

ปฏิบัติการครั้งที่ 4

คำสั่งควบคุม (Nested If, Switch)

โจทย์

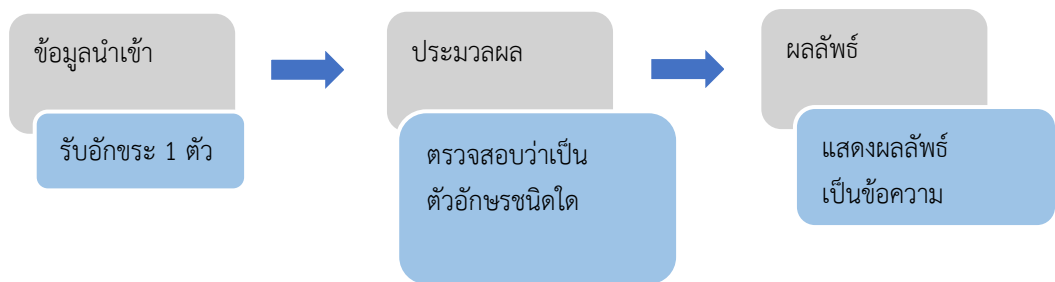
1. โปรแกรมตรวจสอบตัวอักษร (Character)

จงเขียนขั้นตอน (Algorithm) “โดยใช้คำสั่ง Switch Case” เพื่อตรวจสอบว่าอักขระที่รับเข้ามาเป็นตัวอักษรชนิดใด พยัญชนะหรือสระ โดยรับค่าอักขระ 1 ตัวจากคีย์บอร์ด แล้วแสดงผลของการตรวจสอบออกทางจอภาพ

โดยกำหนดให้ อักขระที่รับเข้ามา ถ้าเป็นตัวอักษร มี 2 ชนิด ดังนี้

- สระ (Vowel) คือ ตัวอักษร a, e, i, o, u หรือ A, E, I, O, U
- พยัญชนะ (Consonant) คือ ตัวอักษร a-z หรือ A-Z ยกเว้นตัวที่เป็นสระ

1.1 วิเคราะห์โจทย์



1.2 ผลการรัน

ข้อมูลนำเข้า รับตัวอักขระ 1 ตัว (ch) โดยที่ $a \leq ch \leq Z$

ผลลัพธ์ ระบุประเภทของตัวอักษร [Vowel/Consonant]

ตัวอย่างที่ 1

ข้อมูลนำเข้า	ผลลัพธ์
U	Vowel

ตัวอย่างที่ 2

ข้อมูลนำเข้า	ผลลัพธ์
a	Vowel

ตัวอย่างที่ 3

ข้อมูลนำเข้า	ผลลัพธ์
f	Consonant

2. โปรแกรมคำนวณเลข (Calculator)

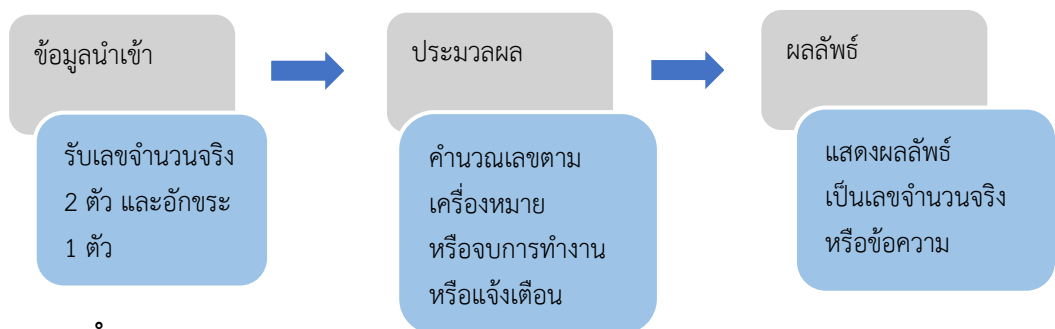
จงเขียนขั้นตอน (Algorithm) “โดยใช้คำสั่ง Switch” เพื่อคำนวณค่าเลขจำนวนจริง 2 ตัว ตามเครื่องหมายที่รับเข้ามา โดยรับค่าเลขจำนวนจริง 2 ตัว และอักขระ 1 ตัวจากคีย์บอร์ด แล้วแสดงผลลัพธ์ของการคำนวณออกทางจอภาพ

โดยกำหนดให้ อักขระที่รับเข้ามา ถ้าเป็นมี 5 ชนิด ดังนี้

- + แทนเครื่องหมายบวก
- - แทนเครื่องหมายลบ
- * แทนเครื่องหมายคูณ
- / แทนเครื่องหมายหาร
- > แทนการจบการทำงาน ให้ออกจากโปรแกรม
- อักขระอื่น ๆ ให้แสดงผลว่า ผิดพลาด (Invalid mark!!)

และกำหนดให้ในการคำนวณนั้น จำนวนเต็มตัวแรกที่รับเข้ามาเป็นตัวตั้งเสมอ

2.1 วิเคราะห์โจทย์



2.2 ขั้นตอนการทำงาน

- 1) เริ่มต้น
- 2) กำหนดตัวแปรชื่อ num1 และ num2 สำหรับเก็บเลขจำนวนเต็ม 2 ตัว
- 3) กำหนดตัวแปรชื่อ mark สำหรับเก็บเครื่องหมาย ซึ่งเป็นตัวอักขระ 1 ตัว (Character)
- 4) รับเลขจำนวนเต็ม 2 ตัว และอักขระ 1 ตัว ใส่ในตัวแปร num1, num2 และ mark ตามลำดับ
- 5) ตรวจสอบว่าเป็นเครื่องหมายอะไร
 - ถ้ารับอักขระเป็นเครื่องหมาย + - * / เครื่องหมายใดเครื่องหมายหนึ่ง ให้คำนวณเลขจำนวนเต็ม 2 ตัวด้วยเครื่องหมายนั้น และแสดงผลออกทางจอภาพ
 - ถ้ารับอักขระเป็นเครื่องหมาย > ให้จบการทำงาน โดยออกจากโปรแกรม
 - ถ้ารับอักขระเป็นตัวอื่น ให้แสดงว่าผลลัพธ์ว่าผิดพลาด (Invalid mark!!) ออกทางจอภาพ และออกจากโปรแกรม
- 6) จบการทำงาน

2.3 ผลการรัน

ข้อมูลนำเข้า รับเลขจำนวนเต็ม 2 ตัว และรับตัวอักขระ 1 ตัว ดังนี้

บรรทัดแรก รับค่าเลขจำนวนเต็มตัวที่ 1 (num1) โดยที่ $0 < \text{num1} < 100,000$

บรรทัดที่สอง รับค่าเลขจำนวนเต็มตัวที่ 2 (num2) โดยที่ $0 < \text{num2} < 100,000$

บรรทัดที่สาม รับค่าตัวอักขระ 1 ตัว (mark)

ผลลัพธ์ ขึ้นอยู่กับอักขระที่รับเข้ามา ดังนี้

- ถ้ารับอักขระเป็นเครื่องหมาย + - * / ให้ระบุผลลัพธ์การคำนวณเลขตามเครื่องหมาย + - * / ที่รับเข้ามา เป็นเลขทศนิยม
- ถ้ารับอักขระเป็นเครื่องหมาย > ให้จบการทำงาน โดยออกจากโปรแกรม
- ถ้ารับอักขระเป็นตัวอื่น ให้แจ้งเป็นข้อความว่า “Invalid mark!!” และออกจากโปรแกรม

ตัวอย่างที่ 1

ข้อมูลนำเข้า	ผลลัพธ์
25 50 /	0.5

ตัวอย่างที่ 2

ข้อมูลนำเข้า	ผลลัพธ์
5 5 \$	Invalid mark!! และออกจากโปรแกรม

ตัวอย่างที่ 3

ข้อมูลนำเข้า	ผลลัพธ์
1000 200 >	ไม่มีค่าผลลัพธ์ และออกจากโปรแกรม

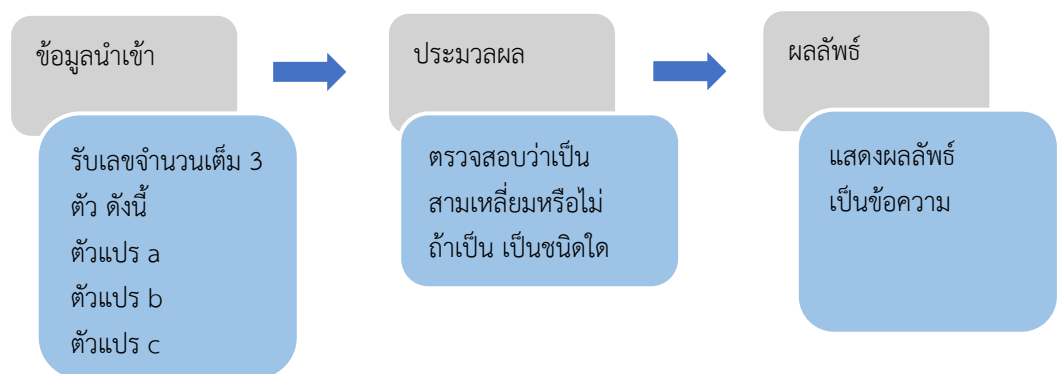
3. โปรแกรมตรวจสอบชนิดของสามเหลี่ยม (triangle)

จงเขียนขั้นตอน (Algorithm) เพื่อตรวจสอบว่าเป็นสามเหลี่ยมหรือไม่ และถ้าเป็น เป็นชนิดใด โดยรับค่าจำนวนเต็ม 3 ตัวจากคีย์บอร์ดเป็นด้านทั้ง 3 ด้านของสามเหลี่ยม แล้วแสดงผลลัพธ์ของการตรวจสอบออกทางจอภาพ

โดยกำหนดให้ ด้าน 3 ด้าน (a, b, c) สามารถสร้างเป็นสามเหลี่ยมได้ ก็ต่อเมื่อ 2 ด้านใด ๆ รวมกันแล้วต้องมากกว่า ด้านที่ 3 และถ้าเป็นสามเหลี่ยม ให้ระบุว่าเป็นชนิดใด ซึ่งมี 3 ชนิด ได้แก่

- หน้าจั่ว (Isosceles) คือ มีด้าน 2 ด้านเท่ากันเท่านั้น
- ด้านเท่า (Equilateral) คือ มีด้าน 3 ด้านเท่ากัน
- ด้านไม่เท่า (Scalene) คือ มีด้านทุกด้านไม่เท่ากัน

3.1 วิเคราะห์โจทย์



3.2 ผลการรัน

ข้อมูลนำเข้า รับเลขจำนวนเต็ม 3 ตัว ดังนี้

บรรทัดแรก รับค่าด้านของสามเหลี่ยมด้านที่ 1 เก็บเลขจำนวนเต็ม

บรรทัดที่สอง รับค่าด้านของสามเหลี่ยมด้านที่ 2 เก็บเลขจำนวนเต็ม

บรรทัดที่สาม รับค่าด้านของสามเหลี่ยมด้านที่ 3 เก็บเลขจำนวนเต็ม

ผลลัพธ์ ระบุประเภทของสามเหลี่ยม [Isosceles/Equilateral/Scalene] แต่ถ้าไม่เป็นให้ระบุว่าเป็น (No)

ตัวอย่างที่ 1

ข้อมูลนำเข้า	ผลลัพธ์
14 14 10	Isosceles

ตัวอย่างที่ 2

ข้อมูลนำเข้า	ผลลัพธ์
5	No
10	
15	

ตัวอย่างที่ 3

ข้อมูลนำเข้า	ผลลัพธ์
21	Scalene
22	
23	