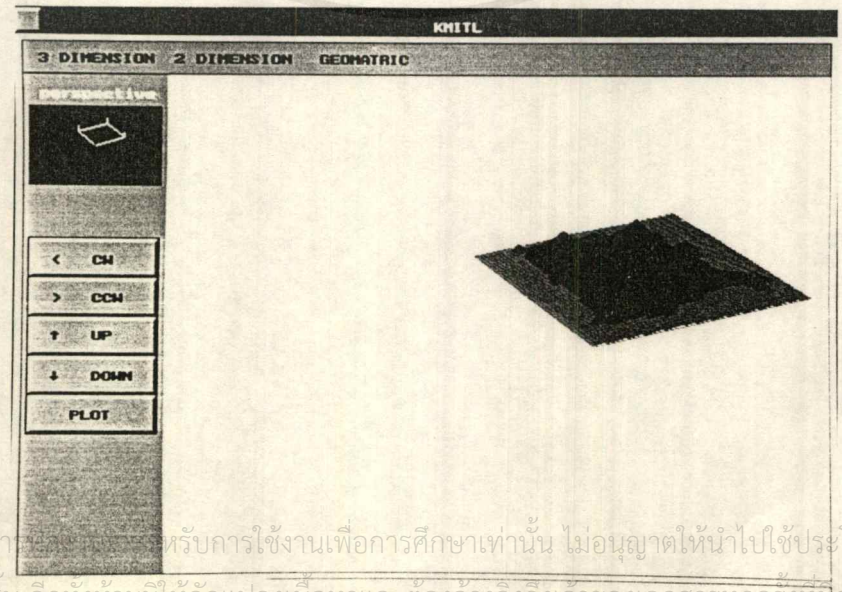




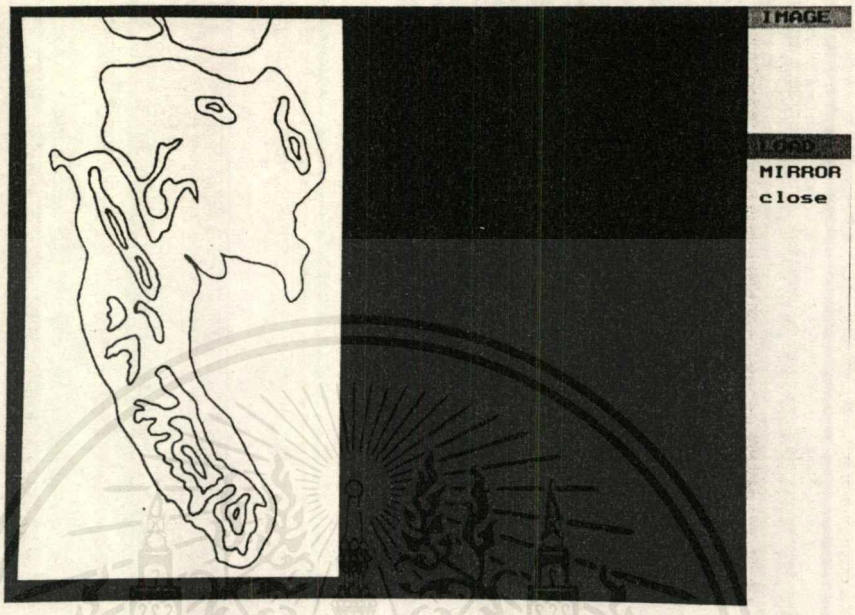
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



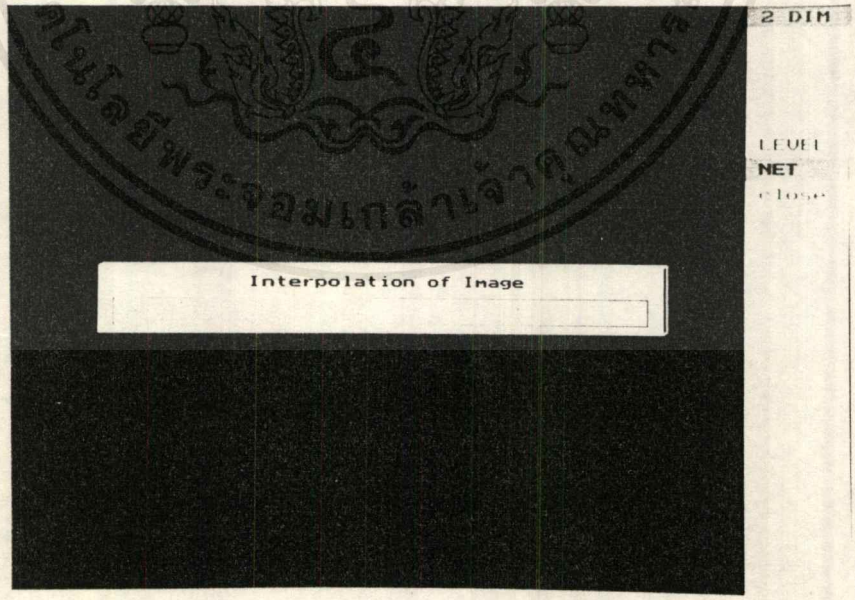




เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

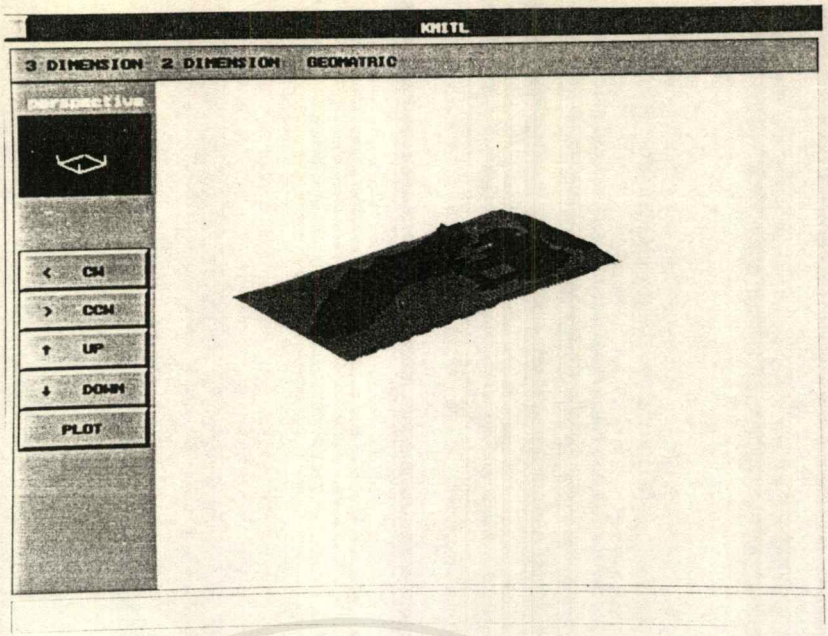


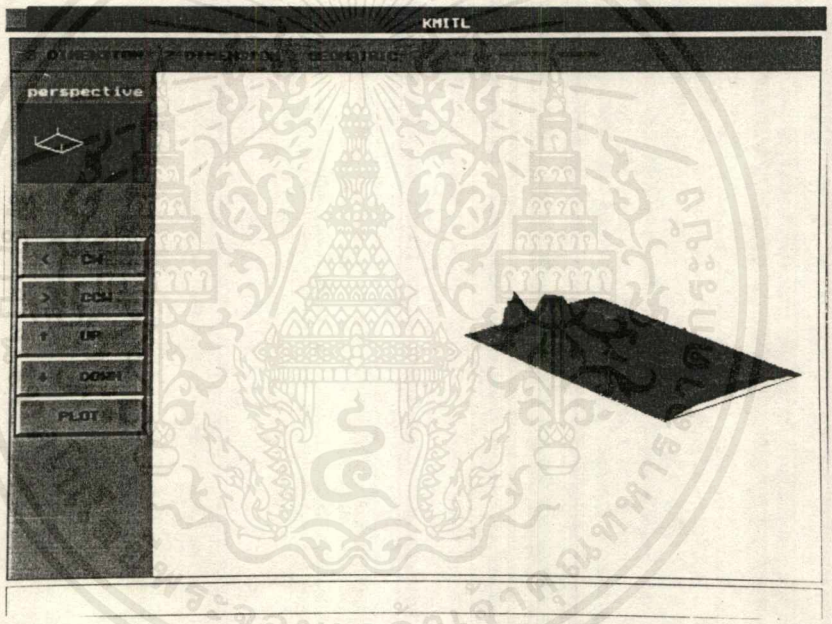
Mirror File Name :lantanoi.bmp



File Name of Net :1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
**********************
**  program  title.h  **
**********************

#include  <stdio.h>
#include  <graphics.h>
#include  <conio.h>
#include  <dos.h>

void  title()
{
  char  *title1 = " TITLE       3-DIMENSION OF CONTOUR                    ";
  char  *title2 = "                                                       ";
  char  *title3 = " PRODUCE BY    นาย ธรพล  ปวตรปก                       ";
  char  *title4 = "                                                       ";
  char  *title5 = " ADVISOR  รศ.ดร. ฟศกด  ชาสวทย                         ";
  char  *title6= "                                                        ";
  char  *title7 = "          ภาควชา เทคโนโลยการวดคมทางอตสาหกรรม          ";
  char  *title8 = "                                                       ";
  char  *title9 = "          คณะ    วศวกรรมศาสตร                         ";
  char  *title10= "                                                       ";
  char  *title11= "          สถาบนเทคโนโลยพระจอมเกลา เจาคณทหารลาดกระบง   ";
  char  *title12= "                                                       ";

  clrscr();
  sound(1000);
  delay(10);
  nosound();
  gotoxy(1,5);
  puts(title1);
  puts(title2);
  puts(title3);
  puts(title4);
  puts(title5);
  puts(title6);
  puts(title7);
  puts(title8);
  puts(title9);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    puts(title10);
    puts(title11);
    puts(title12);
    gotoxy(1,25);

    getch();
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
**************************
**   program  tmenu.c   **
**************************
#include <graphics.h>
#include <stdio.h>
#include <bios.h>
#include <math.h>
#include <process.h>
#include <conio.h>
#define  F3    15616
#define  UP    18432
#define  DOWN  20480
#define  RIGHT 19712
#define  LEFT  19200
#define  ENTER 7181
#define HBAR  bar(getmaxx()-124,0,getmaxx(),getmaxy()-738);     /*(yellow)*/
#define RBAR  bar(getmaxx()-124,getmaxy()-738,getmaxx(),getmaxy()); (white)
#define LBAR  bar(0,getmaxy()-40,getmaxx()-124,getmaxy());       /*(white)*/
#define  MBAR(d) bar(getmaxx()-124,getmaxy()-618+(d*30),getmaxx(),getmaxy()
                  -588+(d*30));                          /*move bar(cyan)*/
#define  PBAR  bar(0,0,getmaxx()-125,getmaxy()-40);

char  H_text(int page)
{
  char *Hd_text[] = {
  " *MENU*",
  " IMAGE",
  " 2 DIN",
  " 3 DIN"
  };
 setcolor(4);
 outtextxy(getmaxx()-124,7,Hd_text[page]);
}


char I_text(int page,int item)
{
  char *text[3][3] = {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    {" IMAGE"," 2  D" ," 3  D"},
    {" LOAD" ," MIRROR"," close"},
    {" LEVEL"," NET"," close"}

    };
  setcolor(1);
  outtextxy(getmaxx()-124,158+(item*30),text[page][item]);
}



/************************************/
/*  initializes the graphics system  */
/************************************/

init()
{
   static int    i,j;
   static int    errornumber;
   static int    graphdriver=IBM8514, graphmode=IBM8514HI;
   static char   driverparth[]="";

   initgraph(&graphdriver,&graphmode,driverparth);

   errornumber=graphresult();

   for(i=64;i<128;i++)
      {
        j = i<<2;
        setrgbpalette(i,j,j,j);
      }
}



/************************************/
/*  define color blackground menu  */
/************************************/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void blackground()
{
  setfillstyle(1,9);
  PBAR;
  setfillstyle(1,14);
  HBAR;
  setfillstyle(1,15);
  RBAR;
  LBAR;
}

int keyin()
{
  int  key,i;
  while(!bioskey(1));
  key = bioskey(0);
  return(key);
}

int menu(int status)
{
  int i,mark,item,page,l_status,l_item,l_page;

  mark = 0x0f;
  item = mark & status;
  page = (status>>4)&mark;

  setfillstyle(1,14);
  HBAR;H_text(page);

  setfillstyle(1,15);
  RBAR;
  for (i=0;i<=2;i++)
     I_text(page,i);

  setfillstyle(1,3);
  MBAR(item);I_text(page,item);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
while((i=keyin())!= ENTER)
{
   setfillstyle(1,15);
   LBAR;

   if(i==F3)
    {
     status = i;
     setcolor(0);
     outtextxy(5,getmaxy()-20,"DO you want to EXIT PROGRAM ? (YES=enter,
               NO=arrew key)");
    }
   if(i==UP)
    {
     l_item=item;
     item--;
     if(item<0)
item+=3;
     status=0;
    }
   if(i==DOWN)
    {
     l_item=item;
     item++;
     if(item>2)
item-=3;
     status=0;
    }
 sound(1000);
 delay(2);
 nosound();

 setfillstyle(1,15);
 MBAR(l_item); l_text(page,l_item);

 setfillstyle(1,3);
 MBAR(item); l_text(page,item);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    sound(1000);
    delay(10);
    nosound();

    if(status==F3)
       item = 3;
    item++;
    status=(item<<4)&0xf0;
    setfillstyle(1,15);
    LBAR;
    return(status);
}

void save_pixel(ux,uy,dx,dy,lx,ly,rx,ry,buffer)
 unsigned char *buffer;
 int ux,uy,dx,dy;
 int lx,ly,rx,ry;
{
    int temp,i,j;
    unsigned char color;

    temp=0;
    for(j=uy;j<=dy;j++)
     {
       color=getpixel(ux,j);
       buffer[temp]=color;
       temp++;
     }
    for(i=lx;i<=rx;i++)
     {
       color=getpixel(i,ly);
       buffer[temp]=color;
       temp++;
     }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void put_pixel(ux,uy,dx,dy,lx,ly,rx,ry,buffer)
 unsigned char *buffer;
 int ux,uy,dx,dy;
 int lx,ly,rx,ry;
{
 int temp,i,j;
   temp=0;
   for(j=uy;j<=dy;j++)
      {
putpixel(ux,j,buffer[temp]);
temp++;
      }

   for(i=lx;i<=rx;i++)
      {
       putpixel(i,ly,buffer[temp]);
       temp++;
      }
}


void move_cross(int *x,int *y,int snap)
{
  static unsigned char buffer[50];
  unsigned char   color;
  int ux,uy,dx,dy,j,i;
  int lx,ly,rx,ry;
  int temp;

  ux=*x;uy=*y-10;
  dx=*x;dy=*y+10;
  lx=*x-10;ly=*y;
  rx=*x+10;ry=*y;

  save_pixel(ux,uy,dx,dy,lx,ly,rx,ry,buffer);
  setcolor(4);                                  /* design shape */
  line(ux,uy,dx,dy);
  line(lx,ly,rx,ry);
```

เอกสารนี้มีลิขสิทธิ์ สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   while( (temp=keyin()) !=ENTER)
   {
       switch(temp)
       {
case UP    :if(uy>0)
    {
    put_pixel(ux,uy,dx,dy,lx,ly,rx,ry,buffer);
    uy-=snap;dy-=snap;ly-=snap;ry-=snap;
    save_pixel(ux,uy,dx,dy,lx,ly,rx,ry,buffer);
    }
    break;
case DOWN :if(dy<768)
    put_pixel(ux,uy,dx,dy,lx,ly,rx,ry,buffer);
    uy+=snap;dy+=snap;ly+=snap;ry+=snap;
                    save_pixel(ux,uy,dx,dy,lx,ly,rx,ry,buffer);
    break;
case LEFT :if(lx>0)
    {
    put_pixel(ux,uy,dx,dy,lx,ly,rx,ry,buffer);
    ux-=snap;dx-=snap;lx-=snap;rx-=snap;
    save_pixel(ux,uy,dx,dy,lx,ly,rx,ry,buffer);
    }
    break;
case RIGHT:if(rx<1024)
    {
    put_pixel(ux,uy,dx,dy,lx,ly,rx,ry,buffer);
    ux+=snap;dx+=snap;lx+=snap;rx+=snap;
    save_pixel(ux,uy,dx,dy,lx,ly,rx,ry,buffer);
    }
    break;
       }
       line(ux,uy,dx,dy);                                    /* shape */
       line(lx,ly,rx,ry);
  }
 put_pixel(ux,uy,dx,dy,lx,ly,rx,ry,buffer);
 *x=ux;
 *y=ly;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}


/*****************************************/
/*  status is status of bar (0,1,2)      */
/*  d is constant value distance mbar    */
/*****************************************/
main()
{
  int  color;
  int  x,y,tempx,tempy;
  int  status,error;
  int  d=30;


  title();
  init();
  blackground();
  status=00;
  do{
     status=menu(status);
     switch (status)
       {
 case 0x10 :    do{
status=menu(status);
switch(status)
  {
   case 0x10: setfillstyle(1,9);
      PBAR;
      file(status);
      break;
   case 0x20: setfillstyle(1,9);
      PBAR;
      file(status);
      status=0x10;
      break;
   case 0x30: status=00; break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   case 0x40: status=-1; break;
  }
}while(status > 00);
break;

 case 0x20 :    x=370;y=530;     /*set position move_cross*/
do{
status=menu(status);
switch(status)
  {
   case 0x10:
        color = level();
        sound(1000);
        delay(10);
        nosound();
        move_cross(&x,&y,5);
        tempx=x;tempy=y;
        if(up_edge(&x,&y,color)==1)
  {
    status=00;
    break;
  }
        edge_detect(x,y,color);
        x=tempx;y=tempy;
        status=0x20;
        break;

    case 0x20:  status=0x22;
        file(status);
        status=0x20;
        break;

    case 0x30:  status=00;
        break;
    case 0x40:  status=-1;
   }
  }while(status > 00);
  break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  case 0x30 : perspective();
      status=00;
      break;

  case 0x40 : status=-1;
      break;
        }



   }while(status >= 00);
  closegraph();
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
**********************
**  promgram  file.h  **
**********************
#include  <stdio.h>
#include  <math.h>
#include  <alloc.h>
#include  <dos.h>
#include  <graphics.h>
#include  <conio.h>
#include  <ctype.h>
#include  <stdlib.h>
#define  PBAR  bar(0,0,getmaxx()-124,getmaxy()-40);
#define   LBAR  bar(0,getmaxy()-40,getmaxx()-124,getmaxy())

long filesize(FILE *stream)
{
 long curpos,length;

 curpos = ftell(stream);
 fseek(stream,0L,SEEK_END);
 length = ftell(stream);
 fseek(stream,curpos,SEEK_SET);
 return(length);
}

void file(int status)
{
  FILE   *f,*f1;
  int    x,y,xpix,ypix,tempx,tempy;
  int    color,t_color;
  static unsigned char buff[1];
  static unsigned char huge *image;
  long   temp,dog,z;
  long   f_size;
  int    i,j,k,l,height;
  float  t,temp_y;
  static char filename[15];
  static char ch[1];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 dof
  setfillstyle(1,15); LBAR;
  if (status==0x22)
     outtextxy(20,getmaxy()-20,"File Name of Net");

  if (status==0x20)
     outtextxy(20,getmaxy()-20,"Mirror File Name");

  if (status==0x10)
     outtextxy(20,getmaxy()-20," Image File Name");
  setcolor(4);
  outtextxy(300,getmaxy()-20,":");
  setcolor(1);

  i=0;
  do {
    k=*ch=getch();       /* normol key */
    dof
     if(k==8) /* check backspece key */
       if(i>0)
         {
  i--;
  filename[i]='  ';
  setfillstyle(1,15);
  bar(320+(i*17),getmaxy()-20,320+20+(i*17),getmaxy());
  k=*ch=getch();
  setcolor(1);
 }
       else  k=*ch=getch();
    if(k==0)             /* arrow key */
     {
      k=getch();
      switch(k)
case 75: if(i>0)
                    {
    i--;
    filename[i]='  ';
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   setfillstyle(1,15);
   bar(320+(i*17),getmaxy()-20,320+20+(i*17),getmaxy());
   k=*ch=getch();
   setcolor(1);
  }
    }
   }while(( k==0)||(k==8));
  if(*ch!='r')
    {
     if(i!=12)
     {
      filename[i]=*ch;
      outtextxy(320+(i*17),getmaxy()-20,ch);
     }
    }
  i++;
  if(i>12)              /* check amount key */
    i=12;
}while(*ch!='r');
if (status!=0x22)
   f = fopen(filename,"rb");
if(i!=12)                    /* clear filename when press again */
  i--;
for(i;i<=12;i++)
   filename[i]='  ';
if (f== NULL) {
   LBAR;
   outtextxy(350,getmaxy()-20,"cannot open file");
   sound(1000);
   delay(1000);
   nosound();
   LBAR;
   return(NULL); }
```

เอกสารนี้สงวนลิขสิทธิ์ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}while(f==NULL);

 if (status==0x22)                /* subprogram of net */
 {
    x=15;y=15;
    do{       /*fine value x*/
x++;
color=getpixel(x,16);
    }while(color!=9);
    x--;
    xpix=x-16;
    do{
y++;
color=getpixel(16,y);
   }while(color!=9);
   y--;
   ypix=y-16;

 }
 else
 {
     fseek(f,22L,SEEK_SET);

     fread(buff,1,1,f);
     temp=(long)buff[0];
     fread(buff,1,1,f);
     temp=(long)temp|(long)(buff[0]<<8);

     f_size=filesize(f);
     f_size-=1078L;

     xpix=tempx=f_size/temp;
     ypix=tempy=temp;

     fseek(f,0L,SEEK_SET);
     fseek(f,1078L,SEEK_SET);
 }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        z=(long)xpix*ypix;                    /* allocate Memory */
        if((image = (char*)farcalloc(z,sizeof(char)))==NULL)
   {
    LBAR;
    outtextxy(325,getmaxy()-20,"Don't Allocation Memory");
    sound(1000);
    delay(1000);
    nosound();
    LBAR;
    return(NULL);
   }

        setfillstyle(1,14);
        bar(getmaxx()-124,getmaxy()-40,getmaxx(),getmaxy());
        outtextxy(getmaxx()-124,getmaxy()-25," WAIT !");
        sound(1000);
        delay(20);;
        nosound();

      l=-1;k=0;
      if(status==0x22)      /* net:save image into memory and file */
      {
        f = fopen(filename,"wb");
        fseek(f,0L,SEEK_SET);
        for (j=16; j<y; j++){
 l++;k=0;
 for (i=16; i<x; i++){
    color=getpixel(i,j);
    if(color==15)
       color=0;
    dog=xpix*(long)l+k;
    image[dog]=buff[0]=color;
    fwrite(buff,1,1,f);
    k++;
    putpixel(i,j,9);
 }
 putpixel(x,j,9);                     /* delete edge of y */
        }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      for (i=16; i<=x; i++)
putpixel(i,y,9);                    /* delete edge of x */

      fclose(f);

    }
    else  for (j=0; j<ypix; j++)
   for (i=0; i<xpix; i++){
     fread(buff,1,1,f);
     color=(int)buff[0];
     dog=xpix*(long)j+i;
     color=color/4+64;
     if(color<110)
color=0;
     else  color=15;
     image[dog]=color;
  }

     setfillstyle(1,15);                    /* erase ' wait! ' */
     bar(getmaxx()-124,getmaxy()-40,getmaxx(),getmaxy());


  if (status==0x10)
     for (j=0; j<ypix; j++)
for (i=0; i<xpix; i++){
    dog=xpix*(long)j+i;
    putpixel(i,j,image[dog]);
}

  if (status==0x20)
     for (j=0; j<ypix; j++){
for (i=0; i<xpix; i++){
     dog=xpix*(long)j+(xpix-i);
            putpixel(i,j,image[dog]);
}
    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if(status==0x22)
    {

      setfillstyle(1,110);          /* display  for working interpulate */
      setcolor(15);
      bar3d(100,319,800,403,5,15);
      setcolor(0);
      outtextxy(280,332,"Interpolation of Image");
      rectangle(118,360,782,393);
      sound(1000);
      delay(20);
      nosound();
      setcolor(1);

      x=0;                          /*interpolate in axis x*/
      for(k=0;k<2;k++)
      {
       for(j=0;j<ypix;j++)
       {
  dog=xpix*(long)j+x;
  if(j==0)
  {
    i=j;

    temp=image[dog];
  }
  else
  {
    if((ypix-j)==1)
      if(image[dog]==0)
image[dog]=15;


    if(image[dog]!=0)
    {
     if((ypix-j)==1)
       if(image[dog]==15)
 image[dog]=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      temp_y = image[dog];
      l=j;
      t = (temp-temp_y)/(j-i);
      for(j--;j>i;j--)
      {
        dog=xpix*(long)j+x;
        temp_y+=t;
        image[dog]=(int)temp_y;
      }
      j=l;
      j++;
      dog=xpix*(long)j+x;
      while(image[dog]!=0)
      {
  j++;
  dog=xpix*(long)j+x;
      }
      j--;
      i=j;
      dog=xpix*(long)j+x;
      temp=image[dog];
   }
 }
        }
        x=xpix-1;
      }

      setfillstyle(1,14);                    /* bar check working */
      bar(120,362,153+(0*33),391);

      for(j=0;j<ypix;j++)                    /* start */
       for(x=0;x<xpix;x++)
{
 dog=xpix*(long)j+x;
 if(x==0)
 {
   i=x;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    temp=image[dog];
  }
  else
  {
    if((xpix-x)==1)
      if(image[dog]==0)
image[dog]=15;


    if(image[dog]!=0)
    {
     if((xpix-x)==1)
       if(image[dog]==15)
 image[dog]=0;

      temp_y = image[dog];
      l=x;
      t = (temp-temp_y)/(x-i);

      for(x--;x>i;x--)
      {
        dog=xpix*(long)j+x;
        temp_y+=t;
        image[dog]=(int)temp_y;
      }
      x=l;
      x++;
      dog=xpix*(long)j+x;
      while(image[dog]!=0)
      {
 x++;
 dog=xpix*(long)j+x;
      }
      x--;
      i=x;
      dog=xpix*(long)j+x;
      temp=image[dog];
```

เอกสารนี้สงวนลิขสิทธิ์ ใช้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  }
}
      }

    setfillstyle(1,14);                          /* bar check working */
    bar(120+(1*33),362,153+(4*33),391);

     f1 = fopen("itpul.pic","wb");

     fseek(f1,0L,SEEK_SET);

     for (j=0; j<ypix; j++)                    /* net:save memory into file */
for (i=0; i<xpix; i++){
    dog=xpix*(long)j+i;
    buff[0]=image[dog];
    fwrite(buff,1,1,f1);
}
     fclose(f1);

    setfillstyle(1,14);                          /* bar check working */
    bar(120+(33*5),362,153+(8*33),391);

     f = fopen(filename,"rb");
     fseek(f,0L,SEEK_SET);
     for (j=0; j<ypix; j++)                     /* load original file */
  for (i=0; i<xpix; i++){
    fread(buff,1,1,f);
    height=(int)buff[0];
    dog=xpix*(long)j+i;
    image[dog]=height;
  }
     fclose(f);

    setfillstyle(1,14);                          /* bar check working */
    bar(120+(33*9),362,153+(12*33),391);


      j=0;                        /* interpolate in axis y */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        for(k=0;k<2;k++)
        {
for(x=0;x<xpix;x++)
{
 dog=xpix*(long)j+x;
 if(x==0)
 {
   i=x;

   temp=image[dog];
 }
 else
 {
   if((xpix-x)==1)
     if(image[dog]==0)
image[dog]=15;


   if(image[dog]!=0)
   {
    if((xpix-x)==1)
      if(image[dog]==15)
 image[dog]=0;

     temp_y = image[dog];
     l=x;
     t = (temp-temp_y)/(x-i);

     for(x--;x>i;x--)
     {
       dog=xpix*(long)j+x;
       temp_y+=t;
       image[dog]=(int)temp_y;
     }
     x=l;
     x++;
     dog=xpix*(long)j+x;
     while(image[dog]!=0)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      {
   x++;
   dog=xpix*(long)j+x;
      }
      x--;
      i=x;
      dog=xpix*(long)j+x;
      temp=image[dog];
    }
 }
        }
        j=ypix-1;
      }

      setfillstyle(1,14);                    /* bar check working */
      bar(120+(13*33),362,153+(13*33),391);


      for(x=0;x<xpix;x++)                    /* start */
       for(j=0;j<ypix;j++)
       {
  dog=xpix*(long)j+x;
  if(j==0)
  {
    i=j;

    temp=image[dog];
  }
  else
  {
    if((ypix-j)==1)
      if(image[dog]==0)
 image[dog]=15;


    if(image[dog]!=0)
    {
     if((ypix-j)==1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      if(image[dog]==15)
  image[dog]=0;

      temp_y = image[dog];
      l=j;
      t = (temp-temp_y)/(j-i);
      for(j--;j>i;j--)
      {
        dog=xpix*(long)j+x;
        temp_y+=t;
        image[dog]=(int)temp_y;
      }
      j=l;
      j++;
      dog=xpix*(long)j+x;
      while(image[dog]!=0)
      {
   j++;
   dog=xpix*(long)j+x;
      }
      j--;
      i=j;
      dog=xpix*(long)j+x;
      temp=image[dog];
    }
  }
        }

      setfillstyle(1,14);                          /* bar check working */
      bar(120+(14*33),362,153+(17*33),391);

        f1=fopen("itpul.pic","rb");
        fseek(f1,0L,SEEK_SET);
        for(j=0;j<ypix;j++)              /* interpolate x,y */
  for(i=0;i<xpix;i++)
  {
   fread(buff,1,1,f1);
   dog=xpix*(long)j+i;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 t=(buff[0]+image[dog])/2;
 image[dog]=(int)t;
}
      fclose(f1);

     setfillstyle(1,14);                          /* bar check working */
     bar(120+(18*33),362,153+(19*33),391);
     delay(1000);

     setfillstyle(1,9);                        /* delete display */
     bar(90,315,810,410);

      setfillstyle(1,14);
      bar(getmaxx()-124,getmaxy()-40,getmaxx(),getmaxy());
      outtextxy(getmaxx()-124,getmaxy()-25," SAVE !");
      sound(1000);
      delay(20);
      nosound();

      f = fopen(filename,"wb");
      fseek(f,0L,SEEK_SET);

      for (j=0; j<ypix; j++)                    /* net:save memory into file */
for (i=0; i<xpix; i++){
     dog=xpix*(long)j+i;
     buff[0]=image[dog];
     fwrite(buff,1,1,f);
}

       tempx=xpix; tempy=ypix;                  /* save  wide and long (x,y) */

       tempx>>=8; tempx&0x00ff;
       fwrite(&tempx,1,1,f);
       tempx=xpix; tempx&0x00ff;
       fwrite(&tempx,1,1,f);

       tempy>>=8; tempy&0x00ff;
       fwrite(&tempy,1,1,f);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        tempy=ypix; tempy&0x00ff;
        fwrite(&tempy,1,1,f);

        setfillstyle(1,15);                /* erase ' save! ' */
        bar(getmaxx()-124,getmaxy()-40,getmaxx(),getmaxy());

    }
    fclose(f);
    farfree(image);

    if(status!=0x22)
    {
     setfillstyle(1,9);                    /* delete edge */
     bar(0,0,15,tempy);
     bar(tempx-15,0,tempx,tempy);
     bar(0,0,tempx,15);
     bar(0,tempy-15,tempx,tempy);
    }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
**********************
**  program  2dim.h **
**********************

#include <graphics.h>
#include <stdlib.h>
#include <conio.h>
#include <math.h>
#include <stdio.h>
#define   LBAR  bar(0,getmaxy()-40,getmaxx()-124,getmaxy())
/*********************************/
/*  status = '0' means fine line   */
/*  status = '1' means delete line */
/*********************************/
int way_cw1(int x,int y,int status)
{
  int color;

  x++; y--;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
  {
    if(status==1)
      putpixel(x,y,15);
    else return(0x14);
  }
  y++;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
  {
    if(status==1)
      putpixel(x,y,15);
    else return(0x10);
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    y++;
    color = getpixel(x,y);
    if(color==9)
      return(-1);
    if(color==Ø)
    {
      if(status==1)
        putpixel(x,y,15);
      else return(Øx11);
    }
    x--;
    color = getpixel(x,y);
    if(color==9)
      return(-1);
    if(color==Ø)
    {
      if(status==1)
        putpixel(x,y,15);
      else return(Øx01);
    }
    x--;
    color = getpixel(x,y);
    if(color==9)
      return(-1);
    if(color==Ø)
    {
      if(status==1)
        putpixel(x,y,15);
      else  return(Øx41);
    }
    y--;
    color = getpixel(x,y);
    if(color==9)
      return(-1);
    if(color==Ø)
    {
      if(status==1)
        putpixel(x,y,15);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    else return(0x40);
  }
  y--;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
  {
    if(status==1)
      putpixel(x,y,15);
    else return(0x44);
  }
  x++;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
  {
    if(status==1)
      putpixel(x,y,15);
    else return(0x04);
  }
  return(-1);
}


int way_cw2(int x,int y,int status)
{
  int color;

  x++; y-=2;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
  {
    if(status==1)
      putpixel(x,y,15);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  else return(0x15);
}
x++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x25);
}
y++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x24);
}
y++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x20);
}
y++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if(status==1)
      putpixel(x,y,15);
    else  return(0x21);
  }
  y++;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
  {
    if(status==1)
      putpixel(x,y,15);
    else return(0x22);
  }
  x--;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
  {
    if(status==1)
      putpixel(x,y,15);
    else return(0x12);
  }
  x--;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
  {
    if(status==1)
      putpixel(x,y,15);
    else return(0x02);
  }
  x--;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x42);
}
x--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x52);
}
y--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x51);
}
y--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x50);
}
y--;
color = getpixel(x,y);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if(color==9)
  return(-1);
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x54);
}
y--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x55);
}
x++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x45);
}
x++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x05);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  return(-1);
}

int way_cw3(int x,int y,int status)
{
  int color;

  x++; y-=3;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
  {
    if(status==1)
      putpixel(x,y,15);
    else return(0x16);
  }
  x++;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
  {
    if(status==1)
      putpixel(x,y,15);
    return(0x26);
  }
  x++;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
  {
    if(status==1)
      putpixel(x,y,15);
    else return(0x36);
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    y++;
    color = getpixel(x,y);
    if(color==9)
      return(-1);
    if(color==0)
    {
      if(status==1)
        putpixel(x,y,15);
      else return(0x35);
    }
    y++;
    color = getpixel(x,y);
    if(color==9)
      return(-1);
    if(color==0)
    {
      if(status==1)
        putpixel(x,y,15);
      else return(0x34);
    }
    y++;
    color = getpixel(x,y);
    if(color==9)
      return(-1);
    if(color==0)
    {
      if(status==1)
        putpixel(x,y,15);
      else return(0x30);
    }
    y++;
    color = getpixel(x,y);
    if(color==9)
      return(-1);
    if(color==0)
    {
      if(status==1)
        putpixel(x,y,15);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    else return(0x31);
  }
  y++;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
  {
    if(status==1)
      putpixel(x,y,15);
    else return(0x32);
  }
  y++;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
  {
    if(status==1)
      putpixel(x,y,15);
    else return(0x33);
  }
  x--;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
  {
    if(status==1)
      putpixel(x,y,15);
    else return(0x23);
  }
  x--;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
  {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  if(status==1)
    putpixel(x,y,15);
  else return(0x13);
}
x--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x03);
}
x--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x43);
}
x--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x53);
}
x--;
color = getpixel(x,y);
if(color==9)
  return(-1);
```

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x63);
}
y--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x62);
}
y--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x61);
}
y--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x60);
}
y--;
color = getpixel(x,y);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if(color==9)
  return(-1);
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x64);
}
y--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x65);
}
y--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x66);
}
x++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
{
  if(status==1)
    putpixel(x,y,15);
  else return(0x56);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  x++;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
  {
    if(status==1)
      putpixel(x,y,15);
    else return(0x46);
  }
  x++;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
  {
    if(status==1)
      putpixel(x,y,15);
    else return(0x06);
  }

  return(-1);
}

int way_ccw1(int x,int y)
{
  int color;

  x--;y--;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
    return(0x44);
  y++;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
```

เอกสารนี้จัดทำขึ้นเพื่อไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if(color==0)
  return(0x40);
y++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x41);
x++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x01);
x++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x11);
y--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x10);
y--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x14);
x--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x04);
```

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  return(-1);
}

int way_ccw2(int x,int y)
{
  int color;

  x--;y-=2;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
    return(0x45);
  x--;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
    return(0x55);
  y++;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
    return(0x54);
  y++;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
    return(0x50);
  y++;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
    return(0x51);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
y++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x52);
x++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x42);
x++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x02);
x++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x12);
x++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x22);
y--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x21);
y--;
color = getpixel(x,y);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  if(color==9)
    return(-1);
  if(color==0)
    return(0x20);
  y--;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
    return(0x24);
  y--;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
    return(0x25);
  x--;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
    return(0x15);
  x--;
  color = getpixel(x,y);
  if(color==9)
    return(-1);
  if(color==0)
    return(0x05);

  return(-1);
}

int way_ccw3(int x,int y)
{
  int color;

  x--;y-=3;
  color = getpixel(x,y);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if(color==9)
  return(-1);
if(color==0)
  return(0x46);
x--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x56);
x--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x66);
y++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x65);
y++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x64);
y++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x60);
y++;
color = getpixel(x,y);
if(color==9)
  return(-1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if(color==0)
  return(0x61);
y++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x62);
y++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x63);
x++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x53);
x++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x43);
x++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x03);
x++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x13);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
x++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x23);
x++;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x33);
y--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x32);
y--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x31);
y--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x30);
y--;
color = getpixel(x,y);
if(color==9)
  return(-1);
if(color==0)
  return(0x34);
y--;
color = getpixel(x,y);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if(color==9)
      return(-1);
    if(color==0)
      return(0x35);
    y--;
    color = getpixel(x,y);
    if(color==9)
      return(-1);
    if(color==0)
      return(0x36);
    x--;
    color = getpixel(x,y);
    if(color==9)
      return(-1);
    if(color==0)
      return(0x26);
    x--;
    color = getpixel(x,y);
    if(color==9)
      return(-1);
    if(color==0)
      return(0x16);
    x--;
    color = getpixel(x,y);
    if(color==9)
      return(-1);
    if(color==0)
      return(0x06);

    return(-1);
}


/* return value '0' when program finish */
/* fill color */
int  ck_end(x,y,x2,y2,x3,y3,color_p)
int x,y,x2,y2,x3,y3,color_p;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
 int  color;

 putpixel(x,y,color_p);
  if((x == x2)&&(y == y2)&&(x2==x3)&&(y2==y3))
      return(0);
  return(1);
}

void  coordinate(int *x,int *y,int way)
{
 int X,Y,mark,temp;
 if(way!=-1)
 {
 mark = 0x0f;
 X    = *x;
 Y    = *y;
 temp = way;
 way &= mark;
 switch(way)
 {
   case(1): Y++; break;
   case(2): Y+=2;break;
   case(3): Y+=3;break;
   case(4): Y--; break;
   case(5): Y-=2;break;
   case(6): Y-=3;break;
 }
 temp >>= 4;
 temp &= mark;
 switch(temp)
 {
   case(1): X++; break;
   case(2): X+=2;break;
   case(3): X+=3;break;
   case(4): X--; break;
   case(5): X-=2;break;
   case(6): X-=3;break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 }
 *x=X;*y=Y;
 }
 else  { *x=-1; *y=-1; }
}

void fill_point(int x,int y,int x2,int y2)
{
  setcolor(0);
  line(x2,y2,x,y);
}
void fill_continue(x,y,x2,y2,x3,y3,color)
int  x,y,x2,y2,x3,y3,color;
{
  if((x!=-1)&&(y!=-1)&&(x2!=-1)&&(y2!=-1))
     fill_point(x,y,x2,y2);
  if((x3!=-1)&&(y3!=-1)&&(x2!=-1)&&(y2!=-1))
    fill_point(x3,y3,x2,y2);
  if((x!=-1)&&(y!=-1)&&(x3!=-1)&&(y3!=-1))
    fill_point(x,y,x3,y3);
}


void edge_detect(int x,int y,int color)
{
  int  way ,way2,way3,count;
  int  x2,y2,x3,y3,X,Y;
  X=x;Y=y;
  count=0;
  while(ck_end(x,y,x2,y2,x3,y3,color)){
       x2=x;y2=y;x3=x;y3=y;
       way  = way_cw1(x,y,0);
       way2 = way_cw2(x,y,0);
       way3 = way_cw3(x,y,0);

       if(count==3)
       {
way_cw1(x,y,1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
way_cw2(x,y,1);
way_cw3(x,y,1);
        }
        if(count<3)
 count++;
        coordinate(&x,&y,way);
        coordinate(&x2,&y2,way2);
        coordinate(&x3,&y3,way3);
        fill_continue(x,y,x2,y2,x3,y3,color);
        if(kbhit())
  x=y=x2=y2=x3=y3;

  }
  x=X;y=Y;
  while(ck_end(x,y,x2,y2,x3,y3,color)){
        x2=x;y2=y;x3=x;y3=y;
        way  = way_ccw1(x,y);
        way2 = way_ccw2(x,y);
        way3 = way_ccw3(x,y);

way_cw1(x,y,1);
way_cw2(x,y,1);
way_cw3(x,y,1);
        coordinate(&x,&y,way);
        coordinate(&x2,&y2,way2);
        coordinate(&x3,&y3,way3);
        fill_continue(x,y,x2,y2,x3,y3,color);
        if(kbhit())
  x=y=x2=y2=x3=y3;
  }
LBAR;
}
int up_edge(int *x,int *y,int color_c)
{
  int color,Y,color_p;
  Y=*y;
  color_p = getpixel(*x,*y);
  if((color_p==0)||(color_p==9))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    {
      setcolor(1);
      setfillstyle(1,15);
      sound(1000);
      delay(50);
      nosound();
      LBAR;
      outtextxy(350,getmaxy()-20,"location ERROR");
      delay(1000);
      LBAR;
      return(1);
    }
   do{
       Y--;
       color = getpixel(*x,Y);
   }while((color==15)||(color==color_p));
   if(color==0)
     do{
        Y--;
        color = getpixel(*x,Y);
     }while(color==0);
 Y++;
 *y=Y;
}

int  level()
{
  char filename[5];
  char ch[1];
  int i,j,k,temp,tempa,mark;

  do{
  setfillstyle(1,15); LBAR;
  outtextxy(20,getmaxy()-20,"height of contour");
  setcolor(4);
  outtextxy(300,getmaxy()-20,":");
  setcolor(1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    i=0;
    do {
      k=*ch=getch();        /* normol key */
      do{
       if(k==8) /* check backspece key */
         if(i>0)
           {
    i--;
    filename[i]='  ';
    setfillstyle(1,15);
    bar(320+(i*17),getmaxy()-20,320+20+(i*17),getmaxy());
    k=*ch=getch();
    setcolor(1);
   }
  else  k=*ch=getch();
      if(k==0)              /* arrow key */
       {
        k=getch();
        switch(k)
case 75: if(i>0)
                  {
      i--;
      filename[i]='  ';
      setfillstyle(1,15);
      bar(320+(i*17),getmaxy()-20,320+20+(i*17),getmaxy());
      k=*ch=getch();
      setcolor(1);
     }
       }
      }while(( k==0)||(k==8));

    if(*ch!='r')
      {
       if(i!=5)
       {
        filename[i]=*ch;
        outtextxy(320+(i*17),getmaxy()-20,&ch);
       }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }
      i++;
      if(i>5)                    /* check amount key */
        i=5;

    }while(*ch!='\r');

    i-=2;j=0;temp=0;                    /* calculate number */
    for(i;i>=0;i--){
       tempa=filename[i];
       if((tempa>=48)&&(tempa<=57))
       {
         tempa&=0x0f;
         tempa = pow10(j)*tempa;
         temp+=tempa;
         j++;
       }
       else
       {
         temp=NULL;
         i=-1;
       }
    }
   if (temp == NULL) {
      LBAR;
      outtextxy(350,getmaxy()-20," define error ! ");
      sound(1000);
      delay(1000);
      nosound();
      }
  else  return(temp);

 }while(temp==NULL);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
'**********************
**  program  3dim.h  **
**********************
#include  <stdio.h>
#include  <graphics.h>
#include  <math.h>
#include  <alloc.h>
#include  <dos.h>
#include  <bios.h>
#include  <conio.h>

#define   LBAR  bar(0,getmaxy()-40,getmaxx(),getmaxy())
#define   PBAR  bar(0,0,getmaxx()-125,getmaxy()-40);
#define   F1    15104
#define   F2    15360
#define   F3    15616
#define   F4    15872
#define   F5    16128
#define   F6    16384
#define   F7    16640

void v_tranformation(int *X,int *Y,int z,double za,double fi,int D,
     int x_c,int y_c,int x_sc,int y_sc)
{
  int   x,y;
  float xe,ye,ze;

  x=*X;y=*Y;

  xe = -(x*sin(za)) + (y*cos(za));        /*  change  wold ->  eye coordinate*/
  ye = -(x*cos(za)*cos(fi)) - (y*sin(za)*cos(fi)) + (z*sin(fi));
  ze = -(x*cos(za)*sin(fi)) - (y*sin(za)*sin(fi)) -  z*cos(fi) + D;

  xe =  0.92*(4)*(xe/ze)*x_sc + x_c;          /*  change  eye -> display screem */
  ye =      -(4)*(ye/ze)*y_sc + y_c;

  *X=xe; *Y=ye;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void perspective()
{
  FILE   *f;
  int x,y,xpix,ypix,temp,temp1,tempx,tempy,av_h;
  static unsigned char buff[1];
  static unsigned char huge *image;
  double  za,fi;
  long   dog,z;
  int    i,j,k,l,height,key,c_x,c_y;
  static char filename[15];
  static char ch[1];
  int    block[10];
  int    tempx1,tempy1,tempx2,tempy2,tempx3,tempy3;
  int    sc_x,sc_y;
  char   start;

  start=1; sc_x=600; sc_y=300;
  setfillstyle(1,9);                    /* new menu */
  bar(0,0,getmaxx(),getmaxy()-40);
  setfillstyle(1,15);
  bar(0,0,200,getmaxy()-40);

  setcolor(1);
  outtextxy(13,20,"perspective");
  setcolor(0);
  rectangle(10,50,190,230);

  setfillstyle(1,115);
  setcolor(0);
  bar3d(10,300,185,330,3,3);
  outtextxy(20,310,"F1.  cw");

  bar3d(10,350,185,380,3,3);
  outtextxy(20,360,"F2.  ccw");

  bar3d(10,400,185,430,3,3);
  outtextxy(20,410,"F3.  up");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  bar3d(10,450,185,480,3,3);
  outtextxy(20,460,"F4.  down");

  bar3d(10,500,185,530,3,3);
  outtextxy(20,510,"F5.  run");

  bar3d(10,550,185,580,3,3);
  outtextxy(20,560,"F6.  load");

  bar3d(10,600,185,630,3,3);
  outtextxy(20,610,"F7. close");

   za=0.92729;          /* ze = 53.12 degree */
   fi=0.81768;          /* fi = 46.85 degree */
   i=-2;
   do{
if((i==-2)||(key==F1)||(key==F2)||(key==F3)||(key==F4))
{
  setfillstyle(1,15);           /* delete  motion of axis */
  bar(11,51,189,229);


 setcolor(4);           /* axis  x,y,z */
 tempx=0; tempy=0; temp=0;
 v_tranformation(&tempx,&tempy,temp,za,fi,3000,100,135,sc_x,sc_y);
 x=70; y=0; temp=0;
 v_tranformation(&x,&y,temp,za,fi,3000,100,135,sc_x,sc_y);
 line(tempx,tempy,x,y);

 x=0; y=70; temp=0;
 v_tranformation(&x,&y,temp,za,fi,3000,100,135,sc_x,sc_y);
 line(tempx,tempy,x,y);

 x=0; y=0; temp=40;
 v_tranformation(&x,&y,temp,za,fi,3000,100,135,sc_x,sc_y);
 setcolor(1);
 line(tempx,tempy,x,y);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
setcolor(4);
tempx=70; tempy=70; temp=0;
v_tranformation(&tempx,&tempy,temp,za,fi,3000,100,135,sc_x,sc_y);
x=70; y=0; temp=0;
v_tranformation(&x,&y,temp,za,fi,3000,100,135,sc_x,sc_y);
line(x,y,tempx,tempy);

x=0; y=70; temp=0;
v_tranformation(&x,&y,temp,za,fi,3000,100,135,sc_x,sc_y);
line(tempx,tempy,x,y);

x=70; y=70; temp=40;
v_tranformation(&x,&y,temp,za,fi,3000,100,135,sc_x,sc_y);
line(tempx,tempy,x,y);

tempx=70; tempy=0; temp=0;
v_tranformation(&tempx,&tempy,temp,za,fi,3000,100,135,sc_x,sc_y);
x=70; y=0; temp=40;
v_tranformation(&x,&y,temp,za,fi,3000,100,135,sc_x,sc_y);
line(tempx,tempy,x,y);

tempx=0; tempy=70; temp=0;
v_tranformation(&tempx,&tempy,temp,za,fi,3000,100,135,sc_x,sc_y);
x=0; y=70; temp=40;
v_tranformation(&x,&y,temp,za,fi,3000,100,135,sc_x,sc_y);
line(tempx,tempy,x,y);
     }
     do{
 key = keyin();
 i=0;
 switch(key){
   case(F1) : i=-1;break;
   case(F2) : i=-1;break;
   case(F3) : i=-1;break;
   case(F4) : i=-1;break;
   case(F5) : i=-1;break;
   case(F6) : i=-1;break;
   case(F7) : i=-1;break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   }
        }while(i=0);

        switch (key){
case(F1) :     setfillstyle(1,15);
        bar(5,295,190,335);

        setfillstyle(1,110);
        setcolor(0);
        bar(13,297,188,327);
        rectangle(13,297,188,327);
        rectangle(14,296,187,326);
        outtextxy(23,307,"F1.  cw");

        delay(100);
        za+=0.087266;              /*increase 5 degree*/
                        if(za>6.2831)
za-=6.2831;
        if(za<0)
za+=6.2831;


                        setfillstyle(1,15);
        bar(5,295,190,335);

        setfillstyle(1,115);
        setcolor(0);
        bar3d(10,300,185,330,3,3);
        outtextxy(20,310,"F1.  cw");

        break;

case(F2) :
        setfillstyle(1,15);
        bar(5,(295+50),190,(335+50));

        setfillstyle(1,110);
        setcolor(0);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        bar(13,297+50,188,327+50);
        rectangle(13,297+50,188,327+50);
        rectangle(14,296+50,187,326+50);
        outtextxy(23,307+50,"F2.  ccw");

        delay(100);
        za-=0.087266;
                        if(za>6.2831)
za-=6.2831;
        if(za<0)
za+=6.2831;


                        setfillstyle(1,15);
        bar(5,(295+50),190,(335+50));

        setfillstyle(1,115);
        setcolor(0);
        bar3d(10,350,185,380,3,3);
        outtextxy(20,360,"F2.  ccw");

        break;
case(F3) :
                        setfillstyle(1,15);
        bar(5,295+(2*50),190,335+(2*50));

        setfillstyle(1,110);
        setcolor(0);
        bar(13,297+(2*50),188,327+(2*50));
        rectangle(13,297+(2*50),188,327+(2*50));
        rectangle(14,296+(2*50),187,326+(2*50));
        outtextxy(23,307+(2*50),"F3.  up");

        delay(100);
        if(fi<1.5158)
 fi+=0.174532;              /* increase  10 degree */

                        setfillstyle(1,15);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        bar(5,295+(2*50),190,335+(2*50));

        setfillstyle(1,115);
        setcolor(0);
                        bar3d(10,400,185,430,3,3);
        outtextxy(20,410,"F3.  up");

        break;
case(F4) :
                        setfillstyle(1,15);
        bar(5,295+(3*50),190,335+(3*50));

        setfillstyle(1,110);
        setcolor(0);
        bar(13,297+(3*50),188,327+(3*50));
        rectangle(13,297+(3*50),188,327+(3*50));
        rectangle(14,296+(3*50),187,326+(3*50));
        outtextxy(23,307+(3*50),"F4.  down");

        delay(100);
        if(fi>0.2941)
 fi-=0.174532;

                        setfillstyle(1,15);
        bar(5,295+(3*50),190,335+(3*50));

        setfillstyle(1,115);
        setcolor(0);
        bar3d(10,450,185,480,3,3);
        outtextxy(20,460,"F4.  down");

        break;


case(F5) :
                        setfillstyle(1,15);
       bar(5,295+(4*50),190,335+(4*50));
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      setfillstyle(1,110);
      setcolor(0);
      bar(13,297+(4*50),188,327+(4*50));
      rectangle(13,297+(4*50),188,327+(4*50));
      rectangle(14,296+(4*50),187,326+(4*50));
      outtextxy(23,307+(4*50),"F5.  run");

setfillstyle(1,9);
bar(201,0,getmaxx(),getmaxy()-40);


setcolor(4);          /* axis  x,y,z */
c_x=612+(100*cos(-za+1.5707));
c_y=314-(100*sin(-za+1.5707));

    i=0;
    do{
     do{
      do{
do{
i++;
tempx=0; tempy=0; temp=0;       /* change  x <--> y */
v_tranformation(&tempx,&tempy,temp,za,fi,3000,c_x,c_y,sc_x,sc_y);
}while( check_limit(tempx,tempy,&c_x,&c_y)&&(i<1000) );

tempx1=ypix-1; tempy1=0; temp=0;
v_tranformation(&tempx1,&tempy1,temp,za,fi,3000,c_x,c_y,sc_x,sc_y);
      }while( check_limit(tempx1,tempy1,&c_x,&c_y)&&(i<1000) );

tempx2=0; tempy2=xpix-1; temp=0;
v_tranformation(&tempx2,&tempy2,temp,za,fi,3000,c_x,c_y,sc_x,sc_y);
     }while( check_limit(tempx2,tempy2,&c_x,&c_y)&&(i<1000) );

tempx3=ypix-1; tempy3=xpix-1; temp=0;
v_tranformation(&tempx3,&tempy3,temp,za,fi,3000,c_x,c_y,sc_x,sc_y);
    }while( check_limit(tempx3,tempy3,&c_x,&c_y)&&(i<1000) );

    if(i<1000){
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  line(tempx,tempy,tempx1,tempy1);
  line(tempx,tempy,tempx2,tempy2);
  line(tempx3,tempy3,tempx1,tempy1);
  line(tempx3,tempy3,tempx2,tempy2);

  if(((za>5.4978)&&(za<=6.2832))||((za>=0)&&(za<=0.7854))) /*  315 <-> 45 degree */
  for (j=10; j<ypix; j+=10){
      k=j-10;
      for (i=0; i<xpix; i+=10){
dog=xpix*(long)j+i;
temp=image[dog];
tempx=j; tempy=i;       /* change  x <--> y */
   v_tranformation(&tempx,&tempy,temp,za,fi,3000,c_x,c_y,sc_x,sc_y);
          block[2] = tempx;
   block[3] = tempy;

dog=xpix*(long)k+i;
temp1=image[dog];
x=k; y=i;
   v_tranformation(&x,&y,temp1,za,fi,3000,c_x,c_y,sc_x,sc_y);
   block[0] = x;
   block[1] = y;

   av_h = (temp+temp1)/2;

l=i+10;
dog=xpix*(long)j+l;
temp=image[dog];
tempx=j; tempy=l;
   v_tranformation(&tempx,&tempy,temp,za,fi,3000,c_x,c_y,sc_x,sc_y);
   block[4] = tempx;
   block[5] = tempy;

dog=xpix*(long)k+l;
temp1=image[dog];
x=k; y=l;
   v_tranformation(&x,&y,temp1,za,fi,3000,c_x,c_y,sc_x,sc_y);
  block[6] = x;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   block[7] = y;

   temp = (temp1+temp)/2;
   av_h = (av_h+temp)/2;

      if(av_h<=20)                  /* set color for detect height */
 setcolor(3);
      if((av_h>20)&&(av_h<256))
 setcolor(1);
      /*if((av_h>180)&&(av_h<256))
 setcolor(2);*/

      setfillstyle(1,9);
      fillpoly(4,block);

      if(kbhit())
      {
 j=ypix;
 i=xpix;
      }

  }
        }


   if((za>0.7854)&&(za<=2.3562))        /* 45 <-> 135 degree */
   for (i=10; i<xpix; i+=10){
      k=i-10;
      for (j=ypix-1; j>=10; j-=10){
 dog=xpix*(long)j+i;
 temp=image[dog];
 tempx=j; tempy=i;
   v_tranformation(&tempx,&tempy,temp,za,fi,3000,c_x,c_y,sc_x,sc_y);
           block[2] = tempx;
   block[3] = tempy;

 dog=xpix*(long)j+k;
 temp1=image[dog];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
y=k; x=j;
  v_tranformation(&x,&y,temp1,za,fi,3000,c_x,c_y,sc_x,sc_y);
  block[0] = x;
  block[1] = y;

  av_h = (temp+temp1)/2;

l=j-10;
dog=xpix*(long)l+i;
temp=image[dog];
tempx=l; tempy=i;
  v_tranformation(&tempx,&tempy,temp,za,fi,3000,c_x,c_y,sc_x,sc_y);
  block[4] = tempx;
  block[5] = tempy;

dog=xpix*(long)l+k;
temp1=image[dog];
y=k; x=l;
  v_tranformation(&x,&y,temp1,za,fi,3000,c_x,c_y,sc_x,sc_y);
  block[6] = x;
  block[7] = y;

  temp = (temp1+temp)/2;
  av_h = (av_h+temp)/2;

     if(av_h<=20)                    /* set color for detect height */
setcolor(3);
     if((av_h>20)&&(av_h<256))
setcolor(1);
     /*if((av_h>180)&&(av_h<256))
setcolor(2);*/

     setfillstyle(1,9);
     fillpoly(4,block);

     if(kbhit())
     {
j=10;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
i=xpix;
     }

 }
       }

   if((za>2.3562)&&(za<=3.7525))          /*135 <-> 215 degree*/J
   for (j=ypix-11; j>=0; j-=10){
      k=j+10;
      for (i=xpix-1; i>=10; i-=10){
dog=xpix*(long)j+i;
temp=image[dog];
tempx=j; tempy=i;
  v_tranformation(&tempx,&tempy,temp,za,fi,3000,c_x,c_y,sc_x,sc_y);
          block[2] = tempx;
  block[3] = tempy;

dog=xpix*(long)k+i;
temp1=image[dog];
x=k; y=i;
  v_tranformation(&x,&y,temp1,za,fi,3000,c_x,c_y,sc_x,sc_y);
  block[0] = x;
  block[1] = y;

  av_h = (temp+temp1)/2;

l=i-10;
dog=xpix*(long)j+l;
temp=image[dog];
tempx=j; tempy=l;
  v_tranformation(&tempx,&tempy,temp,za,fi,3000,c_x,c_y,sc_x,sc_y);
  block[4] = tempx;
  block[5] = tempy;

dog=xpix*(long)k+l;
temp1=image[dog];
x=k; y=l;
  v_tranformation(&x,&y,temp1,za,fi,3000,c_x,c_y,sc_x,sc_y);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   block[6] = x;
   block[7] = y;

   temp = (temp1+temp)/2;
   av_h = (av_h+temp)/2;

      if(av_h<=20)                  /* set color for detect height */
setcolor(3);
      if((av_h>20)&&(av_h<256))
setcolor(1);
      /*if((av_h>180)&&(av_h<256))
setcolor(2);*/

      setfillstyle(1,9);
      fillpoly(4,block);

      if(kbhit())
      {
j= 0;
i= 10;
      }

 }
       }


  if((za>3.7525)&&(za<=5.4978))              /*215 <-> 315 degree*/
  for (i=xpix-11; i>=0; i-=10){
     k=i+10;
     for (j=0; j<=ypix-10; j+=10){
dog=xpix*(long)j+i;
temp=image[dog];
tempx=j; tempy=i;
  v_tranformation(&tempx,&tempy,temp,za,fi,3000,c_x,c_y,sc_x,sc_y);
          block[2] = tempx;
  block[3] = tempy;

dog=xpix*(long)j+k;
```

เอกสารนี้จัดทำขึ้นไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
temp1=image[dog];
y=k; x=j;
  v_tranformation(&x,&y,temp1,za,fi,3000,c_x,c_y,sc_x,sc_y);
  block[0] = x;
  block[1] = y;

  av_h = (temp+temp1)/2;

l=j+10;
dog=xpix*(long)l+i;
temp=image[dog];
tempx=l; tempy=i;
  v_tranformation(&tempx,&tempy,temp,za,fi,3000,c_x,c_y,sc_x,sc_y);
  block[4] = tempx;
  block[5] = tempy;

dog=xpix*(long)l+k;
temp1=image[dog];
y=k; x=l;
  v_tranformation(&x,&y,temp1,za,fi,3000,c_x,c_y,sc_x,sc_y);
  block[6] = x;
  block[7] = y;

  temp = (temp1+temp)/2;
  av_h = (av_h+temp)/2;

     if(av_h<=20)                    /* set color for detect height */
setcolor(3);
     if((av_h>20)&&(av_h<256))
setcolor(1);
     /*if((av_h>180)&&(av_h<256))
setcolor(2);*/

     setfillstyle(1,9);
     fillpoly(4,block);

     if(kbhit())
     {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  j=ypix-10;
  i= 0;
       }

   }
         }

       }  else {
        setfillstyle(1,14);
        bar(getmaxx()-210,getmaxy()-39,getmaxx(),getmaxy());
        outtextxy(getmaxx()-210,getmaxy()-25," OVER range!");
        sound(1000);
        delay(100);;
        nosound();
        delay(1000);
        setfillstyle(1,WHITE);
        bar(getmaxx()-210,getmaxy()-39,getmaxx(),getmaxy());


      }

        setfillstyle(1,15);
        bar(5,295+(4*50),190,335+(4*50));

        setfillstyle(1,115);
        setcolor(0);
        bar3d(10,500,185,530,3,3);
        outtextxy(20,510,"F5.  run");

        break;

 case(F6) :    setfillstyle(1,15);
        bar(5,295+(5*50),190,335+(5*50));

        setfillstyle(1,110);
        setcolor(0);
        bar(13,297+(5*50),188,327+(5*50));
        rectangle(13,297+(5*50),188,327+(5*50));
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        rectangle(14,296+(5*50),187,326+(5*50));
        outtextxy(23,307+(5*50),"F6.  load");


    setfillstyle(1,15); LBAR;
    outtextxy(20,getmaxy()-20,"Load File of Net");
    setcolor(4);
    outtextxy(300,getmaxy()-20,":");


    setcolor(1);
    i=0;
    do {
      k=*ch=getch();        /* normol key */
      do{
       if(k==8) /* check backspece key */
         if(i>0)
           {
    i--;
    filename[i]='  ';
    setfillstyle(1,15);
    bar(320+(i*17),getmaxy()-20,320+20+(i*17),getmaxy());
    k=*ch=getch();
    setcolor(1);
  }
     if(k==0)             /* arrow key */
      {
       k=getch();
       switch(k)
case 75: if(i>0)
                    {
     i--;
     filename[i]='  ';
     setfillstyle(1,15);
     bar(320+(i*17),getmaxy()-20,320+20+(i*17),getmaxy());
     k=*ch=getch();
     setcolor(1);
    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
   }while(( k==0)||(k==8));

  if(*ch!='\r')
     if(i!=12)
     {
      filename[i]=*ch;
      outtextxy(320+(i*17),getmaxy()-20,ch);
     }
  i++;
  if(i>12)              /* check amount key */
    i=12;

}while(*ch!='\r');

f = fopen(filename,"rb");

if(i!=12)                       /* clear filename when press again */
  i--;
for(i;i<=12;i++)
   filename[i]='  ';


if (f== NULL) {
   LBAR;
   outtextxy(350,getmaxy()-20,"cannot open file");
   sound(1000);
   delay(1000);
   nosound();
   LBAR;
   }

if(f!=NULL)
{
    fseek(f,-4,SEEK_END);

    fread(buff,1,1,f);                          /* read 'x' from file */
    xpix=buff[0];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        fread(buff,1,1,f);
        xpix = (xpix<<=8) | (buff[0]&0x00ff);

        fread(buff,1,1,f);
        ypix=buff[0];                              /* read 'y' from file */
        fread(buff,1,1,f);
        ypix = (ypix<<=8) | (buff[0]&0x00ff);


        fseek(f,0L,SEEK_SET);

        if(start!=1){                             /* first pass [farfree(image)]*/
farfree(image);
        }
        start=0;
        z=(long)xpix * ypix;                       /* allocate Memory */
        if((image = (char*)farcalloc(z,sizeof(char)))==NULL)
 {
  LBAR;
  outtextxy(325,getmaxy()-20,"Don't Allocation Memory");
  delay(300);
          LBAR;
 }
        if(image!=NULL)
        {
        setfillstyle(1,14);
        bar(getmaxx()-124,getmaxy()-40,getmaxx(),getmaxy());
        outtextxy(getmaxx()-124,getmaxy()-25," WAIT !");
        sound(1000);
        delay(20);;
        nosound();

       for (j=0; j<ypix; j++)
    for (i=0; i<xpix; i++){
      fread(buff,1,1,f);
      height=(int)buff[0];
      dog=xpix*(long)j+i;
      image[dog]=height;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
       fclose(f);
       setfillstyle(1,15);                    /* erase ' wait! ' */
       bar(getmaxx()-124,getmaxy()-40,getmaxx(),getmaxy());
       }
     }

                        setfillstyle(1,15);
       bar(5,295+(5*50),190,335+(5*50));
       setfillstyle(1,115);
       setcolor(0);
       bar3d(10,550,185,580,3,3);
       outtextxy(20,560,"F6.  load");

       break;


case(F7) :    setfillstyle(1,15);
       bar(5,295+(6*50),190,335+(6*50));

       setfillstyle(1,110);
       setcolor(0);
       bar(13,297+(6*50),188,327+(6*50));
       rectangle(13,297+(6*50),188,327+(6*50));
       rectangle(14,296+(6*50),187,326+(6*50));
       outtextxy(23,307+(6*50),"F7. close");

       j=-1;
       delay(300);

                        setfillstyle(1,15);
       bar(6,295+(6*50),190,335+(6*50));
       setfillstyle(1,115);
       setcolor(0);
       bar3d(10,600,185,630,3,3);
       outtextxy(20,610,"F7. close");

       break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 }

   }while(j!=-1);
   fclose(f);
   farfree(image);
   setfillstyle(1,9);
   PBAR;
   setfillstyle(1,15);
   LBAR;

}
/***************************/
/*   '0' is  inside renge  */
/*   '1' is  over renge    */
/***************************/

int check_limit(int x,int y,int *c_x,int *c_y)
{

   if((x>=210)&&(x<=1000)) ;
   else
      if(x>1000){
 x-=1000;
 abs(x);
 *c_x-=x;
 return(1);
      }
      else {
 x-=210;
         abs(x);
 *c_x+=x;
 return(1);
      }

   if((y>=20)&&(y<=718)) ;
   else
      if(y>718){
 y-=718;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    abs(y);
    *c_y-=x;
    return(1);
         }
         else {
    y-=20;
    abs(y);
    *c_y+=y;
    return(1);
         }
      return(0);

  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้