

บทที่ 4

กระบวนการความ

4.1 รู้จักกับกระบวนการความ

คำสั่งต่าง ๆ ที่ใช้สั่งเตาให้ทำงาน สามารถนำมารวมกันสร้างเป็นชุดคำสั่งเพื่อทำงานตามต้องการ และตั้งชื่อชุดคำสั่งไว้เพื่อความสะดวกในการเรียกใช้ ชุดคำสั่งในลักษณะนี้เรียกว่า กระบวนการความ เมื่อกระบวนการความถูกสร้างขึ้น ชื่อของกระบวนการความจะเป็นคำสั่งที่เตารู้จัก ชุดคำสั่งของกระบวนการความจะถูกเก็บไว้ในหน่วยความจำขณะทำงาน และจะถูกลบทิ้งเมื่อเลิกใช้โปรแกรมโลโก หรือสั่งลบทิ้ง หากต้องการเก็บกระบวนการความไว้อย่างถาวร สามารถทำได้โดยการบันทึกกระบวนการความเป็นแฟ้มไว้ในหน่วยความจำรอง เช่นแผ่นบันทึก

รูปแบบของกระบวนการความเป็นดังนี้

TO ชื่อกระบวนการความ

คำสั่ง 1

คำสั่ง 2

.

.

.

END

ตัวอย่างที่ 4.1 กระบวนการความ SQUARE

TO SQUARE

FORWARD 50

RIGHT 90

FORWARD 50

RIGHT 90

FORWARD 50

RIGHT 90

FORWARD 50

RIGHT 90

END

- หมายเหตุ 1. กระบวนการในภาษาโลโกจะเริ่มต้นด้วย TO ตามด้วยชื่อกระบวนการ และจบด้วย END
2. ชื่อของกระบวนการที่ตั้งจะต้องไม่ซ้ำกับคำสั่งในภาษา โลโก

ในการทำงานครั้งหนึ่ง ๆ นั้น อาจมีการสร้างกระบวนการหลาย ๆ กระบวนการ โดยแต่ละจะรู้จักชื่อกระบวนการที่สร้างขึ้นเป็นคำสั่งเสมอ และเมื่อมีการสั่งบันทึกกระบวนการลงแผ่นบันทึกในชื่อใดชื่อหนึ่ง เพิ่มกระบวนการที่ได้จะประกอบด้วยกระบวนการต่าง ๆ ที่ถูกจำไว้ในหน่วยความจำขณะทำงานในขณะนั้น

สำหรับกรณีที่เราบรรจ (load) เพิ่มกระบวนการเข้ามาใช้งาน กระบวนการต่าง ๆ ในเพิ่มกระบวนการจะเข้ามาเพิ่มเติมให้กับกระบวนการที่มีอยู่เดิม โดยจำไว้ในหน่วยความจำแรม และแต่ละจะรู้จักชื่อกระบวนการทั้งหมดเป็นคำสั่งแต่ละคำสั่ง

4.2 การสร้างและแก้ไขกระบวนการ

การสั่งเตาให้วาดรูปสี่เหลี่ยมจัตุรัสเป็นเรื่องที่ไม่ยาก แต่การที่จะต้องพิมพ์คำสั่งหลาย ๆ คำสั่งทุกครั้งที่ต้องการวาดสี่เหลี่ยมจัตุรัส จะทำให้การสั่งเตาวาดภาพทำได้ช้า ดังนั้นถ้าต้องการสั่งเตาให้วาดรูปสี่เหลี่ยมจัตุรัสอีกโดยไม่ต้องเขียนคำสั่งใหม่ก็ทำได้โดยเขียนโปรแกรมหรือสร้างกระบวนการให้เตาวาดสี่เหลี่ยมจัตุรัสเก็บไว้ แล้วเรียกมาใช้งานเมื่อต้องการรูปสี่เหลี่ยมจัตุรัส จะทำให้การสั่งงานได้เร็วและ สะดวกสบายขึ้น เพราะไม่ต้องสั่งคำสั่งซ้ำ ๆ จำนวนมาก

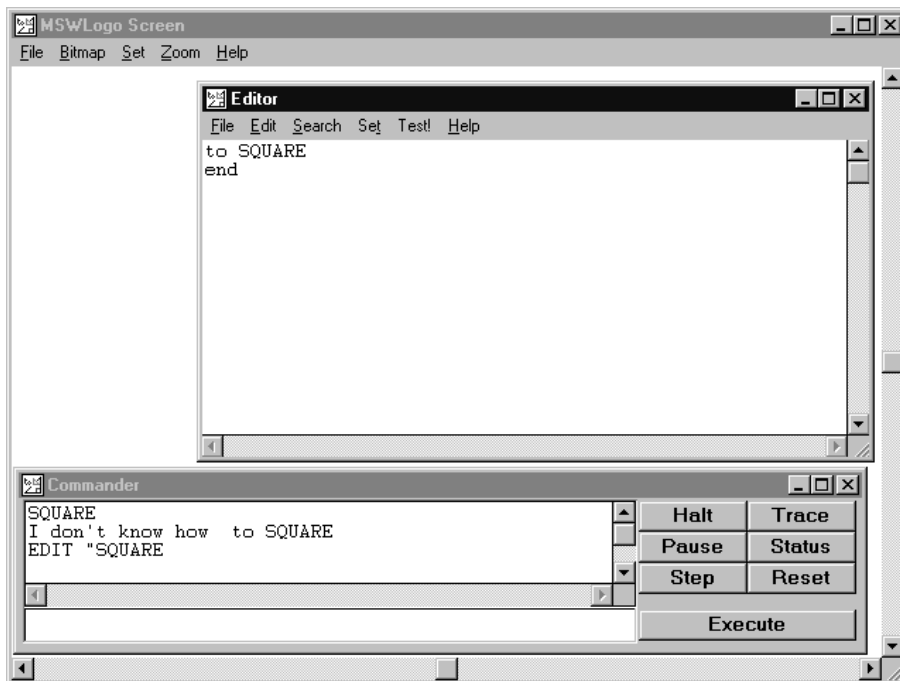
วิธีหนึ่งในการสร้างหรือแก้ไขกระบวนการที่สามารถทำได้ง่าย ๆ ในภาษาโลโก คือการใช้คำสั่ง EDIT ตามด้วยชื่อกระบวนการ โดยมีเครื่องหมาย " นำหน้าชื่อ และชื่อที่ใช้จะต้องไม่ซ้ำกับคำสั่งพื้นฐานของภาษา โลโก แต่ถ้าชื่อกระบวนการเป็นชื่อกระบวนการที่เตารู้จัก จะเป็นการเรียกกระบวนการนั้นมาแก้ไข จากนั้นจะปรากฏหน้าต่างแก้ไขให้ใช้ในการดำเนินการต่อไป

การเรียกหน้าต่างเพื่อใช้สร้างกระบวนการหรือแก้ไขกระบวนการทำได้ดังนี้ จากแถบเมนูในหน้าต่าง MSWLogo Screen ให้คลิกคำสั่ง File และคำสั่ง Edit จะปรากฏหน้าต่าง Edit Procedure ให้พิมพ์ชื่อกระบวนการที่ต้องการสร้าง หรือเลือกชื่อกระบวนการที่ต้องการแก้ไข จะปรากฏหน้าต่าง Edit Procedure ซึ่งมีชุดคำสั่งของกระบวนการที่ต้องการแก้ไข

กิจกรรมที่ 4.1

การสร้างกระบวนการความวาดรูปสี่เหลี่ยมจัตุรัส

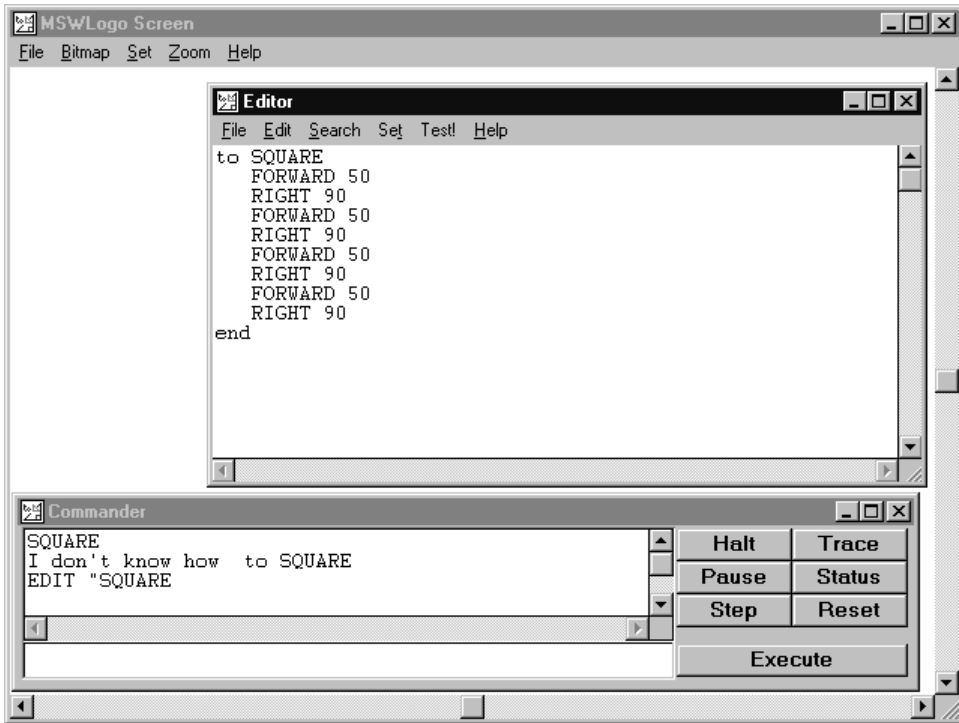
- 1) ต้องการทดสอบว่าเตารู้จักคำสั่ง SQUARE หรือไม่ ให้พิมพ์ SQUARE ในช่องป้อนคำสั่ง ผลที่ได้คือ I don't know how TO SQUARE
แสดงว่า เต่า.....
- 2) ต้องการสร้างกระบวนการความ SQUARE เพื่อให้เตารู้จักคำสั่งนี้ ให้พิมพ์ EDIT "SQUARE ในช่องป้อนคำสั่ง จะปรากฏหน้าต่างแก้ไข (Editor) สำหรับสร้างหรือแก้ไขกระบวนการ ดังรูปที่ 4.1



รูปที่ 4.1 หน้าต่างแก้ไข สำหรับสร้างหรือแก้ไขกระบวนการความ SQUARE

จากรูปที่ 4.1 จะเห็นว่าในหน้าต่างแก้ไขมีข้อความอยู่ 2 บรรทัด บรรทัดแรกคือ TO SQUARE บอกให้เตารู้ว่าชุดคำสั่งสำหรับกระบวนการความ SQUARE จะเริ่มต้นที่นี้ และคำสั่งในบรรทัดที่สองคือ END เพื่อบอกเต่าว่า ชุดคำสั่งสำหรับ SQUARE จะจบที่ตำแหน่งนี้

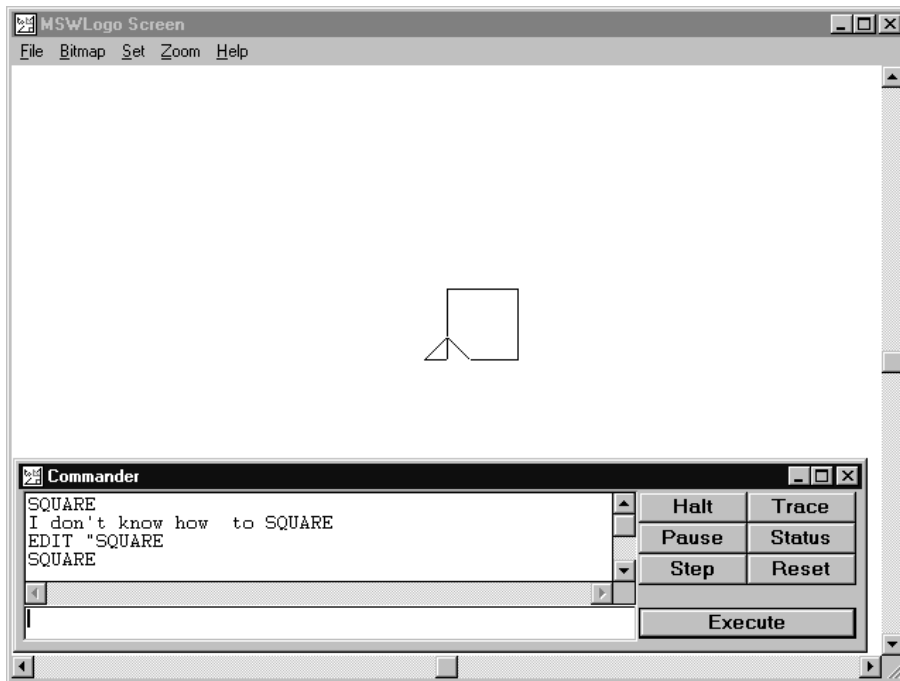
- 3) พิมพ์ชุดคำสั่งสำหรับสร้างกระบวนการความ SQUARE ในหน้าต่างแก้ไข ดังรูปที่ 4.2



รูปที่ 4.2 ชุดคำสั่งสำหรับสร้างกระบวนการความ SQUARE ในหน้าต่างแก้ไข

จากรูปที่ 4.2 คำสั่ง 8 บรรทัดที่พิมพ์เพิ่มแทรกระหว่าง TO SQUARE กับ END คือชุดคำสั่งที่สั่งให้เต่าวาดรูปสี่เหลี่ยมจัตุรัสที่มีพื้นที่ขนาด 50 X 50 ตารางหน่วย

- 4) ต้องการออกจากการสร้างกระบวนการความ SQUARE ให้คลิกคำสั่ง File และคลิกคำสั่ง Exit ผลคือ กระบวนการความ SQUARE จะถูกเก็บไว้ในหน่วยความจำแรม และเต่ารู้จักคำสั่ง SQUARE
- 5) ต้องการทดสอบอีกครั้งว่าเต่ารู้จักคำสั่ง SQUARE หรือไม่ ให้สั่งเต่าด้วยคำสั่ง SQUARE ในช่องป้อนคำสั่ง จะได้ผลดังรูปที่ 4.3



รูปที่ 4.3 เต่าวาดรูปสี่เหลี่ยมจัตุรัสตามคำสั่งในกระบวนความ SQUARE

6) พิมพ์ข้อความ square ในช่องป้อนคำสั่ง

ผลที่ได้.....

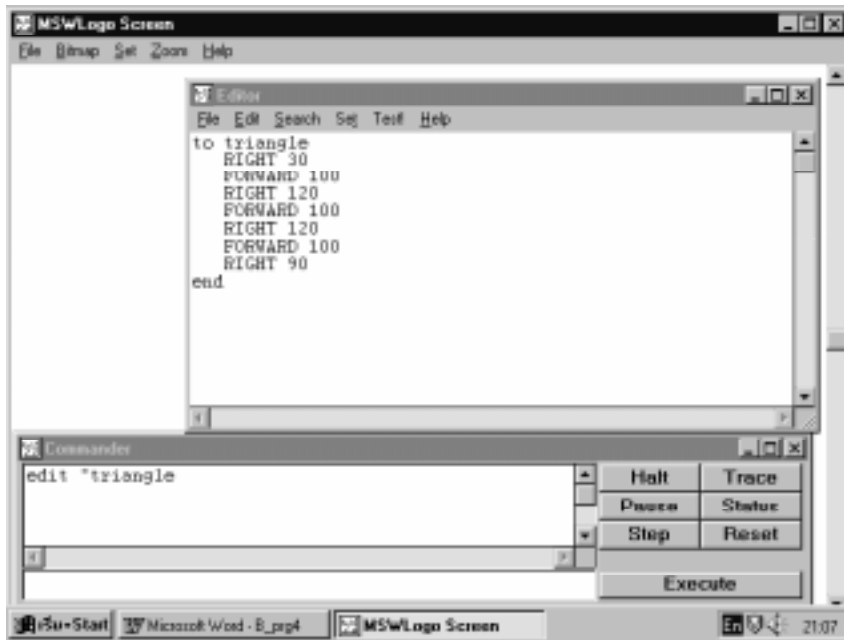
แสดงว่า เต่ารับรู้คำสั่ง SQUARE ทั้งอักษรตัวพิมพ์ใหญ่หรือพิมพ์เล็กโดยถือว่าเป็นคำสั่งเดียวกัน

กิจกรรมที่ 4.2

การสร้างกระบวนความวาดรูปสามเหลี่ยมด้านเท่า

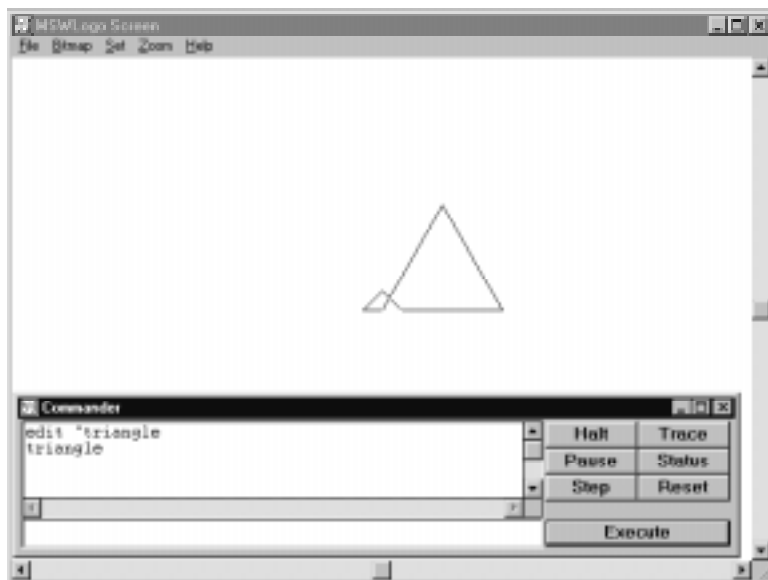
สร้างกระบวนความชื่อ triangle สำหรับวาดรูปสามเหลี่ยมด้านเท่า ดังต่อไปนี้

- 1) พิมพ์คำสั่ง edit “triangle”ในช่องป้อนคำสั่ง แล้วกด Enter
- 2) เมื่อปรากฏหน้าต่างแก้ไข ให้พิมพ์ชุดคำสั่งสำหรับสร้างกระบวนความ triangle ดังรูปที่ 4.4



รูปที่ 4.4 ชุดคำสั่งสำหรับสร้างกระบวนความ triangle ในหน้าต่างแก้ไข

- 3) ออกจากการสร้างกระบวนความ triangle โดยคลิกคำสั่ง File และคลิกคำสั่ง Exit ผลคือ กระบวนความ triangle ถูกจำไว้ในหน่วยความจำแรม และเตารู้อักคำสั่ง triangle แล้ว
- 4) พิมพ์คำสั่ง triangle ในช่องป้อนคำสั่ง จะได้ผลดังรูปที่ 4.5



รูปที่ 4.5 ผลลัพธ์จากคำสั่ง triangle

4.3 การบันทึกกระบวนการความลงในแผ่นบันทึก

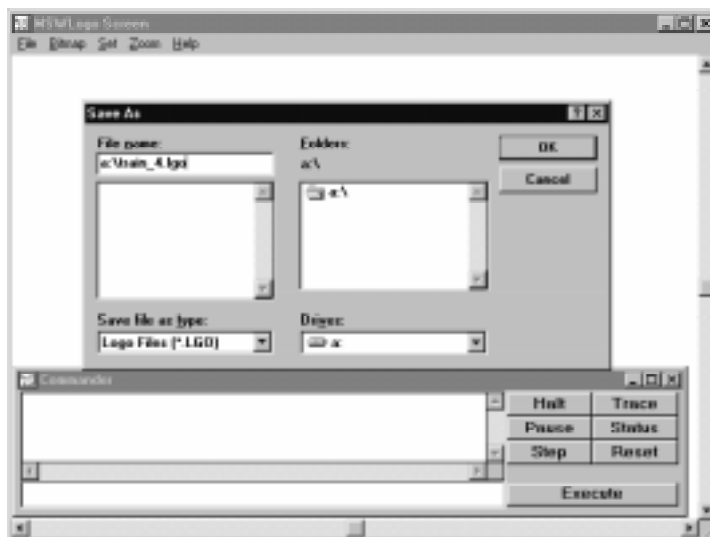
เมื่อออกจากการแก้ไขกระบวนการ ความ กระบวนการจะถูกจำไว้ในหน่วยความจำแรม และจะหายไปถ้าออกจากโปรแกรม โลโก หากต้องการเก็บกระบวนการไว้ใช้งานต่อไป ก็ต้องทำการบันทึกกระบวนการความลงในแผ่นบันทึก โดยคลิกคำสั่ง File บนแถบเมนูในหน้าต่าง MSWLogo Screen และคลิกคำสั่ง Save เมื่อปรากฏกรอบโต้ตอบ (Save as) ให้ตั้งชื่อแฟ้ม (file) กำหนดชื่อเครื่องจับแผ่นบันทึก (drive) และกำหนดสถานที่เก็บแฟ้ม (folder) แล้วคลิกปุ่ม OK

กิจกรรมที่ 4.3

การบันทึกแฟ้มกระบวนการความ

ให้บันทึกกระบวนการความในกิจกรรมที่ 4.1 และ 4.2 ลงบนแผ่นบันทึก ตามขั้นตอนดังนี้

- 1) ในหน้าต่าง MSWLogo Screen ให้คลิกคำสั่ง File บนแถบเมนู และคลิกคำสั่ง Save จะปรากฏกรอบโต้ตอบ Save as
- 2) ตั้งชื่อแฟ้มโดยคลิกในช่อง File name แล้วพิมพ์ชื่อ TRAIN_4.LGO
- 3) กำหนดชื่อเครื่องจับแผ่นบันทึกที่ใช้เก็บแฟ้มโดยคลิกในช่อง Drives แล้วเลือกหน่วยจับ A ถ้าต้องการระบุสารบย่อ ให้คลิกที่ช่อง Folders แล้วเลือกสารบย่อที่ต้องการ ดังรูปที่ 4.6 แล้วคลิกปุ่ม OK ผลที่ได้คือ แฟ้ม train_4.lgo จะประกอบด้วยกระบวนการ 2 กระบวนการ คือ SQUARE และ triangle ถูกจัดเก็บบนแผ่นบันทึกในหน่วยจับ A



รูปที่ 4.6 หน้าต่างสำหรับบันทึกแฟ้มกระบวนการความ

4.4 การเริ่มต้นงานใหม่และการดึงเพิ่มกระบวนการความมาใช้งาน

หลังจากทำกิจกรรมที่ 4.1 และ 4.2 แล้ว เตาจะรู้จักคำสั่ง SQUARE และคำสั่ง triangle อยู่เสมอ ถ้าต้องการเริ่มต้นทำชิ้นงานใหม่โดยไม่ต้องปิดเครื่องและให้เตาลืมคำสั่งต่าง ๆ ทั้งหมด ทำได้ในหน้าต่าง MSWLogo Screen โดยให้คลิกคำสั่ง File บนแถบเมนู และคลิกคำสั่ง New จะทำให้หน่วยความจำแรมไม่มีกระบวนการความใด ๆ ถูกเก็บไว้ หมายถึงเตาจะไม่รู้จักกระบวนการความใด ๆ รวมทั้งกระบวนการความ SQUARE และ triangle ด้วย

ในการทำงานเรามักจะต้องบรรจุเพิ่มกระบวนการความที่บันทึกเอาไว้ในแผ่นบันทึกมาเก็บไว้ในหน่วยความจำแรมเพื่อใช้งานต่อไป เพราะทำให้ประหยัดเวลา กำลังงาน และค่าใช้จ่ายที่จะต้องไปดำเนินการสร้างกระบวนการความใหม่ทุกครั้งที่ต้องการใช้งาน การบรรจุเพิ่มกระบวนการความสามารถทำได้ง่าย ๆ ในหน้าต่าง MSWLogo Screen โดยให้คลิกคำสั่ง File บนแถบเมนู และคลิกคำสั่ง Load จะได้กรอบโต้ตอบ Open ให้ระบุชื่อแฟ้ม กำหนดชื่อเครื่องขับแผ่นบันทึก กำหนดสถานที่เก็บแฟ้ม แล้วคลิกปุ่ม OK

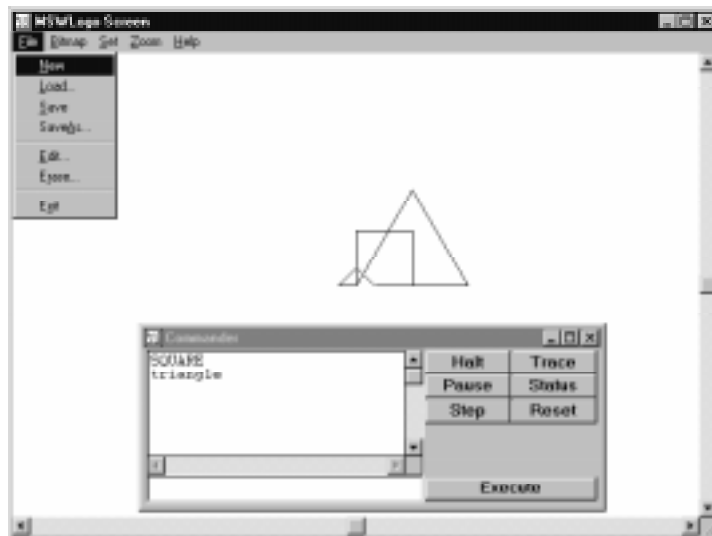
ในการบรรจุเพิ่มกระบวนการความเข้ามาใช้งาน ถ้าหน่วยความจำแรมในขณะนั้นจำกระบวนการความอื่นอยู่ กระบวนการความในแผ่นบันทึกจะเข้ามาเพิ่มเติมให้กับกระบวนการความที่มีอยู่ในหน่วยความจำขณะทำงาน ซึ่งเตาจะรู้จักชื่อกระบวนการความทั้งหมดเป็นคำสั่งด้วย

กิจกรรมที่ 4.4

การเริ่มต้นงานใหม่และการดึงเพิ่มกระบวนการความมาใช้งาน

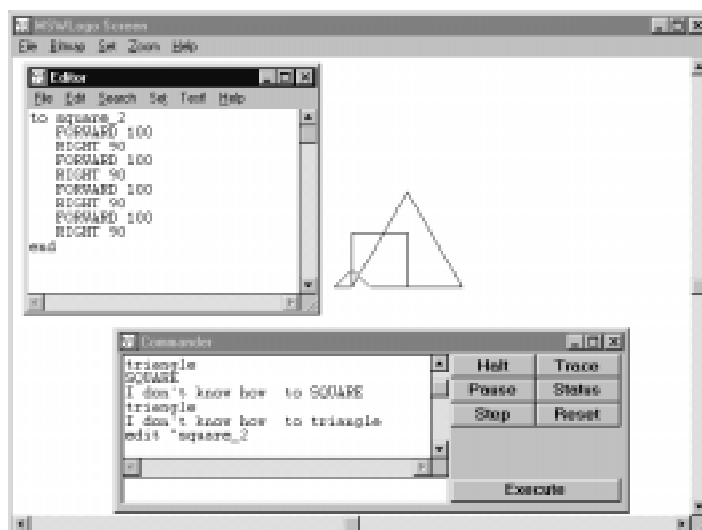
กิจกรรมนี้จะทำต่อจากกิจกรรมที่ 4.1, 4.2 และ 4.3 ซึ่งหมายถึงเตารู้จักคำสั่ง SQUARE, TRIANGLE

- 1) พิมพ์คำสั่ง SQUARE ในช่องป้อนคำสั่งเพื่อสั่งเตาวาดรูปสี่เหลี่ยม จะได้รูปสี่เหลี่ยม
- 2) พิมพ์คำสั่ง triangle จะได้รูปสามเหลี่ยม แสดงว่า ขณะนี้เตารู้จักคำสั่ง SQUARE และ triangle
- 3) ต้องการเริ่มต้นทำงานชิ้นใหม่ ให้คลิกคำสั่ง File บนแถบเมนูในหน้าต่าง MSWLogo และคำสั่ง New ดังรูปที่ 4.7



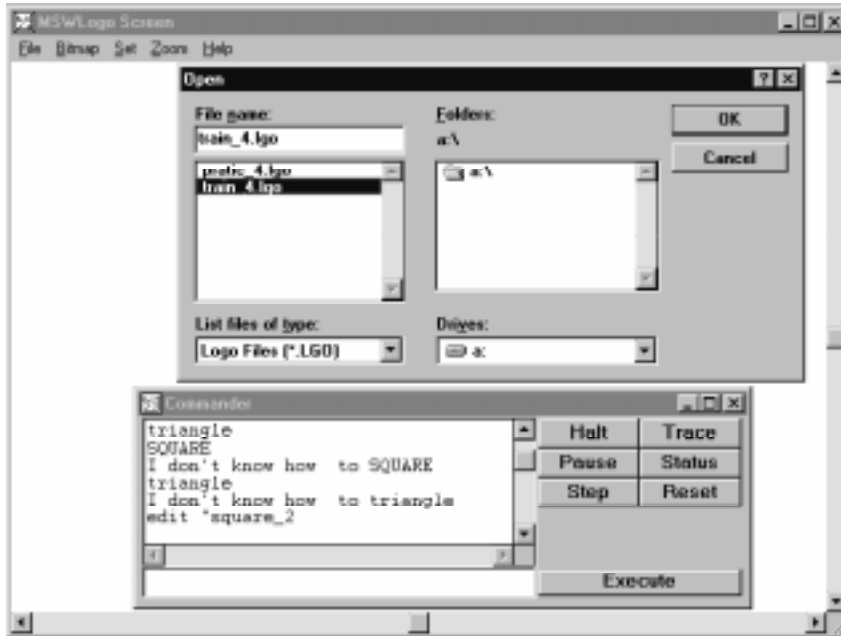
รูปที่ 4.7 การใช้คำสั่ง File และ New เพื่อเริ่มต้นงานใหม่

- 4) ในช่องป้อนคำสั่ง พิมพ์คำสั่ง SQUARE อีกครั้ง ผลคือ เต่าไม่รู้จักคำสั่งนี้
- 5) ทดสอบป้อนคำสั่ง triangle ในช่องคำสั่ง ผลคือ เต่าไม่รู้จักคำสั่งนี้เช่นกัน
- 6) ต้องการสร้างกระบวนการความชื่อ SQUARE_2 ซึ่งเป็นกระบวนการสั่งให้เต่าวาดรูปสี่เหลี่ยมที่มีพื้นที่ขนาด 100 X 100 ตารางหน่วย โดยพิมพ์ชุดคำสั่งในหน้าต่างแก้ไข ดังรูปที่ 4.8 แล้วคลิกคำสั่ง File และ คำสั่ง Exit



รูปที่ 4.8 ลำดับคำสั่งในกระบวนการ SQUARE_2

- 7) ต้องการบรรจุแฟ้ม train4.LGO ให้คลิกคำสั่ง File บนแถบเมนู และคลิกคำสั่ง Load ในหน้าต่าง MSWLogo Screen จะปรากฏกรอบโต้ตอบ Open ให้คลิกในช่อง drives เลือกหน่วยขับ A แล้วคลิกชื่อแฟ้ม train_4.LGO ในช่อง file name คลิกปุ่ม OK ดังรูปที่ 4.9



รูปที่ 4.9 การบรรจุแฟ้มกระบวนการความจากแผ่นบันทึกเข้าสู่หน่วยความจำแรม

- 8) ต้องการทดสอบว่า เตารู้จักคำสั่ง SQUARE และ triangle ซึ่งเป็นกระบวนการที่อยู่ในแฟ้มชื่อ train4.LGO และคำสั่ง square_2 ที่สร้างขึ้นใหม่ ให้พิมพ์คำสั่งทั้งสามในช่องป้อนคำสั่ง ผลที่ได้คือ.....
- 9) ต้องการบันทึกแฟ้มกระบวนการทั้งสามในชื่อเดิมอีกครั้ง ทำได้โดยคลิกคำสั่ง File บนแถบเมนูและคลิกคำสั่ง Save

4.5 การลบกระบวนการบางกระบวนการ

การเริ่มต้นงานใหม่โดยใช้คำสั่ง New จะเป็นการลบคำสั่งทั้งหมดที่อยู่ในหน่วยความจำแรม ซึ่งบางครั้งไม่ตรงกับความต้องการที่จะลบบางกระบวนการที่ไม่ต้องการเท่านั้น ภาษาโลโก้จึงมีคำสั่งช่วยลบเฉพาะกระบวนการที่ไม่ต้องการ คือ Erase ตามด้วยชื่อกระบวนการ ซึ่งจะทำให้กระบวนการที่มีชื่อตามที่ระบุจะถูกลบออกจากหน่วยความจำแรม และแต่จะไม่รู้จักกระบวนการนี้อีก

การลบกระบวนการโดยใช้คำสั่ง Erase เป็นการลบออกจากหน่วยความจำแรมเท่านั้น แต่จะไม่ลบออกจากแฟ้มกระบวนการในแผ่นบันทึก ถ้าต้องการลบออกจากแฟ้มกระบวนการในแผ่นบันทึกด้วย จะต้องทำการบันทึกแฟ้มกระบวนการใหม่

วิธีการใช้คำสั่ง Erase นอกจากพิมพ์คำสั่งในช่องป้อนคำสั่งแล้ว ยังทำได้ในหน้าต่าง MSWLogo Screen โดยคลิกคำสั่ง File บนแถบเมนู และคลิกคำสั่ง Erase จะปรากฏกรอบโต้ตอบ Erase Procedure ซึ่งจะแสดงชื่อกระบวนการที่มีอยู่ในหน่วยความจำแรมในขณะนั้นให้เลือก เมื่อเลือกชื่อกระบวนการที่ต้องการลบแล้ว ให้คลิกปุ่ม OK แต่ถ้าต้องการลบกระบวนการทั้งหมดให้คลิกปุ่ม ALL

กิจกรรมที่ 4.5

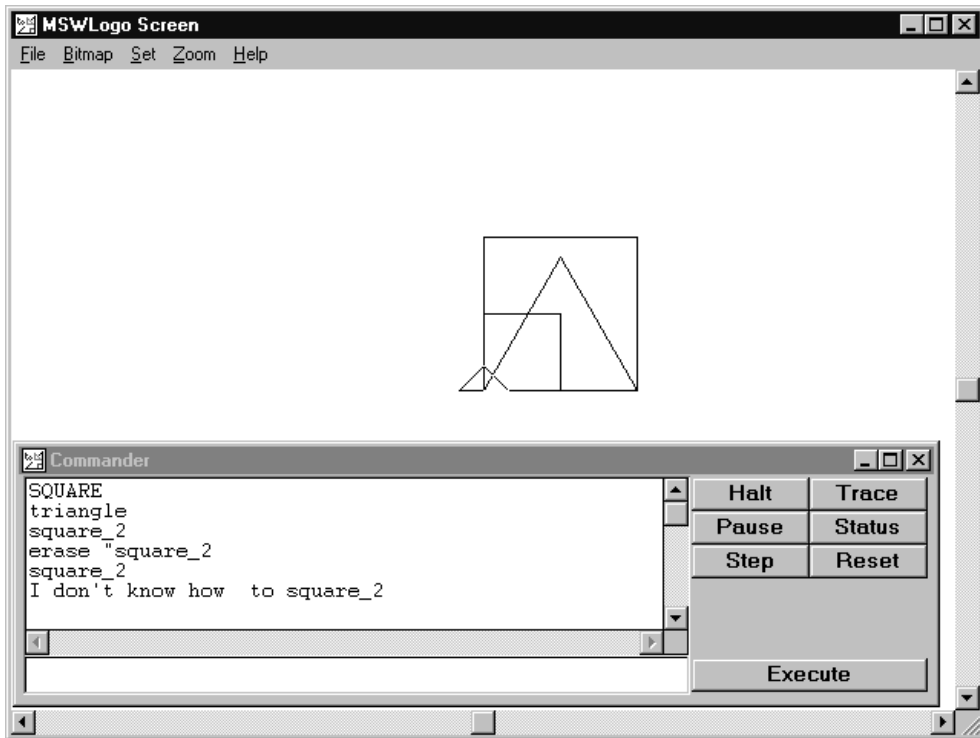
การลบกระบวนการบางกระบวนการ

ให้ลบกระบวนการบางกระบวนการ โดยสั่งแต่เริ่มต้นชิ้นงานใหม่ แล้วจึงเพิ่มกระบวนการ train_4 เข้ามาในหน่วยความจำ แล้วลบกระบวนการ square_2 โดยใช้คำสั่ง erase และลบกระบวนการ triangle โดยกระทำในหน้าต่าง MSWLogo Screen

การลบกระบวนการ square_2 โดยใช้คำสั่ง erase ตามขั้นตอน ดังนี้

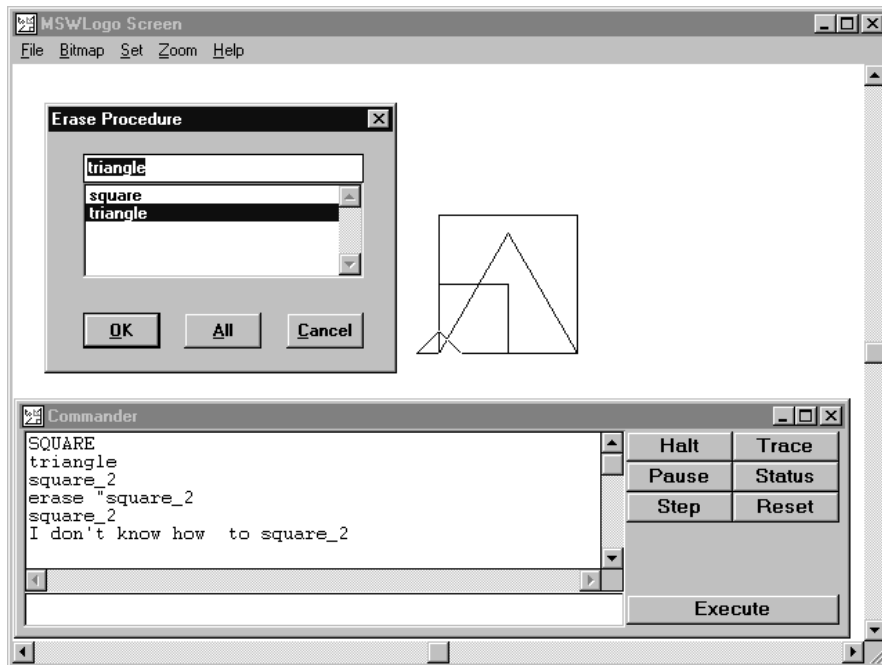
- 1) ในหน้าต่าง MSWLogo Screen บนแถบรายการเมนู ให้คลิกคำสั่ง File และคลิกคำสั่ง New
- 2) คลิกคำสั่ง File และคลิกคำสั่ง Load เมื่อปรากฏหน้าต่าง Open ให้คลิกในช่อง drives เลือกช่องขับ A แล้วคลิกชื่อแฟ้ม train_4 ในช่อง file name คลิกปุ่ม OK

- 3) ทดสอบว่า เตารู้จักคำสั่ง SQUARE คำสั่ง triangle และคำสั่ง square_2 โดยการสั่งเต่าในช่องป้อนคำสั่งสังเกตผลที่ได้
- 4) พิมพ์คำสั่งerase “ square_2 ในช่องป้อนคำสั่ง
- 5) ทดสอบป้อนคำสั่ง square_2 ในช่องป้อนคำสั่ง ผลคือ เต่าไม่รู้จักคำสั่งนี้ ดังรูปที่ 4.10



รูปที่ 4.10 ผลลัพธ์ที่ได้เมื่อลบกระบวนการความ square_2

- 6) ให้คลิกคำสั่ง File บนแถบเมนู และคลิกคำสั่ง Erase
- 7) เมื่อปรากฏกรอบโต้ตอบ Erase Procedure ให้คลิกชื่อกระบวนการ triangle ดังรูปที่ 4.11 แล้วคลิกปุ่ม OK (ถ้าต้องการลบกระบวนการทั้งหมดให้คลิกปุ่ม ALL ถ้าไม่ต้องการลบกระบวนการใด ๆ ให้คลิกปุ่ม Cancel)



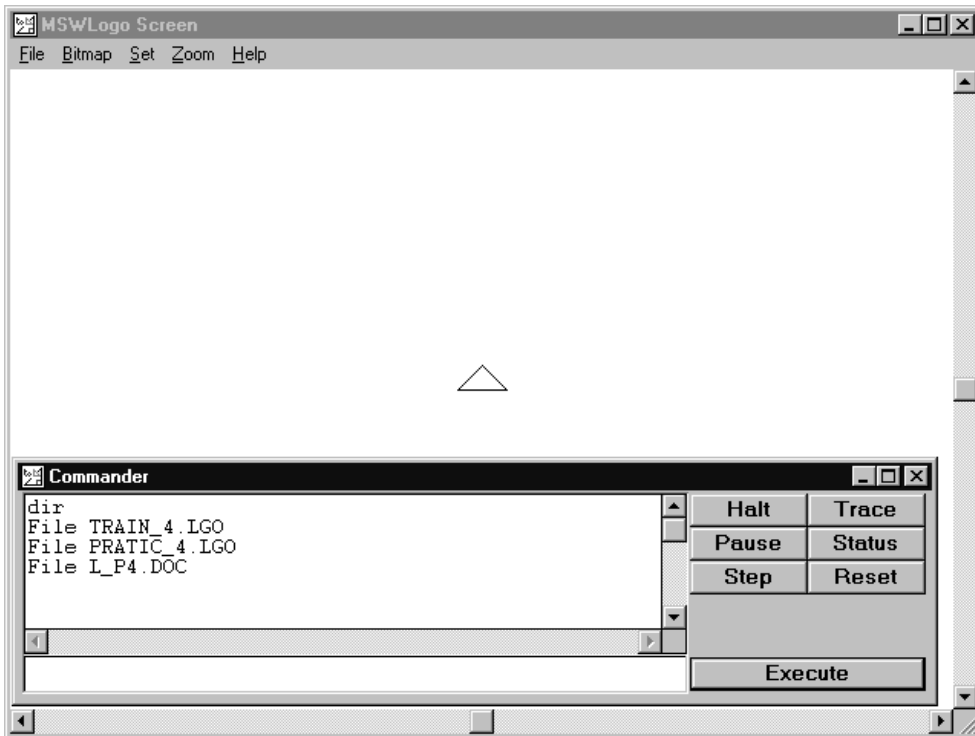
รูปที่ 4.11 ผลลัพธ์ที่ได้เมื่อลบกระบวนการความ square_2

8) ทดสอบป้อนคำสั่ง triangle ในช่องป้อนคำสั่ง ผลคือ เต่าไม่รู้จักคำสั่งนี้

4.6 การเรียกดูรายชื่อแฟ้มข้อมูล

เมื่อสั่งบันทึกแฟ้มกระบวนการลงในหน่วยความจำรองที่กำลังติดต่อกแล้ว จะสามารถตรวจสอบได้ว่า แฟ้มกระบวนการได้รับการบันทึกจริงหรือไม่ โดยใช้คำสั่ง DIR ในช่องป้อนคำสั่ง ซึ่งจะเป็นการเรียกดูรายชื่อแฟ้มข้อมูลทั้งหมดที่บันทึกไว้ในแผ่นบันทึกที่กำลังติดต่อใช้งานในขณะนั้น

ตัวอย่างที่ 4.2 ผลที่ได้จากการเรียกดูรายชื่อแฟ้มข้อมูลที่บันทึกไว้ในแผ่นบันทึกที่อยู่ในหน่วยขับ A ซึ่งเป็นหน่วยขับที่กำลังติดต่อใช้งานในขณะนั้น จะพบแฟ้ม train_4.lgo ดังรูปที่ 4.12



รูปที่ 4.12 การเรียกดูรายชื่อแฟ้มข้อมูลด้วยคำสั่ง DIR

4.7 การพิมพ์กระบวนการความด้วยเครื่องพิมพ์

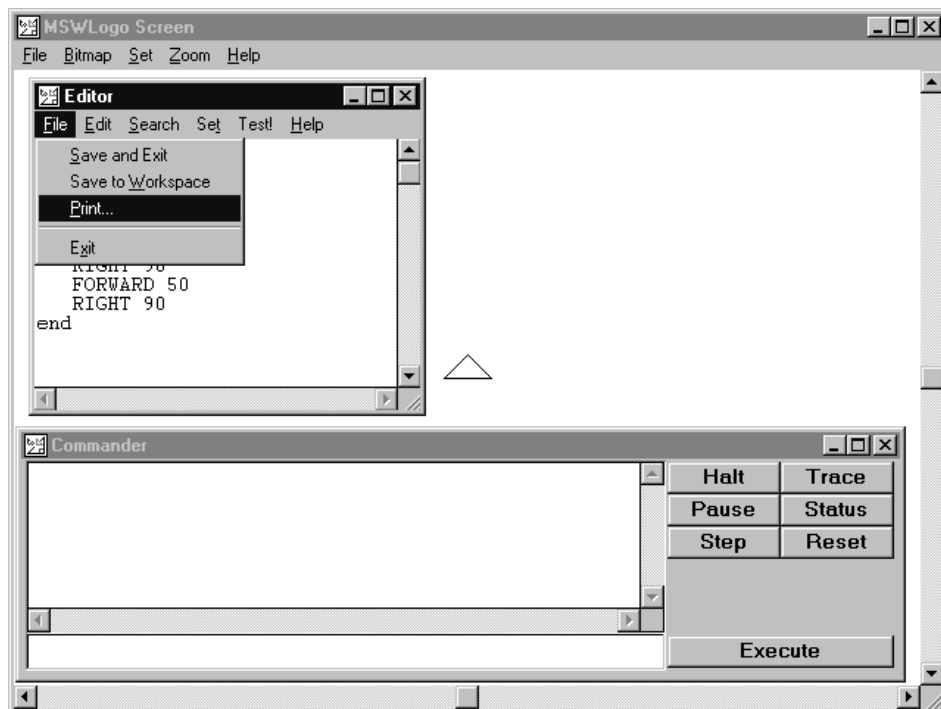
ถ้ามีเครื่องพิมพ์ต่ออยู่กับเครื่องคอมพิวเตอร์ จะสามารถพิมพ์ชุดคำสั่งในกระบวนการความต่าง ๆ ที่อยู่ในหน้าต่าง Editor ลงบนกระดาษได้ โดยการคลิกคำสั่ง File และคำสั่ง Print บนแถบเมนูของหน้าต่าง Editor

กิจกรรมที่ 4.6

การพิมพ์กระบวนการทางเครื่องพิมพ์

ให้พิมพ์กระบวนการ SQUARE ทางเครื่องพิมพ์ ตามขั้นตอนดังนี้

- 1) เรียกกระบวนการ SQUARE มาแสดงในหน้าต่างแก้ไข โดยพิมพ์ edit “SQUARE ในช่องป้อนคำสั่ง
- 2) ถ้าแต่ไม่รู้จักรบวนการ SQUARE ให้บรรจุเพิ่มกระบวนการ train_4 ลงในหน่วยความจำแรม โดยคลิกคำสั่ง File และคำสั่ง Load เมื่อปรากฏกรอบโต้ตอบ Open ให้คลิกชื่อเพิ่ม train_4 คลิกปุ่ม OK แล้วจึงกลับไปทำข้อที่ 1 ใหม่
- 3) สั่งพิมพ์กระบวนการที่อยู่ในหน้าต่าง Editor โดยคลิกคำสั่ง File บนแถบเมนู และคลิกคำสั่ง Print ดังรูปที่ 4.13



รูปที่ 4.13 การพิมพ์กระบวนการ SQUARE ทางเครื่องพิมพ์

4.8 การใช้กระบวนการความสร้างชิ้นงาน

กระบวนการความใด ๆ ที่บรรจุอยู่ในหน่วยความจำแรม เค้าจะถือว่าเป็นคำสั่งที่รู้จัก การใช้กระบวนการหรือคำสั่งเหล่านี้ ทำได้โดยการพิมพ์ชื่อกระบวนการที่ต้องการในช่องคำสั่ง เค้าก็จะทำงานตามชุดของคำสั่งที่นำมาเรียงกันในกระบวนการนั้น เช่น กระบวนการ SQUARE จะได้รูปสี่เหลี่ยมจัตุรัสที่มีพื้นที่ขนาด 50 x 50 ตารางหน่วย เป็นต้น

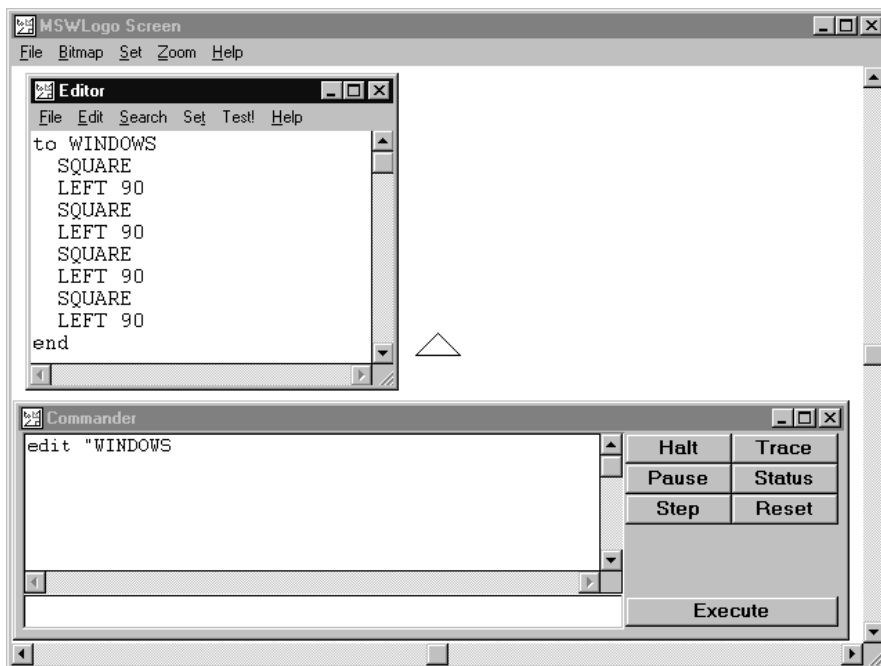
ในการสร้างชิ้นงานใด ๆ สามารถนำกระบวนการที่สร้างขึ้นก่อนมาเป็น คำสั่งในกระบวนการใหม่ได้ ซึ่งทำให้สามารถสร้างชิ้นงานที่ซับซ้อนได้ง่ายขึ้น

กิจกรรมที่ 4.7

กระบวนการวาดหน้าต่าง

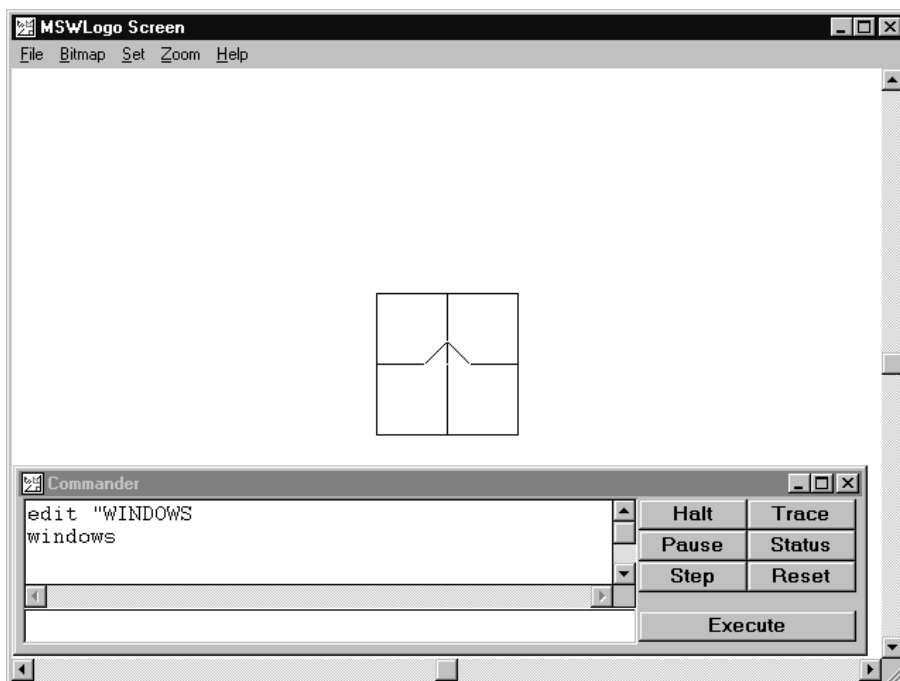
สร้างกระบวนการเพื่อวาดหน้าต่างชื่อ WINDOWS โดยใช้กระบวนการ SQUARE ที่อยู่ในแฟ้มกระบวนการที่ชื่อ POLYGON มาช่วย ทำได้ตามขั้นตอน ดังนี้

- 1) บรรจุแฟ้มกระบวนการ train_4 ลงในหน่วยความจำแรม โดยคลิกคำสั่ง File และคำสั่ง Load เมื่อปรากฏกรอบโต้ตอบ Open ให้คลิกชื่อแฟ้ม train_4 แล้วคลิกปุ่ม OK
- 2) พิมพ์คำสั่ง edit “WINDOWS ในช่องป้อนคำสั่ง
- 3) เมื่อปรากฏหน้าต่าง Editor ให้พิมพ์ชุดคำสั่งดังรูปที่ 4.14 แล้วคลิกคำสั่ง File และคำสั่ง Exit



รูปที่ 4.14 กระบวนการความ WINDOW ที่ใช้กระบวนการความ SQUARE มาช่วย

- 4) สั่งเต้าวาดภาพ โดยพิมพ์คำสั่ง windows ในช่องป้อนคำสั่ง จะได้น้าต่างดังรูป 4.15



รูปที่ 4.15 การวาดภาพด้วยกระบวนการความ WINDOWS

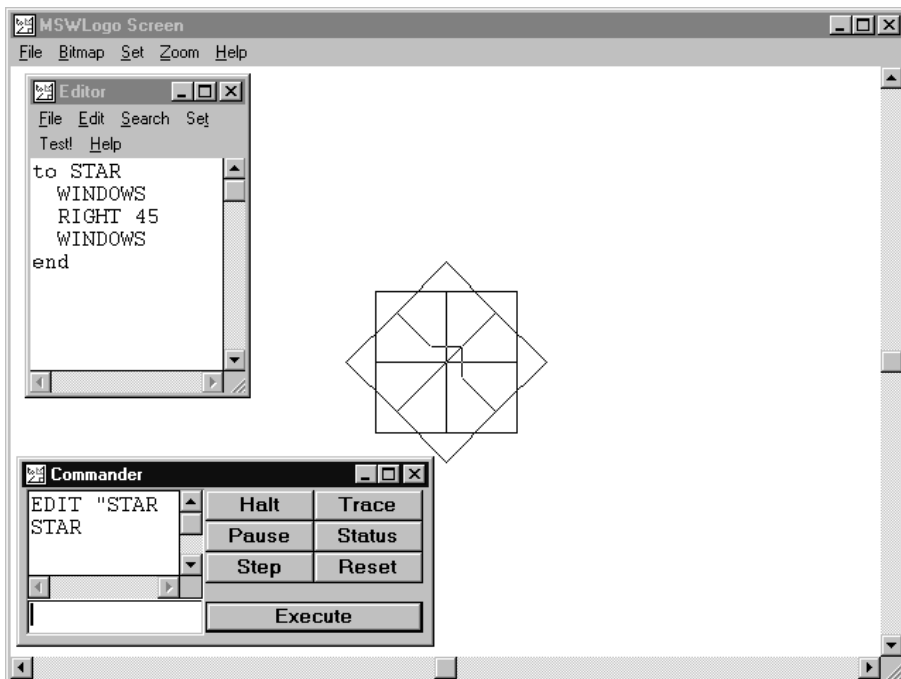
- 5) ต้องการบันทึกกระบวนการความ WINDOWS เก็บเข้าไปในแฟ้ม train_4 เพื่อนำมาใช้ในกิจกรรมต่อไป

กิจกรรมที่ 4.8

การสร้างกระบวนการความวาดดาว

สร้างกระบวนการความวาดดาวชื่อ STAR โดยใช้กระบวนการความ WINDOWS มาช่วย ตามขั้นตอน ดังนี้

- 1) ถ้าแต่ยังไม่รู้จักกระบวนการความ WINDOWS ให้บรรจุแฟ้มกระบวนการความ train_4.lgo ซึ่งมีกระบวนการความ WINDOWS
- 2) พิมพ์คำสั่ง edit “STAR ในช่องป้อนคำสั่ง
- 3) เมื่อปรากฏหน้าต่าง Editor ให้พิมพ์ชุดคำสั่งดังรูปที่ 4.16 แล้วคลิกคำสั่ง File และคำสั่ง Exit จะทำให้เตารู้จักคำสั่ง STAR เพิ่มมาอีกคำสั่ง
- 4) พิมพ์คำสั่ง star ในช่องป้อนคำสั่ง จะได้รูปดาว ดังรูป 4.16



รูปที่ 4.16 ชุดคำสั่งสำหรับวาดดาวและภาพดาวที่เกิดกระบวนการความ STAR

5) บันทึกกระบวนการความ STAR เก็บเข้าไปในแฟ้มกระบวนการ train_4.lgo เพื่อนำมาใช้ในกิจกรรมต่อไป

ข้อสังเกต การวาดรูปดาวนี้ ถ้าไม่ใช้กระบวนการ WINDOWS มาช่วยจะยุ่งยากมาก การวาดดาวทำได้โดยการให้เตาวาดรูปหน้าตาจากรูปแรกแล้วหันหัวเตาไปทางขวา 45 องศา จึงวาดหน้าตาอีกอันซ้อนทับกัน

4.9 การทำงานซ้ำ

การเรียกกระบวนการหรือคำสั่งมาทำงานซ้ำตามจำนวนครั้งที่ต้องการ จะช่วยให้การเขียนโปรแกรมหรือกระบวนการต่าง ๆ ทำได้ง่ายขึ้น โปรแกรมมีขนาดเล็กลง ภาษาโลโกมีคำสั่งที่ใช้เพื่อจุดประสงค์นี้ คือ คำสั่ง REPEAT ซึ่งมีรูปแบบดังนี้

REPEAT n [คำสั่งหรือกระบวนการหรือชุดคำสั่ง]

เมื่อ n คือจำนวนครั้งที่ต้องการทำซ้ำ

ตัวอย่างที่ 4.2 การใช้คำสั่ง REPEAT

จากกระบวนการ SQUARE จะเห็นได้ว่า มีการใช้คำสั่ง FORWARD 50 RIGHT 90 ซ้ำกัน 4 ครั้ง ดังนั้นเพื่อให้ไม่ต้องเขียนคำสั่งทั้งสองซ้ำกัน 4 ครั้ง แต่ให้ทำงาน 4 ครั้งเหมือนเดิม ก็ใช้คำสั่ง REPEAT มาช่วย เขียนเป็นกระบวนการใหม่ชื่อ SQUARE1 ดังนี้

TO SQUARE1

REPEAT 4 [FORWARD 50 RIGHT 90]

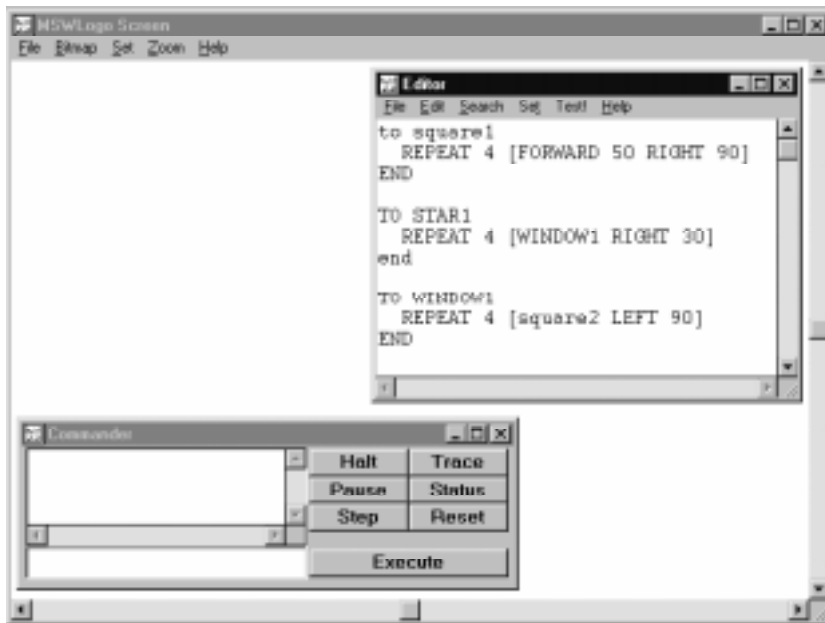
END

กิจกรรมที่ 4.9

การใช้คำสั่ง REPEAT สร้างกระบวนการ

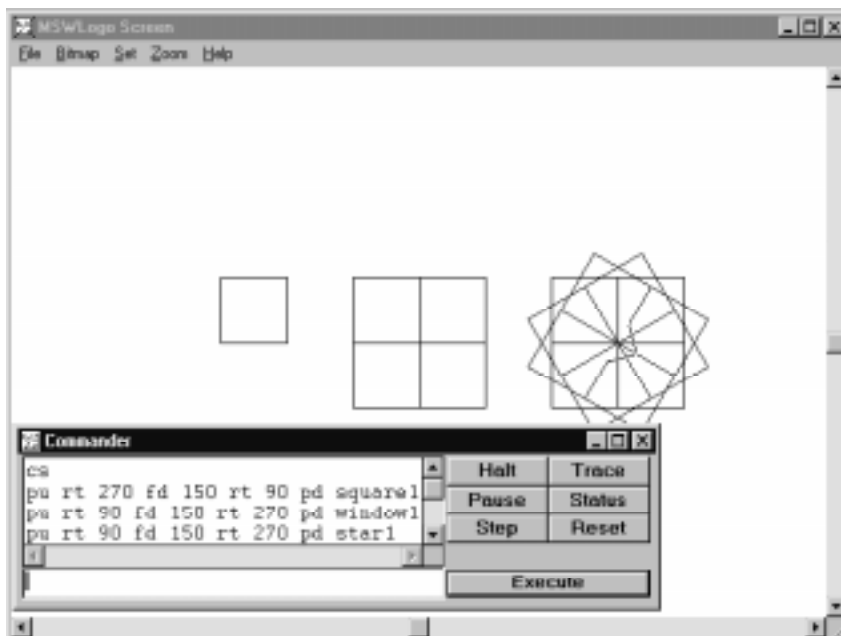
ต้องการสร้างกระบวนการ SQUARE1 WINDOW1 และ STAR1 เพื่อวาดรูปสี่เหลี่ยมจัตุรัสโดยใช้คำสั่ง REPEAT ทำตามขั้นตอน ดังนี้

1) สร้างกระบวนการ SQUARE1 WINDOW1 และ STAR1 โดยพิมพ์คำสั่งตามรูปที่ 4.17



รูปที่ 4.17 ชุดคำสั่งสำหรับสร้างกระบวนการ SQUARE1 WINDOW1 และ STAR1

- 2) พิมพ์คำสั่งเพื่อทดสอบการใช้กระบวนการ SQUARE1 WINDOW1 และ STAR1 ในช่องป้อนคำสั่ง ดังรูปที่ 4.18



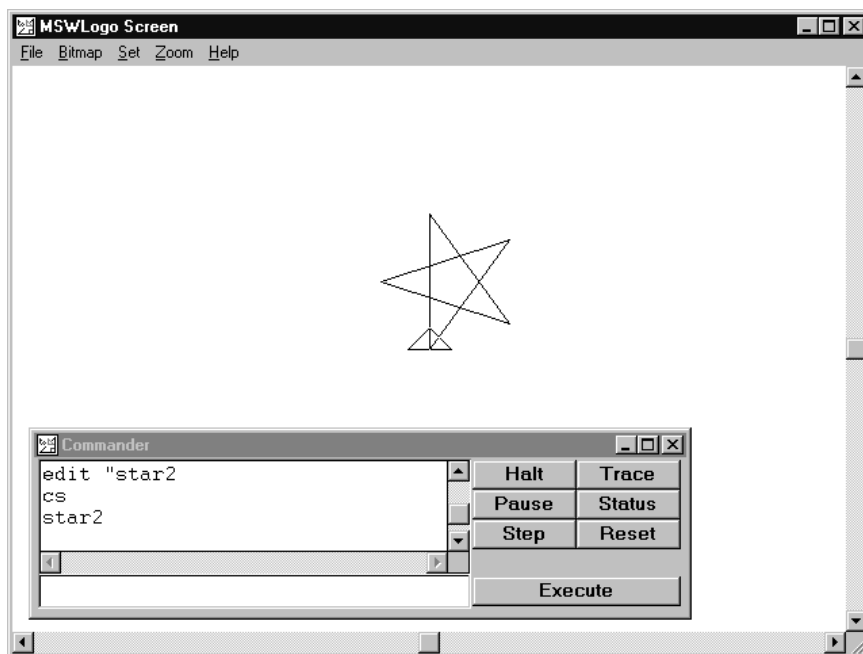
รูปที่ 4.18 รูปที่วาดจากกระบวนการ SQUARE1 WINDOW1 และ STAR1 ตามลำดับ

- 3) ให้พิจารณาผลที่ได้เปรียบเทียบกับกระบวนการความ SQUARE WINDOW และ STAR
- 4) ให้บันทึกกระบวนการทั้งหมดเก็บเข้าไปในแฟ้มกระบวนการชื่อ train_41.lgo เพื่อนำมาใช้ในกิจกรรมอื่นต่อไป

กิจกรรมที่ 4.10

กระบวนการวาดดาวห้าแฉก

- 1) พิจารณาคำสั่งต่อไปนี้ซึ่งเป็นคำสั่งวาดรูปดาวห้าแฉกชื่อดังรูป 4.19 แต่ยังคงขนาดของมุม REPEAT 5 [FORWARD 50 RIGHT...] ให้หาว่ามุมนี้มีค่าเท่ากับ องศา



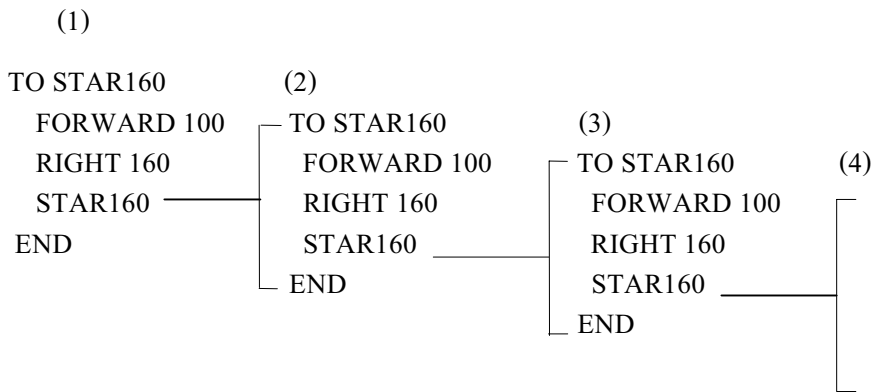
รูปที่ 4.19 การวาดดาวห้าแฉก

- 2) พิมพ์คำสั่งวาดรูปดาวห้าแฉกที่สมบูรณ์ในช่องป้อนข้อมูล
- 3) พิจารณาผลที่ได้แล้วเปรียบเทียบกับรูปที่ 4.19

4.10 การเรียกซ้ำ

การเรียกซ้ำ (recursion) คือการเรียกกระบวนการความซึ่งเป็นตัวเองมาทำงาน เพื่อให้เข้าใจความหมายของการเรียกซ้ำ ให้พิจารณาตัวอย่างแสดงการทำงานของกระบวนการ STAR160 ซึ่งเป็นกระบวนการที่มีการเรียกซ้ำ

ตัวอย่างที่ 4.3 แสดงการทำงานของกระบวนการ STAR160 มีการเรียกใช้ตัวเองดังนี้



จะเห็นว่าเมื่อมีการสั่งให้เต่าทำงานตามกระบวนการความนี้ เต่าจะเริ่มโดยทำคำสั่ง FORWARD 100 และ RIGHT 160 เมื่อมาพบคำสั่ง STAR160 ซึ่งได้นิยามไว้แล้วในกระบวนการความ TO STAR160 ก็จะไปทำกระบวนการความนี้อีก โดยทำคำสั่ง FORWARD 100 RIGHT 160 แล้ว STAR160 อีก เช่นนี้เรื่อยไป โดยไม่มีการหยุด ถ้าต้องการให้หยุดทำได้ดังนี้

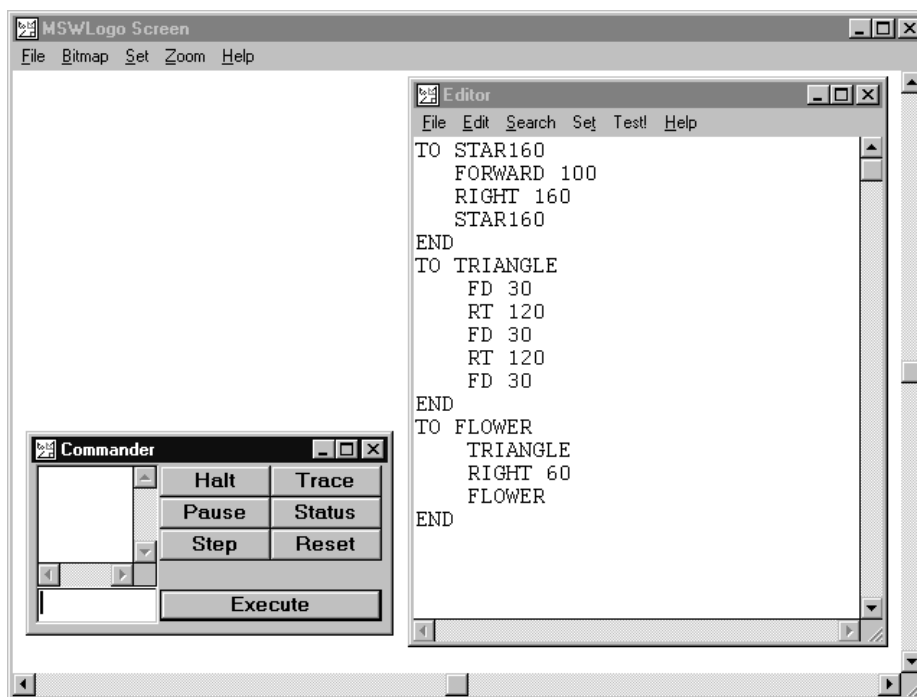
- 1) ถ้าต้องการจะหยุดชั่วคราวเพื่อดูขั้นตอนการทำงานของคำสั่ง ให้คลิกปุ่ม Pause ในหน้าต่างบังคับการ โปรแกรมจะหยุดทำงานชั่วคราวและจะปรากฏกรอบโต้ตอบ Pause mode เพื่อรอรับการป้อน ข้อมูลเพื่อให้โปรแกรมทำงานต่อไป
- 2) ถ้าต้องการหยุดการทำงานให้คลิกปุ่ม Halt ในหน้าต่างบังคับการ

กิจกรรมที่ 4.11

กระบวนการวาดรูปดาวและดอกไม้

สั่งให้เต่าวาดรูปดาวและดอกไม้ดังรูป 4.21 ทำตามขั้นตอนดังนี้

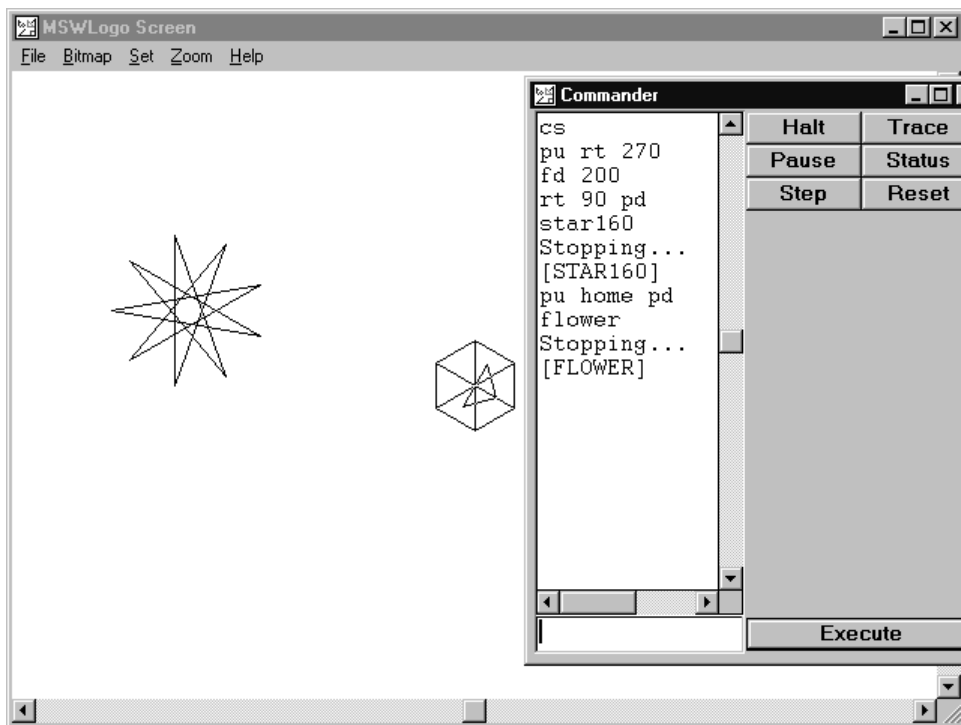
- 1) สร้างกระบวนการต่าง ๆ เพื่อวาดรูปดาวและดอกไม้ ดังรูปที่ 4.20



รูปที่ 4.20 กระบวนการวาดภาพดาวและดอกไม้

ในรูปที่ 4.20 มีจำนวนกระบวนการทั้งสิ้นกระบวนการ ความชื่อของกระบวนการคือ.....

- 2) พิมพ์คำสั่ง CS PU RT 270 FD 200 RT 90 PD และคำสั่ง STAR160 ในช่องป้อนคำสั่ง ดูผลที่ได้ และหยุดการทำงานของกระบวนการ STAR160 โดยการคลิกปุ่ม Halt
- 3) สั่งคำสั่ง PU HOME PD และคำสั่ง FLOWER ในช่องป้อนคำสั่ง ดูผลที่ได้ และหยุดการทำงานของกระบวนการ FLOWER โดยการคลิกปุ่ม Halt



รูปที่ 4.21 รูปที่ได้จากกระบวนการ STAR160 และ FLOWER

แบบฝึกหัดบทที่ 4

1. จงวาดภาพตามคำสั่งต่อไปนี้

```
TO FAN
  FORWARD 50
  LEFT 90
  FORWARD 20
  LEFT 120
  FORWARD 30
END
```

ใช้กระบวนความ FAN สร้างกระบวนความต่อไปนี้

```
TO WINDMILL
  FAN
  WINDMILL
END
```

ตั้งเต้าวาดภาพด้วยคำสั่ง WINDMILL แล้วพิมพ์ออกทางกระดาษ

2. จงสร้างกระบวนความวาดรูปดาวหรือรูปหลายเหลี่ยม โดยใช้คำสั่งต่อไปนี้

```
TO STAR135
  FORWARD 60
  LEFT 135
  STAR135
END
```

```
TO STAR45
  FORWARD 30
  RIGHT 45
  STAR45
END
```

ตั้งเต้าวาดภาพด้วยคำสั่ง STAR135 และ STAR45 แล้วพิมพ์ออกทางกระดาษ

3. การสั่งให้เต่าวาดรูปวงกลมนอกจากใช้คำสั่ง CIRCLE แล้ว ยังสามารถใช้แนวคิดการสร้างรูปหลายเหลี่ยมที่มีจำนวนด้านมากที่สุด การสร้างรูปเหลี่ยมที่มีจำนวนด้านมากที่สุดโดยกำหนดให้มุมเปลี่ยนไปที่ละน้อย

3.1 จงสร้างกระบวนการวาดวงกลมจากกระบวนการของรูปหลายเหลี่ยมนี้

```
TO CIRCLES  
  FORWARD 10  
  RIGHT 20  
  CIRCLES  
END
```

3.2 ลองปรับกระบวนการข้างต้นเพื่อให้วงกลมเรียบขึ้น