

ใบความรู้หน่วยการเรียนรู้ที่ 2

เรื่อง การเขียนรหัสจำลอง

ขั้นตอนที่สำคัญในการแก้ปัญหาคือ การวางแผน การวางแผนที่ดีจะช่วยให้การแก้ปัญหาเป็นไปได้อย่างง่ายดาย ผู้ที่สามารถวางแผนในการแก้ปัญหาได้นอกจากจะต้องใช้ประสบการณ์ ความรู้ และความมีเหตุผลแล้ว ยังควรรู้จักวางแผนให้เป็นขั้นตอนอย่างเป็นระเบียบด้วย การจำลองความคิดเป็นส่วนหนึ่งในขั้นตอนการวางแผนในการแก้ปัญหา การจำลองความคิดออกมาในลักษณะเป็นข้อความ หรือเป็นผังงานจะช่วยให้สามารถแก้ปัญหาได้ดีโดยเฉพาะปัญหาที่ยุ่งยากซับซ้อน การวางแผนจะเป็นแนวทางในการดำเนินการแก้ปัญหาต่อไป อีกทั้งเป็นการแสดงแบบเพื่อให้ผู้ที่เกี่ยวข้องได้เข้าใจ และสามารถปฏิบัติตามในแนวทางเดียวกัน



การจำลองความคิด

การจำลองความคิดเป็นส่วนหนึ่งในขั้นตอนที่สองของการแก้ปัญหา การจำลองความคิดออกมาใน ลักษณะข้อความ หรือเป็นผังงานจะช่วยให้สามารถแก้ปัญหาได้ดี โดยเฉพาะปัญหาที่ยุ่งยากซับซ้อน

เครื่องมือ

เครื่องมือที่ใช้ในการจำลองความคิด มักจะประกอบขึ้นด้วยเครื่องหมายที่แตกต่างกันหลาย พอสรุป ได้เป็น 2 ลักษณะ ได้แก่

1. ข้อความหรือคำบรรยาย หรือรหัสจำลอง (Pseudo Code)
2. สัญลักษณ์ (Flowchart)

ข้อความหรือคำบรรยาย หรือรหัสจำลอง (Pseudo Code)

เป็นการเขียนเค้าโครงด้วยการบรรยายเป็นภาษาที่มนุษย์ใช้สื่อสารกัน เพื่อให้ทราบถึงขั้นตอนการ ทำงานของโปรแกรมแต่ละตอน ในบางครั้งอาจใช้คำสั่ง ของภาษาที่ใช้เขียนโปรแกรมก็ได้

ตัวอย่าง Pseudo Code แสดงขั้นตอนการไปทำข้อสอบ

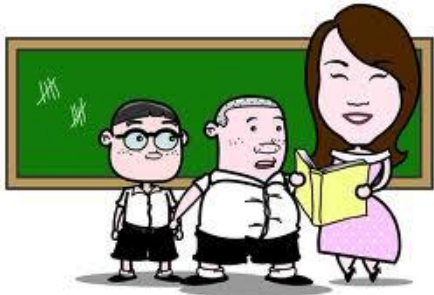
- ☞ อ่านคำชี้แจงของข้อสอบ
- ☞ คิดถึงหน้าอาจารย์ผู้สอน
- ☞ ลงมือทำข้อสอบตั้งแต่ข้อแรกถึงข้อสุดท้าย
- ☞ มองหน้าเพื่อนข้างๆ
- ☞ กลับมามองข้อสอบของตัวเอง แล้วตรวจคำตอบตั้งแต่ข้อแรกถึงข้อสุดท้าย
- ☞ ตัดสินใจส่งข้อสอบให้อาจารย์ผู้คุมสอบ



การเขียนผังงาน (Flowchart)

ผังงาน คือ แผนภาพที่มีการใช้สัญลักษณ์รูปภาพและลูกศรที่แสดงถึงขั้นตอนการทำงานของโปรแกรมหรือระบบทีละขั้นตอน รวมไปถึงทิศทางการไหลของข้อมูลตั้งแต่แรกจนได้ผลลัพธ์ตามที่ต้องการ

ผังงานแบ่งได้ 2 ประเภท



1. ผังงานระบบ (System Flowchart) คือ ผังงานที่แสดงขั้นตอนการทำงานในระบบอย่างกว้าง ๆ แต่ไม่เจาะลงในระบบงานย่อย

2. ผังงานโปรแกรม (Program Flowchart) คือ ผังงานที่แสดงถึงขั้นตอนในการทำงานของโปรแกรม ตั้งแต่รับข้อมูล คำนวณ จนถึงแสดงผลลัพธ์

ประโยชน์ของผังงาน

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


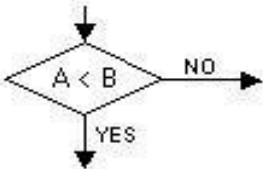



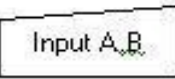



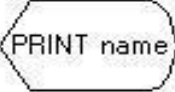

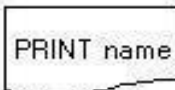
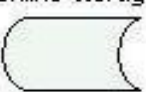

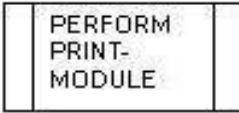



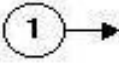


วิธีการเขียนผังงานที่ดี

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



ผังงานโปรแกรม (Program Flowchart)

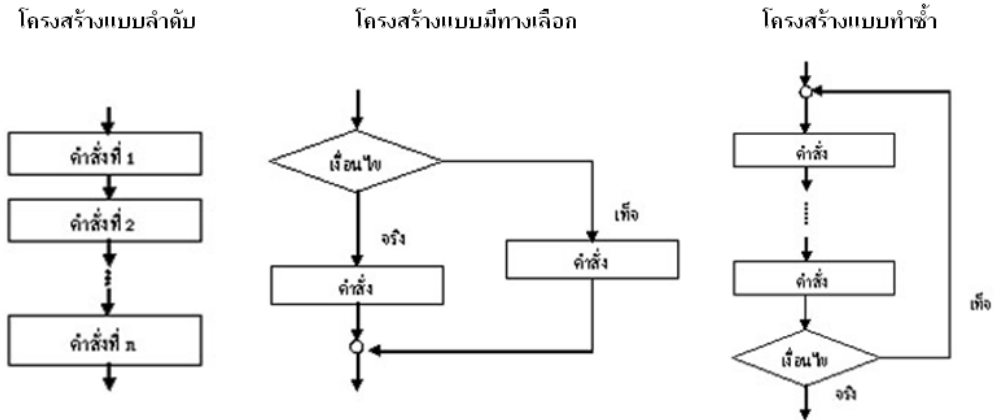
การเขียนผังโปรแกรมจะประกอบไปด้วยการใช้สัญลักษณ์มาตรฐานต่าง ๆ ที่เรียกว่า สัญลักษณ์ ANSI (American National Standards Institute) ในการสร้างผังงาน ดังตัวอย่างที่แสดงในรูปต่อไปนี้

สัญลักษณ์	ความหมาย	ตัวอย่าง
Decision 	กำหนดเงื่อนไข ทางเลือก การเปรียบเทียบทางตรรกศาสตร์ เพื่อการตัดสินใจ	
Terminal, Interrupt 	แสดงจุดเริ่มต้นหรือสิ้นสุดของโปรแกรม	
Manual Input 	การป้อนข้อมูลผ่านทางแป้นพิมพ์	
Magnetic Tape 	การรับหรือแสดงผลของข้อมูลทางเทปแม่เหล็ก	
Display 	แสดงผลจอทางจอภาพ	
Printer 	แสดงผลทางเครื่องพิมพ์	
Online Storage 	แหล่งเก็บข้อมูล Online หน่วยความจำสำรอง	
Predefined Process 	โปรแกรมย่อย หรือ โมดูล เริ่มทำงาน หลังจากจบคำสั่งในโปรแกรมย่อยแล้ว จะกลับมาทำคำสั่งต่อไป	
Preparation 	การเตรียมทำงานลำดับต่อไป	
Connector 	จุดเชื่อมต่อผังงานในหน้าเดียวกัน	
Offpage Connector 	จุดเชื่อมต่อผังงานที่อยู่ต่างหน้ากัน	
Arrowheads 	หัวลูกศร แสดงทิศทางการทำงานของผังงาน	

ลักษณะโครงสร้างของผังงาน

ผังงานทั่วไปจะประกอบด้วยโครงสร้างพื้นฐาน 3 รูปแบบต่อไปนี้คือ

1. โครงสร้างแบบเป็นลำดับ (sequence structure)
2. โครงสร้างแบบมีการเลือก (selection structure)
3. โครงสร้างแบบทำซ้ำ (iteration structure)



1. โครงสร้างแบบลำดับ (Sequential structure) คือ โครงสร้างแสดงขั้นตอนการทำงานที่เป็นไปตามลำดับก่อนหลัง และแต่ละขั้นตอนจะถูกประมวลผลเพียงครั้งเดียวเท่านั้น

ตัวอย่างที่ 1.1 การวางแผนไปโรงเรียน

1. การจำลองความคิดเป็นข้อความ

เริ่มต้น

1. ตื่นนอน
2. อาบน้ำแต่งตัว
3. ไปโรงเรียน

จบ

2. การจำลองความคิดเป็นสัญลักษณ์



ตัวอย่างที่ 1.2 ลำดับงานการพูดทางโทรศัพท์

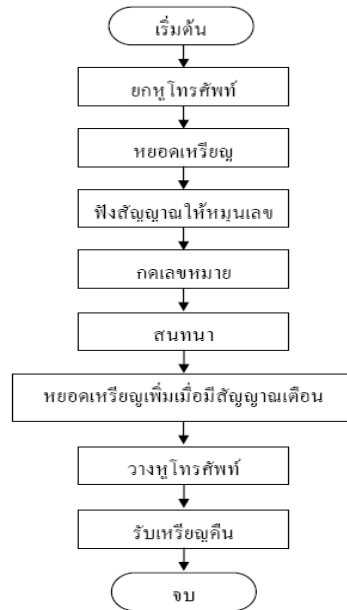
การจำลองความคิดเป็นข้อความ

เริ่มต้น

1. ยกหูโทรศัพท์
2. หยอดเหรียญ
3. ฟังสัญญาณให้หมุนเลข
4. กดเลขหมาย
5. สนทนา
6. หยอดเหรียญเพิ่มเมื่อมีสัญญาณเตือน
7. วางหูโทรศัพท์
8. รับเหรียญคืน

จบ

การจำลองความคิดเป็นผังงาน (ผังงานแบบลำดับ)



2. โครงสร้างแบบเลือก

(Selection structure) คือ โครงสร้างที่มีเงื่อนไข ขั้นตอนการทำงานบางขั้นตอนต้องมีการตัดสินใจเพื่อเลือกวิธีการประมวลผลขั้นต่อไป และจะมีบางขั้นตอนที่จะไม่ได้รับการประมวลผล การตัดสินใจอาจมีทางเลือก 2 ทาง หรือมากกว่าก็ได้ โครงสร้างที่มีทางเลือกเพียง 2 ทาง เราเรียกชื่อว่า โครงสร้างแบบ if...them...else และโครงสร้างที่มีทางเลือกมากกว่า 2 ทาง เราเรียกชื่อว่า โครงสร้างแบบ case

ตัวอย่างที่ 2.1 ลำดับของการทำบัตรประชาชน ของกรมการปกครอง กระทรวงมหาดไทย

การจำลองความคิดเป็นข้อความ

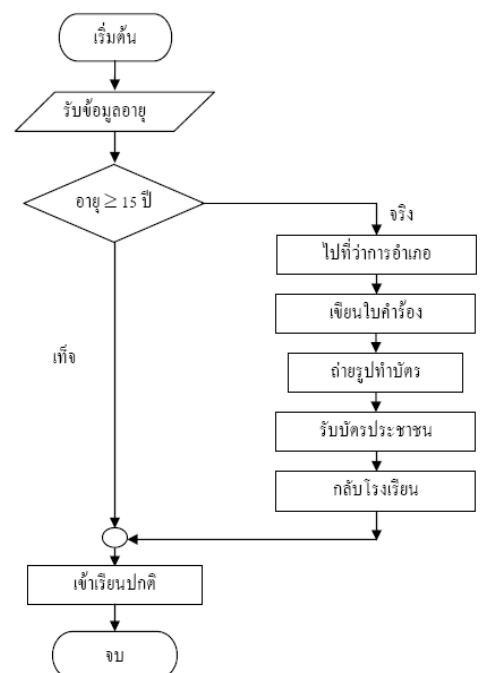
เริ่มต้น

- รับข้อมูลอายุ
- ตรวจสอบอายุเท่ากับหรือมากกว่า 15 ปี
- ถ้าจริง
- ไปที่ว่าการอำเภอ
- เขียนใบคำร้อง
- ถ่ายรูปทำบัตร
- รับบัตรประชาชน
- กลับโรงเรียน
- ถ้าเท็จ
- เข้าเรียนปกติ

จบ

การจำลองความคิดเป็นผังงาน

ผังงานแบบทางเลือก 2 ทาง (IF...THEN)



ตัวอย่างที่ 2.2 ลำดับขั้นตอนการประเมินผลสอบ

การจำลองความคิดเป็นข้อความ

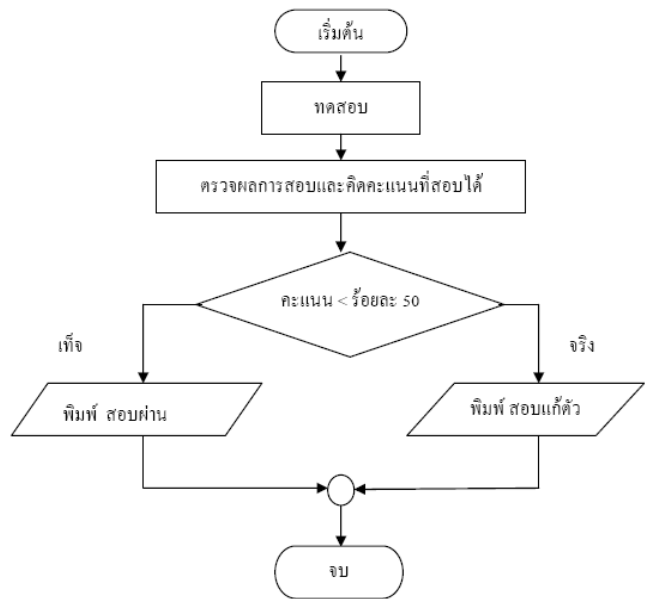
เริ่มต้น

ทดสอบ
 ตรวจสอบผลการสอบและคิดคะแนนที่สอบได้
 คะแนนน้อยกว่าร้อยละ 50 ใช่ไหม
 ถ้าจริง
 พิมพ์ สอบแก้ตัว
 ถ้าไม่จริง
 พิมพ์ สอบผ่าน

จบ

การจำลองความคิดเป็นผังงาน

ผังงานแบบทางเลือก 2 ทาง (IF...THEN...ELSE)



ตัวอย่างที่ 2.3 ลำดับขั้นตอนการประเมินผลการเรียน

การจำลองความคิดเป็นข้อความ

เริ่มต้น

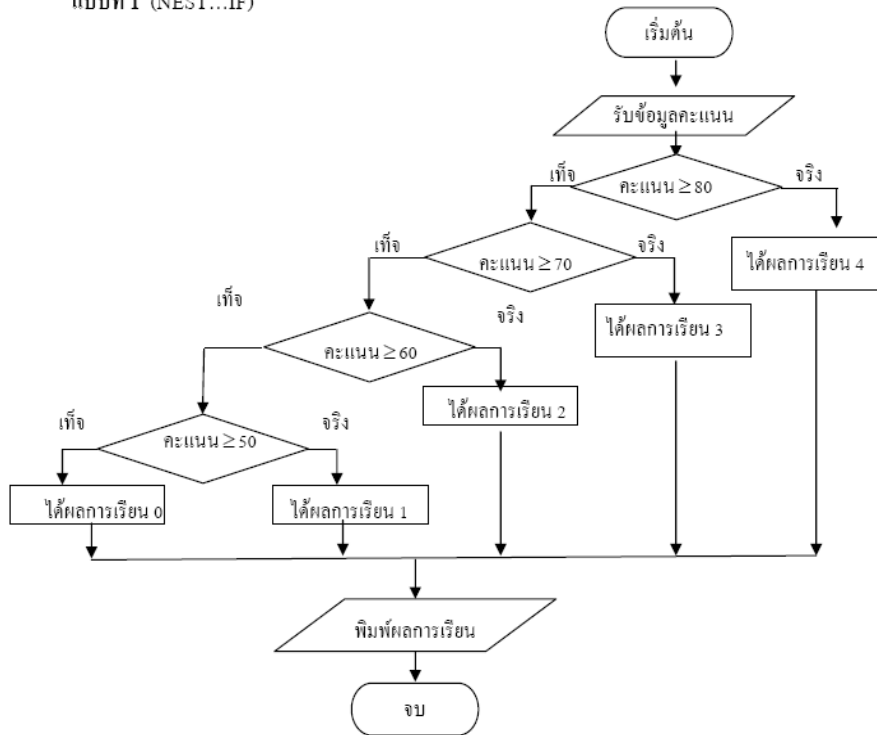
รับข้อมูลคะแนน
 ตรวจสอบคะแนน
 ถ้าคะแนนเท่ากับ 80 ขึ้นไป ได้ระดับผลการเรียน 4
 ถ้าคะแนนเท่ากับ 70-79 ได้ระดับผลการเรียน 3
 ถ้าคะแนนเท่ากับ 60-69 ได้ระดับผลการเรียน 2
 ถ้าคะแนนเท่ากับ 50-59 ได้ระดับผลการเรียน 1
 ถ้าคะแนนน้อยกว่า 50 ได้ระดับผลการเรียน 0
 พิมพ์ระดับผลการเรียน

จบ

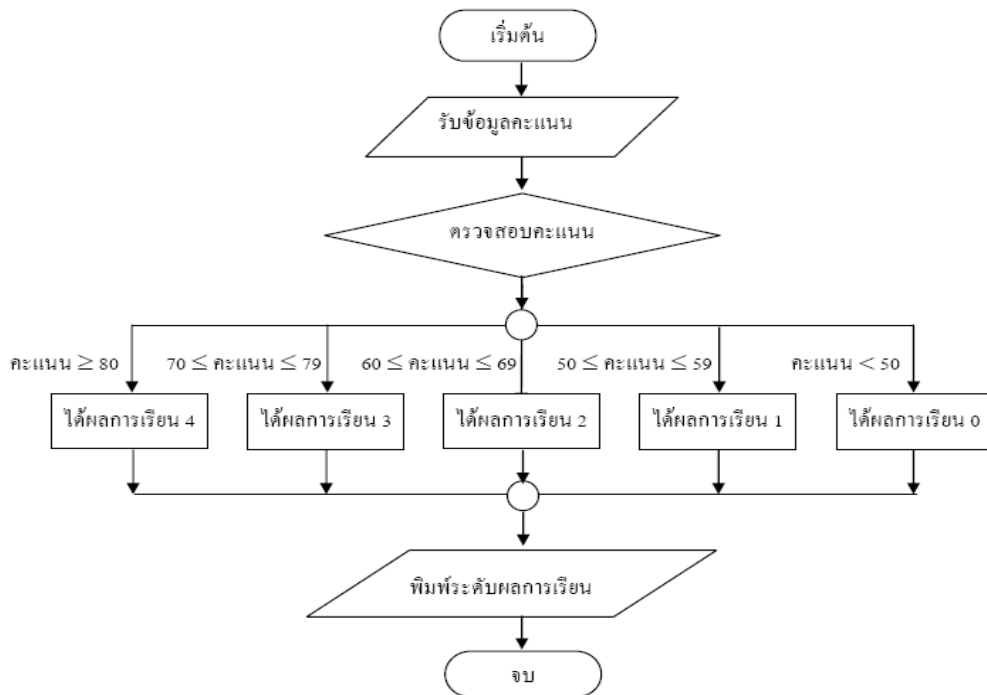
การจำลองความคิดเป็นผังงาน

แบบที่ 1 (NEST...IF)

แบบที่ 1 (NEST...IF)



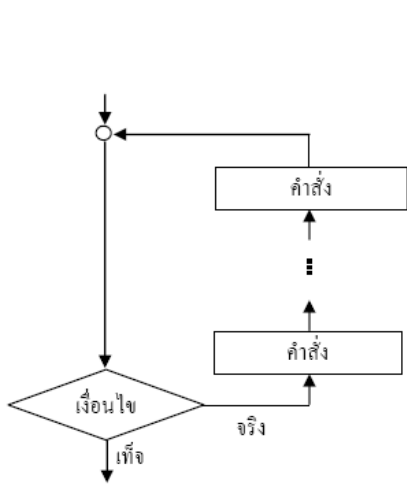
แบบที่ 2 (CASE...OF) กรณีนี้ไม่สามารถใช้กับตัวแปรหรือค่าคงที่ที่เป็นจำนวนจริงได้



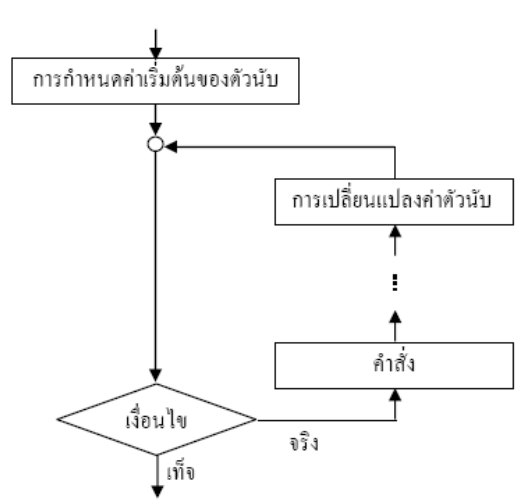
3. โครงสร้างแบบทำซ้ำ (Repetition structure) คือ โครงสร้างที่ขั้นตอนการทำงานบางขั้นตอนได้รับการประมวลผลมากกว่า 1 ครั้ง ทั้งนี้ขึ้นอยู่กับเงื่อนไขบางประการ โครงสร้างแบบทำซ้ำนี้ต้องมีการตัดสินใจในการทำงานซ้ำ และลักษณะการทำงานของ

โครงสร้างแบบนี้มี 2 ลักษณะ ได้แก่

- แบบที่มีการตรวจสอบเงื่อนไขในการทำซ้ำทุกครั้งก่อนดำเนินการกิจการใดๆ ถ้าเงื่อนไขเป็นจริงจะทำงานซ้ำไปเรื่อยๆ และหยุดเมื่อเงื่อนไขเป็นเท็จ เรียกการทำงานลักษณะนี้ว่า การทำซ้ำแบบ do while -

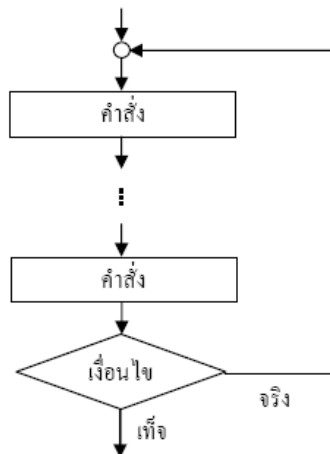


รูปแสดงการทำงานของกรวนซ้ำแบบ while



รูป แสดงการทำงานของกรวนซ้ำแบบ for

- แบบที่ทำกิจกรรมซ้ำเรื่อยๆ จนกระทั่งเงื่อนไขที่กำหนดทั้งหมดเป็นจริงแล้วจึงหยุดการทำงาน โดยแต่ละครั้งที่เสร็จสิ้นการดำเนินการแต่ละรอบจะต้องมีการตรวจสอบเงื่อนไข เรียกการทำซ้ำลักษณะนี้ว่า การทำซ้ำแบบ do until



รูป แสดงการทำงานของกรวนซ้ำแบบ do while

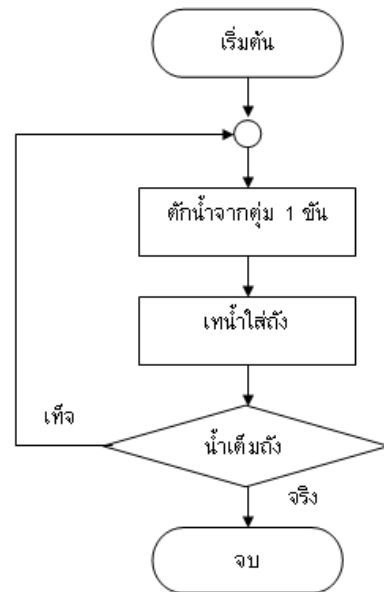
ตัวอย่าง ลำดับขั้นตอนการตักน้ำจากตุ่มครึ่งละ 1 ชั้นใส่ถังน้ำจนเต็ม

• การจำลองความคิดเป็นข้อความ
เริ่มต้น

- ตักน้ำจากตุ่ม 1 ชั้น
- เทน้ำใส่ถัง
- ตรวจสอบน้ำเต็มถึงหรือไม่
- ถ้าไม่เต็มตักน้ำต่ออีก
- ถ้าเต็มหยุดตักน้ำ

จบ

• การจำลองความคิดเป็นผังงาน (แบบทำซ้ำ)



ตัวอย่าง ลำดับขั้นตอนการนับจำนวนตั้งแต่ 1 – 10

การจำลองความคิดเป็นข้อความ

1. เริ่มต้น
2. กำหนดค่า x ให้มีค่าเท่ากับ 1
3. เปรียบเทียบค่า x น้อยกว่า 10 หรือ ไม่
 - ถ้าเป็นจริง (x น้อยกว่า 10)
 - แสดงค่า x
 - กำหนดให้ค่า x เพิ่มค่าขึ้นอีก 1
 - กลับไปทำงานขั้นที่ 3
 - ถ้าเป็นเท็จ (x ไม่น้อยกว่า 10)
 - ไปทำงานขั้นที่ 4
4. จบ

การจำลองความคิดเป็นผังงาน

(แบบตรวจสอบเงื่อนไขก่อนวนซ้ำ)

