

การเขียนโปรแกรมภาษา



1. การเขียนผังงาน (Flowchart) จัดอยู่ในกระบวนการใดของขั้นตอนการแก้ปัญหา

- ก) การวิเคราะห์และกำหนดรายละเอียดของปัญหา
- ข) การเลือกเครื่องมือและออกแบบขั้นตอนวิธีในการแก้ปัญหา
- ค) การดำเนินการแก้ปัญหา
- ง) การตรวจสอบและปรับปรุงวิธีการ



ขั้นตอนการแก้ปัญหา

1. การวิเคราะห์และกำหนด
รายละเอียดของปัญหา

2. การเลือกเครื่องมือ
และออกแบบขั้นตอนวิธี
ในการแก้ปัญหา

3. การดำเนินการแก้ปัญหา

4. การตรวจสอบ
และปรับปรุงวิธีการ



1. การวิเคราะห์และกำหนดรายละเอียดของปัญหา

1.1 การระบุข้อมูลออก >> สิ่งที่โจทย์ต้องการในการแก้ปัญหาด้วยคอมพิวเตอร์

1.2 การระบุข้อมูลเข้า >> ข้อมูลเริ่มต้นหรือเงื่อนไขที่โจทย์กำหนดให้

1.3 รายละเอียดของปัญหา >> การพิจารณาความต้องการของปัญหาและวิธีการประมวลผล

2. การเลือกเครื่องมือและออกแบบขั้นตอนวิธีในการแก้ปัญหา

2.1 รหัสจำลอง (Pseudo code)

2.2 ผังงาน (Flowchart)



3. การดำเนินการแก้ปัญหา

เป็นขั้นตอนของการใช้โปรแกรมประยุกต์หรือเขียนโปรแกรมขึ้นเอง โดยใช้ภาษาคอมพิวเตอร์เพื่อแก้ปัญหา ซึ่งต้องอาศัยความเชี่ยวชาญในการใช้โปรแกรมหรือภาษาคอมพิวเตอร์นั้นๆ

4. การตรวจสอบและปรับปรุงวิธีการ

เป็นการตรวจสอบให้แน่ใจว่าปัญหารวมทั้งโปรแกรมที่พัฒนาขึ้นให้ผลลัพธ์ถูกต้อง



2. ข้อใดเป็นสัญลักษณ์ที่ใช้ในการตรวจสอบเงื่อนไขหรือตัดสินใจ

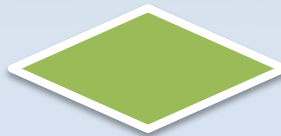
ก)



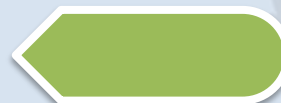
ข)



ค)



ง)



ผังงาน (Flowchart)






การอธิบายขั้นตอนวิธีการแก้ปัญหา โดยใช้รูปสัญลักษณ์มาเรียงต่อกัน สัญลักษณ์แต่ละแบบจะมีความหมายถึงกระบวนการที่แตกต่างกัน โดยมีการเขียนคำอธิบายสั้นๆ เพิ่มเติมในสัญลักษณ์ สัญลักษณ์ของผังงานถูกกำหนดโดยสถาบันมาตรฐานแห่งชาติอเมริกา (The American National Standard Institute: ANSI) เพื่อให้สามารถสื่อความหมายได้ตรงกัน

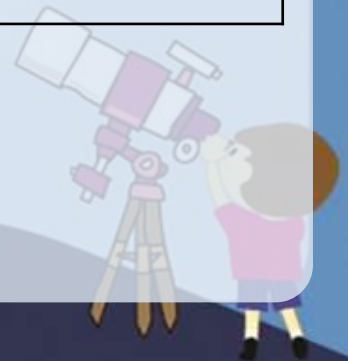
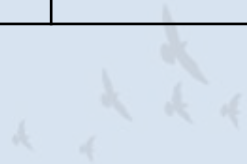


สัญลักษณ์ผังงาน (Flowchart)

ชื่อ	สัญลักษณ์	ความหมาย
Terminal		จุดเริ่มต้น/สิ้นสุดของผังงาน
Input/Output		รับข้อมูล/แสดงผลข้อมูลแบบไม่ระบุชื่อ
Manual Input		รับข้อมูลเข้าทางแป้นพิมพ์
Process		กระบวนการทำงาน/การประมวลผล
Decision		ตัดสินใจ/เปรียบเทียบ
Display		แสดงผลข้อมูลทางจอภาพ
Document		แสดงผลข้อมูลทางเครื่องพิมพ์/กระดาษ

สัญลักษณ์ผังงาน (Flowchart)

ชื่อ	สัญลักษณ์	ความหมาย
Connector		เชื่อมต่อผังงานภายในหน้ากระดาษเดียวกัน
Page connector		เชื่อมต่อผังงานคนละหน้ากระดาษ
Preparation		กำหนดจำนวนรอบในการทำงานซ้ำ
Flow line		ทิศทางการไหลของข้อมูล
Comment		อธิบายรายละเอียดเพิ่มเติม



3. หากต้องการเขียนโปรแกรมเพื่อคำนวณหาโบนัสของพนักงานในบริษัททั้งหมด โดยพนักงานจะได้รับโบนัสไม่เท่ากันซึ่งขึ้นอยู่กับเงินเดือนที่ได้รับ นักเขียนควรเลือกโครงสร้างการเขียนโปรแกรมแบบใดจึงจะเหมาะสมที่สุด

- ก) โครงสร้างแบบลำดับ
- ข) โครงสร้างแบบเลือกทำ
- ค) โครงสร้างแบบทำซ้ำ
- ง) ข้อ ข และ ค ถูกต้อง

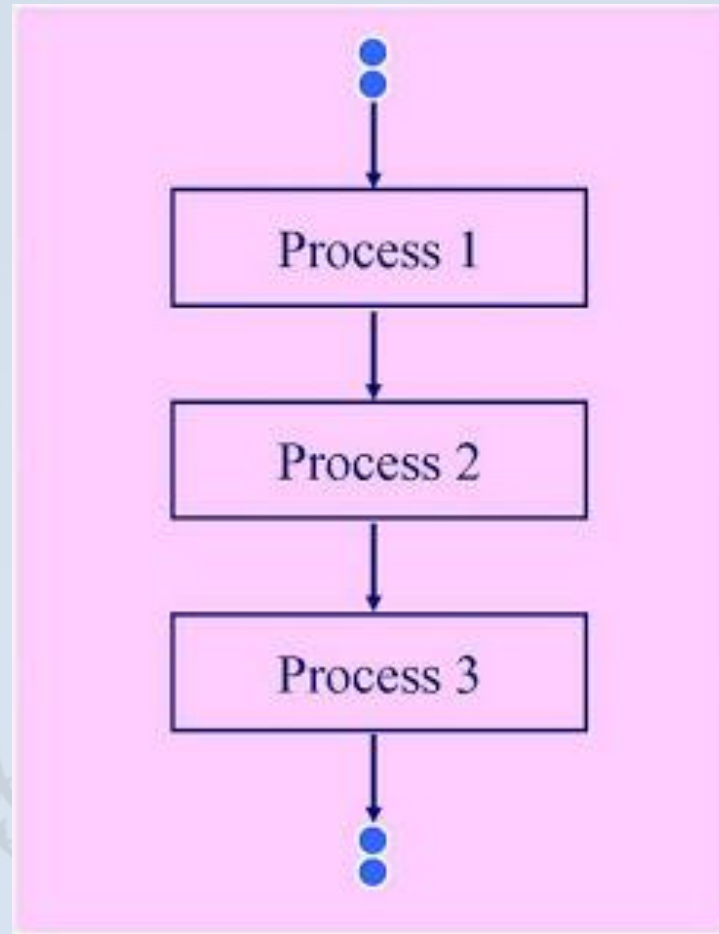


โครงสร้างผังงาน

1. โครงสร้างแบบลำดับ (Sequential structure)
2. โครงสร้างแบบเลือกทำหรือมีเงื่อนไข (Selection structure)
3. โครงสร้างแบบวนซ้ำ (Repetition structure)

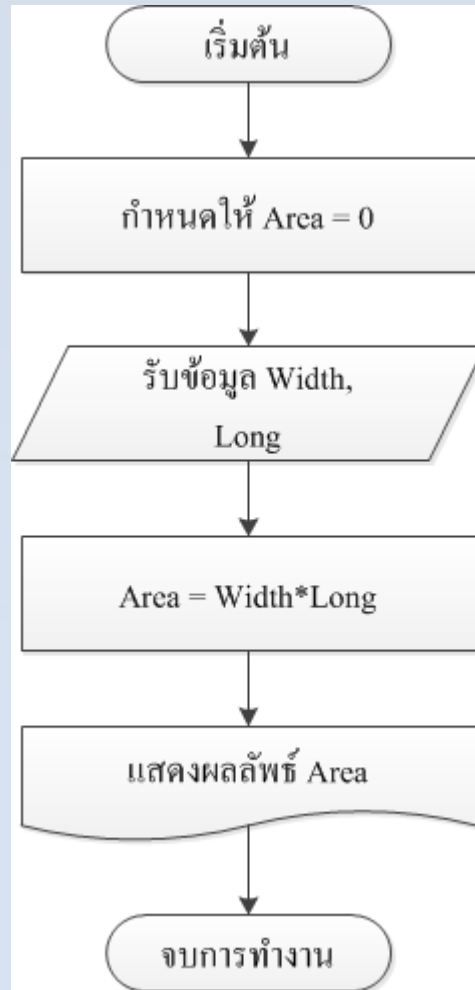


1. โครงสร้างแบบลำดับ (Sequential structure)



ตัวอย่าง ฟังก์ชันการคำนวณหาพื้นที่สี่เหลี่ยม

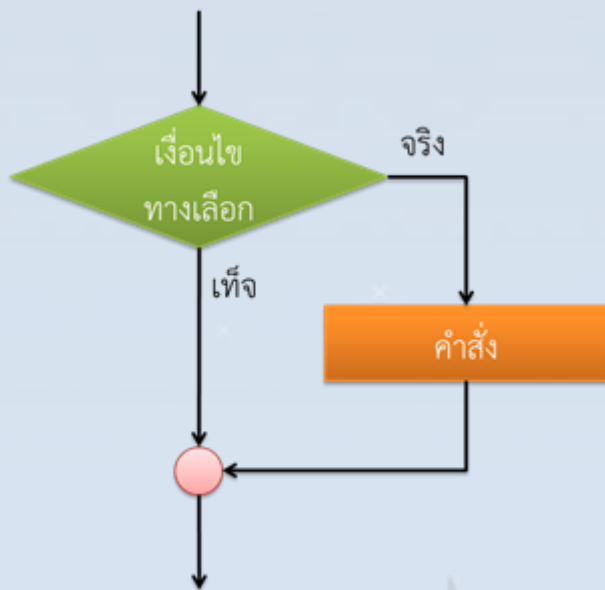
PHILIPPA FUNDIT UNIVERSITY
THAILAND UNIVERSITY



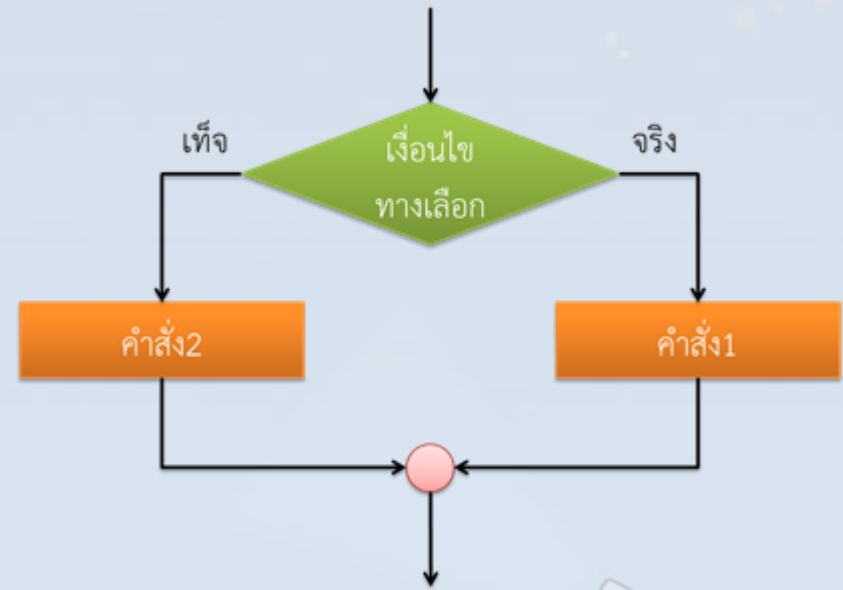
2. โครงสร้างแบบเลือกทำหรือมีเงื่อนไข (Selection structure)

BHUMBUK UNIVERSITY
BANGKOK UNIVERSITY

2.1 โครงสร้างแบบ 1 ทางเลือก



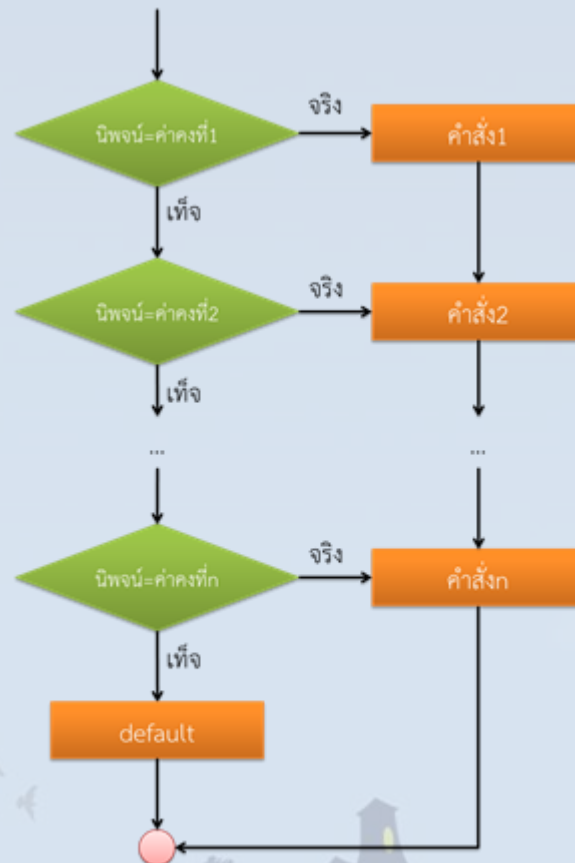
2.2 โครงสร้างแบบ 2 ทางเลือก



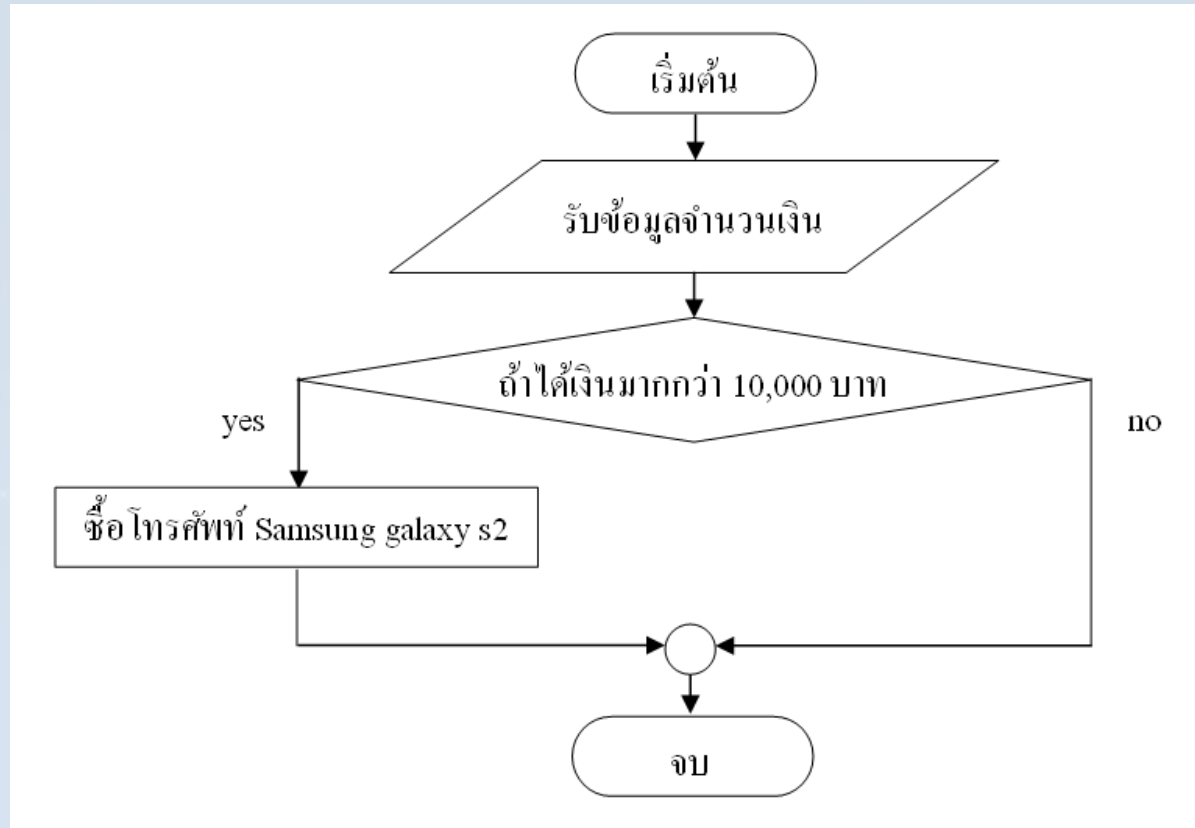
2. โครงสร้างแบบเลือกทำหรือมีเงื่อนไข (Selection structure)

BUENAVISTA UNIVERSITY
COLLEGE OF ENGINEERING

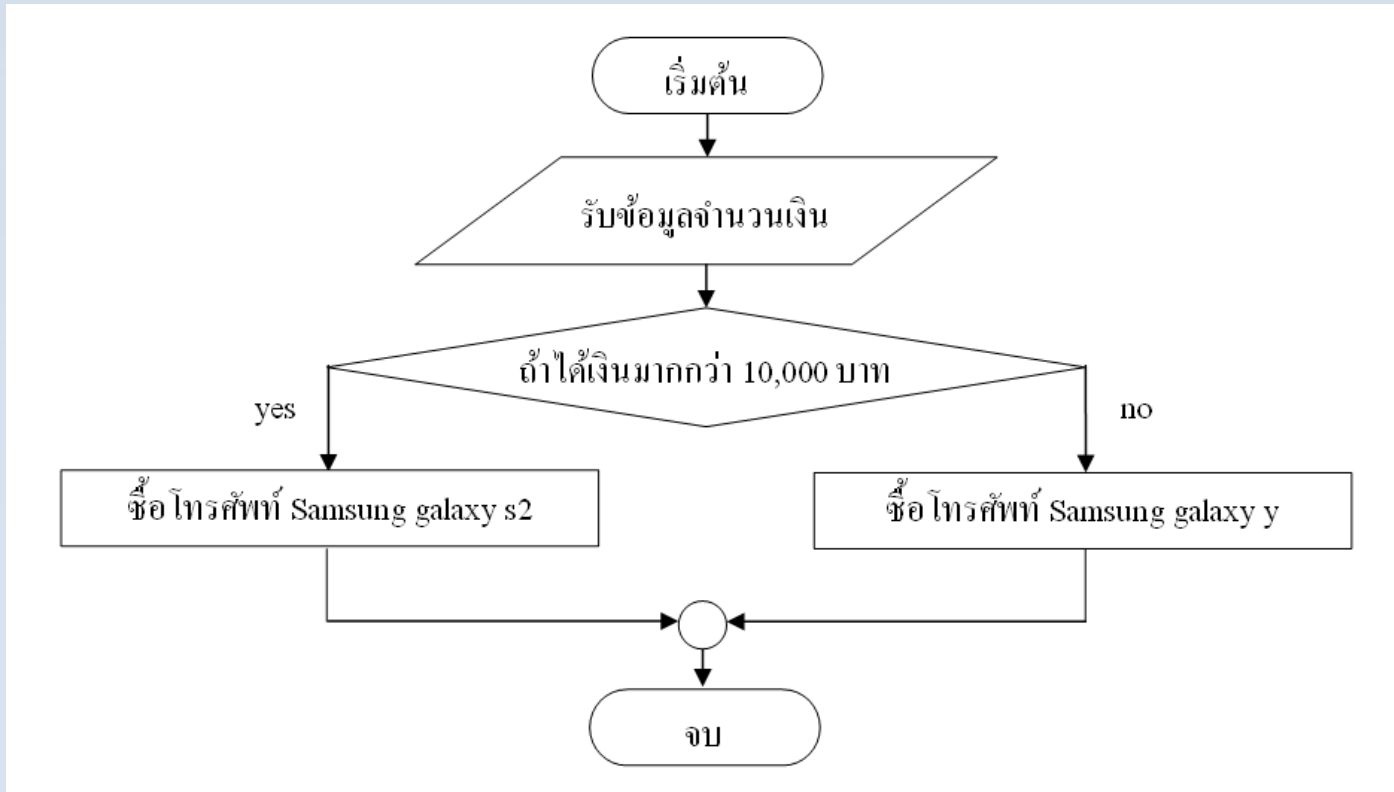
2.3 โครงสร้างแบบหลายทางเลือก



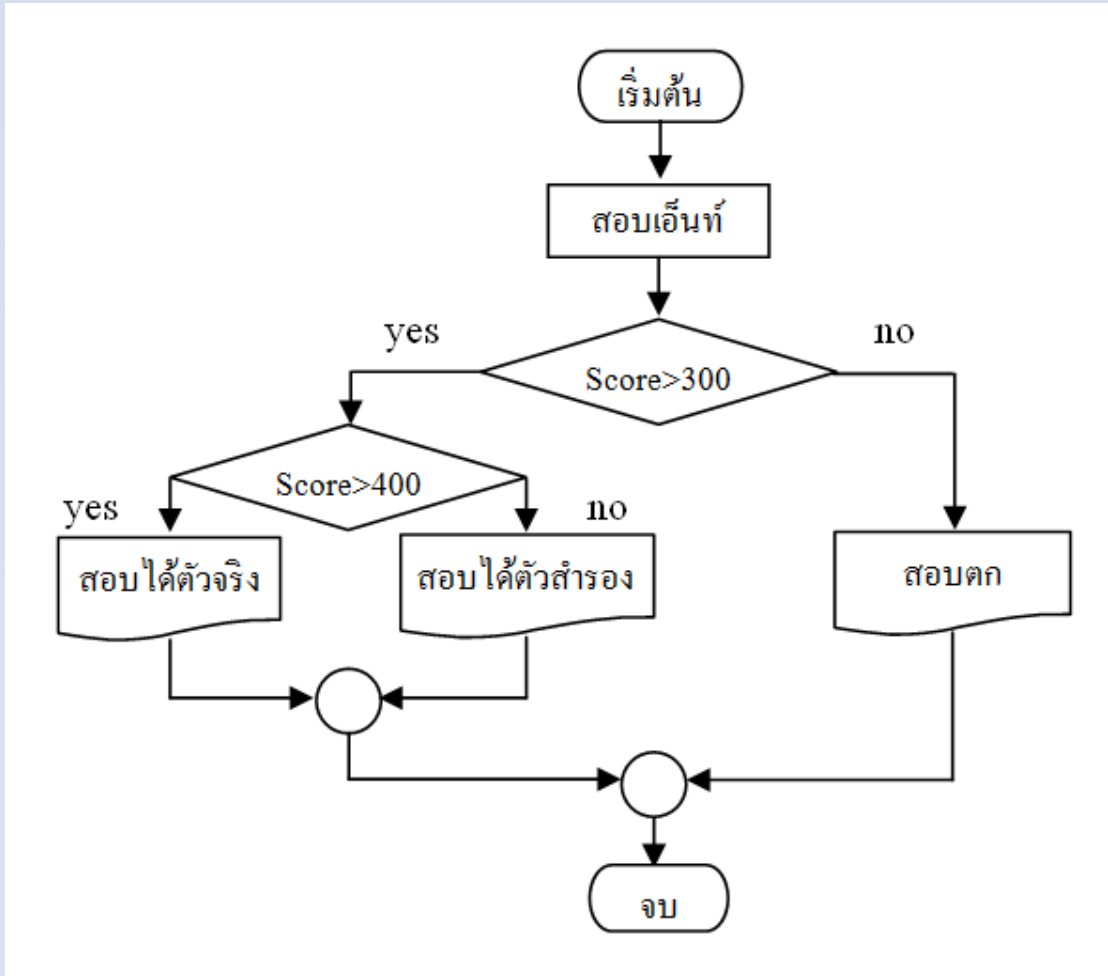
ตัวอย่างผังงานแบบ 1 ทางเลือก



ตัวอย่างผังงานแบบ 2 ทางเลือก

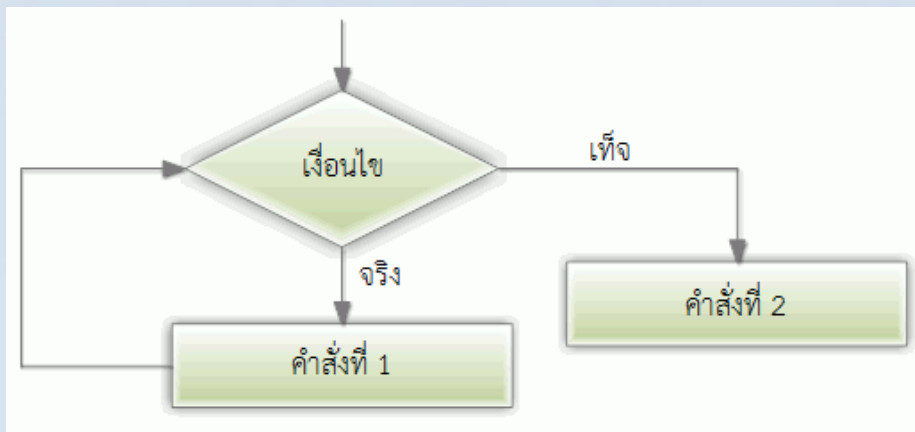


ตัวอย่างผังงานแบบหลายทางเลือก

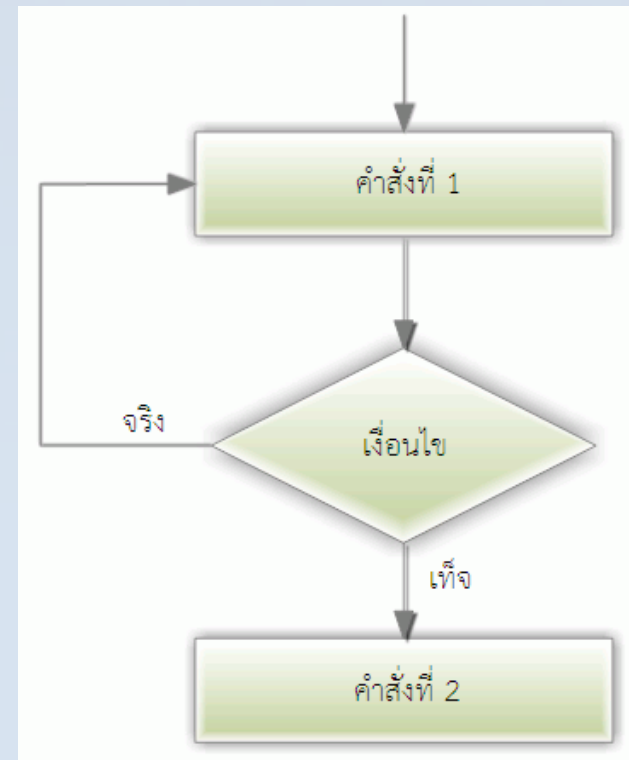


2.3 โครงสร้างแบบวนซ้ำ (Repetition structure)

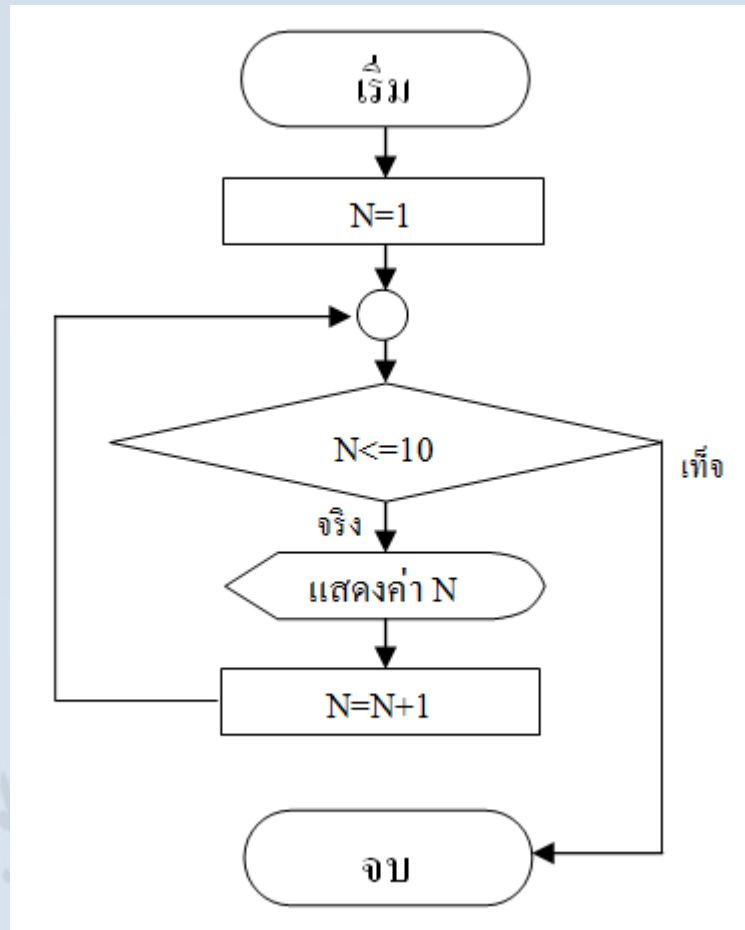
3.1 วนซ้ำแบบลูป while



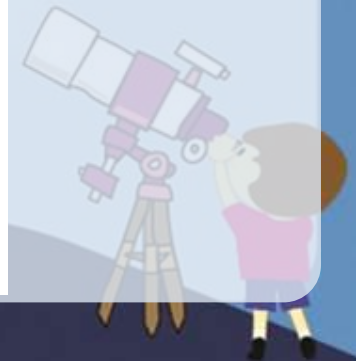
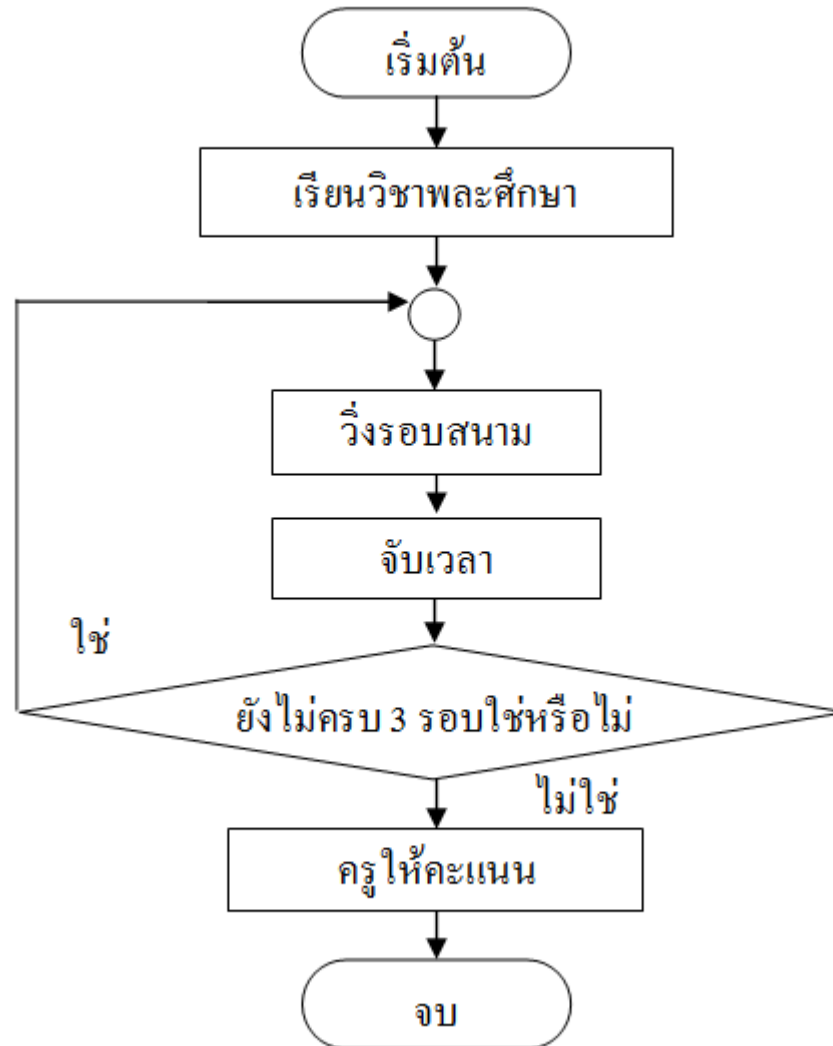
3.2 วนซ้ำแบบลูป do...while



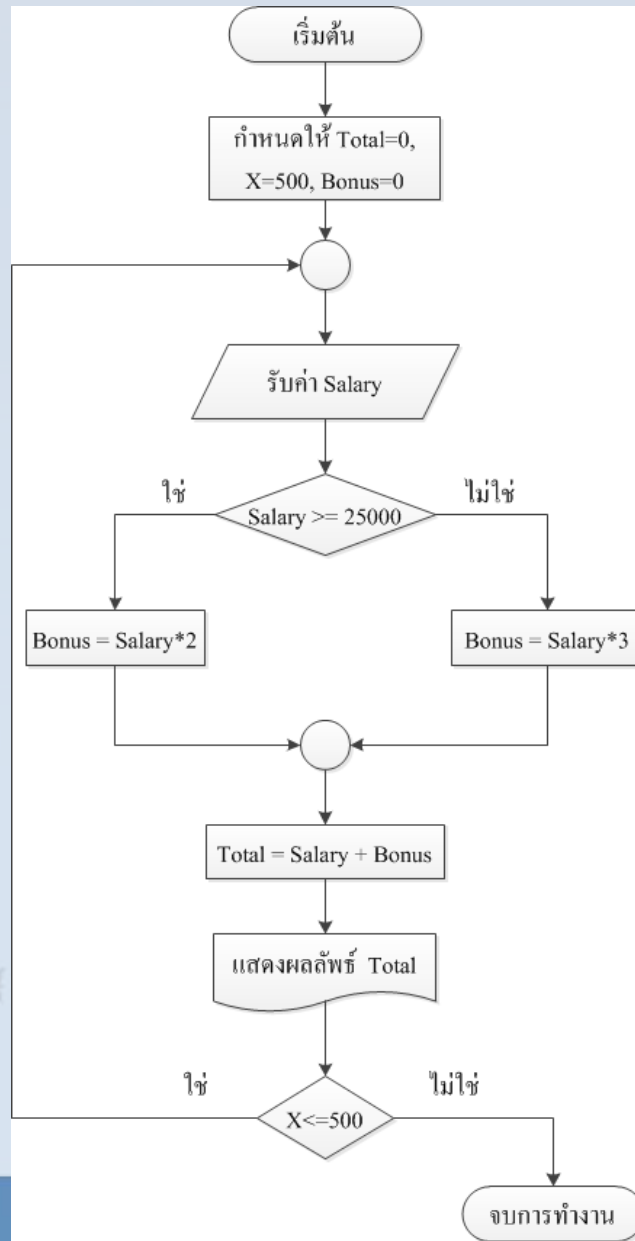
ตัวอย่างผังงานวนซ้ำแบบลูป while



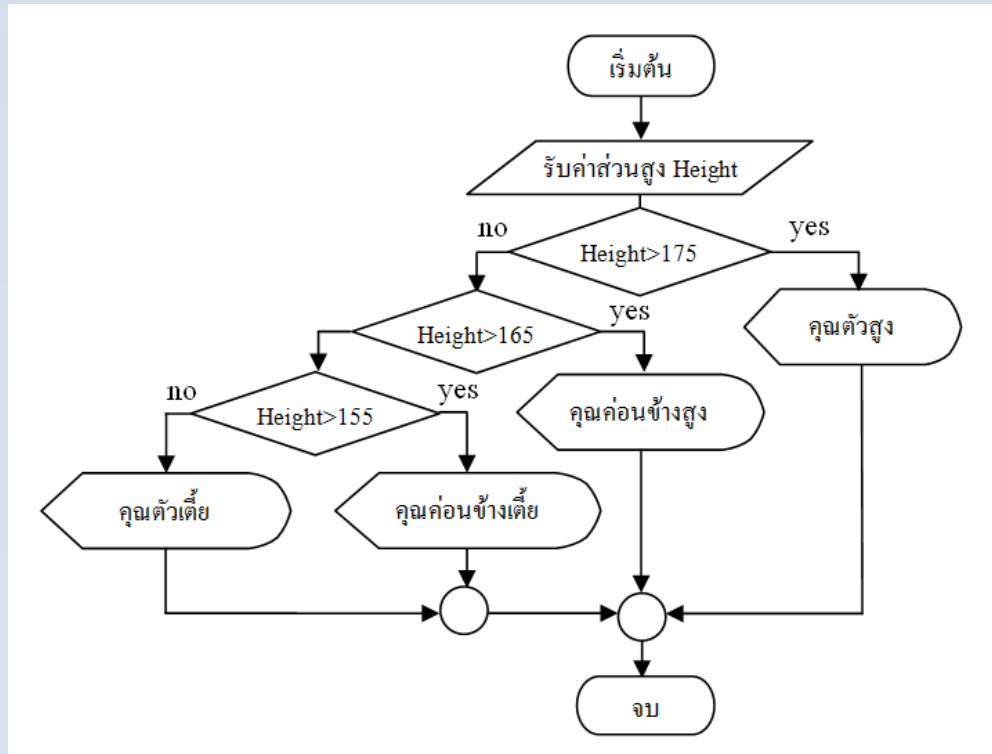
ตัวอย่างผังงานวนซ้ำแบบลูป do...while



การประยุกต์ใช้โครงสร้างแบบเลือกทำร่วมกับโครงสร้างแบบวนซ้ำในการหาโบนัสพนักงานทั้งหมด

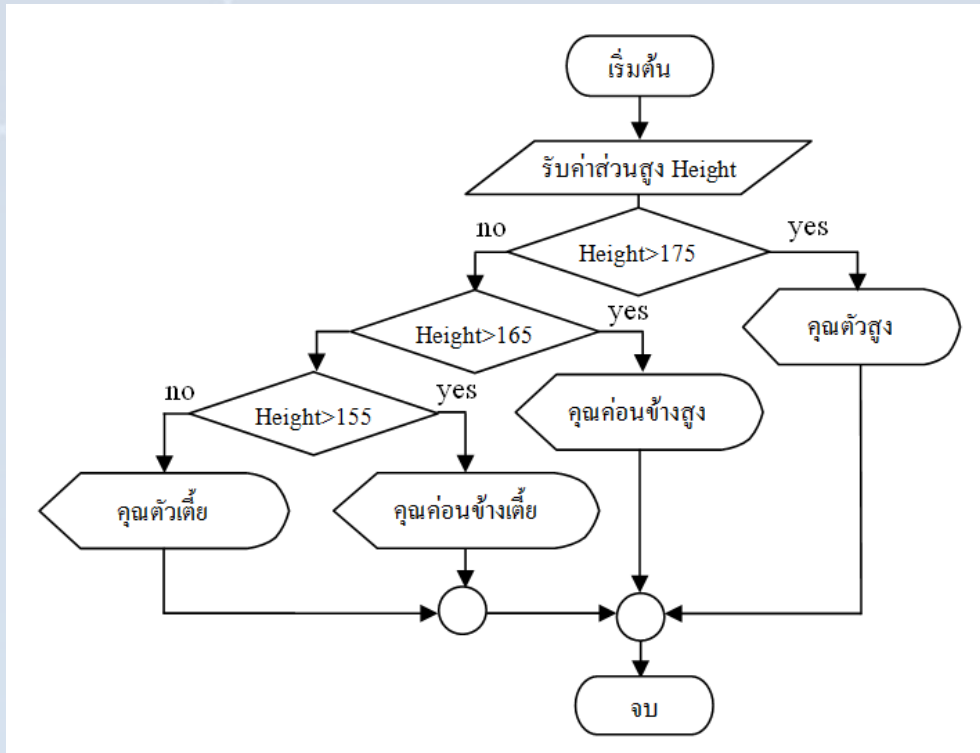


4. จากภาพหากกำหนดให้ค่า Height = 158 จะมีรูปแบบการแสดงผลอย่างไร

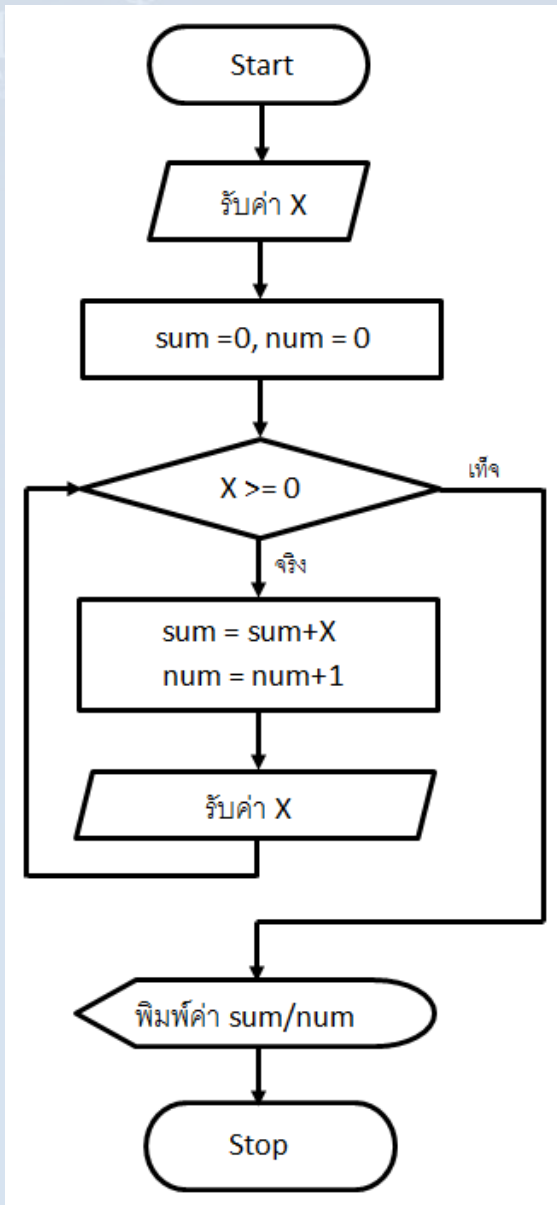


- ก) แสดงข้อความ “คุณตัวสูง” ทางจอภาพ
- ข) แสดงข้อความ “คุณค่อนข้างสูง” ทางเครื่องพิมพ์
- ค) แสดงข้อความ “คุณค่อนข้างเตี้ย” ทางจอภาพ
- ง) แสดงข้อความ “คุณตัวเตี้ย” ทางเครื่องพิมพ์





ผังงานที่กำหนดให้มีโครงสร้างการทำงานแบบหลายทางเลือก จะเริ่มต้นจากรับค่าส่วนสูงแล้วเก็บไว้ในตัวแปร Height จากนั้นนำตัวแปร Height ไปตรวจสอบเงื่อนไข โดยเริ่มจากเงื่อนไขที่ 1 คือ ตรวจสอบว่า $Height > 175$ หรือไม่ หากมากกว่าจะแสดงข้อความ “คุณตัวสูง” ออกทางจอภาพ แต่หากไม่ใช่ก็จะไปตรวจสอบเงื่อนไขที่ 2 ต่อ คือ $Height > 165$ หรือไม่ หากมากกว่า จะแสดงข้อความ “คุณค่อนข้างสูง” ออกทางจอภาพ หากไม่ใช่ให้ตรวจสอบเงื่อนไขที่ 3 ต่อ คือ $Height > 155$ หรือไม่ หากมากกว่าให้แสดงข้อความ “คุณค่อนข้างเตี้ย” แต่หากไม่ใช่ให้แสดงข้อความ “คุณตัวเตี้ย” ออกทางจอภาพ ***ผังงานที่กำหนดมีทั้งหมด 4 ทางเลือก***

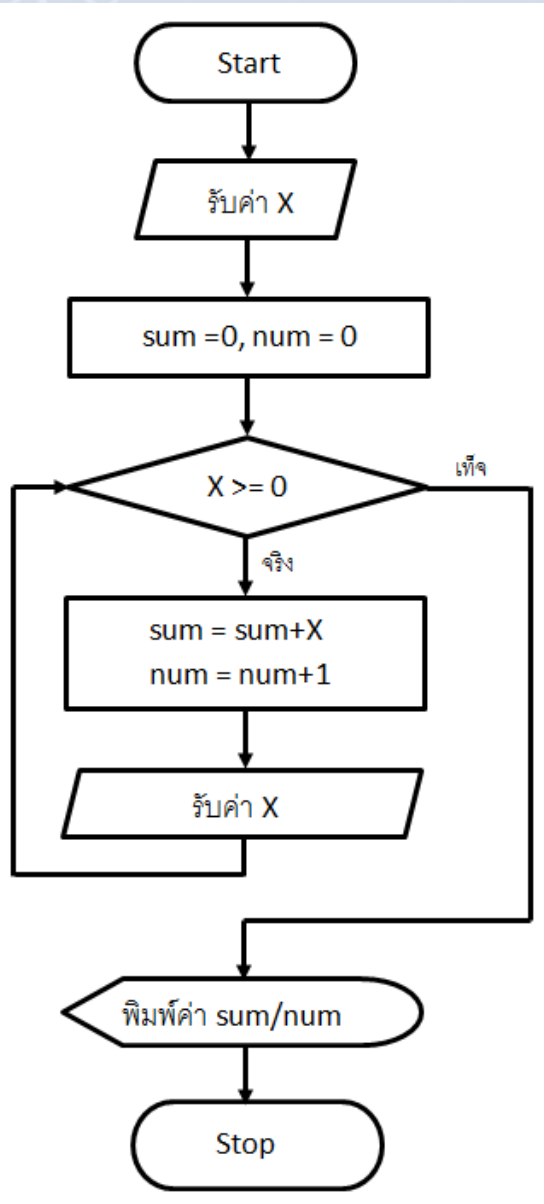


5. โครงสร้างการวนซ้ำในผังงานหยุดการวนซ้ำเมื่อใด

- ก) ผู้ใช้ป้อนข้อมูลครบ 5 รอบ
- ข) ผู้ใช้ป้อนข้อมูลที่เป็น 0
- ค) ผู้ใช้ป้อนข้อมูลที่เป็น -1
- ง) ผู้ใช้ไม่ป้อนข้อมูลเป็นเวลา 5 วินาที



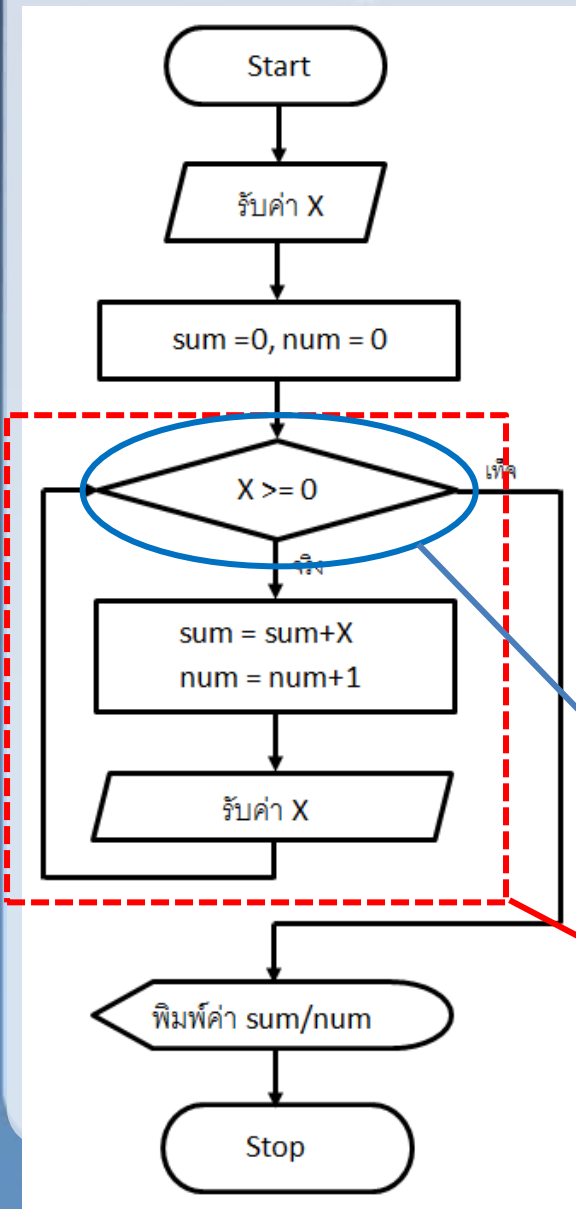
กระบวนการทำงานของผังงาน



1. เริ่มต้นการทำงาน
2. รับค่า x จากผู้ใช้งาน
3. กำหนดค่าเริ่มต้นให้ตัวแปร $sum=0$ และตัวแปร $num=0$
4. ตรวจสอบเงื่อนไขว่าค่า x ที่รับเข้ามามากกว่าหรือเท่ากับ 0 หรือไม่
5. กรณีค่า x มากกว่าหรือเท่ากับ 0 ให้คำนวณหาค่า sum และ num จากนั้นทำการรับค่า x ใหม่อีกครั้ง แล้ววนไปทำขั้นตอนที่ 4 ใหม่
6. กรณีค่า x น้อยกว่า 0 แสดงผลลัพธ์ที่ได้จาก sum/num
7. จบการทำงาน



คำอธิบายข้อ 5.

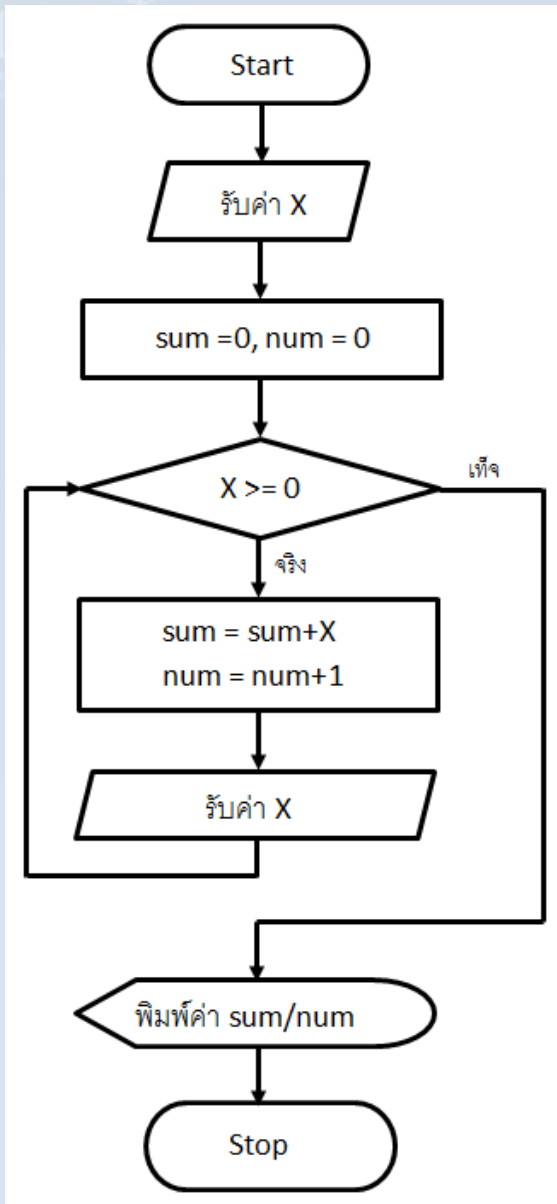


จากผังงานแสดงให้เห็นว่าการทำงานซ้ำของ
ผังงานจะทำเมื่อเงื่อนไขที่ตรวจสอบเป็นจริง
เท่านั้น นั่นคือ ค่า $x \geq 0$ ดังนั้น หากผู้ใช้
ป้อนค่า x ที่น้อยกว่า 0 ผังงานจะหยุด
กระบวนการทำงานซ้ำทันที ซึ่งในที่นี้ไม่
สามารถระบุได้ว่าการทำงานซ้ำนั้นจะทำจำนวน
กี่รอบ เพราะหากค่า x ที่ป้อนยังมากกว่าหรือ
เท่ากับ 0 อยู่ ผังงานก็ยังคงทำงานต่อไป
เรื่อยๆ

เงื่อนไขที่ใช้ในการทำงานซ้ำ คือค่า $x \geq 0$

วงรอบการทำงานซ้ำ





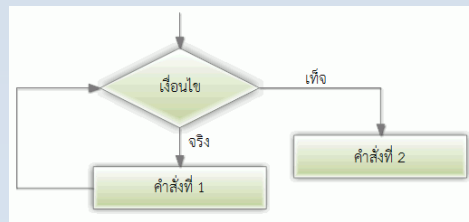
6. โครงสร้างดังกล่าวจัดอยู่ในกลุ่มใด

- ก) โครงสร้างการวนซ้ำแบบ while
- ข) โครงสร้างการวนซ้ำแบบ do...while
- ค) โครงสร้างทางเลือกแบบ if-then
- ง) โครงสร้างทางเลือกแบบ if-then-else



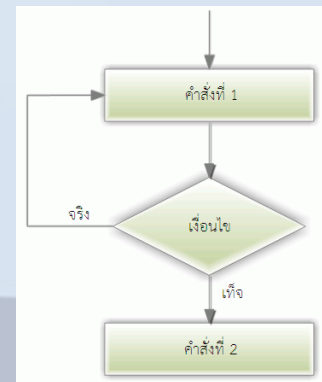
โครงสร้างการทำงานซ้ำแบบ while

การทำงานซ้ำแบบ while เมื่อเข้าไปทำงานในวงจรทำซ้ำจะเริ่มด้วยการตรวจสอบเงื่อนไขก่อน ถ้าเงื่อนไขเป็น “จริง” จะมีการทำซ้ำ แต่ถ้าเงื่อนไขเป็น “เท็จ” ก็จะเลิกทำคำสั่ง



โครงสร้างการทำงานซ้ำแบบ do...while

การทำงานซ้ำแบบ do...while เมื่อเข้าไปทำงานในวงจรทำซ้ำจะเริ่มด้วยการทำงานก่อนอย่างน้อย 1 รอบ จากนั้นค่อยทำการตรวจสอบเงื่อนไข ถ้าเงื่อนไขเป็น “จริง” จะมีการทำซ้ำรอบถัดไป แต่ถ้าเงื่อนไขเป็น “เท็จ” ก็จะเลิกทำคำสั่ง

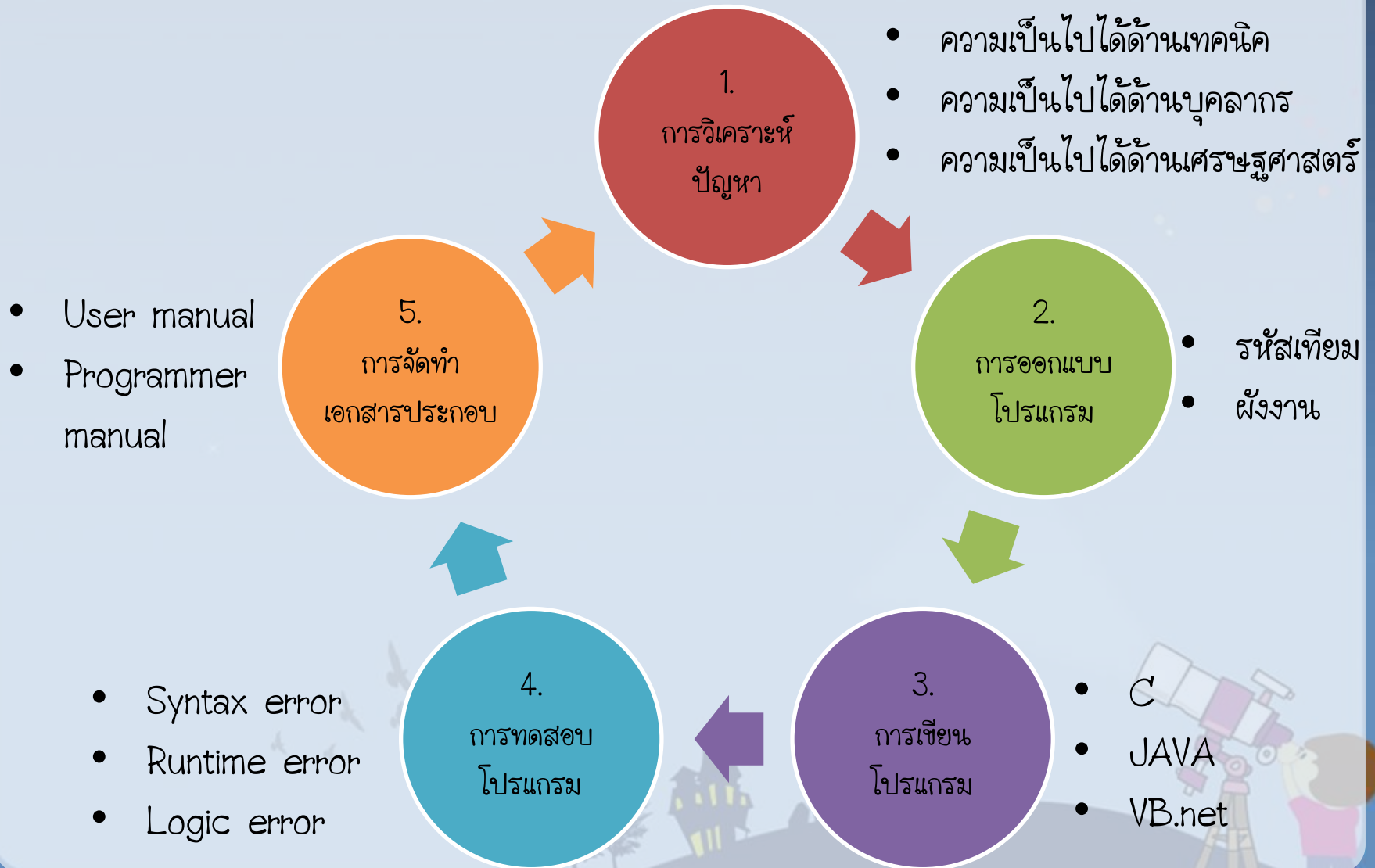


7. ขั้นตอนใดในวงจร SDLC ที่ถือว่ามีความสำคัญมากที่สุดในการพัฒนาโปรแกรม

- ก) การวิเคราะห์ปัญหา
- ข) การเขียนโปรแกรม
- ค) การทดสอบโปรแกรม
- ง) การจัดทำเอกสารประกอบโปรแกรม



วงจรการพัฒนาโปรแกรม (Development Life Cycle: SDLC)



8. จากระหัสเทียมที่กำหนดให้จะเกิดข้อผิดพลาดแบบใด

```
begin
    number1 = 50
    number2 = 0
    number3 = number1/number2
    print number3
end.
```

- ก) Syntax error
- ข) **Runtime error**
- ค) Logic error
- ง) Pseudo code error



ข้อผิดพลาดในการเขียนโปรแกรม

- 1. ข้อผิดพลาดทางไวยากรณ์ (Syntax error)** เป็นข้อผิดพลาดที่เกิดจากเขียนภาษาคอมพิวเตอร์ไม่ถูกต้องตามหลักไวยากรณ์ เช่น เขียนชื่อคำสั่งผิดหรือสะกดผิด และลืมใส่ () , { } เป็นต้น ข้อผิดพลาดรูปแบบนี้ตัวแปลภาษาจะแจ้งให้เราทราบ
- 2. ข้อผิดพลาดที่เกิดขึ้นขณะรันโปรแกรม (Runtime error)** เป็นข้อผิดพลาดที่เกิดขึ้นขณะการรันโปรแกรม ส่วนใหญ่เกิดจากการคำนวณตัวเลข เช่น การหารตัวเลขด้วยศูนย์ เช่น $10/0$ เป็นต้น ข้อผิดพลาดรูปแบบนี้ตัวแปลภาษาจะแจ้งให้เราทราบ
- 3. ข้อผิดพลาดทางตรรกะ (Logic error)** เป็นข้อผิดพลาดที่เกิดจากการเขียนสูตรคำนวณผิดและโปรแกรมให้ผลลัพธ์ที่ผิด เช่น ต้องการหาค่าเฉลี่ยของเลข 3 จำนวน แต่เขียนเป็น $x=a+b+c/3$ ซึ่งที่ถูกต้องคือ $x=(a+b+c)/3$ เป็นต้น ข้อผิดพลาดรูปแบบนี้ตัวแปลภาษาจะไม่แจ้งให้เราทราบต้องทำการตรวจสอบเอง เช่น นั่งดูโค้ดทั้งหมดอีกรอบเพื่อหาจุดที่ผิด



9. ในการเขียนโปรแกรมภาษา หากคำสั่งที่เขียนประกอบด้วยเลขฐานสอง (0 กับ 1) จัดเป็นภาษาคอมพิวเตอร์ประเภทใด

- ก) ภาษาเครื่อง
- ข) ภาษาระดับต่ำ
- ค) ภาษาระดับกลาง
- ง) ภาษาระดับสูง



ภาษาคอมพิวเตอร์ (Computer Language)

คือ ภาษาที่ใช้กับคอมพิวเตอร์หรือใช้เพื่อสื่อสารให้คอมพิวเตอร์เข้าใจ และสามารถปฏิบัติตามคำสั่งได้ ในการเลือกภาษาคอมพิวเตอร์สำหรับเขียนโปรแกรมนั้นผู้เขียนจะต้องคำนึงถึงปัจจัยที่สำคัญ คือ ลักษณะของปัญหา ความถนัดของผู้เขียนเอง สภาพแวดล้อมในการทำงานของระบบ นโยบายของหน่วยงาน เป็นต้น





ประกอบด้วยเลขฐานสอง คือ 0 กับ 1
เป็นภาษาที่เครื่องคอมพิวเตอร์สามารถเข้าใจ
ได้ทันที



ลักษณะเหมือนภาษาอังกฤษทั่วไป
เช่น C, Pascal, BASIC, C++



ใช้สัญลักษณ์ ข้อความแทนเลขฐานสอง
เช่น แอสเซมบลี



10. คำสั่งใดในภาษาซีใช้สำหรับการรับข้อมูลเข้า

- ก) printf
- ข) scanf
- ค) main
- ง) stdio.h



โครงสร้างของภาษาซี

โครงสร้างพื้นฐานของภาษาซีจะต้องประกอบด้วยฟังก์ชันอย่างน้อย 1 ฟังก์ชัน

```
1 int main() {  
2     ประกาศตัวแปร;  
3     คำสั่งต่างๆ;  
4 }
```

ฟังก์ชัน **main()** ประกอบด้วย 3 ส่วนหลัก

1. ส่วนหัวของฟังก์ชัน บรรทัดที่ 1
2. ส่วนการประกาศตัวแปร บรรทัดที่ 2
3. ส่วนคำสั่ง บรรทัดที่ 3

****** ขอบเขตการทำงานของฟังก์ชันจะอยู่ระหว่าง { และ }



องค์ประกอบของภาษาซี

การประกาศรวมแฟ้มส่วนหัว

- `#include<stdio.h>`
- `#include<conio.h>`

การประกาศตัวแปร

- `int score;`
- `char name[30];`

การรับข้อมูลเข้า

- `scanf("%d",&score);`
- `scanf("%s",name);`

การพิมพ์ผลลัพธ์

- `printf("Score = %d ",score);`
- `printf("Welcome to my program");`

องค์ประกอบของภาษาซี

คำสั่งการกำหนดค่าให้ตัวแปร

- `score = 85;`
- `name = "Yothinburana"`

คำสั่งควบคุมการทำงานแบบวนซ้ำ

- `while`
- `do...while`

คำสั่งควบคุมการทำงานแบบมีทางเลือก

- `if`
- `if- else`
- `if- else if - else`

คำสั่ง `Return`

- `return 0;`

11. คำใดในภาษาซีใช้ในการระบุการประกาศตัวแปรชนิดจำนวนเต็ม

ก) integer

ข) int

ค) float

ง) char



ชนิดของตัวแปร

ชนิด	คำสั่ง (เขียนโค้ด)	รูปแบบการแสดงผล (%)
จำนวนเต็ม	<code>int</code>	<code>%d</code>
ทศนิยม	<code>float</code>	<code>%f</code>
อักขระ	<code>char</code>	<code>%c</code>
สายอักขระ	<code>char</code> ตัวแปร[ขนาด]	<code>%s</code>



การตั้งชื่อตัวแปร

1. ประกอบด้วย a-z, A-Z, 0-9, _ เท่านั้น
2. อักขระตัวแรกต้องเป็น a-z, A-Z และ _
3. ห้ามใช้ชื่อเฉพาะ
4. ตัวอักษรพิมพ์เล็กและพิมพ์ใหญ่มีความหมายที่แตกต่างกัน
5. ยาวสูงสุด 31 ตัวอักษร



12. เครื่องหมายในข้อใดไม่ใช่เครื่องหมายที่ใช้ในการเปรียบเทียบ

ก) $>=$

ข) $<=$

ค) $=$

ง) $!=$



เครื่องหมายในการเปรียบเทียบทางคณิตศาสตร์

ตัวดำเนินการ	ความหมาย	ตัวอย่าง
$=$	เท่ากับ	$x = y$
$>$	มากกว่า	$x > y$
$<$	น้อยกว่า	$x < y$
\geq	มากกว่าหรือเท่ากับ	$x \geq y$
\leq	น้อยกว่าหรือเท่ากับ	$x \leq y$
\neq	ไม่เท่ากับ	$x \neq y$



```

1  #include<stdio.h>
2  int main() {
3      int a,b;
4      scanf("%d",&a);
5      scanf("%d",&b);
6      if(a==b)
7          printf("The two number are equal.\n");
8      else
9          printf("The two number are different.\n");
10 }

```

13. คำสั่งในข้อใดจะไม่ถูกเรียกใช้งานพร้อมกันในการรันโปรแกรมครั้งหนึ่งๆ

- ก) บรรทัดที่ 4 และ 5
- ข) บรรทัดที่ 6 และ 7
- ค) บรรทัดที่ 6 และ 8
- ง) บรรทัดที่ 7 และ 9



```

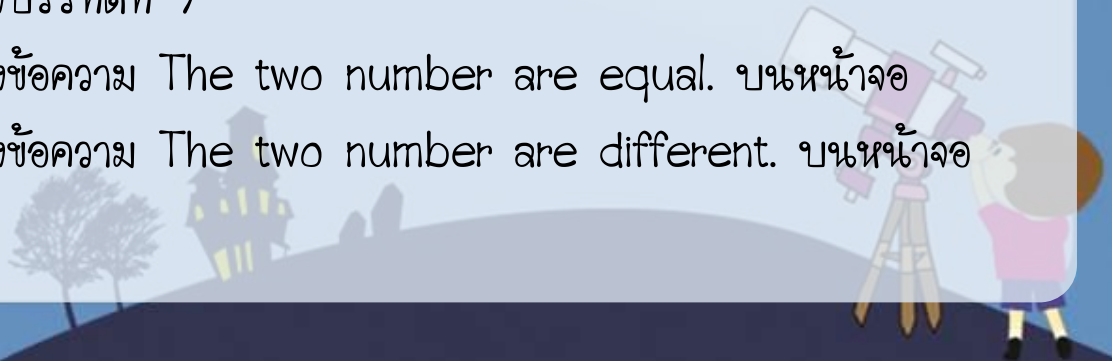
1  #include<stdio.h>
2  int main() {
3      int a,b;
4      scanf("%d",&a);
5      scanf("%d",&b);
6      if(a==b)
7          printf("The two number are equal.\n");
8      else
9          printf("The two number are different.\n");
10 }

```

โครงสร้างแบบ 2 ทางเลือก
if - else

จากโปรแกรมที่กำหนดให้มีโครงสร้างการทำงานแบบ 2 ทางเลือก

- | | |
|---------------|--|
| บรรทัดที่ 3 | ประกาศตัวแปรชนิดจำนวนเต็ม 2 ตัว คือ a และ b |
| บรรทัดที่ 4,5 | รับค่าตัวเลขตัวที่ 1 เก็บใน a ตัวที่ 2 เก็บใน b |
| บรรทัดที่ 6 | เปรียบเทียบค่า a และ b เท่ากันหรือไม่ โดยแบ่งเป็น 2 กรณี
ดังนี้ กรณี $a = b$ ให้ทำคำสั่งบรรทัดที่ 7 และกรณี $a \neq b$ ให้ทำ
คำสั่งบรรทัดที่ 9 |
| บรรทัดที่ 7 | แสดงข้อความ The two number are equal. บนหน้าจอ |
| บรรทัดที่ 9 | แสดงข้อความ The two number are different. บนหน้าจอ |



14. ทำการแปลภาษาและตรวจสอบคำสั่งที่ละบรรทัด คือการทำงานของตัวแปลภาษาแบบใด

- ก) Compiler
- ข) Interpreter
- ค) Assembly
- ง) ถูกทุกข้อ



ประเภทของตัวแปลภาษา

1. **Assembly** >> เป็นตัวแปลภาษาแอสแซมบลีซึ่งเป็นภาษาระดับต่ำทำให้เป็นภาษาเครื่อง
2. **Interpreter** >> เป็นตัวแปลภาษาระดับสูงไปเป็นภาษาเครื่อง โดยใช้หลักการแปลพร้อมกันทำงานตามคำสั่งทีละบรรทัดตลอดทั้งโปรแกรมทำให้การแก้ไขโปรแกรมทำได้ง่ายและรวดเร็ว
3. **Compiler** >> เป็นตัวแปลภาษาระดับสูงเช่นเดียวกับอินเตอร์พรีเตอร์แต่จะใช้วิธีแปลโปรแกรมทั้งโปรแกรมให้เป็นออบเจกต์โค้ด



15. ภาษาใดไม่ได้เป็นภาษาที่ใช้เขียนโปรแกรมสั่งงานคอมพิวเตอร์ที่แท้จริง

- ก) Java
- ข) C
- ค) HTML
- ง) SQL



Java >>> ภาษาที่ใช้ในการเขียนโปรแกรมสั่งงานคอมพิวเตอร์ ซึ่งเป็นลักษณะภาษาแบบ Object Oriented Language คือการเขียนโปรแกรมเชิงวัตถุ ช่วยอำนวยความสะดวกให้ผู้เขียนโปรแกรมสามารถเขียนโปรแกรมง่ายขึ้น เหมาะกับการพัฒนาโปรแกรมขนาดใหญ่

C >>> ภาษาที่ใช้ในการเขียนโปรแกรมสั่งงานคอมพิวเตอร์ ซึ่งเป็นลักษณะภาษาเชิงกระบวนการ (Procedural Language) มีลักษณะการทำงานตามลำดับของคำสั่ง จากคำสั่งแรกจนถึงคำสั่งสุดท้าย เหมาะสำหรับผู้ที่เริ่มต้นในการเขียนโปรแกรม

HTML >>> ภาษาที่ใช้ในการจัดรูปแบบแสดงผลของข้อความและรูปภาพ รวมถึงสื่อประสมบนหน้าเว็บ ไม่จัดเป็นภาษาที่ใช้ในการเขียนโปรแกรม

SQL >>> ภาษาที่ใช้ในการเขียนโปรแกรมสั่งงานคอมพิวเตอร์ โดยใช้ในการเข้าถึงข้อมูลในฐานข้อมูล และการสืบค้นข้อมูลจากฐานข้อมูล

