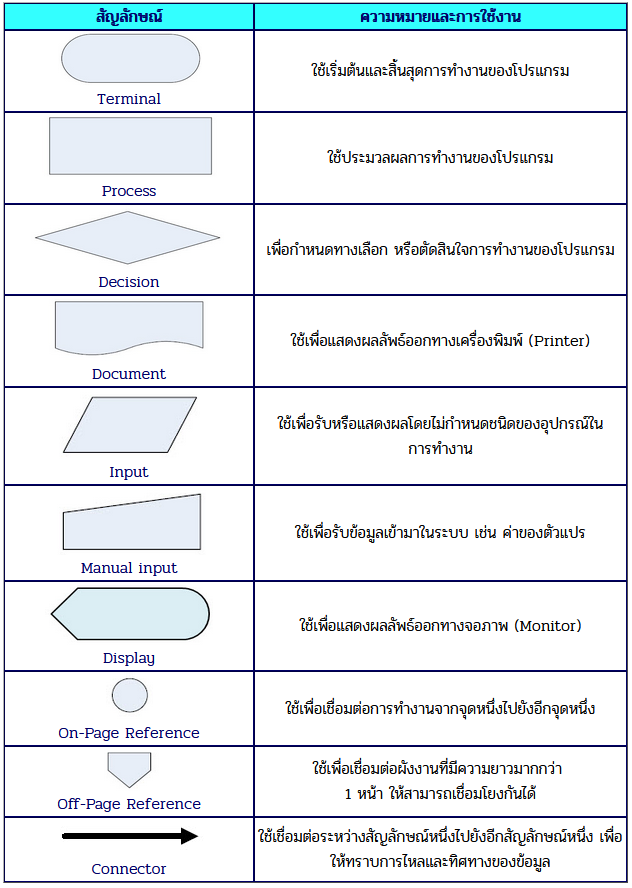
**การเขียนผังงาน****(Flowchart)**

            ผังงาน (Flowchart) เป็นผังงานรูปภาพที่ใช้แสดงแนวคิด หรือขั้นตอนการทำงานของโปรแกรม และเป็นเครื่องมือที่ช่วยให้มองเห็นภาพรวมของโปรแกรมที่ทำให้เราเขียนโปรแกรมได้ง่ายยิ่งขึ้น เนื่องจากเราสามารถมองเห็นแนวคิด และทิศทางการทำงานของโปรแกรมนั้นเอง

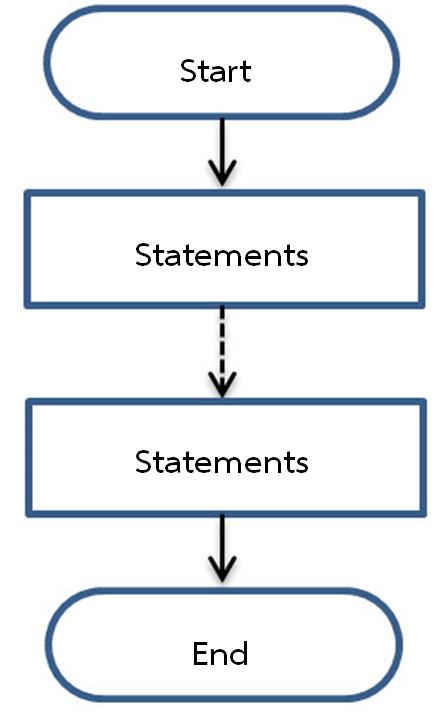
**หลักการเขียนผังงาน (Flowchart)**

            ผังงาน (Flowchart) เป็นผังงานที่ใช้แสดงแนวความคิด หรือขั้นตอนการทำงานของโปรแกรม โดยใช้สัญลักษณ์แทนคำอธิบาย ไม่ว่าจะเป็นการใช้กรอบสี่เหลี่ยมเป็นสัญลักษณ์แทนการประมวลผล หรือจะเป็นการใช้ลูกศรแทนทิศทางการทำงานของโปรแกรม ซึ่งเราสามารถสรุปสัญลักษณ์การทำงานที่ควรทราบได้ ดังนี้

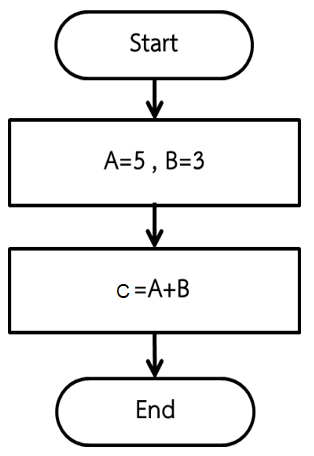


การเขียนผังงานจะมีโครงสร้างพื้นฐาน 3 รูปแบบต่อไปนี้คือ

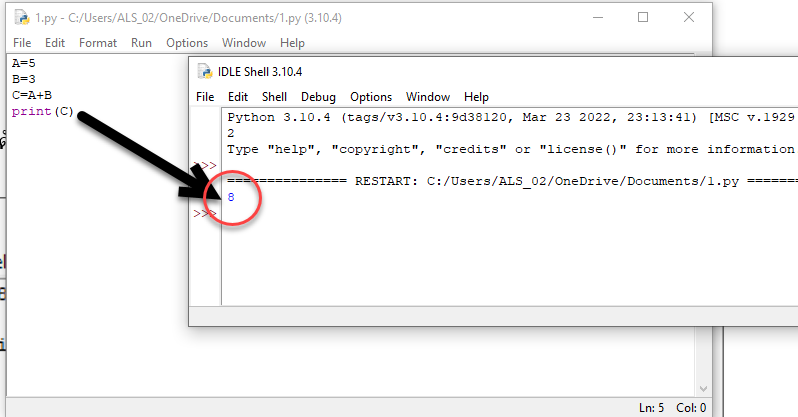
1. การทำงานแบบตามลำดับ (Sequence)   
      เป็นรูปแบบการเขียนโปรแกรมที่ง่ายที่สุด และไม่มีความซับซ้อน มีลำดับการทำงานจากบนลงล่าง   
มีการทำงานทีละคำสั่งจนจบการทำงาน ซึ่งมีรูปแบบการทำงานดังรูปที่ 1

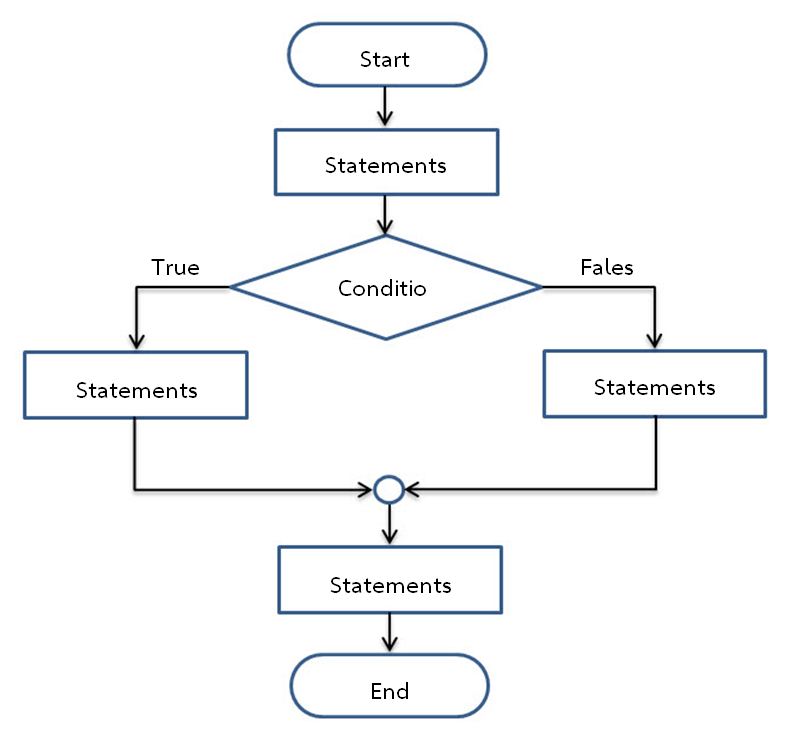
****  
รูป การทำงานแบบตามลำดับ (Sequence)

โครงสร้างลักษณะนี้เป็นโครงสร้างพื้นฐานของผังงาน และเป็นลักษณะขั้นตอนการทำงานที่พบมากที่สุด คือ ทำงานทีละขั้นตอนตามลำดับ   
—————————————————————————————————————  
**ตัวอย่างที่ การเขียนและอธิบายการทำงานของ Flowchart แบบตามลำดับดังนี้**

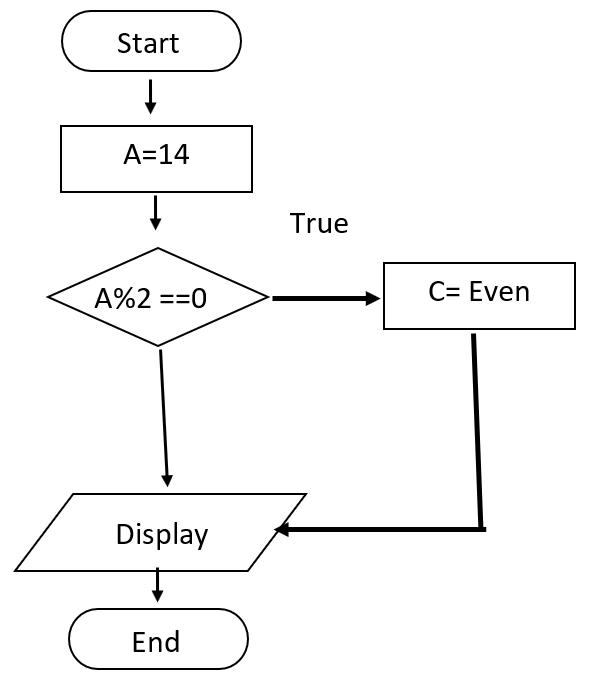


**อธิบายการทำงานของ Flowchart ได้ดังนี้**  
1. เริ่มการทำงานของโปรแกรม โดยกำหนดค่าตัวแปร A และ B มีค่าเท่ากับ 5 และ 3   
ตามลำดับ  
 2. กำหนดตัวแปร A มีค่าเท่ากับ A+B จบการทำงานของโปรแกรม  
  
**จาก Flowchart สามารถเขียนโค้ดโปรแกรมดังต่อไปนี้**

  
**2. การเลือกกระทำตามเงื่อนไข (Decision)**การเลือกทำงานตามเงื่อนไข (Decision) เป็นรูปแบบการเขียนโปรแกรมที่มีทางเลือกเพื่อตัดสินใจซึ่งโปรแกรมจะตรวจสอบเงื่อนไข เพื่อเลือกทิศทางการทำงานของโปรแกรม โดยเลือกทางเลือกใดทางเลือกหนึ่ง จากสองทางเลือกเท่านั้นคือ  
> ทำงานในทางเลือกหนึ่งเมื่อเงื่อนไขเป็นจริง และ  
      > ทำงานในอีกทางเลือกหนึ่งเมื่อเขื่อนไขเป็นเท็จ เมื่อทำงานในแต่ละทางเลือกเสร็จแล้ว โปรแกรมก็จะทำงานในขั้นตอนต่อไป

*****รูป การเลือกกระทำตามเงื่อนไข (Decision)*

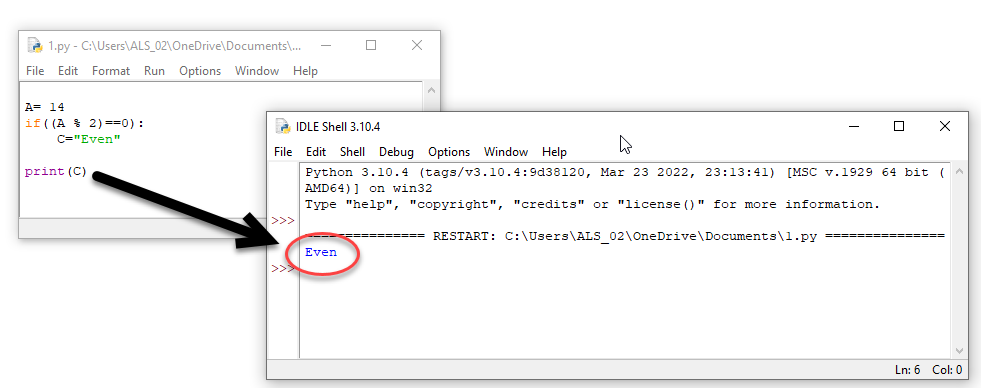
**ตัวอย่างที่ การเขียนและอธิบายการทำงานของ Flowchart แบบเลือกการทำงานตามเงื่อนไขดังนี้**



**อธิบายการทำงานของ Flowchart ได้ดังนี้**1. เริ่มการทำงานของโปรแกรม โดยกำหนดค่าตัวแปร A มีค่าเท่ากับ 14  
              2. ตรวจสอบเงื่อนไขว่าค่าตัวแปร A/2 มีเศษหรือไม่ โดยใช้ โค้ด A%2  
       3. ถ้าเงื่อนไงว่าA/2 มีเศษ กำหนดให้ C= คู่/Even

4. แสดงผลลัพท์ทางจอภาพ จบการทำงานของโปรแกรม

**จาก Flowchart สามารถเขียนโค้ดโปรแกรมได้ดังนี้**



**ตัวอย่าง การเขียนและอธิบายการทำงานของ Flowchart แบบเลือกการทำงานตามเงื่อนไขดังนี้**

Start

A=7,B=3

True

False

sA>B

B-A

A-B

7-3

Display

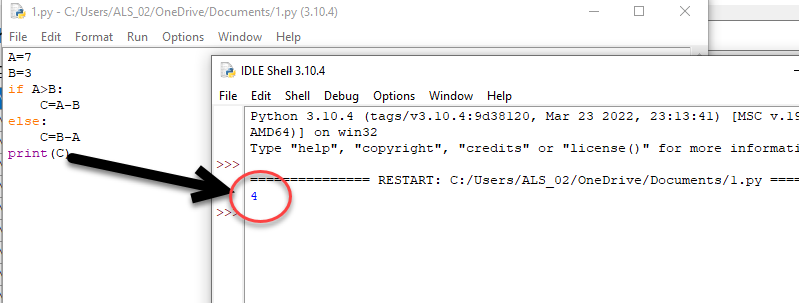
End

**อธิบายการทำงานของ Flowchart ได้ดังนี้**1. เริ่มการทำงานของโปรแกรม โดยกำหนดค่าตัวแปร A และ B มีค่าเท่ากับ 7 และ 3  
ตามลำดับ  
              2. ตรวจสอบเงื่อนไขว่าค่าตัวแปร A มากกว่าค่าตัวแปร B หรือไม่   
       3. ถ้าเงื่อนไงว่าค่าตัวแปร A มากกว่าค่าตัวแปร B เป็นจริง กำหนดให้ A-B

4.ถ้าเงื่อนไงว่าค่าตัวแปร A มากกว่าค่าตัวแปร B เป็นจริง กำหนดให้ B-A

5. แสดงผลลัพท์ทางจอภาพ จบการทำงานของโปรแกรม

**จาก Flowchart สามารถเขียนโค้ดโปรแกรมได้ดังนี้**



**นอกจากนี้เงื่อนไขยังสามารถใช้ตรวจสอบหลายๆช่วงเงื่อนไขได้ดังต่อไปนี้**

Start

Statements

Condit\_1

Statements

Condit\_1

True

False

True

Condit\_3

Condit\_2

Statements

Condit\_2

False

True

Condit\_3

Statements

Statements

End

**ผังการกำหนดเกรด นักเรียน ที่ได้ คะแนน 65 คะแนน โดยมีเงื่อนไขคือ ถ้ามากกว่า 80 คะแนน ได้ A มากกว่า 70 คะแนน ได้ B**

**มากกว่า 60 คะแนน ได้ C 60 คะแนน ได้ F**

Start

A = 65

C=A

A>=80

True

False

True

A>=60

C=B

A>=70

False

C=C

True

False

C=F

print C

End

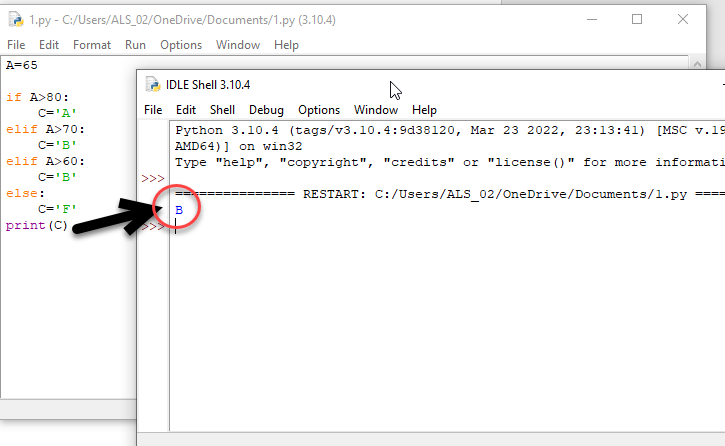
**อธิบายการทำงานของ Flowchart ได้ดังนี้**1. เริ่มการทำงานของโปรแกรม โดยกำหนดค่าตัวแปร A มีค่าเท่ากับ 65  
ตามลำดับ  
              2. ตรวจสอบเงื่อนไขว่าค่าตัวแปร A มากกว่าค่าตัวแปร B หรือไม่   
       3. ถ้าเงื่อนไงว่าค่าตัวแปร A มากกว่า 80 กำหนดให้ C=A

4. ถ้าเงื่อนไงว่าค่าตัวแปร A มากกว่า 70 กำหนดให้ C=B

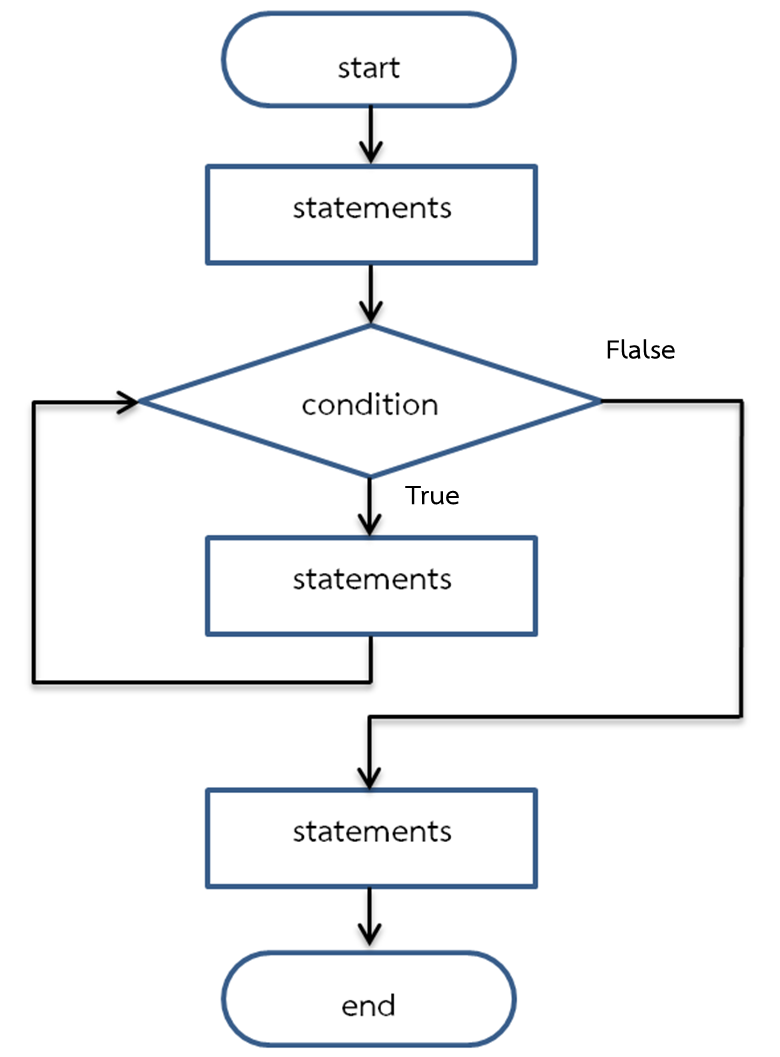
5. ถ้าเงื่อนไงว่าค่าตัวแปร A มากกว่า 60 กำหนดให้ C=C

6. ถ้าเงื่อนไงว่าค่าตัวแปร A ไม่ตรงตามข้อ 2-5 กำหนดให้ C=F

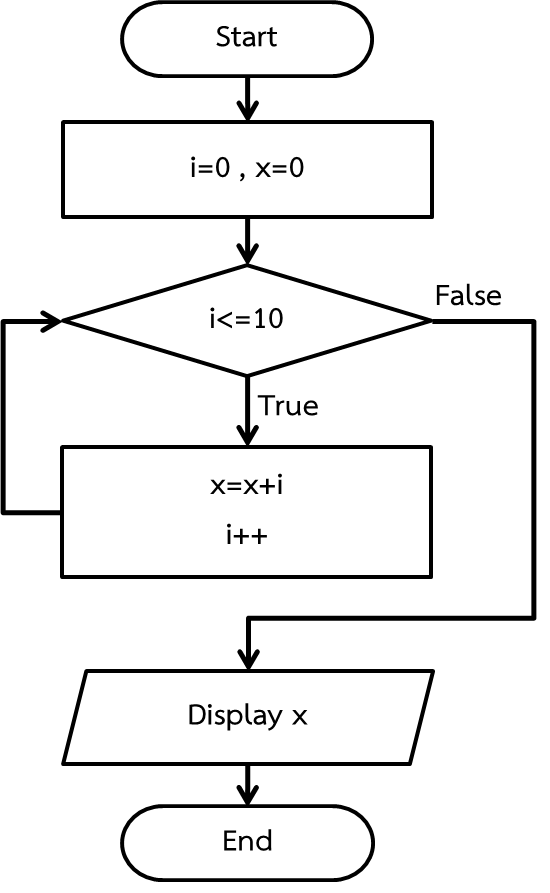
5. แสดงผลลัพท์ทางจอภาพ จบการทำงานของโปรแกรม  
  
**จาก Flowchart สามารถเขียนโค้ดโปรแกรมได้ดังนี้**



3. การทำซ้ำ (Loop)   
การทำซ้ำ (Loop) เป็นรูปแบบการเขียนโปรแกรมที่มีการทำงานในขั้นตอนเดิมซ้ำ ๆ กัน หลาย ๆ รอบ ซึ่งการทำงานของโปรแกรมจะมีการตรวจสอบเงื่อนไข เพื่อกำหนดให้เข้าทำงานในลูป หรือออกจากลูป  
การทำงาน ซึ่งมีรูปแบบการทำงานดังรูปที่ 1

  
รูปการทำซ้ำ(Loop)

**ตัวอย่าง การเขียนและอธิบายการทำงานของ Flowchart แบบทำซ้ำดังนี้**



**อธิบายการทำงานของ Flowchart ได้ดังนี้**  
1. เริ่มการทำงานของโปรแกรม โดยกำหนดค่าตัวแปร i มีค่าเท่ากับ 0 และ x มีค่าเท่ากับ 0 ตามลำดับ  
2. ตรวจสอบเงื่อนไขว่าค่าตัวแปร i น้อยกว่าหรือเท่ากับ 10 หรือไม่  
3. ถ้าเงื่อนไขค่าตัวแปร i น้อยกว่าหรือเท่ากับ 10 เป็นจริง กำหนดให้ตัวแปร x มีค่า  
เท่ากับ x+i และตัวแปร i เพิ่มค่า 1 ค่า คือ i++   
4. เมื่อตรวจสอบเงื่อนไขว่าค่าตัวแปร i น้อยกว่าหรือเท่ากับ 10 เป็นเท็จ ให้ออก  
จากลูปการทำงาน  
5. แสดงค่าตัวแปร x ทางจอบภาพ จบการทำงานของโปรแกรม

**จาก Flowchart สามารถเขียนโค้ดโปรแกรมได้ดังนี้**

