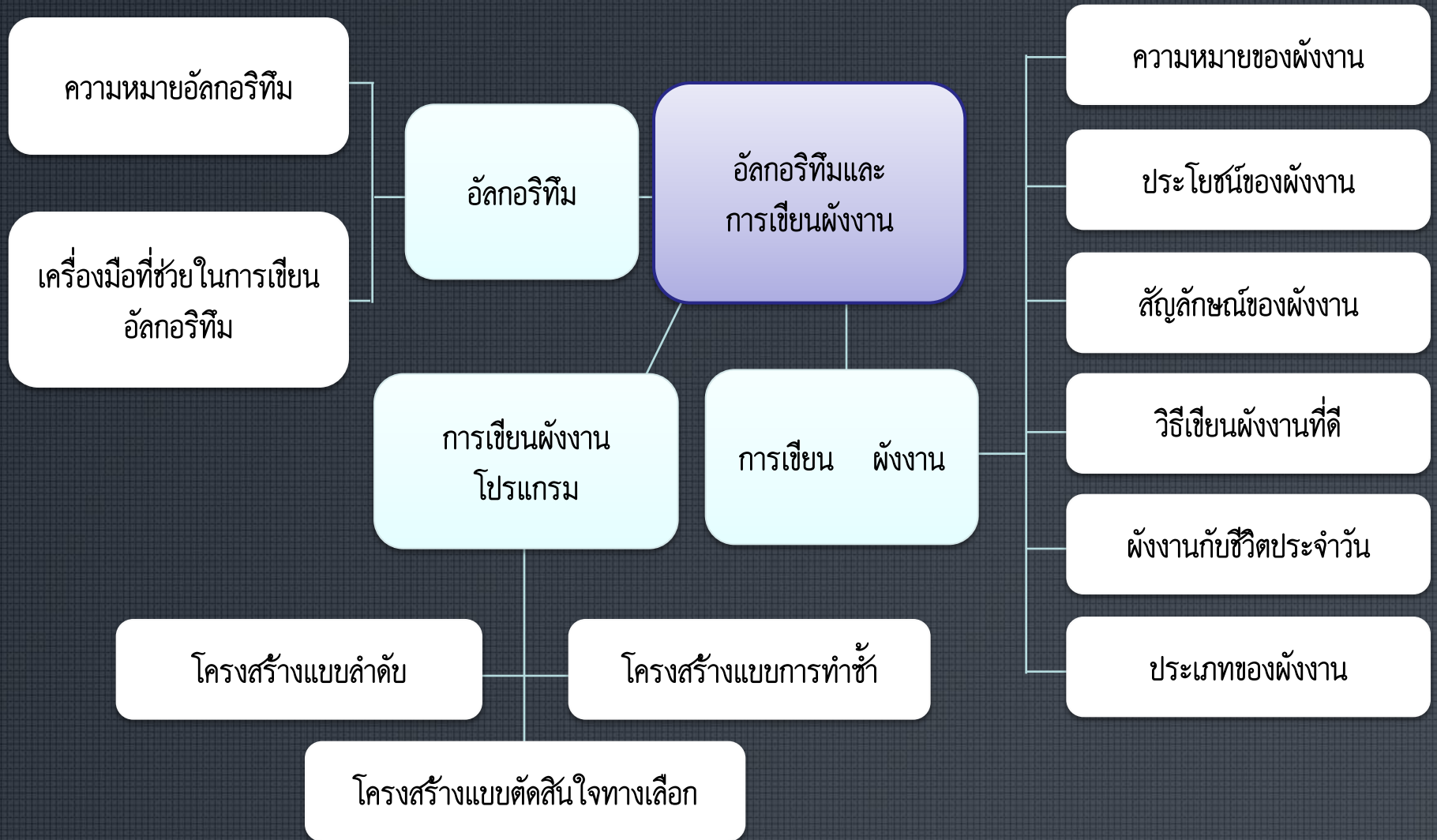


บทที่ 3 ขั้นตอนวิธีการแก้ไข้ปัญหา (Algorithm) และการเขียนผังงาน (Flow Chart)

สาระการเรียนรู้

- ความหมายของอัลกอริทึม
- เครื่องมือที่ช่วยในการเขียนอัลกอริทึม
- ความหมายของผังงาน
- ประโยชน์ของผังงาน
- วิธีการเขียนผังงานที่ดี
- สัญลักษณ์ของผังงาน
- ผังงานกับชีวิตประจำวัน
- การเขียนผังงานโปรแกรม

แผนผังความคิด (Mind Mapping) ของหน่วยการเรียนรู้



แนวคิดการออกแบบอัลกอริทึม

หลังจากที่ได้ผ่านการวิเคราะห์งานมาแล้ว ขั้นตอนต่อไปก็คือการออกแบบอัลกอริทึมหรือขั้นตอนวิธีการแก้ไขปัญหา ซึ่งผังงานเป็นเครื่องมือหนึ่งที่ใช้แทนอัลกอริทึม การเขียนผังงานโดยทั่วไปมีโครงสร้างอยู่ 3 แบบ คือ โครงสร้างแบบลำดับ โครงสร้างแบบเลือกกระทำตามเงื่อนไขหรือแบบตัดสินใจทางเลือก และโครงสร้างแบบทำซ้ำ ผังงานเป็นเครื่องมือที่ช่วยผู้เขียนโปรแกรมสามารถนำไปเขียนโปรแกรมได้ง่าย และสามารถตรวจสอบข้อผิดพลาดและตรวจสอบความถูกต้องได้ง่าย

ความหมายของอัลกอริทึม (Algorithm)

มีผู้ให้ความหมายของอัลกอริทึมหรือขั้นตอนวิธีการแก้ไข้ปัญหา ดังนี้

- อัลกอริทึม คือ กลุ่มของขั้นตอนหรือกฎเกณฑ์ที่จะนำไปสู่การแก้ไข้ปัญหา
- อัลกอริทึม คือ ขั้นตอนวิธีที่ประกอบด้วยชุดคำสั่งเป็นขั้นเป็นตอนที่ชัดเจนและรับประกันว่าเมื่อได้ปฏิบัติตามขั้นตอนจนครบก็จะได้ผลลัพธ์ที่ถูกต้องตามต้องการ
- อัลกอริทึม คือ รูปแบบของการกำหนดการทำงานอย่างเป็นขั้นตอน ซึ่งผ่านการวิเคราะห์และแยกแยะเพื่อการแก้ไข้ปัญหาต่าง ๆ ตามลำดับขั้น อาจเลือกใช้ภาษาไทยหรือภาษาอังกฤษตามความถนัดเพื่อนำเสนอขั้นตอนของกิจกรรมก็ได้

สรุปอัลกอริทึม หมายถึง วิธีการทางคอมพิวเตอร์ ซึ่งมีลำดับขั้นตอนการทำงานเพื่อทำให้ได้ผลลัพธ์ถูกต้องตามความต้องการ

เครื่องมือที่ช่วยในการเขียนอัลกอริทึม

ผังงานและรหัสเทียม เป็นตัวแทนของอัลกอริทึม ที่นิยมใช้ในปัจจุบัน

ความแตกต่างระหว่าง Algorithm กับ Pseudo Code ก็คือการแสดงความคิดที่ได้จากการจินตนาการถึงขั้นตอน ซึ่งขั้นตอนที่อยู่ในความคิดก็คือ Algorithm ที่ผ่านการแยกและจัดลำดับแล้ว เมื่อนำเสนอก็คอาจใช้ภาษาง่าย ๆ แต่หากนำเสนอด้วยการเขียนเป็นภาษาที่สื่อให้ทุกคนเข้าใจตรงกันได้ง่ายก็คือ Pseudo Code นั่นเอง สำหรับหนังสือหลายเล่มแสดง Algorithm ด้วย Pseudo Code ก็ยังเรียกว่า Algorithm ได้เช่นกัน

ผังงาน (Flowchart)

คือ แผนภาพที่มีการใช้สัญลักษณ์รูปภาพและคำอธิบายสั้น ๆ และใช้ลูกศรที่แสดงถึงขั้นตอนการทำงานของโปรแกรมหรือระบบที่ละขั้นตอนรวมไปถึงทิศทางการไหลของข้อมูลตั้งแต่แรกจนได้ผลลัพธ์ตามที่ต้องการ การแสดงขั้นตอนวิธีการที่ใช้สัญลักษณ์ที่เข้าใจได้ง่าย แต่จะให้รายละเอียดได้น้อยกว่า

ประโยชน์ของผังงาน

- ช่วยลำดับขั้นตอนการทำงานของโปรแกรม สามารถนำไปเขียนโปรแกรมได้โดยไม่สับสน
- ช่วยในการตรวจสอบ และแก้ไขโปรแกรมได้ง่าย เมื่อเกิดข้อผิดพลาด
- ช่วยให้การดัดแปลง แก้ไข ทำได้อย่างสะดวกและรวดเร็ว
- ช่วยให้ผู้อื่นสามารถศึกษาการทำงานของโปรแกรมได้อย่างง่าย และรวดเร็วมากขึ้น

สัญลักษณ์ของผังงาน (Flow Chart Symbol)

การเขียนผังงานโปรแกรมจะประกอบไปด้วยการใช้สัญลักษณ์มาตรฐานต่าง ๆ ที่เรียกว่าสัญลักษณ์ ANSI (American National Standards Institute) ในการสร้างผังงาน ดังตัวอย่างที่แสดงในรูปต่อไปนี้

แสดงสัญลักษณ์ที่ใช้ในการเขียนผังงานโปรแกรม

สัญลักษณ์ (Symbol)

คำอธิบาย



Start / End : จุดเริ่มต้น / สิ้นสุดของโปรแกรม



ลูกศรแสดงทิศทางการทำงานของโปรแกรมและการไหล (Flow) ของข้อมูล



ใช้แสดงขั้นตอนในการประมวลผล (Process) หรือการกำหนดค่าข้อมูลให้กับตัวแปร (Assignment)

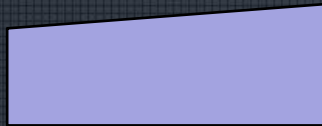


Input / Output : การรับค่า (Input) หรือการแสดงผลลัพธ์ (Output) ที่ได้จากการประมวลผล ใช้ในกรณีที่ยังไม่ระบุชนิดของอินพุตหรือเอาต์พุต

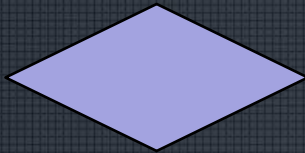
แสดงสัญลักษณ์ที่ใช้ในการเขียนผังงานโปรแกรม (ต่อ)

สัญลักษณ์ (Symbol)

คำอธิบาย



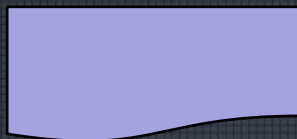
การรับค่า จากอุปกรณ์อินพุต (Manual Input) เช่น การป้อนค่าผ่านทางคีย์บอร์ด



การตรวจสอบเงื่อนไขเพื่อตัดสินใจ (Decision) โดยจะมีเส้นทางออกจากรูปเพื่อแสดงทิศทางการทำงานตามเงื่อนไข เป็นจริงหรือเป็นเท็จ



แสดงผลหรือรายงานผลออกทางจอภาพ (Display) เป็นการรายงานผลข้อมูลที่ยังมีการเปลี่ยนแปลงได้ตลอดเวลา



แสดงผลหรือรายงานผลออกทางเครื่องพิมพ์ (Printer) ให้อยู่ในรูปแบบของเอกสาร เป็นการรายงานผลข้อมูลที่ไม่มีการเปลี่ยนแปลงอีกแล้ว

แสดงสัญลักษณ์ที่ใช้ในการเขียนผังงานโปรแกรม (ต่อ)

สัญลักษณ์ (Symbol)

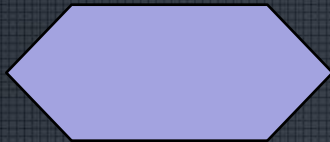
คำอธิบาย



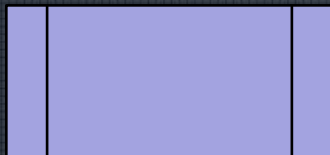
แสดงจุดเชื่อมต่อของผังงานภายในหรือที่อยู่ในหน้าเดียวกัน หรือเป็นที่บรรจบของเส้นหลายเส้นที่มาจากหลายทิศทาง เพื่อจะไปสู่การทำงานอย่างใดอย่างหนึ่งที่เหมือนกัน



แสดงจุดเชื่อมต่อของผังงานภายนอกหรือที่อยู่คนละหน้า



การเตรียมการไว้ล่วงหน้าเพื่อการทำงานซ้ำ เช่น คำสั่ง For



การเรียกใช้โปรแกรมย่อย หรือการเรียกใช้ฟังก์ชันที่กำหนดขึ้นเอง

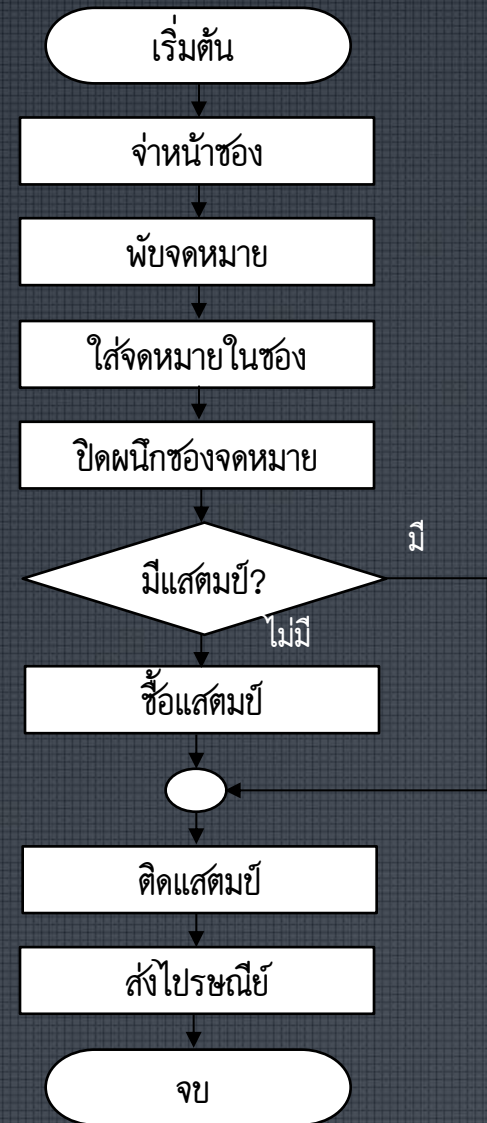
วิธีการเขียนผังงานที่ดี

- ใช้สัญลักษณ์ตามที่กำหนดไว้
- ใช้ลูกศรแสดงทิศทางการไหลของข้อมูลจากบนลงล่าง หรือจากซ้ายไปขวา
- คำอธิบายในภาพควรสั้นกะทัดรัด และเข้าใจง่าย
- ทุกแผนภาพต้องมีลูกศรแสดงทิศทางเข้า-ออก
- ไม่ควรโยงเส้นเชื่อมผังงานที่อยู่ไกลมาก ๆ ควรใช้สัญลักษณ์จุดเชื่อมต่อแทน
- ผังงานควรมีการทดสอบความถูกต้องของการทำงานก่อนนำไปเขียนโปรแกรม

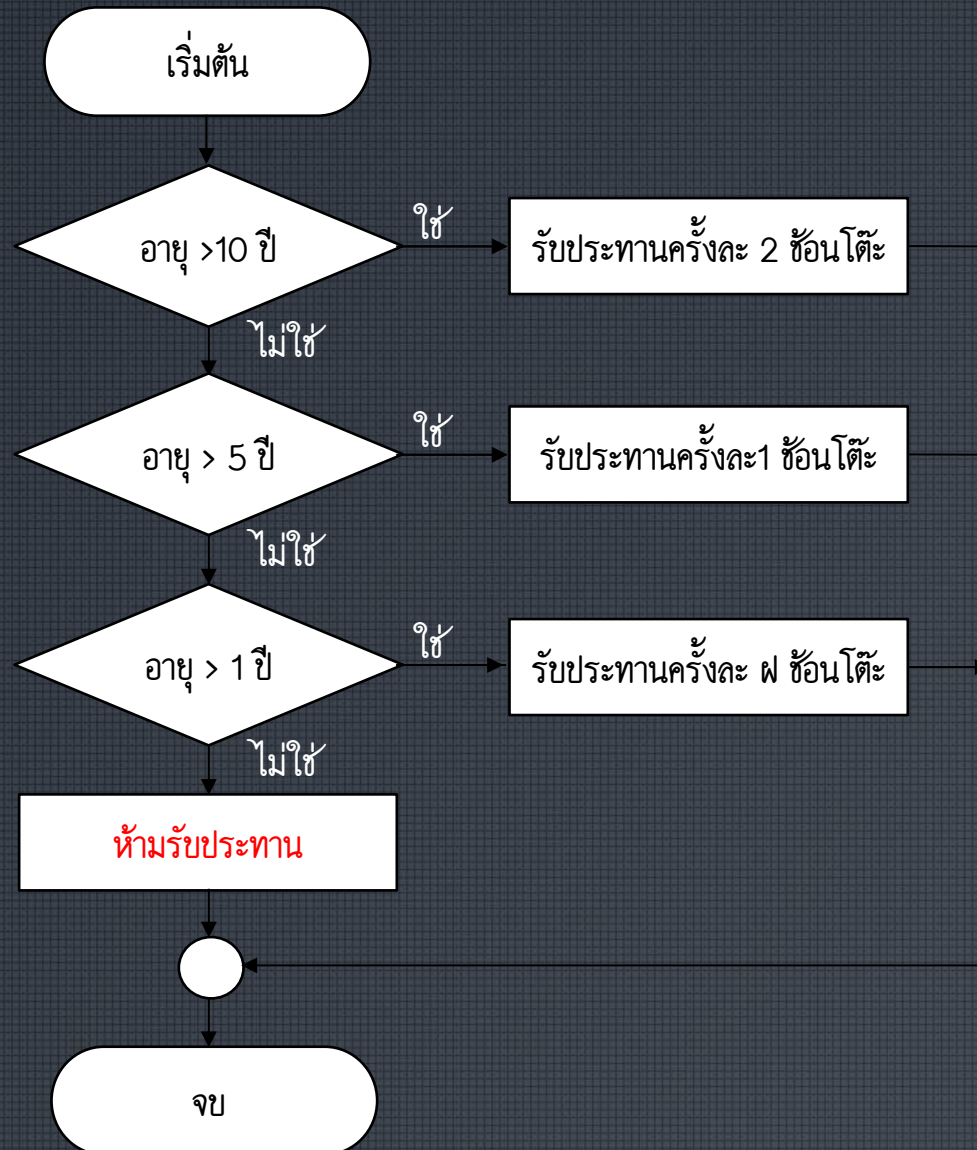
ผังงานกับชีวิตประจำวัน

การทำงานหลายอย่างในชีวิตประจำวัน จะมีลักษณะที่เป็นลำดับขั้นตอนซึ่งก่อนที่ท่านจะได้ศึกษาวิธีการเขียนผังงานโปรแกรมจะแนะนำให้ท่านลองฝึกเขียนผังงานที่แสดงการทำงานในชีวิตประจำวันวันก่อน เพื่อเป็นการสร้างความคุ้นเคยกับสัญลักษณ์รูปภาพต่าง ๆ ที่จะมีใช้ในผังงานโปรแกรมต่อไป

ตัวอย่างที่ 1 เขียนผังงานที่แสดงขั้นตอนการสั่งจดหมาย



ตัวอย่างที่ 2 เขียนผังงานแสดงวิธีการรักษาพยาบาล



ประเภทของผังงาน

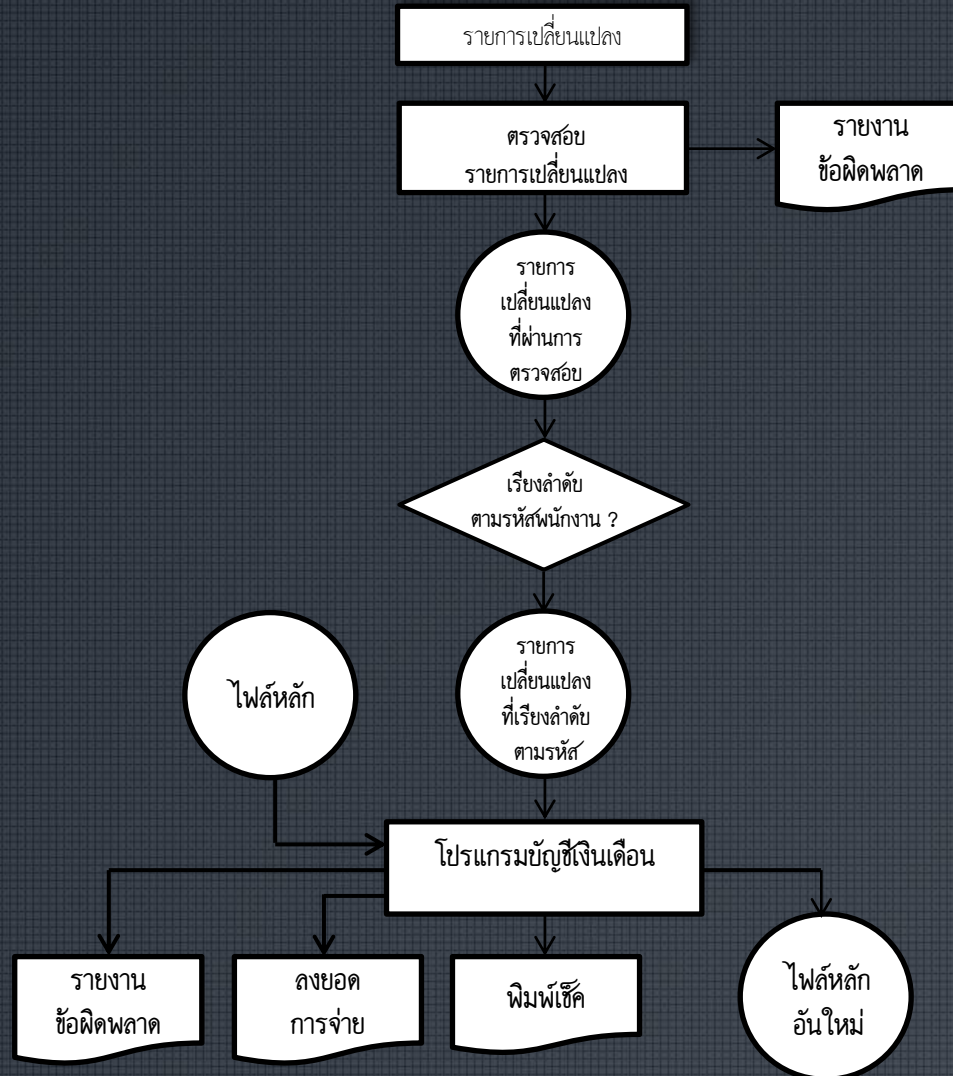
ผังงานแยกออกเป็น 2 ประเภท

1. ผังงานระบบ (System Flowchart)
2. ผังงานโปรแกรม (Program Flowchart)

ผังงานระบบ (System Flowchart)

เป็นผังงานซึ่งแสดงขอบเขตและลำดับขั้นตอนการทำงานของระบบหนึ่ง ๆ รวมทั้งแสดงรูปแบบของข้อมูลเข้าและข้อมูลออกว่าถูกรับเข้าหรือแสดงผลโดยผ่านสื่อประเภทใด เนื่องจากผังงานระบบเป็นแผนภาพที่แสดงถึงระบบโดยรวม ซึ่งรวมถึงการทำงานที่เกี่ยวข้องกับคน วัสดุ อุปกรณ์หรือเครื่องจักร

ตัวอย่างผังงานระบบ



ผังงานโปรแกรม (Program Flowchart)

เป็นผังงานซึ่งแสดงลำดับขั้นตอนของโปรแกรมหนึ่ง ๆ โดยใช้ภาพสัญลักษณ์และคำประกอบ ในการสื่อความหมายลำดับขั้นตอนการทำงาน เป็นเครื่องมือช่วยให้การเขียนและตรวจสอบโปรแกรมทำได้โดยสะดวก รวดเร็วยิ่งขึ้น ดังนั้นผังงานโปรแกรมจึงเป็นเพียงส่วนหนึ่งในผังงานระบบเท่านั้น

การเขียนผังงานโปรแกรม

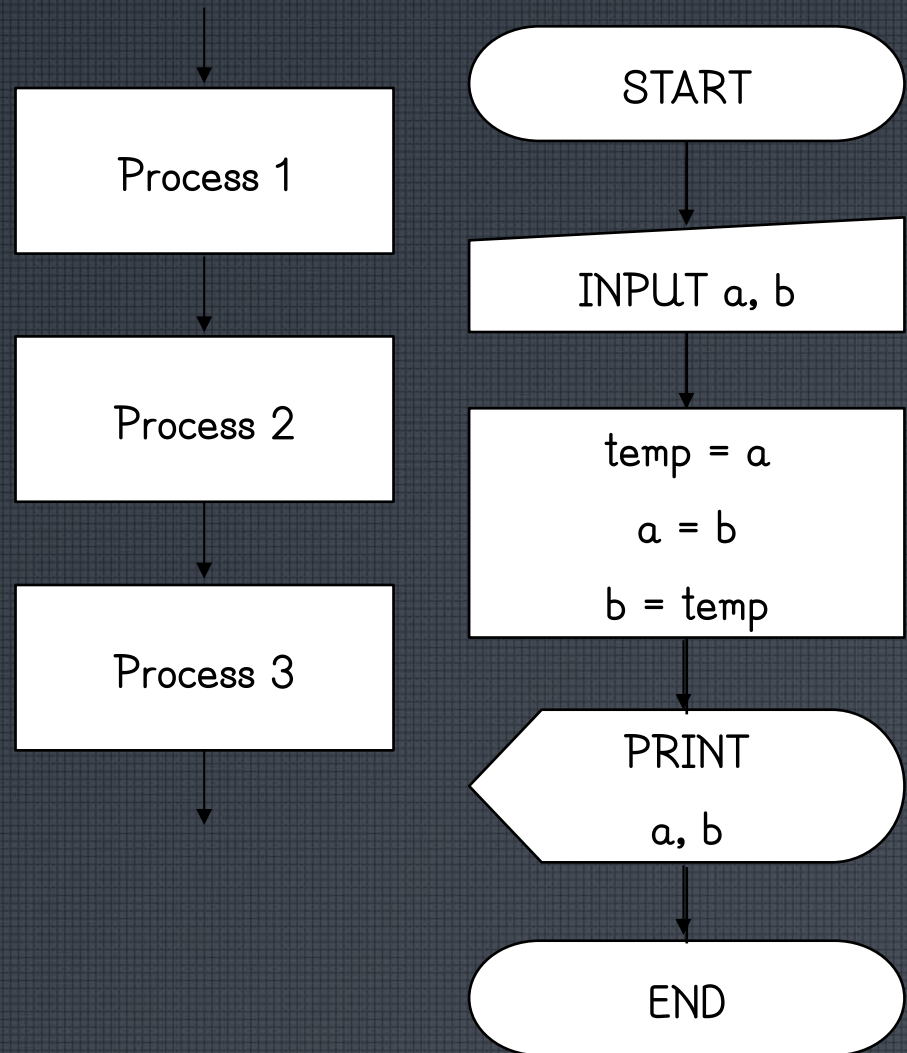
การเขียนผังงานโปรแกรม สามารถแบ่งได้ 3 โครงสร้างหลัก ๆ คือ โครงสร้างแบบลำดับ (Sequence) โครงสร้างแบบการเลือกกระทำตามเงื่อนไข (Decision) และโครงสร้างแบบการทำซ้ำ (Repeation) ทั้ง 3 แบบ จะมี ส่วนประกอบขั้นตอนภายในโครงสร้างที่เหมือนกัน

ประกอบด้วย ส่วนในการรับข้อมูลเข้า (Input) การประมวลผลข้อมูล (Process) และการรายงานผลข้อมูล (Output) ดังนั้นในการเขียนผังงานโปรแกรมจึงต้องประกอบด้วยทั้ง 3 ส่วนนี้เสมอ



โครงสร้างแบบลำดับ (Sequence Structure)

เป็นโครงสร้างพื้นฐานที่สุด มีขั้นตอนเรียงลำดับจากบนลงล่าง ลักษณะการไหล (Flow) แบบทางเดียว จึงมีชื่อเรียกอีกอย่างว่า โครงสร้างแบบทางเดียว (Single Pass)

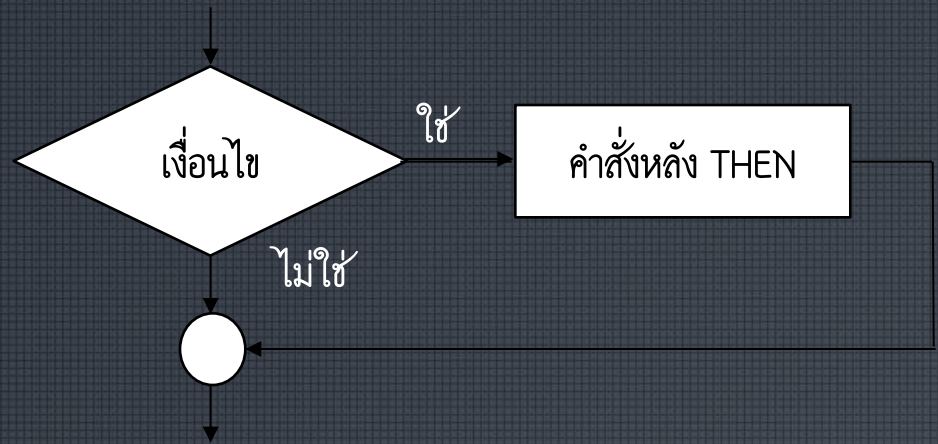


ตัวอย่างผังงานโปรแกรมสลับค่าข้อมูลระหว่างตัวแปร a และ b

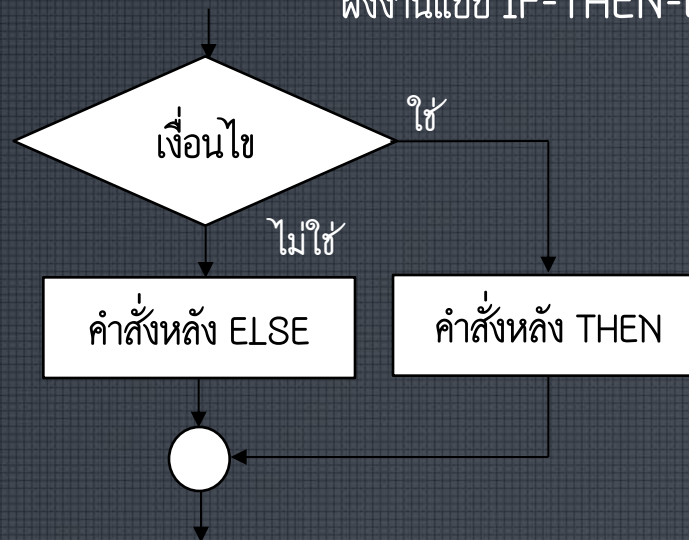
โครงสร้างแบบตัดสินใจทางเลือกตามเงื่อนไข (Decision Structure)

เป็นโครงสร้างที่ใช้การตรวจสอบเงื่อนไขเพื่อการทำงานอย่างใดอย่างหนึ่งมีโครงสร้างพื้นฐานอยู่ด้วยกัน 2 รูปแบบ คือ โครงสร้างแบบ IF - THEN และ โครงสร้างแบบ IF - THEN - ELSE

ผังงานแบบ IF-THEN



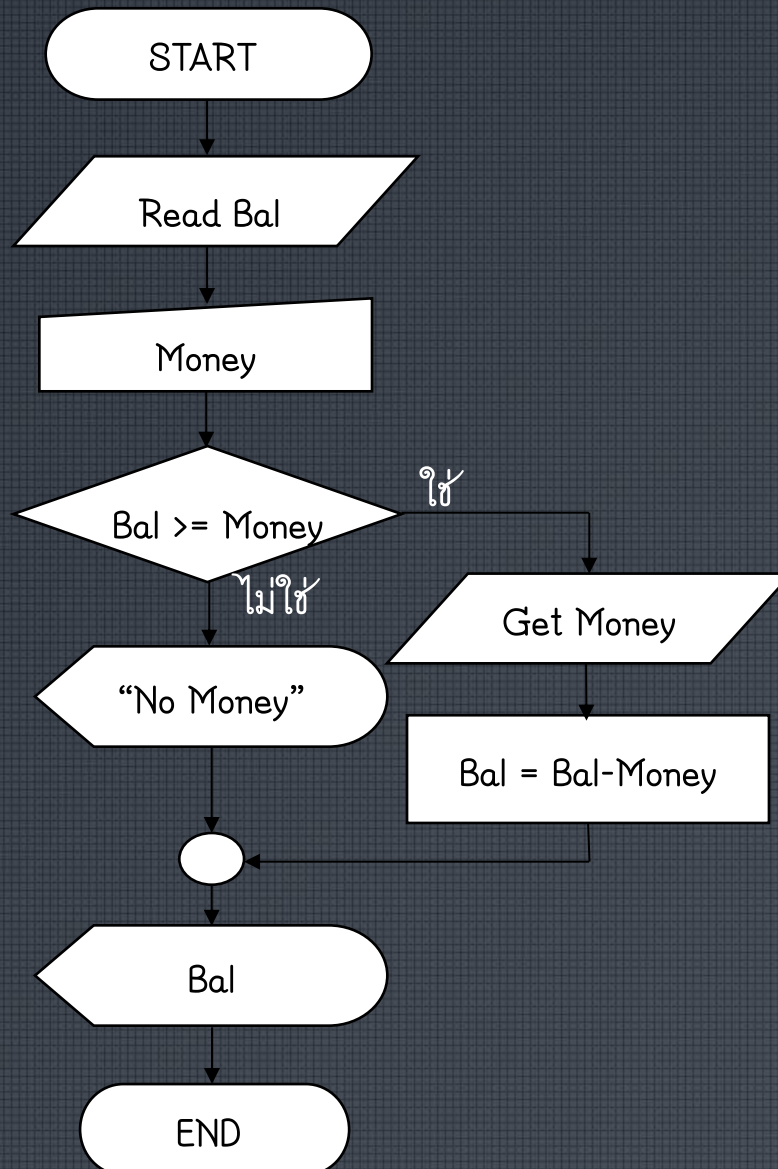
ผังงานแบบ IF-THEN-ELSE



ตัวอย่าง

การเขียนผังงานอ่านค่าข้อมูลเงินในบัญชี (Bal) และรับจำนวนเงิน (Money) ที่จะเบิก โดยช้อนจำนวนเงินผ่านทางแป้นพิมพ์ มีเงื่อนไขดังนี้

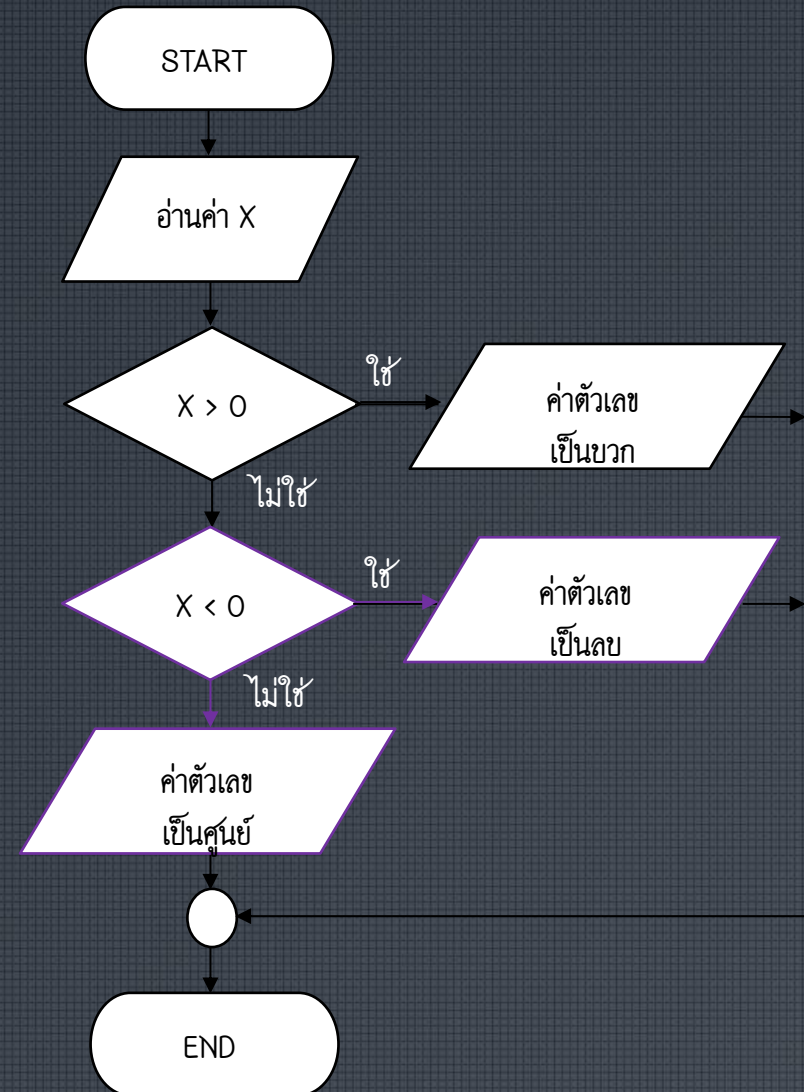
- ถ้า Bal มากกว่าหรือเท่ากับ Money ให้เบิกเงินได้ แล้วตัดยอดเงินที่เหลือในบัญชี (Bal)
- ถ้า Bal น้อยกว่า Money ให้แจ้งว่าไม่มีเงิน (No Money) แล้วแจ้งยอดเงินที่เหลือในบัญชี



ตัวอย่าง การเขียนผังงานเปรียบเทียบค่าข้อมูลที่เก็บอยู่ในตัวแปร x

โดยมีเงื่อนไขดังนี้

- ถ้า $X > 0$ ให้พิมพ์คำว่า " ค่าตัวเลขเป็นบวก "
- ถ้า $X < 0$ ให้พิมพ์คำว่า " ค่าตัวเลขเป็นลบ "
- ถ้า $X = 0$ ให้พิมพ์คำว่า " ค่าตัวเลขเป็นศูนย์ "



โครงสร้างแบบทำซ้ำ (Repeation Structure)

เป็นโครงสร้างที่มีการประมวลผลกลุ่มคำสั่งซ้ำหลายครั้งตามลักษณะเงื่อนไขที่กำหนด อาจเรียก การทำงานซ้ำแบบนี้ได้อีกแบบว่าการวนลูป (Looping)
โครงสร้างแบบการทำงานซ้ำนี้จะมีอยู่

4 ประเภท คือ

- DO-WHILE
- WHILE-LOOP or WHILE-DO or WHILE
- DO-UNTIL or REPEAT-UNTIL
- UNTIL-LOOP or UNTIL-DO

DO-WHILE

เป็นโครงสร้างที่มีการเข้ามาทำงานในกลุ่มคำสั่งที่ต้องทำซ้ำซึ่งเรียกว่า การเข้าลูปก่อน หลังจากนั้นจึงจะทดสอบเงื่อนไขถ้าเงื่อนไขเป็นจริงก็จะย้อนกลับไปทำงานในกลุ่มคำสั่งที่ต้องทำซ้ำอีกแล้วทดสอบเงื่อนไขยังคงเป็นจริงอยู่ก็ยังคงต้องทำกลุ่มคำสั่งซ้ำหรือเข้าลูปต่อไปอีก จนกระทั่งเงื่อนไขเป็นเท็จ ก็จะออกจากลูปไปทำคำสั่งถัดไปที่อยู่ถัดจาก DO WHILE หรืออาจเป็นการจบการทำงาน

WHILE-LOOP

เป็นโครงสร้างที่มีการทดสอบเงื่อนไขก่อนถ้าเงื่อนไขเป็นจริงก็จะเข้ามาทำงานในกลุ่มคำสั่งที่ต้องทำซ้ำซึ่งเรียกว่าการเข้าลูป หลังจากนั้นก็จะย้อนกลับไปตรวจสอบเงื่อนไขใหม่อีกถ้าเงื่อนไขยังคงเป็นจริงอยู่ก็ยังคงต้องทำกลุ่มคำสั่งซ้ำหรือเข้าลูปต่อไปอีก จนกระทั่งเงื่อนไขเป็นเท็จ ก็จะออกจากลูปไปทำคำสั่งถัดไปที่อยู่ถัดจาก DO WHILE หรืออาจเป็นการจบการทำงาน

เป็นโครงสร้างที่มีการเข้ามาทำงานในกลุ่มคำสั่งที่ต้องทำซ้ำซึ่งเรียกว่าการเข้าลูปก่อน หลังจากนั้นจึงจะทดสอบเงื่อนไขถ้าเงื่อนไขเป็นเท็จก็จะย้อนกลับไปทำงานในกลุ่มคำสั่งที่ต้องทำซ้ำอีกแล้วทดสอบเงื่อนไขถ้าเงื่อนไขยังคงเป็นจริงอยู่ก็ยังคงต้องทำกลุ่มคำสั่งซ้ำหรือเข้าลูปต่อไปอีก จนกระทั่งเงื่อนไขเป็นจริง ก็จะออกจากลูปไปทำคำสั่งถัดไปที่อยู่ถัดจาก DO UNTIL หรืออาจเป็นการจบการทำงาน

UNTIL-LOOP

เป็นโครงสร้างที่มีการทดสอบเงื่อนไขก่อนถ้าเงื่อนไขเป็นเท็จก็จะเข้ามาทำงานในกลุ่มคำสั่งที่ต้องทำซ้ำซึ่งเรียกว่าการเข้าลูป หลังจากนั้นก็จะย้อนกลับไปตรวจสอบเงื่อนไขใหม่อีกถ้าเงื่อนไขยังคงเป็นเท็จอยู่ก็ยังคงต้องทำกลุ่มคำสั่งซ้ำหรือเข้าลูปต่อไปอีก จนกระทั่งเงื่อนไขเป็นจริง ก็จะออกจากลูปไปทำคำสั่งถัดไปที่อยู่ถัดจาก DO UNTIL หรืออาจเป็นการจบการทำงาน

สรุปเปรียบเทียบทั้ง 4 แบบ

- DO-WHILE

ประมวลผลในลูปก่อนตรวจสอบจะวนลูปเมื่อตรวจสอบเงื่อนไขว่าเป็นจริง

- WHILE-LOOP

ตรวจสอบก่อนประมวลผลในลูปจะวนลูปเมื่อตรวจสอบเงื่อนไขว่าเป็นจริง

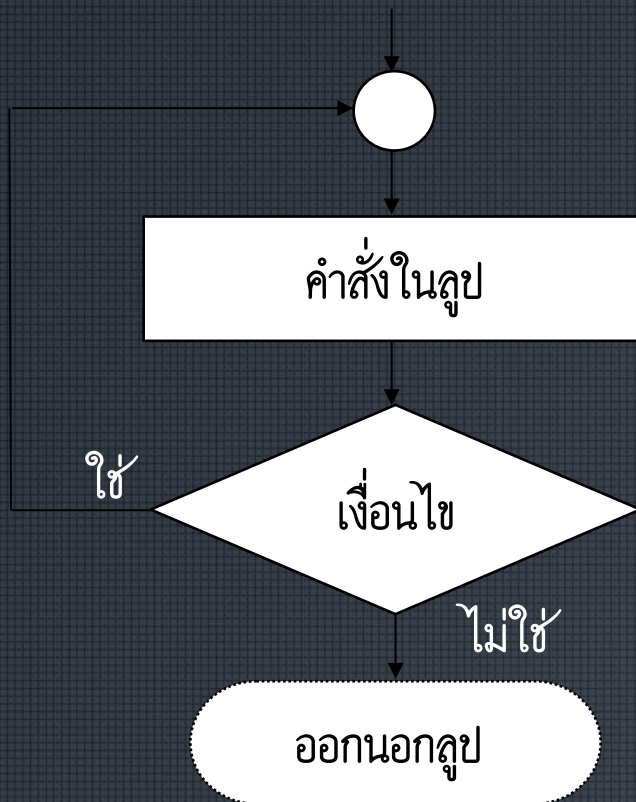
- DO-UNTIL

ประมวลผลในลูปก่อนตรวจสอบจะวนลูปเมื่อตรวจสอบเงื่อนไขเป็นเท็จ

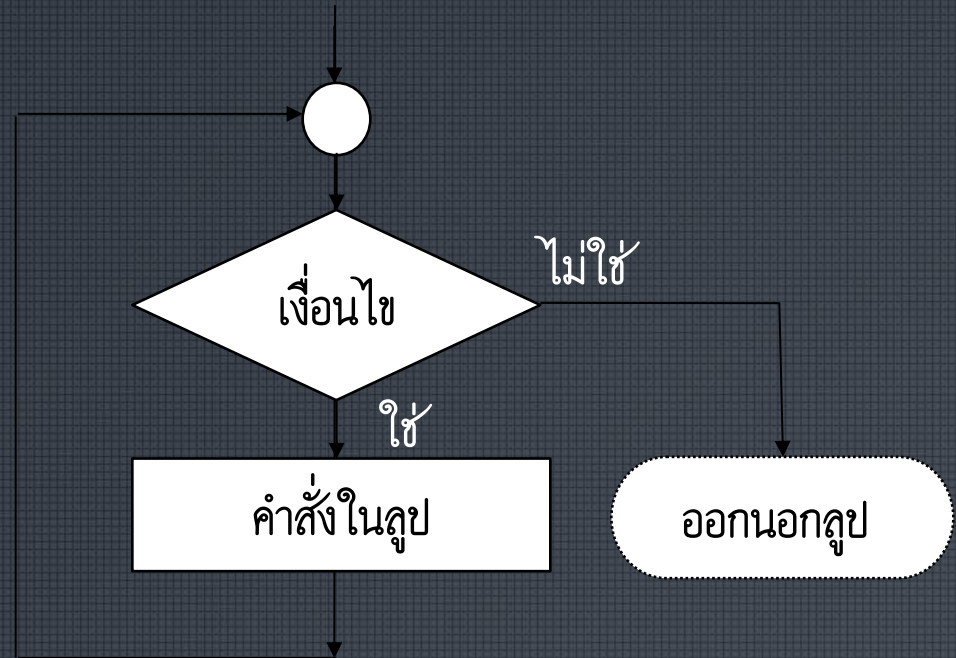
- UNTIL-LOOP

ตรวจสอบก่อนประมวลผลในลูปจะวนลูปเมื่อตรวจสอบเงื่อนไขเป็นเท็จ

โครงสร้างผังงานแบบ DO-WHILE และแบบ WHILE-LOOP

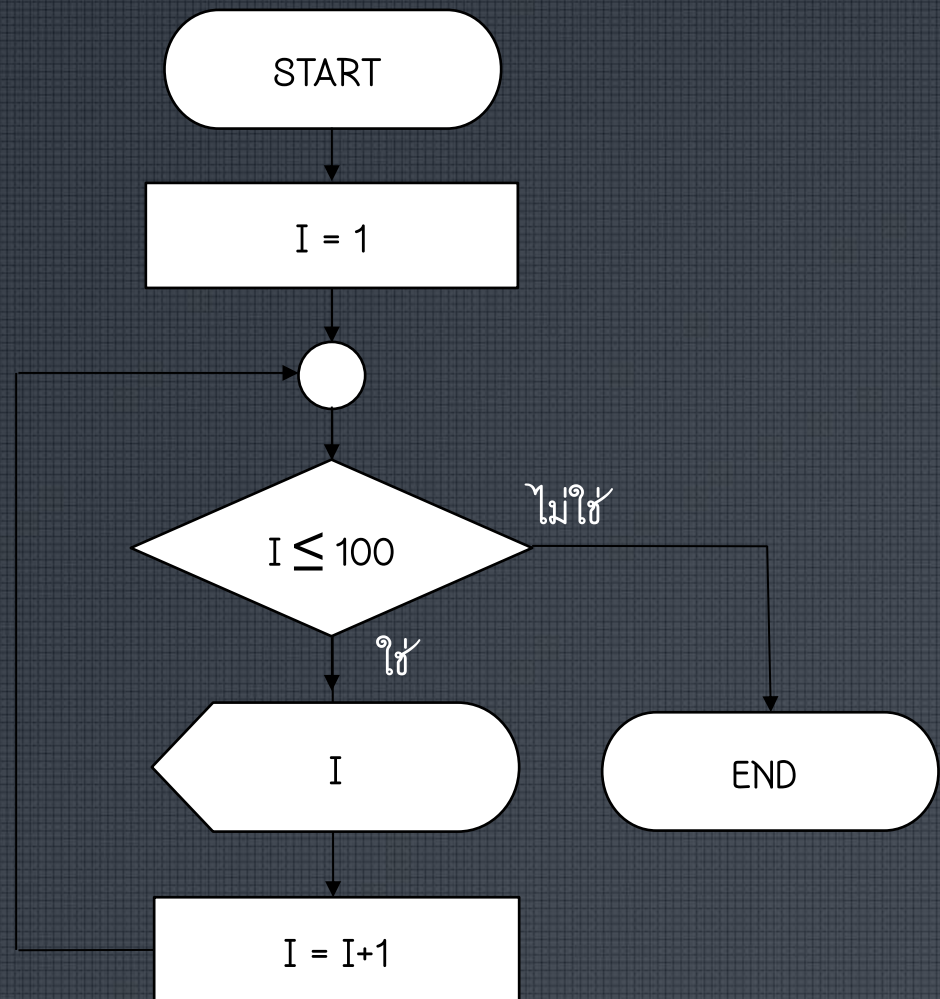
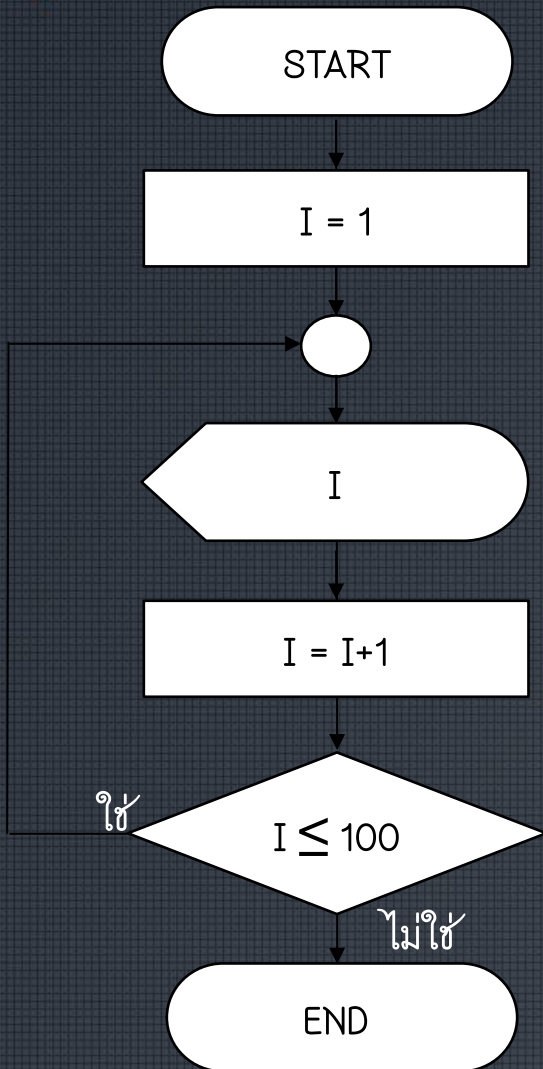


DO-WHILE



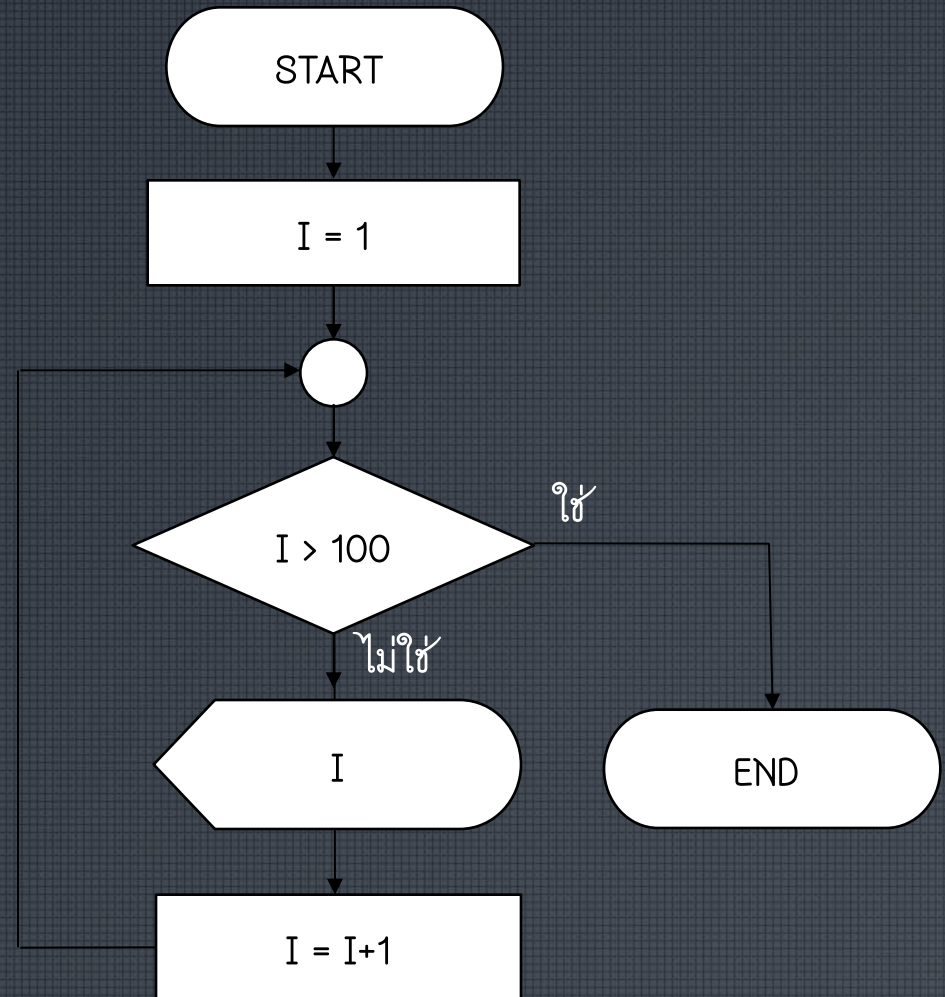
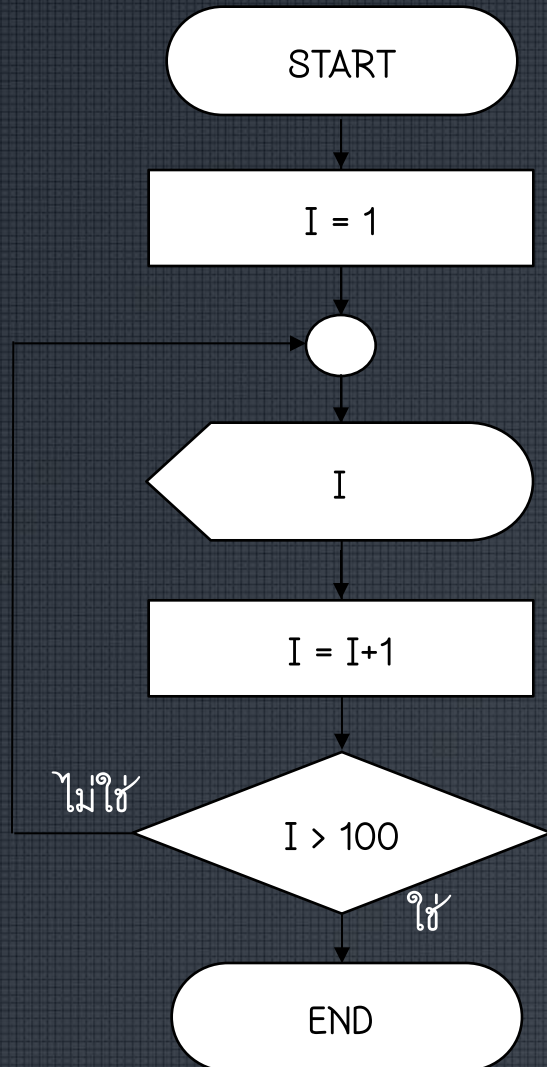
WHILE-LOOP

ตัวอย่าง ฟังก์ชันแสดงการพิมพ์ตัวเลขจาก 1 ถึง 100



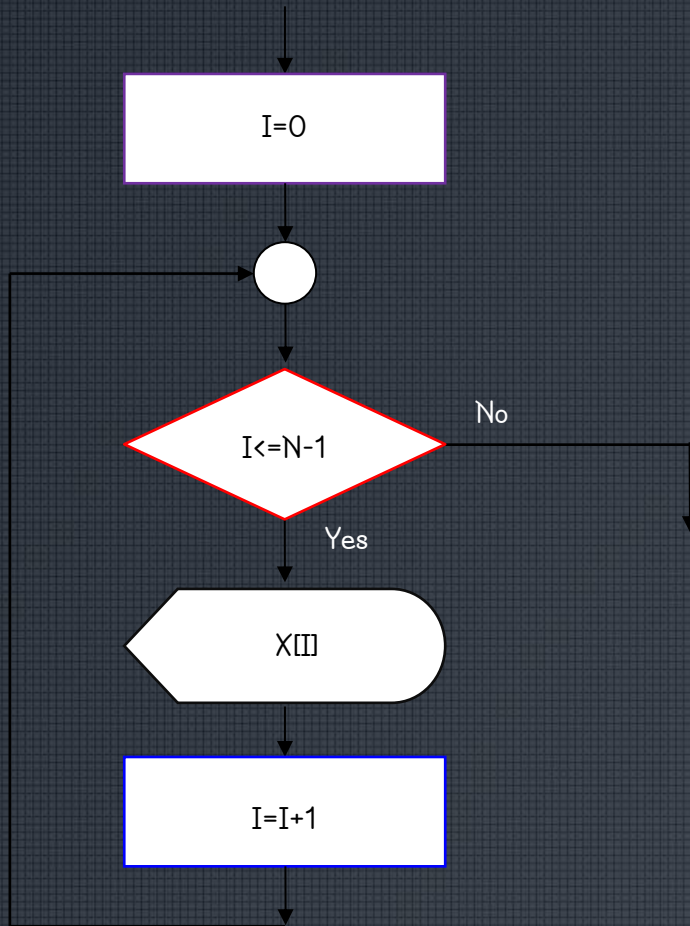
โครงสร้างแบบ DO-WHILE และ WHILE-LOOP

ตัวอย่าง ฟังก์ชันแสดงการพิมพ์ตัวเลขจาก 1 ถึง 100



โครงสร้างแบบ DO-UNTIL และ UNTIL-LOOP

โครงสร้างการทำงานซ้ำที่มีการเตรียมการไว้ล่วงหน้า(คำสั่ง For)



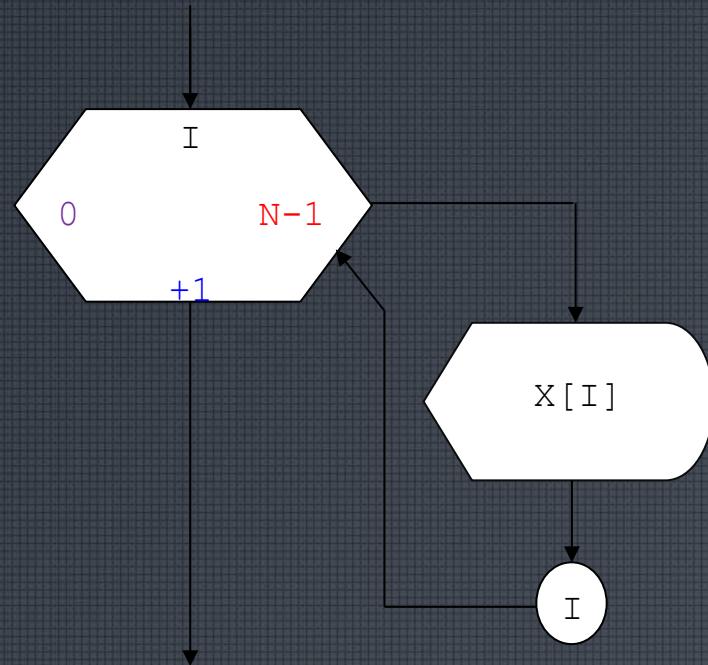
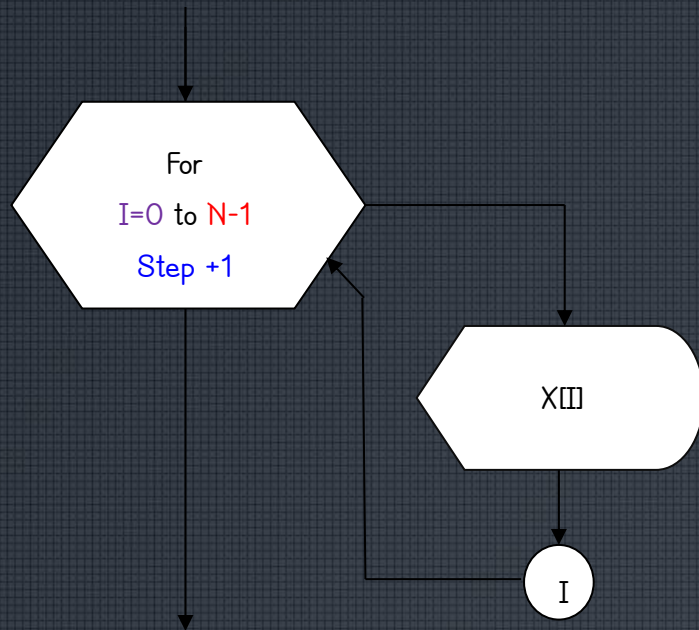
1) กำหนดค่าเริ่มต้น

2) ตรวจสอบเงื่อนไข ถ้าเป็นจริงจะทำงานใน Loop

3) คำสั่งที่ต้องการกระทำใน Loop

4) ค่าเพิ่ม หรือ ค่าลด
(เปลี่ยนแปลงค่าตัวแปร ที่ใช้ตรวจสอบ)

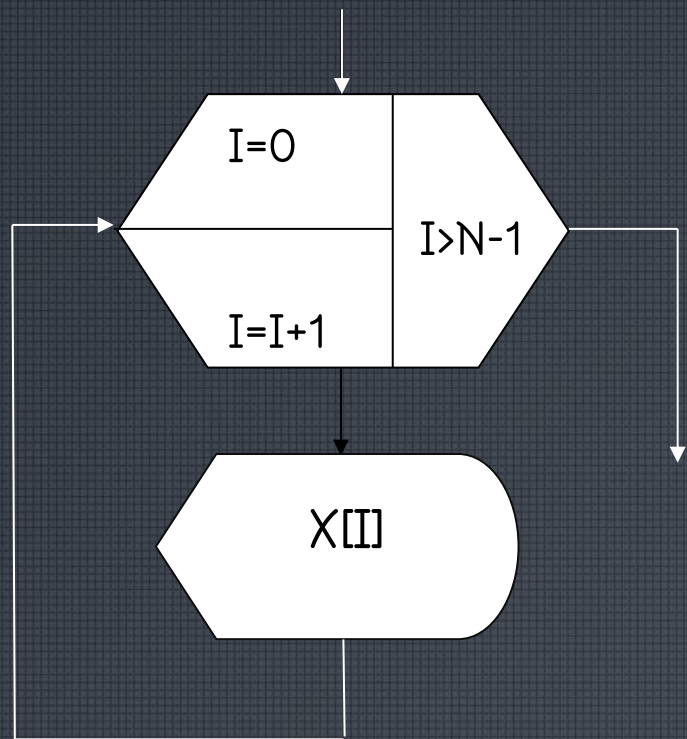
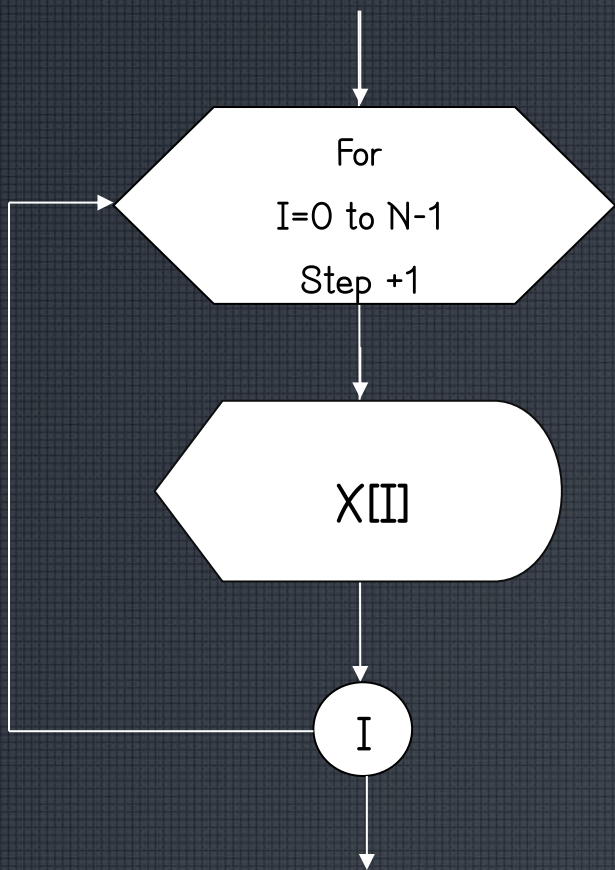
การเขียนผังงานคำสั่ง For



ผังงาน (Flowchart) เป็นตัวแทนของอัลกอริทึม เป็นเครื่องมือในการเขียนอัลกอริทึมซึ่งใช้ภาพและคำอธิบายสั้น ๆ และลูกศรแสดงทิศทางลำดับขั้นตอน ซึ่งมี 3 โครงสร้างหลัก คือ

1. โครงสร้างแบบลำดับ
2. โครงสร้างแบบตัดสินใจทางเลือกตามเงื่อนไข
3. โครงสร้างแบบการทำซ้ำ

การเขียนผังงานคำสั่ง For



สรุป

- โครงสร้างแบบลำดับเป็นโครงสร้างพื้นฐาน ที่มีลำดับการทำงานตั้งแต่ขั้นตอนแรกไปจนถึงขั้นตอนสุดท้าย โดยทำตามลำดับจากบนลงล่าง
- โครงสร้างแบบตัดสินใจทางเลือก มี 2 ลักษณะ แบบแรกเป็นแบบเลือกทำกับไม่ทำ และแบบที่สองคือเลือกทำอย่างใดอย่างหนึ่ง
- โครงสร้างแบบการทำงานซ้ำ มี 2 ลักษณะใหญ่ ๆ คือ ทำในลูปก่อนแล้วตรวจสอบภายหลัง (Post-test) และแบบตรวจสอบก่อนจึงทำในลูป (Pre-test) ซึ่งทั้งสองลักษณะยังสามารถแบ่งได้อีก 2 แบบ คือ จะทำการวนรอบเมื่อเงื่อนไขเป็นจริง ก็จะทำการวนรอบเมื่อเงื่อนไขเป็นเท็จ แต่ไม่ว่าจะใช้วิธีใดก็สามารถทำงานได้ผลลัพธ์เดียวกันได้
นอกจากนี้ยังมีคำสั่งที่มีการเตรียมการไว้ล่วงหน้าในการทำงานซ้ำ ก็คือคำสั่ง For ซึ่งเป็นคำสั่งที่นิยมใช้กันมาก

บทที่ 3

ขั้นตอนวิธีการแก้ไข้ปัญหา (Algorithm) และ การเขียนผังงาน (Flow Chart)