

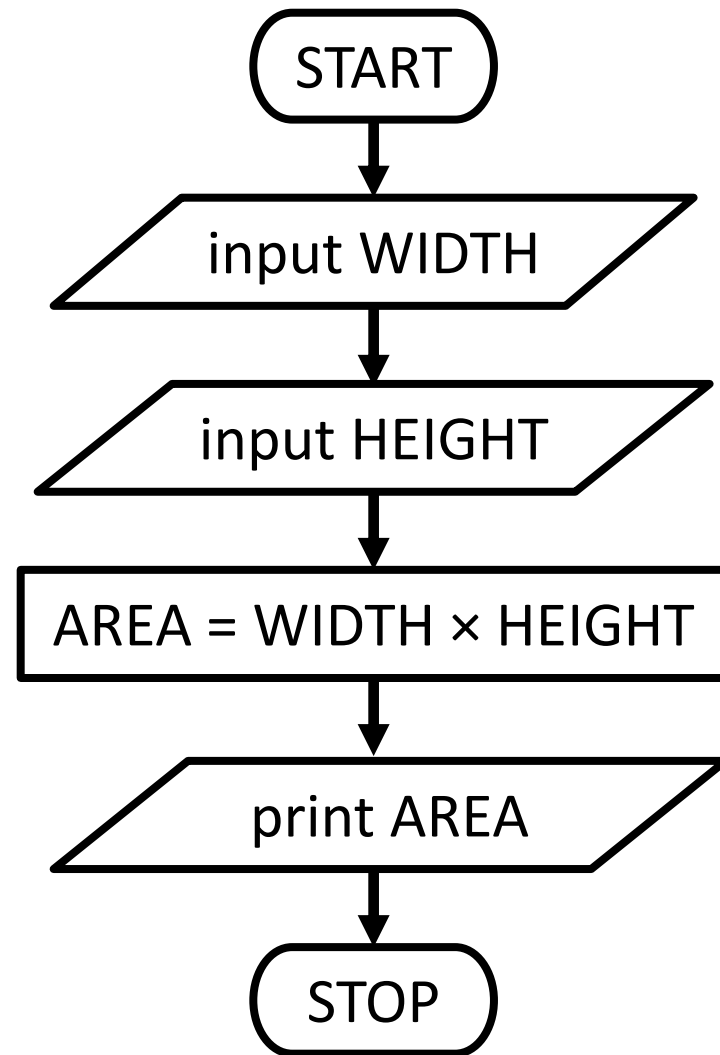
การแสดงขั้นตอนวิธีด้วยผังงานหรือรหัสเทียม

Flowchart

Part 1: Sequential Program

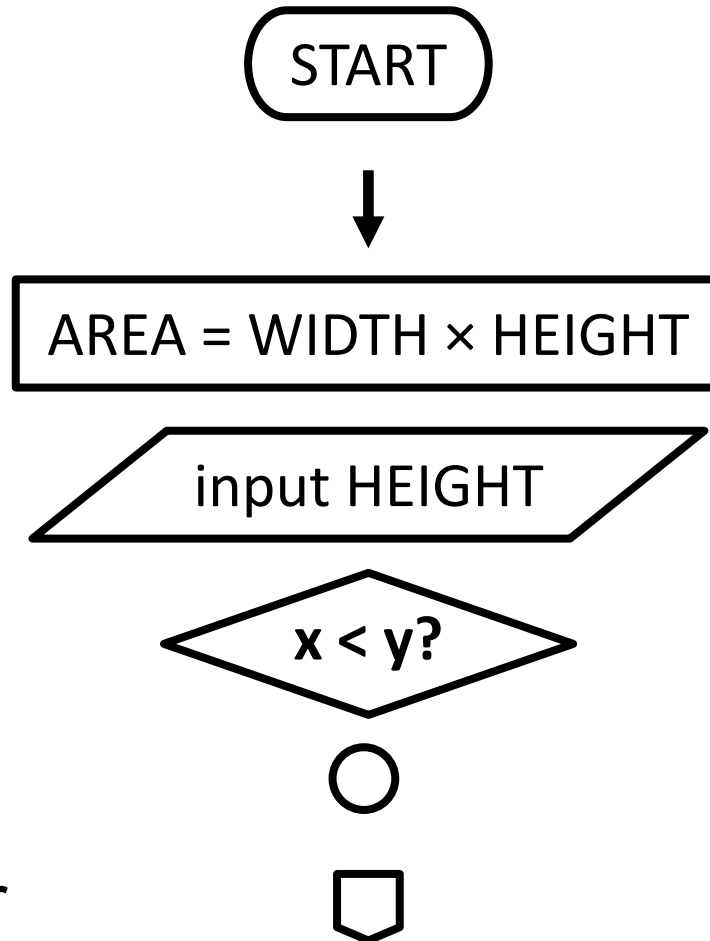
ผังงาน (Flowchart)

- อธิบายด้วยรูปภาพโดยใช้สัญลักษณ์ต่าง ๆ
- มีเส้นแสดงทิศทางการทำงาน
- จะเห็นภาพรวมของโปรแกรมได้ง่าย
- ตัวอย่าง: ผังงานของตัวอย่างปัญหาการหาพื้นที่สี่เหลี่ยมผืนผ้า



สัญลักษณ์ของผังงาน

- Terminator
- Direction
- Process
- Input/Output
- Decision
- Connector
- Offpage Connector



สัญลักษณ์ของผังงาน (ต่อ)

- Annotation



Comments

- สำหรับใส่คอมเมนต์ที่ไม่มีส่วนเกี่ยวข้องกับลำดับวิธีโดยตรง

- Predefined Process / Module



Call Sum
function.

- สำหรับเรียกโปรแกรมย่อยที่เตรียมไว้แล้ว

- Preparation

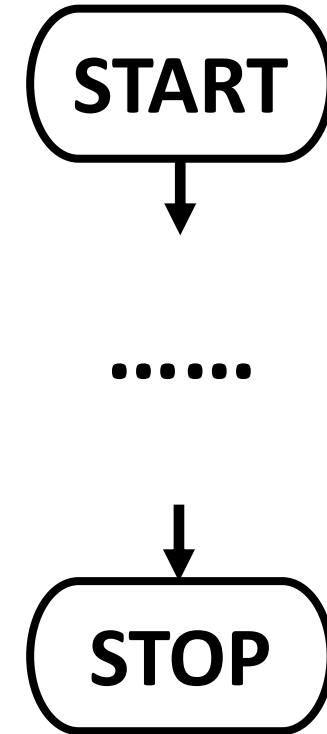


For i in range(10)

- สำหรับเตรียมตัวแปร ใน การทำงานวนซ้ำแบบ for loop

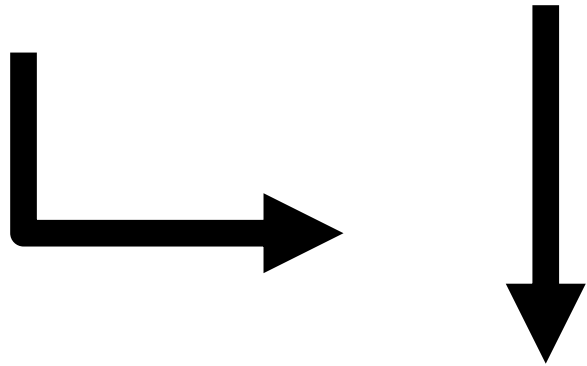
Terminator

- เพื่อบอกให้ทราบถึงจุดเริ่มต้นและสิ้นสุดทำงานของโปรแกรม
 - จุดเริ่มต้น โดยปกติใช้คำว่า START หรือ BEGIN หรือชื่อของโปรแกรม/โปรแกรมน้อย
 - จุดสิ้นสุด โดยปกติใช้คำว่า STOP หรือ END
- ทุกผังงานจะต้องมี Terminator ทั้ง จุดเริ่มต้น และ จุดสิ้นสุด



Direction

- บอกทิศทางการไหลของข้อมูลหรือทิศทางของโปรแกรม
- การทำงานต้องไปตามแนวของลูกศรเท่านั้น ห้ามย้อนกลับ



- เป็นส่วนของการประมวลผลเพื่อคำนวณหรือกำหนดค่าโดยใช้เครื่องหมายดำเนินการให้ค่าคือ เครื่องหมาย “=”
 - จะนำค่าทางด้านขวาของเครื่องหมายเท่ากับ ส่งให้ค่าทางด้านซ้ายเท่านั้น
 - ถ้าด้านขวาของเครื่องหมายเท่ากับ มีการคำนวณ จะต้องทำการคำนวณทางด้านขวาให้เสร็จก่อน

Process (ต่อ)

ตัวอย่างของ Process

AGE = 19

- เป็นการกำหนดค่าให้กับ AGE โดยให้มีค่าเป็น 19

AREA = WIDTH × HEIGHT

- เป็นการคำนวณค่าให้กับตัวแปร AREA โดยนำค่า WIDTH คูณกับค่า HEIGHT ได้ค่าเท่าไรให้นำไปเก็บไว้ในค่า AREA

- ตัวแปร (Variable)

- สิ่งที่ใช้สำหรับเก็บค่าต่าง ๆ เช่น ตัวเลข ตัวอักษร ข้อความ
- สามารถนำไปคำนวณหรือเปรียบเทียบค่าได้
- ชื่อตัวแปรปกติจะใช้ตัวอักษรผสมกับตัวเลข หรือตัวอักษรอย่างเดียวก็ได้ เช่น AB215 และ Salary
- การตั้งชื่อตัวแปร ควรจะเกี่ยวข้องกับการใช้งาน

Process (ต่อ)

- เครื่องหมายดำเนินการ (Operator)

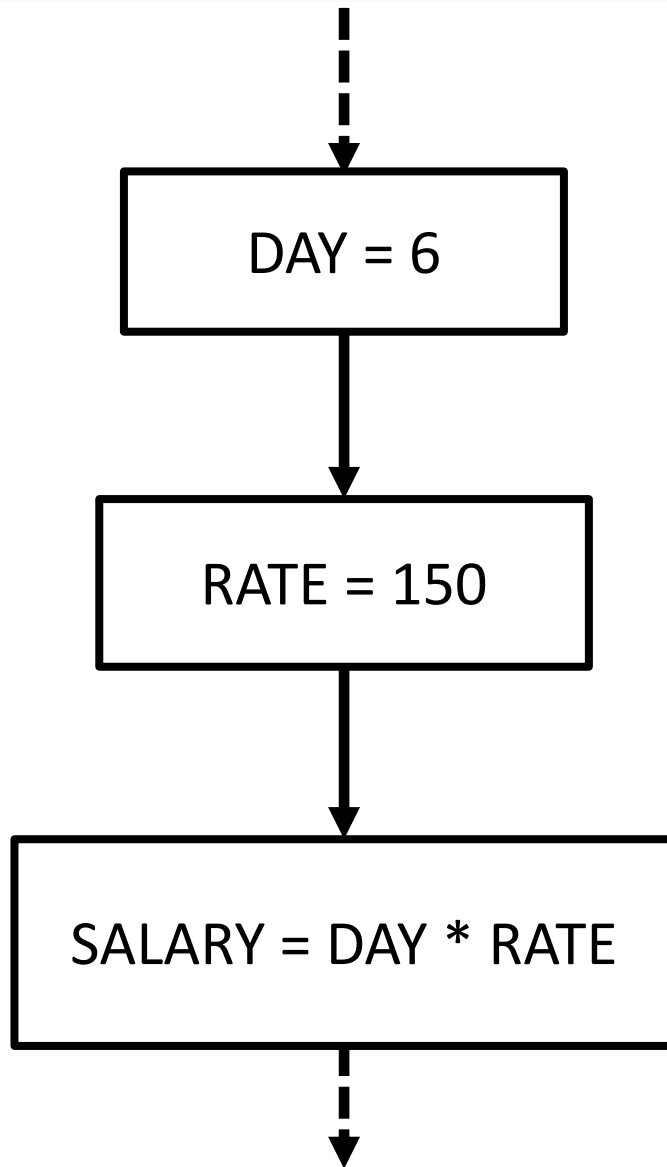
- เครื่องหมายดำเนินการให้ค่า

- =

- เครื่องหมายดำเนินการคำนวณ

- + - * / % (หาเศษจากการหาร)

Process (ต่อ) — ตัวอย่าง



กำหนดค่า DAY เป็น 6

กำหนดค่า RATE เป็น 150

นำค่า DAY คูณกับ RATE ก่อน คือ $6 * 150$ ได้ 900 แล้วนำค่า 900 ไปเก็บไว้ใน SALARY สุดท้าย SALARY มีค่า 900

กำหนดให้

- DAY (จำนวนวันที่ทำงาน) เป็น 6 วัน
- RATE (อัตราค่าแรง) เป็น 150 บาท/วัน

จงคำนวณหา SALARY (ค่าแรงที่ได้) จากสูตร:

$$\text{SALARY} = \text{DAY} * \text{RATE}$$

Input/Output

- เป็นส่วนของรับข้อมูลเข้า หรือ การแสดงผลข้อมูลออก
- การรับข้อมูลเข้า โดยทั่วไปจะพิมพ์จากแป้นพิมพ์
 - ใช้คำสั่งว่า `input` ตามด้วยตัวแปร
- การแสดงผล โดยทั่วไปจะแสดงผลทางจอภาพ
 - ใช้คำสั่งว่า `print` ตามด้วยตัวแปรหรือข้อความ ถ้าเป็นข้อความต้องอยู่ในเครื่องหมาย “ ”

Input/Output (ต่อ) — ตัวอย่าง

กำหนดให้รับค่า

- DAY (จำนวนวันที่ทำงาน)
- RATE (อัตราค่าแรง)

จงคำนวณหา SALARY (ค่าแรงที่ได้) จากสูตร:

$$\text{SALARY} = \text{DAY} * \text{RATE}$$

แล้วให้แสดงค่า SALARY

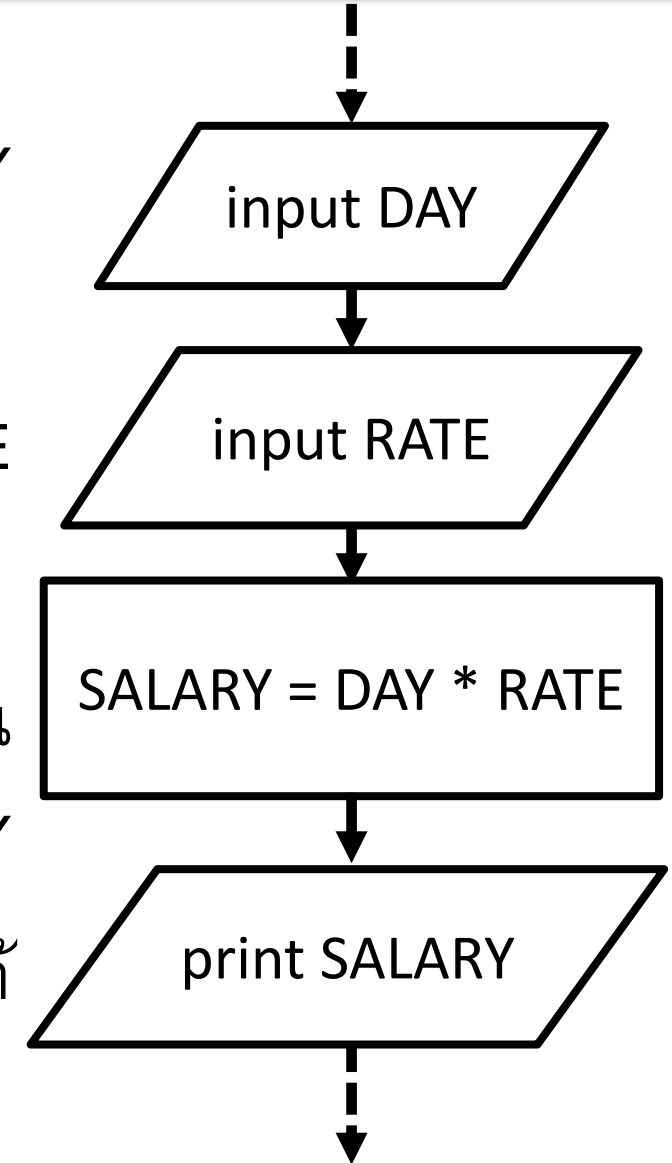
นำค่า DAY คูณกับ RATE แล้วนำไปเก็บไว้ใน

SALARY

แสดงค่า SALARY ที่คำนวณได้

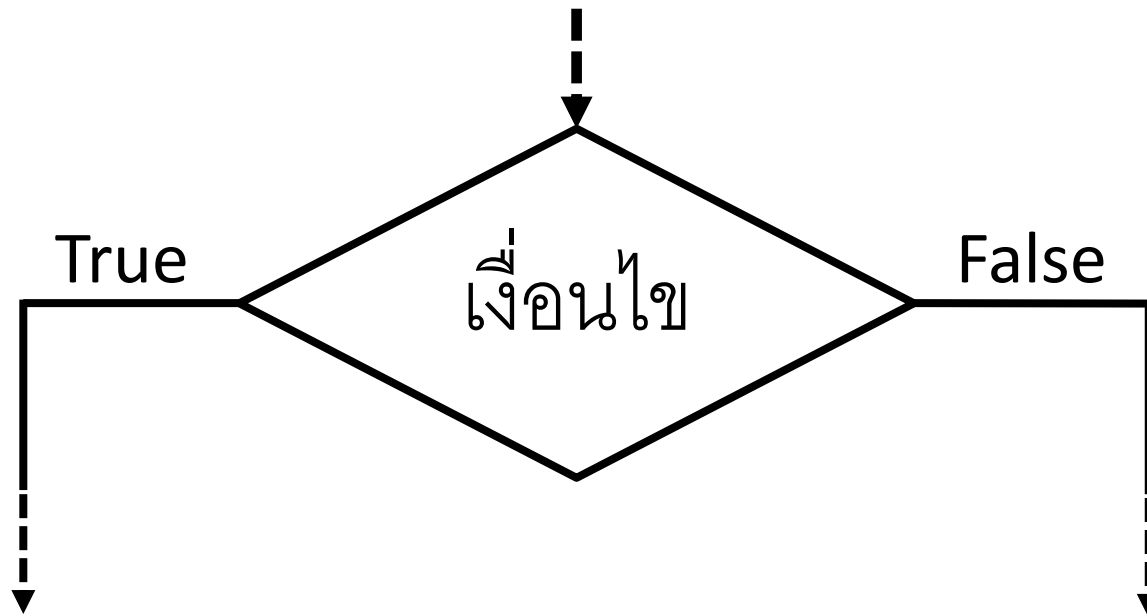
รับค่า DAY

รับค่า RATE



Decision

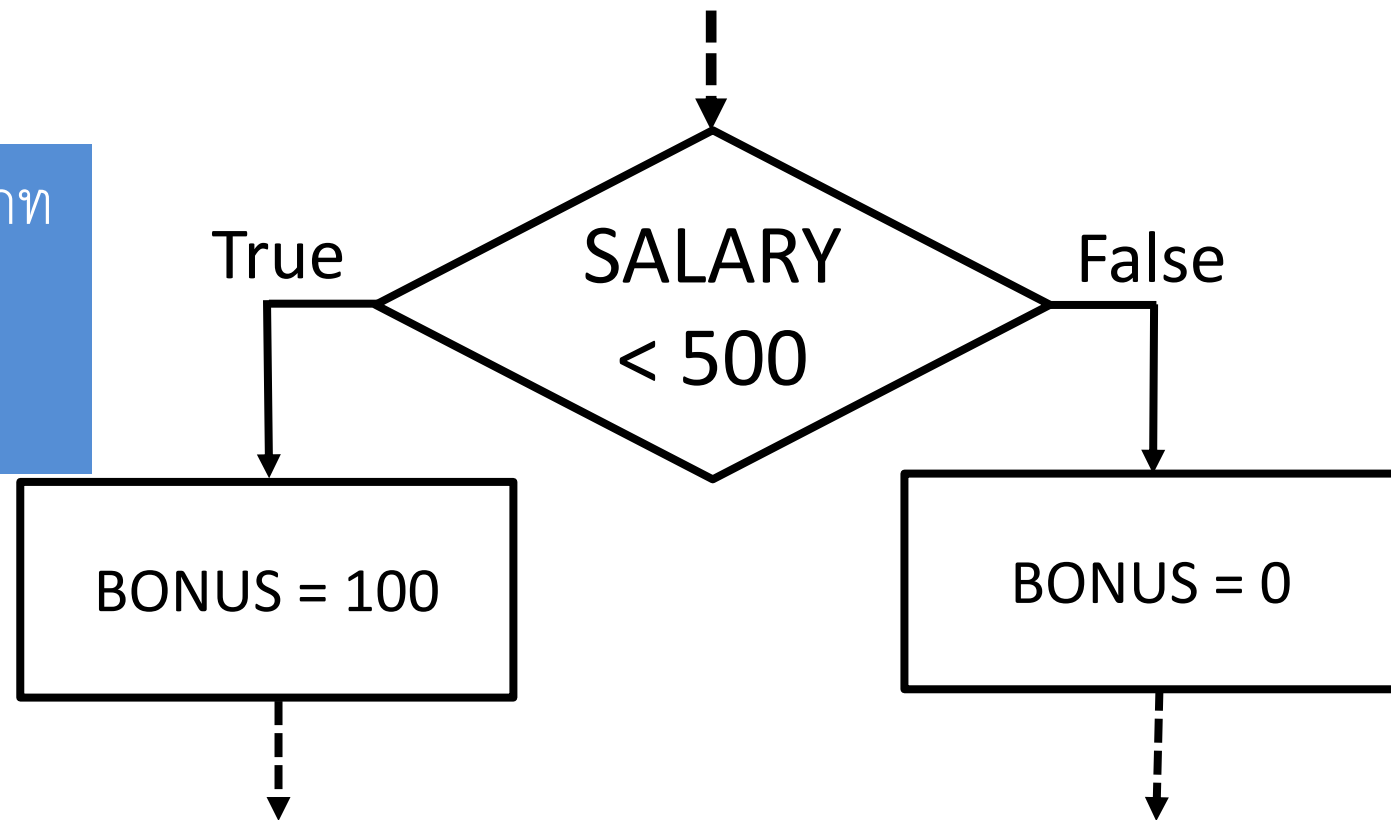
- ตัดสินใจเลือกทิศทางการทำงานของโปรแกรม โดยตรวจสอบจากเงื่อนไข
- เงื่อนไขที่ตรวจสอบจะเป็นจริงหรือเท็จเท่านั้น



Decision (ต่อ)

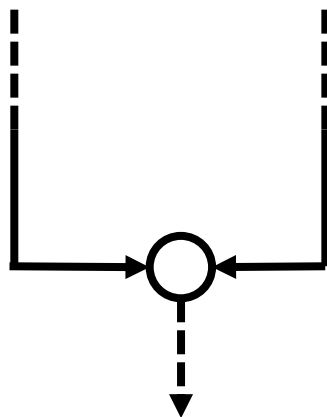
- ตัวอย่าง

ถ้า SALARY มีค่าน้อยกว่า 500 บาท
จะมี BONUS จำนวน 100 บาท
นอกจากนั้นจะไม่มีโบนัส



Connector

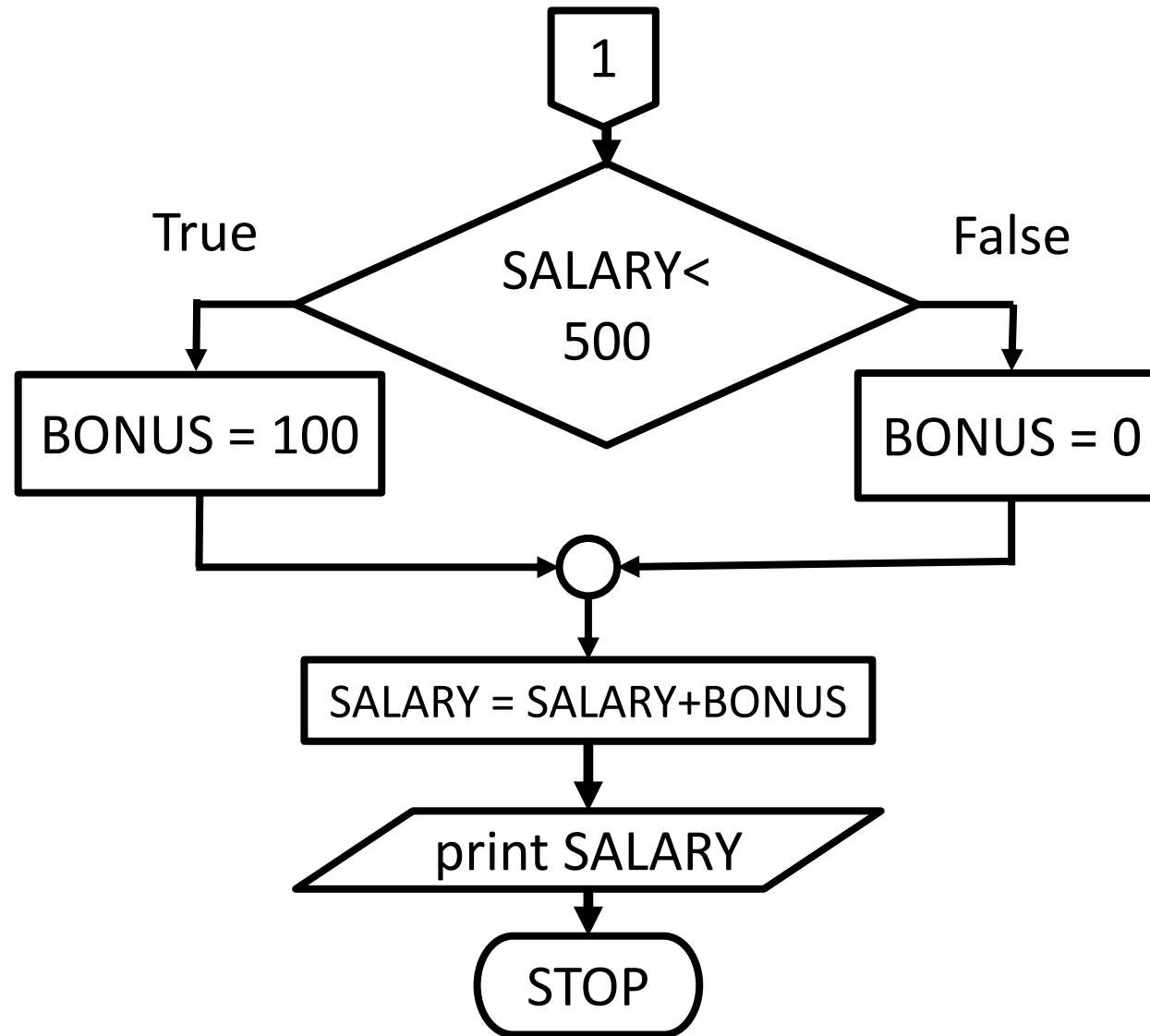
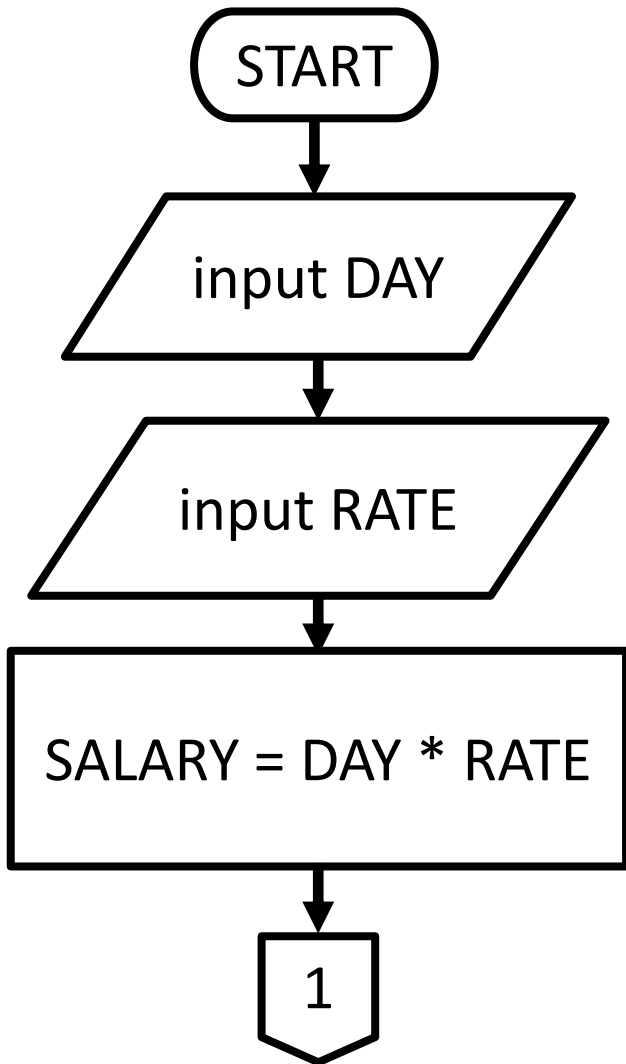
- จุดเชื่อมต่อในผังงาน
 - เป็นจุดรวมของเส้นเชื่อมที่มาจากหลายทิศทาง เพื่อไปสู่เส้นทางเดียวกัน
- CONNECTOR 1 ตัวมีเส้นทางเข้าได้ตั้งแต่ 1 เส้นขึ้นไป แต่มีเส้นทางออกได้เส้นเดียวเท่านั้น
- ไม่ว่าโปรแกรมจะไปทางไหนสุดท้ายจะกลับมารวมกันที่ CONNECTOR แล้วทำงานอื่น ๆ ต่อไป



ตัวอย่างของผังงาน — ลำดับขั้นตอนวิธี

1. กำหนดให้รับค่า
 - DAY (จำนวนวันที่ทำงาน)
 - RATE (อัตราค่าแรง)
2. จงคำนวณหา SALARY (ค่าแรงที่ได้) จากสูตร:
$$\text{SALARY} = \text{DAY} * \text{RATE}$$
3. ถ้า SALARY มีค่าน้อยกว่า 500 บาท จะมี BONUS จำนวน 100 บาท นอกจากนั้นจะไม่มีโบนัส
4. จากนั้นให้บวกโบนัสเข้าไปในเงินค่าแรง ตามสูตร:
$$\text{SALARY} = \text{SALARY} + \text{BONUS}$$
5. สุดท้าย ให้แสดงค่า SALARY

ตัวอย่างของผังงาน — ตัวผังงาน



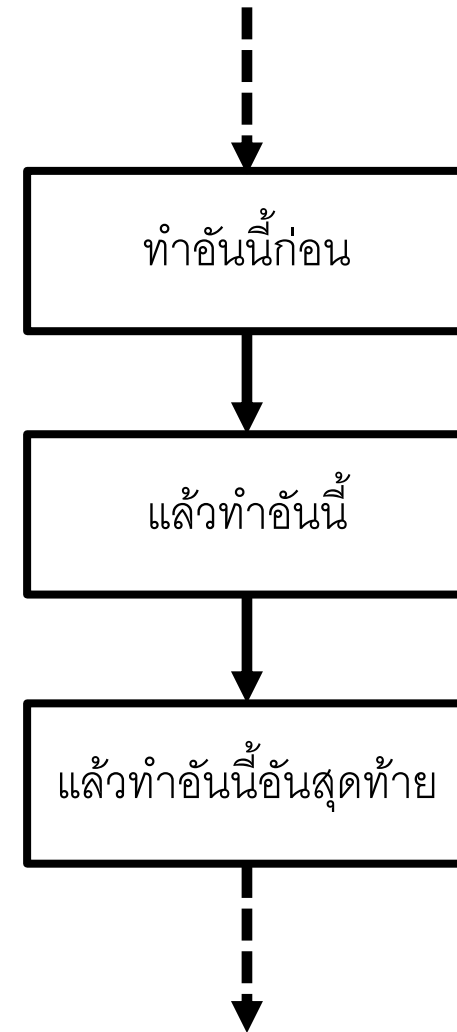
อุปกรณ์การเขียนผังงาน

- กระดาษ + ปากกา / ดินสอ
- PowerPoint
- โปรแกรมเฉพาะ เช่น Flowgorithm (<http://www.flowgorithm.org/>)

1. ผังงานแบบลำดับ (Sequence Flowchart)
 - ทำงานเรียงตามลำดับ ในผังงานจะไม่มีทางแยก
2. ผังงานแบบเลือก (Selection Flowchart)
 - มี Decision เป็นตัวตัดสินใจเลือกทิศทางของโปรแกรม
3. ผังงานแบบวนซ้ำ (Loop Flowchart)
 - ทำงานลักษณะเดียวกันหลาย ๆ ครั้ง

ผังงานแบบลำดับ (Sequence Flowchart)

- ทำงานเรียงตามลำดับ ในผังงานไม่มีทางแยก
 - ลำดับการทำงาน เป็นไปตามทิศทางที่กำหนด
 - ทำงานตามลำดับที่กำหนดตามทิศทางลูกศร
- ไม่มีส่วนของการตัดสินใจเลือกทิศทางของโปรแกรม



ตัวอย่าง 1

- กำหนดและแสดงค่าพื้นที่รูปสี่เหลี่ยมผืนผ้า

การวิเคราะห์

Input

- LENGTH (ความยาว)
- WIDTH (ความกว้าง)

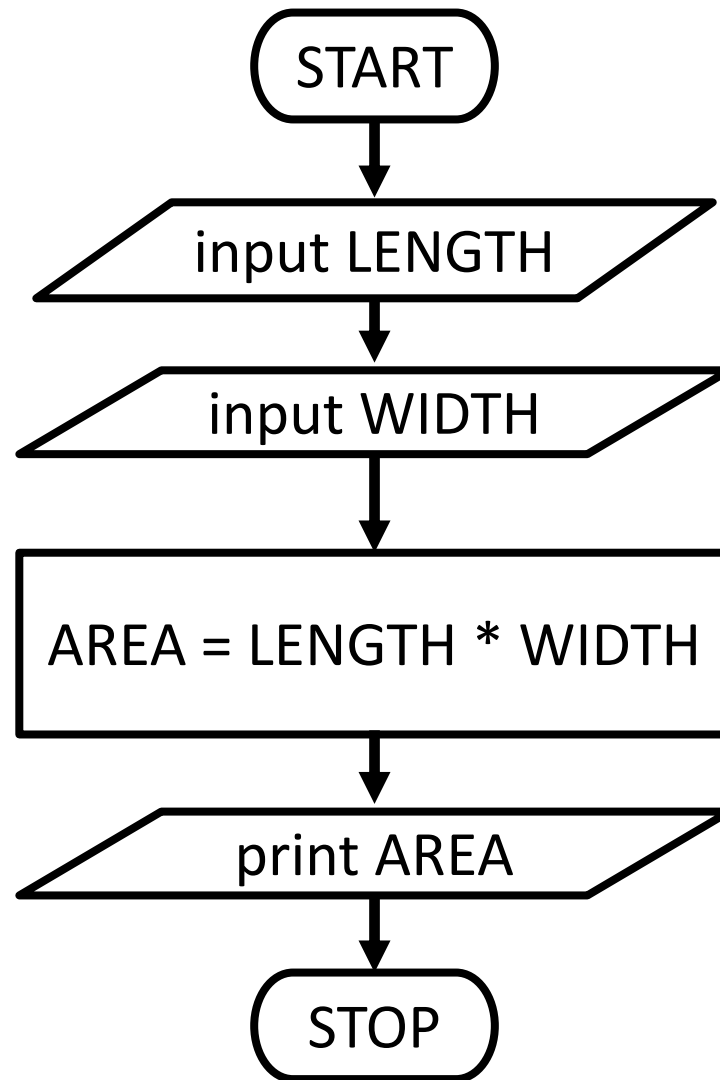
Output

- AREA (พื้นที่ของสี่เหลี่ยม)

Process

$$\text{AREA} = \text{LENGTH} * \text{WIDTH}$$

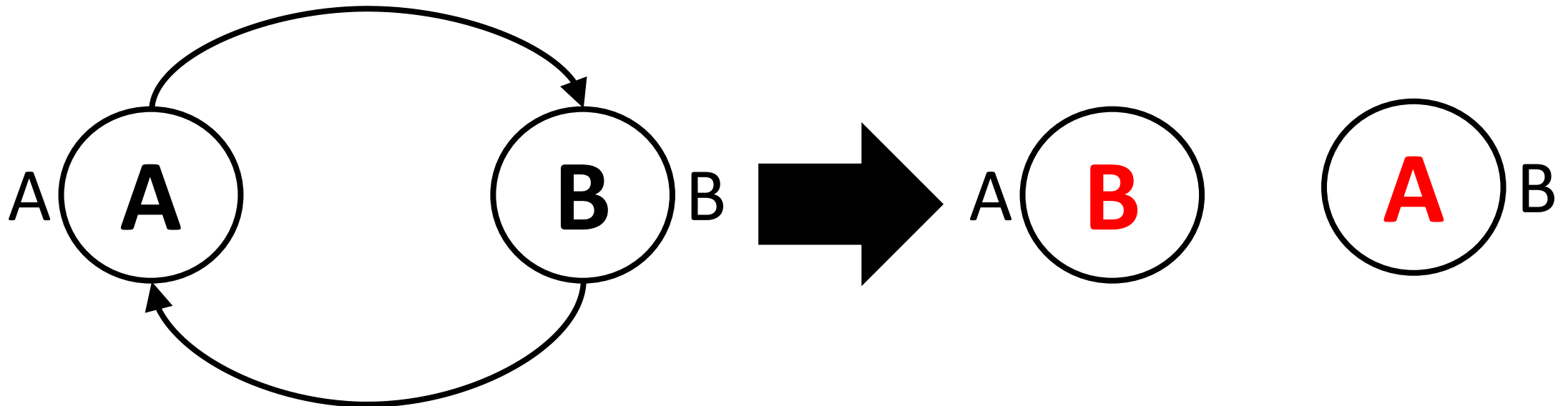
ผังงานของตัวอย่าง 1



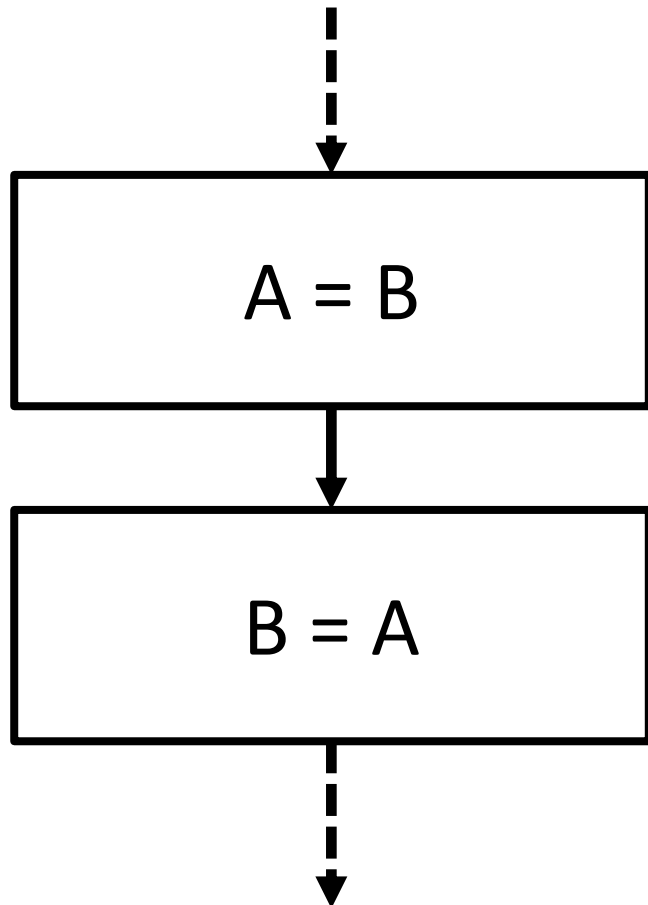
ตัวอย่าง 2: ความสำคัญของลำดับ

- การสลับค่าระหว่างสองตัวแปร

“ให้นำตัวแปรสองตัว A และ B มาสลับค่ากัน โดยที่ตัวแปร A จะมีค่าที่ตัวแปร B เคยมีอยู่ และ ตัวแปร B จะมีค่าที่ตัวแปร A เคยมีอยู่”

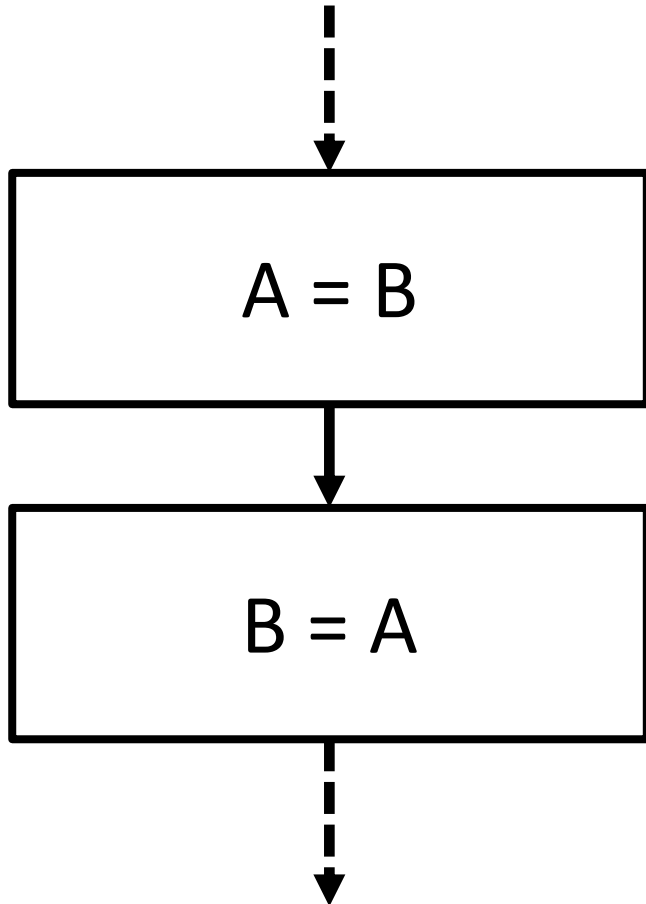


ผังงานของตัวอย่าง 2



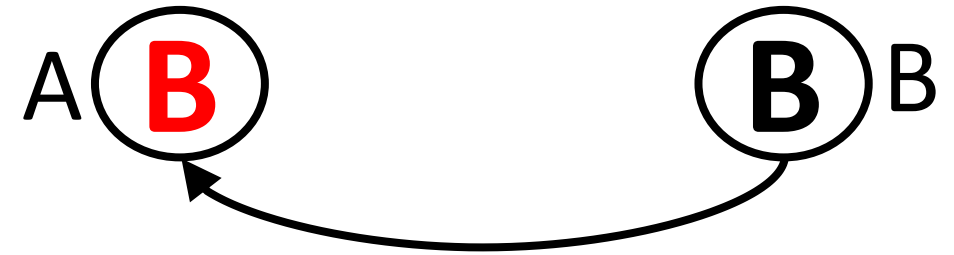
- ถูกต้องไหม
- ผิดตรงไหน

ผังงานของตัวอย่าง 2 (ต่อ)

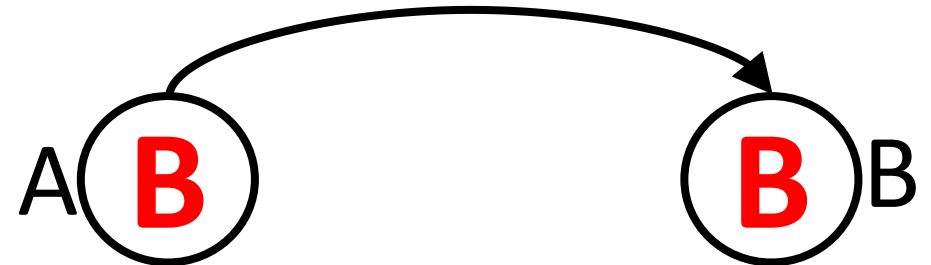


- ลำดับการทำงาน

- นำค่าของตัวแปร B ไปใส่ในตัวแปร A

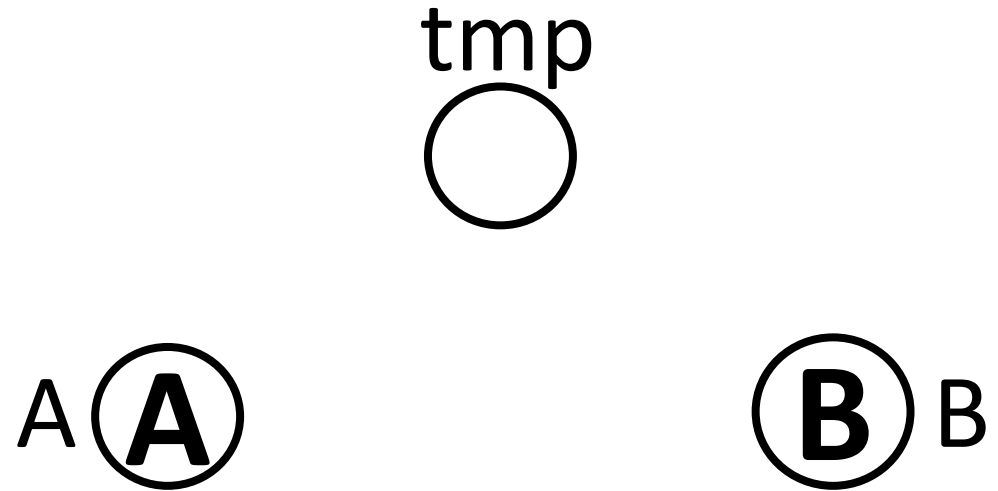


- นำค่าของตัวแปร A ในขณะนั้น ไปใส่ในตัวแปร B

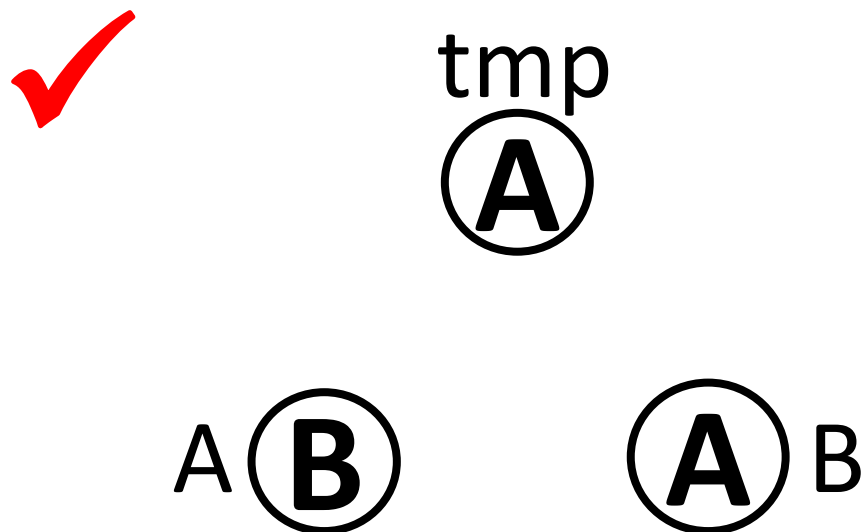
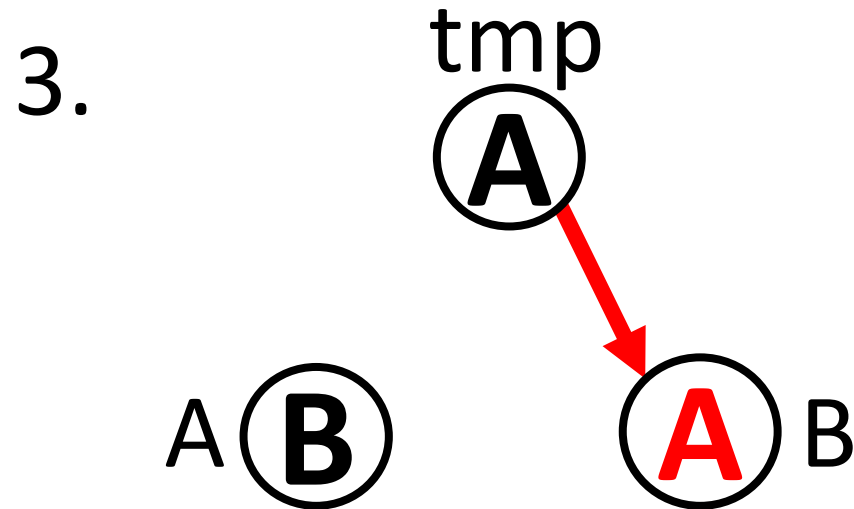
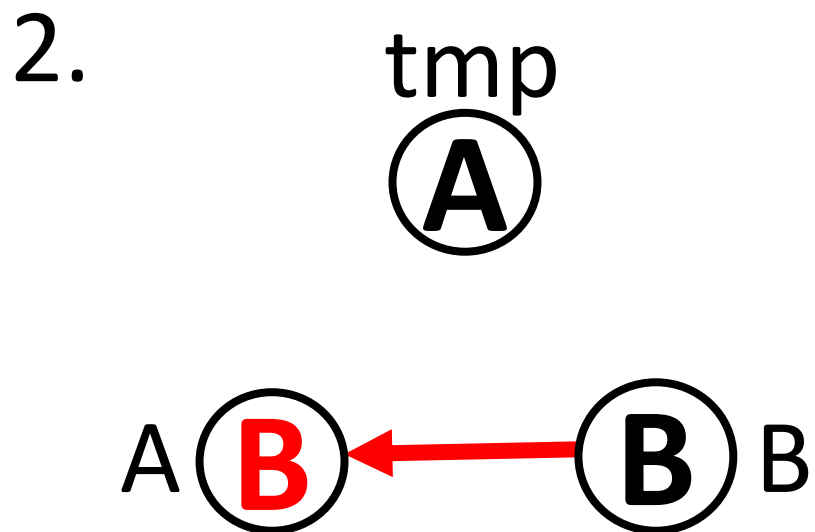
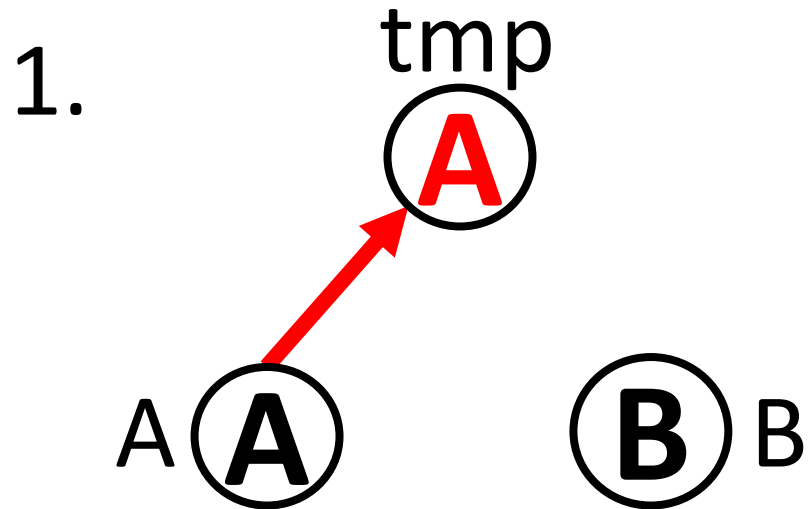


ตัวอย่าง 2 — ปัญหาและการแก้ไข

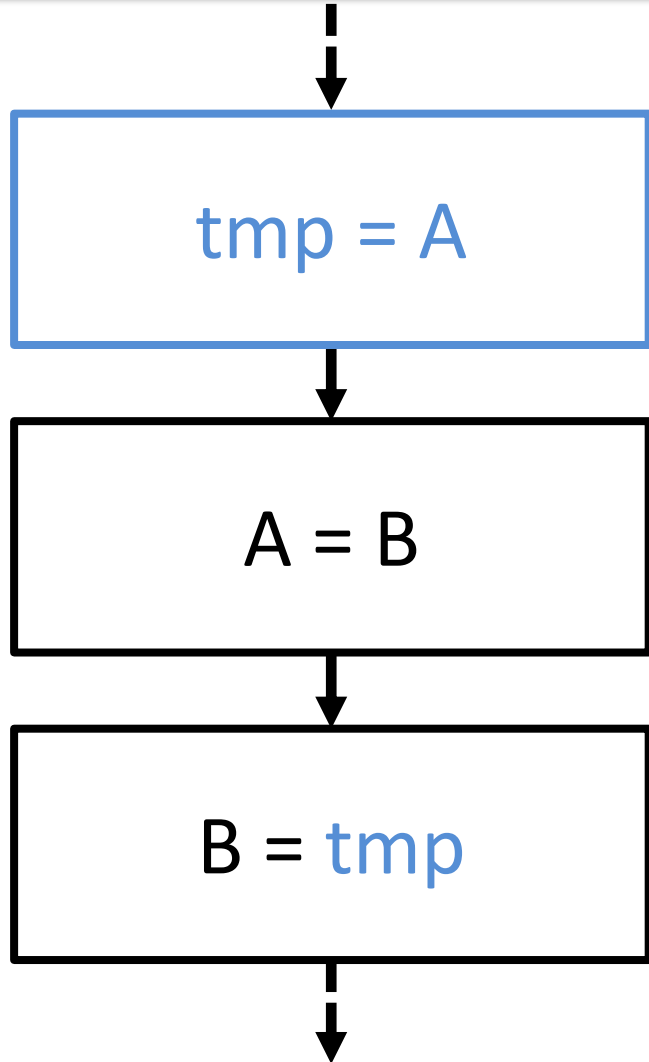
- จะเห็นว่า ค่าของตัวแปรหายไป
- จะต้องมิตัวแปรชั่วคราวมาเก็บค่าของตัวแปรตัวหนึ่งไว้ ก่อนที่ค่าจะถูกเขียนทับ
- การทำงานใหม่
 - ย้ายค่าจาก A ไปเก็บไว้ในที่ tmp ที่ตั้งไว้เป็นตัวแปรสำรอง
 - ย้ายค่าจาก B ไปที่ A
 - ย้ายค่าจาก tmp (ค่าเก่าของ A) ไปที่ B
- จะสามารถสลับค่าได้ถูกต้อง



ตัวอย่าง 2 — ลำดับการทำงานใหม่



ผังงานใหม่ของตัวอย่าง 2



- การทำงานของโปรแกรมตามลำดับ จะเกิดขึ้นทีละคำสั่ง/บรรทัด ไม่เกิดขึ้นพร้อมกัน
 - อะไรที่เกิดก่อน จะมีผลกระทบกับสิ่งที่เกิดทีหลัง
 - ต้องคำนึงถึงผลกระทบของลำดับให้ดี ๆ

“จงสร้างสมการเชิงเส้น $y = ax+b$ จากจุดพิกัด (x, y) สองจุด”

- Input คืออะไร
- Output คืออะไร
- Process คืออะไร
- เขียน Flowchart