

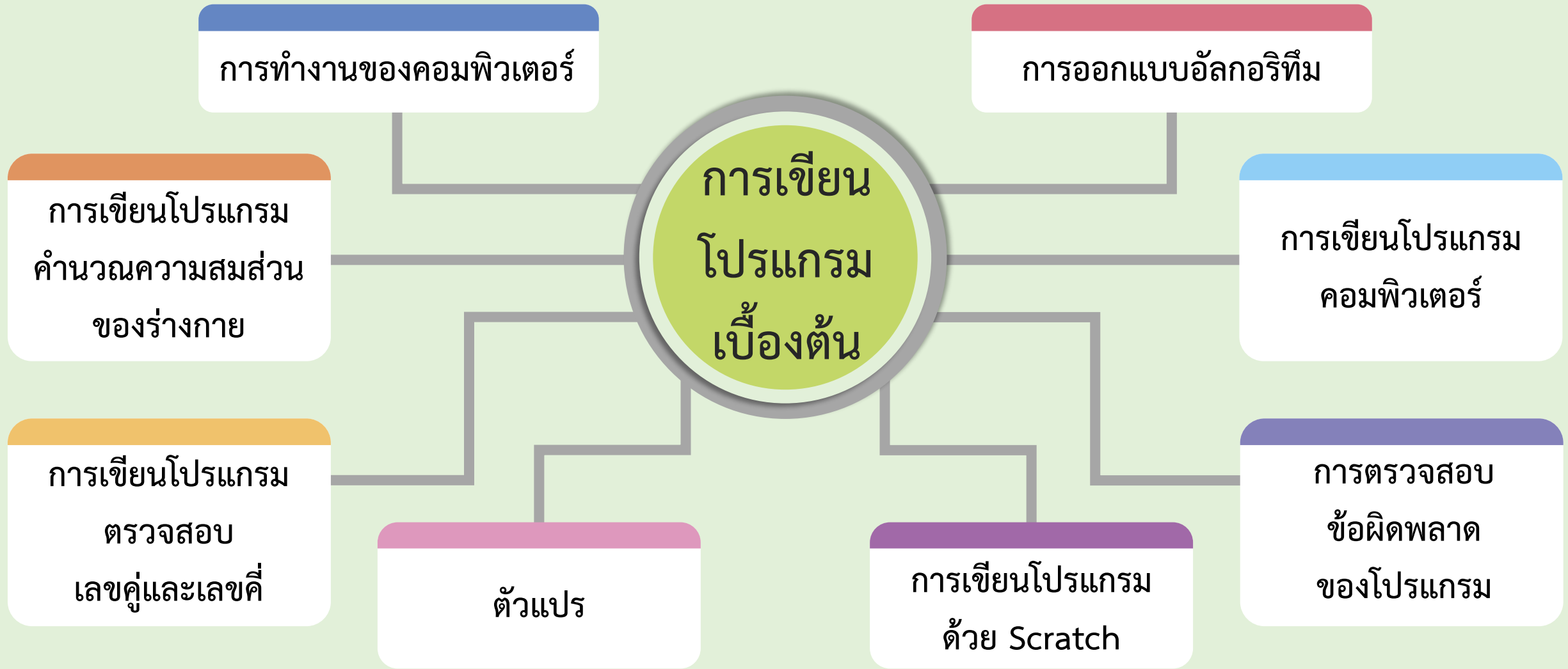


+

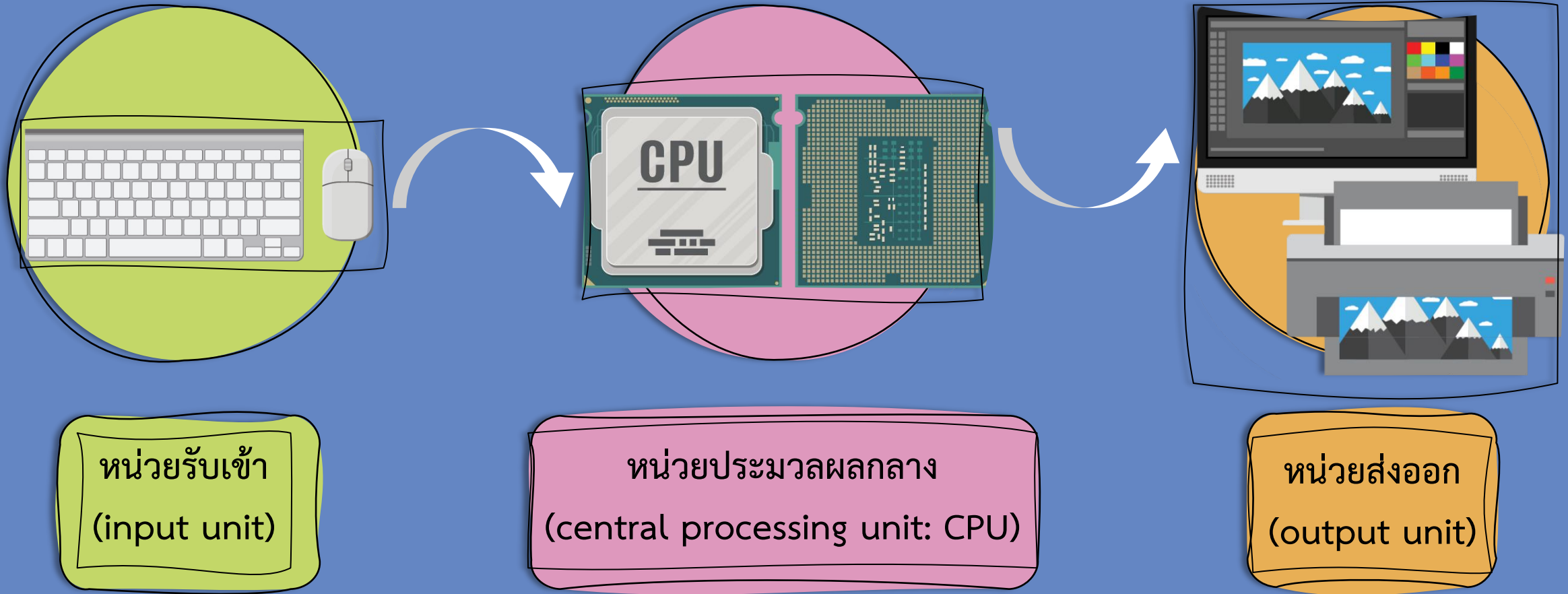
MISS SIRILUCK LEARTHIRUNSAP



การเขียนโปรแกรมเบื้องต้น

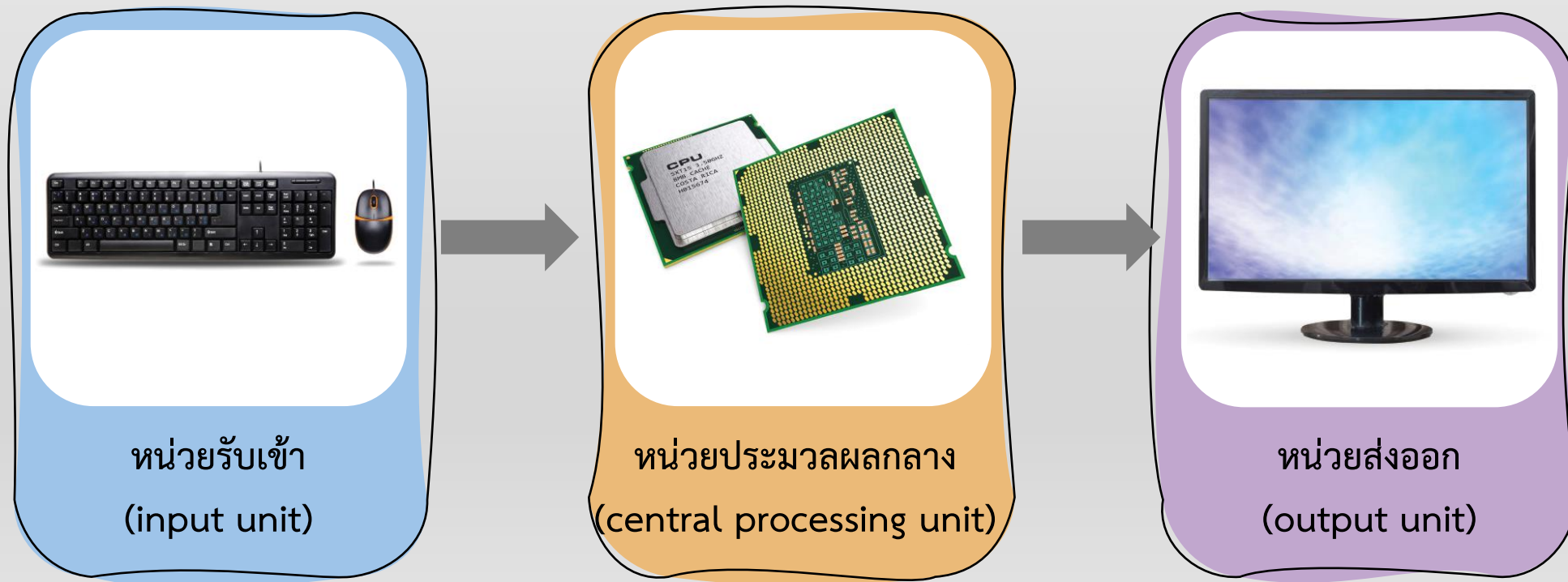


การทำงานของคอมพิวเตอร์



การทำงานของคอมพิวเตอร์

การทำงานของคอมพิวเตอร์จะรับข้อมูลเข้ามา จากนั้นประมวลผลแล้วส่งข้อมูลออกไป โดยการรับข้อมูล จะรับผ่านทางหน่วยรับเข้า เช่น แป้นพิมพ์ เมาส์ แล้วมาประมวลผลด้วยหน่วยประมวลผลกลางหรือซีพียู แล้วส่งข้อมูลออกมาทางหน่วยส่งออก เช่น จอภาพ เครื่องพิมพ์ ลำโพง





การทำงานของคอมพิวเตอร์

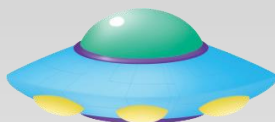
คอมพิวเตอร์จะทำงานได้จะต้องมีโปรแกรมเป็นตัวสั่งงาน ถ้าต้องการพัฒนาโปรแกรมขึ้นมา จะมีสองขั้นตอนหลัก คือ

ขั้นตอนแรก ต้องนึกถึงสิ่งที่ต้องการให้คอมพิวเตอร์ทำงาน โดยกำหนดออกมาเป็นขั้นตอนการทำงาน

ขั้นตอนที่สอง เปลี่ยนจากขั้นตอนการทำงานให้เป็นภาษาที่คอมพิวเตอร์เข้าใจ

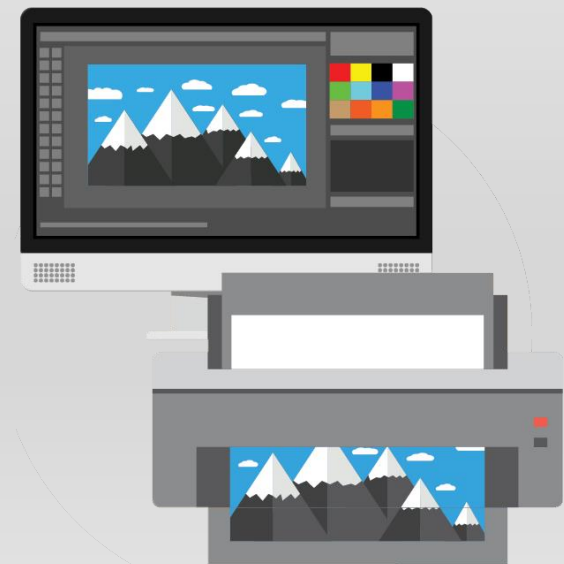
ลำดับขั้นตอนการทำงานของคอมพิวเตอร์จะคล้ายกับการทำงานของมนุษย์ เมื่อมนุษย์จะทำงานอย่างใดอย่างหนึ่งต้องมีลำดับการทำงานซึ่งออกแบบไว้อย่างเป็นขั้นตอนเช่นเดียวกับคอมพิวเตอร์ ถ้าต้องการให้ทำงานลักษณะใดจะต้องมีโปรแกรมที่ออกแบบการทำงานอย่างเป็นขั้นตอนมาให้คอมพิวเตอร์ทำงานตามที่ได้ออกแบบไว้

การออกแบบขั้นตอนการทำงานเบื้องต้นให้กับคอมพิวเตอร์ เราอาจพิจารณาว่า ต้องการให้คอมพิวเตอร์รับค่าใดเข้าไป แล้วต้องการให้ประมวลผลอย่างไร ผลลัพธ์ที่ได้ออกมาเป็นแบบใด



คำถามสำคัญ

การทำงานของคอมพิวเตอร์ หากเปรียบเทียบกับ
การทำงานของร่างกายเราจะเปรียบได้กับอวัยวะ
ใดบ้าง

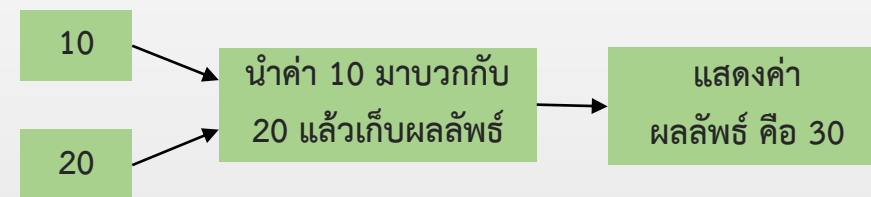




ตัวอย่าง การออกแบบขั้นตอนการบวกเลข

ตัวอย่าง ถ้าต้องการออกแบบโปรแกรมให้คอมพิวเตอร์ทำงาน คล้ายเครื่องคิดเลข โดยบวกตัวเลขสองจำนวน ระหว่าง 10 กับ 20 ออกแบบขั้นตอนการทำงานได้ ดังนี้

การรับข้อมูลจากหน่วยรับเข้า เช่น แป้นพิมพ์ มาเก็บในเครื่องคอมพิวเตอร์ จะต้องสร้างตัวแปรขึ้นมา จากตัวอย่างนี้ จะต้องมิตัวแปรสองตัวเพื่อเก็บข้อมูลสองค่า โดยการออกแบบขั้นตอน

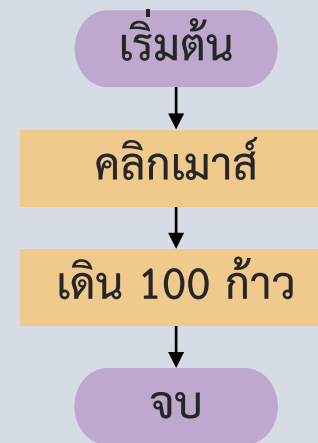




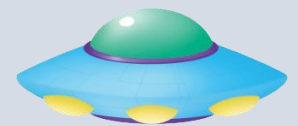
การออกแบบอัลกอริทึม

การเริ่มเขียนโปรแกรมต้องออกแบบการทำงานของโปรแกรมขึ้นมาก่อน หรือเรียกว่า การออกแบบอัลกอริทึม ทำได้โดยสร้างลำดับของคำสั่งให้คอมพิวเตอร์ทำงาน เช่น ถ้าหากต้องการให้ตัวละครเคลื่อนที่ 100 ก้าวอาจทำได้ ดังนี้

คอมพิวเตอร์จะทำงานตามคำสั่งหรืออัลกอริทึมที่กำหนดการให้คอมพิวเตอร์ทำงานอย่างใดอย่างหนึ่งอย่างถูกต้องนั้น ต้องออกแบบอัลกอริทึมให้สมบูรณ์ โดยตรวจสอบเงื่อนไขให้ครอบคลุมทุกกรณี แล้วจึงพัฒนาเป็นโปรแกรมคอมพิวเตอร์ ถ้าทำงานผิดพลาดหรือมีส่วนใดที่ทำงานไม่สมบูรณ์ เราสามารถปรับปรุงอัลกอริทึมได้ แล้วจึงแก้ไขโปรแกรมต่อไป



อัลกอริทึมมีความสำคัญอย่างไรกับการออกแบบการทำงานของโปรแกรม



คำถามสำคัญ

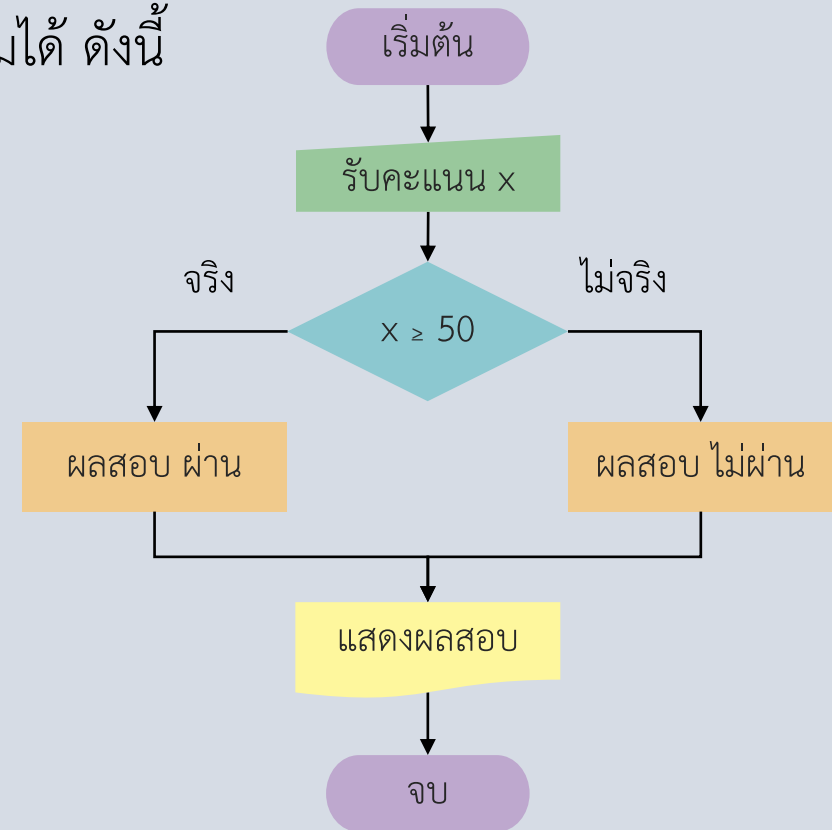
ตัวอย่าง

ถ้าต้องการออกแบบอัลกอริทึมสำหรับประมวลผลการสอบ โดยวิชาที่สอบจะมีคะแนนเต็ม 100 คะแนน ถ้าทำคะแนนได้มากกว่าหรือเท่ากับ 50 คะแนนจะผ่าน ดังนั้นเงื่อนไขของปัญหานี้ คือ ถ้าคะแนนมากกว่าหรือเท่ากับ 50 คะแนน ผ่าน และ ถ้าคะแนนตั้งแต่ 0 ถึง 49 จะไม่ผ่านนั่นเอง ปัญหานี้ข้อมูลรับเข้า คือ คะแนน ส่วนข้อมูลส่งออก คือ ผลการสอบ เราอาจเขียนอัลกอริทึมได้ ดังนี้

เริ่มต้น

1. รับค่าคะแนนมาเก็บใน x
2. ถ้า $x \geq 50$
ผลสอบผ่าน
3. ถ้า $x < 50$
ผลสอบไม่ผ่าน
4. แสดงผลสอบ

จบ



การออกแบบอัลกอริทึมควรเขียนการทำงานหลัก ๆ ออกมาก่อน สำหรับการทำงานย่อย ๆ อาจแทรกเข้าไปภายหลังได้

วัตถุประสงค์

คิดเปรียบเทียบความสัมพันธ์ของปัญหา แล้วสรุปผลได้

วิธีปฏิบัติ

นักเรียนวิเคราะห์ตัวอย่างผังงานอัลกอริทึมการประมวลผล คะแนนสอบ แล้วตอบคำถาม ดังนี้

1. จากอัลกอริทึมการประมวลผลการสอบ นักเรียนคิดว่า ถ้าคะแนน x เป็นข้อมูลที่ผู้ใช้ป้อนเข้าไปทางแป้นพิมพ์ ของคอมพิวเตอร์ดังตารางต่อไปนี้ ผลการสอบ จะเป็นอย่างไร
2. นักเรียนมีแนวคิดการปรับปรุงอัลกอริทึม การประมวลผลการสอบอย่างไร

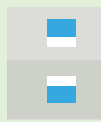
วัสดุอุปกรณ์

1. สมุดสำหรับจดบันทึก หรือแบบฝึกหัด 1 เล่ม
2. ปากกา 1 ด้าม

ตอบ 1.

คะแนนการสอบ (X)	ผลการสอบ (ผ่าน/ไม่ผ่าน)
89	ผ่าน
100	ผ่าน
50	ผ่าน
49	ไม่ผ่าน
45	ไม่ผ่าน

2. การปรับปรุงอัลกอริทึมควรเขียนการทำงานหลัก ๆ ออกมาก่อน สำหรับการงานย่อย ๆ อาจแทรกเข้าไปภายหลังได้

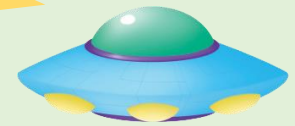


การเขียนโปรแกรมคอมพิวเตอร์

การเขียนโปรแกรมคอมพิวเตอร์เป็นการสร้างลำดับของคำสั่งให้คอมพิวเตอร์ทำงาน เพื่อให้ได้ผลลัพธ์ตามที่ต้องการ โดยทั่วไปการเขียนโปรแกรมจะต้องออกแบบอัลกอริทึมขึ้นมาก่อน แล้วนำมาเขียนเป็นคำสั่งคอมพิวเตอร์ โดยอัลกอริทึมจะต้องมีการตรวจสอบเงื่อนไขที่ครอบคลุมทุกกรณี จากตัวอย่างอัลกอริทึมการประมวลผลการสอบเป็นตัวอย่างที่มีเงื่อนไขไม่ซับซ้อน ถ้าเป็นปัญหาที่ซับซ้อน อัลกอริทึมจะซับซ้อนตามไปด้วย สำหรับการเขียนคำสั่งให้โปรแกรมทำงานทำได้หลายวิธี เช่น เว็บไซต์สำหรับฝึกเขียนโปรแกรม เช่น www.code.org ใช้ซอฟต์แวร์สำหรับเขียนโปรแกรม เช่น Scratch หรือการเขียนโปรแกรมด้วยภาษาระดับสูง โดยเขียนตามอัลกอริทึมที่ได้ออกแบบไว้



นักเรียนจะเขียนโปรแกรมอย่างไรไม่ให้เกิด
ความผิดพลาดหรือถ้าเกิดก็มีความผิดพลาดน้อยที่สุด



คำถามสำคัญ

วัตถุประสงค์

เขียนโปรแกรมโดยใช้เหตุผลเชิงตรรกะอย่างง่ายได้

วัสดุอุปกรณ์

คอมพิวเตอร์ 1 เครื่อง

วิธีปฏิบัติ

1. นักเรียนเปิดเว็บไซต์ <https://code.org>
2. เลือกหมวดนักเรียน แล้วเลือกคอร์ส 3

Course Catalog | ไทยแลนด์

เริ่มต้นการเรียนรู้ที่เลือก

ดูวิดีโอ

Computer science empowers students to create the world of tomorrow!

Satya, CEO of Microsoft

นักเรียนทุกคนจากทุกโรงเรียนควรมีโอกาสที่จะได้เรียนวิทยาการคอมพิวเตอร์

คลิกเพื่อดู -

นักเรียน สำหรับนักเรียนประถมศึกษา

นักเรียน สำหรับนักเรียนมัธยมศึกษา

พื้นฐานด้านวิทยาการคอมพิวเตอร์

Start learning an introduction to computer science on Code Studio with these 20 hour courses for all ages.

View my recent courses >

← 2.1 เลือกบทเรียนสำหรับนักเรียน

2.2 เลือกคอร์ส 3 →

คอร์ส 1	คอร์ส 2	คอร์ส 3	คอร์ส 4
คอร์ส 1 is designed for early readers.	คอร์ส 2 is designed for students who can read.	คอร์ส 3 is a follow-up to คอร์ส 2.	Students taking คอร์ส 4 should have already taken คอร์ส 2 and 3.
อายุ 4-6 ปี	Ages 6+ (reading required)	อายุ 8-18 ปี	อายุ 10-18 ปี

Accelerated Course

Learn basic computer science in an accelerated version of courses 2-4.

อายุ 10-18 ปี

Unplugged Lessons

If you don't have computers, try these unplugged lessons in your classroom.

Ages 4+

วิธีปฏิบัติ (ต่อ)

- เลือกหัวข้อที่ 7 เรื่อง ผีง: เงื่อนไข แล้วคลิกเมาส์เลือก บทเรียนลำดับ 1

2. เชาวงกด	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
3. คิลปิน	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
4. Functional Suncatchers	กิจกรรมถอดปลั๊ก 1 2
5. คิลปิน: Functions	1 2 3 4 5 6 7 8 9 10
6. ผีง: ฟังก์ชัน	1 2 3 4 5 6 7 8 9 10 11
7. ผีง: เงื่อนไข	1 2 3 4 5 6 7 8 9 10
8. เชาวงกด: เงื่อนไข	1 2 3 4 5 6 7 8 9 10 11 12
9. Songwriting	กิจกรรมถอดปลั๊ก 1
10. Dice Race	กิจกรรมถอดปลั๊ก 1 2
11. คิลปิน: ซุปช้อนซุป	1 2 3 4 5 6 7 8 9 10 11 12

↑ 3. เลือกหัวข้อที่ 7 เรื่อง ผีง: เงื่อนไข

- ขั้นตอนของบทเรียนนี้จะเป็นการฝึกคิดอัลกอริทึมสำหรับการเขียนโปรแกรมแบบมีเงื่อนไขเพื่อให้ผีงเดินไปรับ น้ำหวาน ดังภาพ

ขั้นตอน 7: ผีง: เงื่อนไข

เมทริกซ์นี้กำลังบันทึกของงั้น! ตรวจสอบดูว่ามีดอกไม้ที่อยู่ภายใต้เมฆ และเก็บน้ำหวานเพียงอย่างเดียวก้ามดอกไม้จะพังให้ดี เพราะมันสามารถเปลี่ยนแปลงได้ตลอด!

กล่องเครื่องมือ: ฟังก์ชันการวน: 1 / 5 บล็อก

ปุ่ม: เริ่ม, เป็นขั้นตอน

ปุ่ม: ไปข้างหน้า, หันขวา, หันซ้าย, เก็บน้ำหวาน, สลัดน้ำผึ้ง, ทำซ้ำ (???) ครั้ง, ทำ, ถ้า... ออกไป, ทำ (do)

จากภาพข้อ 4 จะพบว่าผีงจะต้องเดินทางไปรับน้ำหวาน โดยเดินไปข้างหน้า 3 ครั้ง และตำแหน่งของดอกไม้นั้นมีเมฆบังอยู่ ทำให้ไม่รู้ว่าเป็นน้ำหวานจริงหรือไม่ ดังนั้นในการออกแบบอัลกอริทึมจะต้องให้ผีงเดินไปถึงจุดที่กำหนด แล้วตรวจสอบว่าถ้าน้ำหวานยังอยู่ที่ดอกไม้ ให้ผีงเก็บน้ำหวาน


วิธีปฏิบัติ (ต่อ)

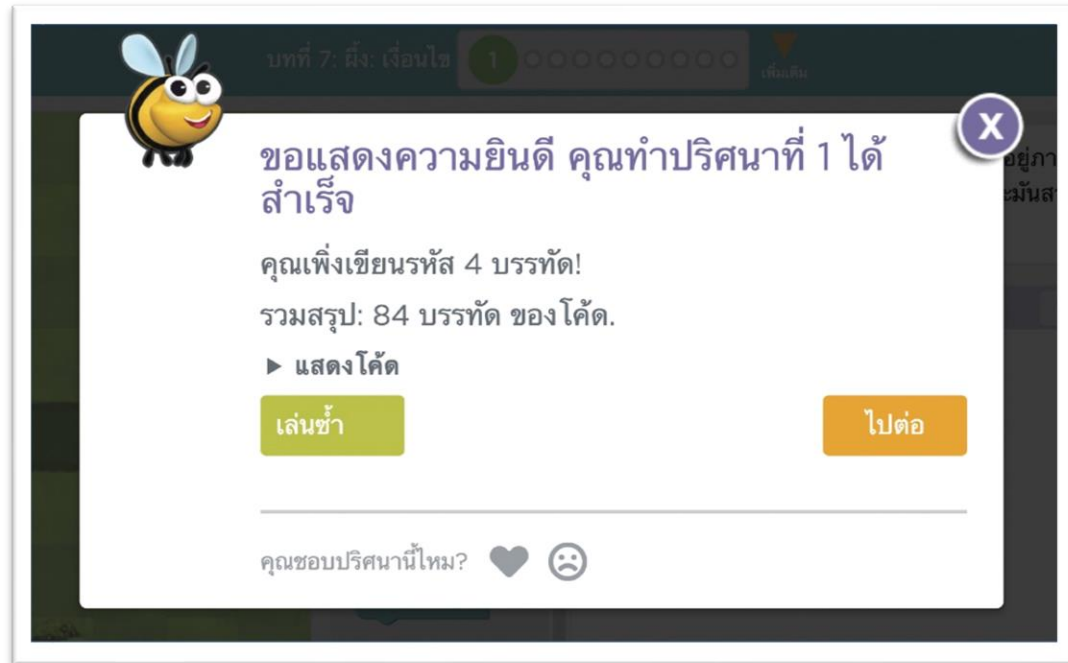
5. จากอัลกอริทึมสามารถเขียนโปรแกรมได้ ดังนี้


The image displays four sequential screenshots of a programming environment, illustrating the step-by-step construction of a program. Each screenshot shows a list of objects on the left and a workspace with code blocks. Callout boxes explain the logic behind each step.

- Screenshot 1 (Step 2 of 5):** Shows the initial setup. A 'เมื่อเรียกใช้งาน' (When clicked) block is connected to a 'ทำซ้ำ 3 ครั้ง' (Repeat 3 times) block. A callout box (5.2) explains that the number 3 is set to repeat the action 3 times.
- Screenshot 2 (Step 3 of 5):** Shows the addition of a 'ไปข้างหน้า' (Move forward) block inside the repeat loop. A callout box (5.3) explains that the 'ไปข้างหน้า' block is placed inside the loop to make the program move forward 3 times.
- Screenshot 3 (Step 4 of 5):** Shows the addition of an 'ถ้า อยู่ที่ตอกไม้' (If on the nail) block and a 'ทำ (do)' block. A callout box (5.4) explains that the 'ถ้า อยู่ที่ตอกไม้' block is used to check if the character is on the nail, and the 'ทำ (do)' block is used to perform the action.
- Screenshot 4 (Step 5 of 5):** Shows the final program. A callout box (5.5) explains that the 'เก็บน้ำหวาน' (Collect nectar) block is placed inside the 'ทำ (do)' block to collect nectar when the character is on the nail.

วิธีปฏิบัติ (ต่อ)

6. จากโปรแกรมที่เขียนขึ้น เมื่อผึ้งเดินทางไปถึงดอกไม้ ผึ้งจะตรวจสอบว่าตำแหน่งนั้นมีน้ำหวานอยู่หรือไม่ ถ้ามีผึ้งจะเก็บน้ำหวาน เมื่อเขียนโปรแกรมเรียบร้อยแล้ว คลิกเมาส์ที่ปุ่ม  จะพบว่าโปรแกรมทำงานตามขั้นตอนได้อย่างถูกต้อง และแสดงข้อความแสดงความยินดี ดังภาพ



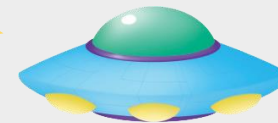
7. ถ้าคลิกเมาส์ที่คำสั่ง  จะพบว่าหน้าจอจะแสดงรหัสคำสั่งที่ใช้ควบคุมการทำงานของผึ้ง ซึ่งเป็นคำสั่งสำหรับการเขียนโปรแกรมด้วยภาษาระดับสูงที่เป็นภาษาใกล้เคียงกับภาษามนุษย์ ดังภาพ





การตรวจสอบข้อผิดพลาดของโปรแกรม

ถ้านักเรียนเขียนโปรแกรมแล้วเกิดปัญหาโปรแกรม
ไม่ทำงานตามที่เราเขียน นักเรียนจะมีวิธีการ
ตรวจสอบความผิดพลาดของโปรแกรมอย่างไร



คำถามสำคัญ

สำหรับปัญหาที่มีความซับซ้อนมากขึ้น การเขียนโปรแกรม
อาจมีข้อผิดพลาดเกิดขึ้นได้ การตรวจสอบข้อผิดพลาด อาจทำได้
โดยทดลองตรวจสอบการทำงานของโปรแกรมที่ละขั้นตอน
หรือทีละคำสั่ง เมื่อพบจุดที่ทำงานไม่ถูกต้อง ให้ทำการแก้ไข
จนกว่าจะได้ผลลัพธ์ที่ต้องการ

ตัวอย่าง

การเขียนโปรแกรมให้ผึ้งเดินไปเก็บน้ำหวาน จะต้องให้ผึ้งเดินไปข้างหน้า 2 ครั้ง แล้วเลี้ยวขวา จากนั้นเดินไปข้างหน้าอีก 2 ครั้ง พร้อมทั้งตรวจสอบว่าตำแหน่งนั้นเป็นน้ำหวานหรือไม่ ถ้าใช่ให้เก็บน้ำหวาน ดังภาพ



The image shows a programming environment with a grid-based scene on the left and a code editor on the right. The scene features a bee character on a green grid with several trees and two question marks. The code editor displays a sequence of blocks: a 'เมื่อเรียกใช้งาน' (When clicked) block, followed by 'ทำซ้ำ 2 ครั้ง' (Repeat 2 times) containing 'ไปข้างหน้า' (Move forward) blocks, then 'หันขวา ๙๐' (Turn right 90 degrees), another 'ไปข้างหน้า' block, a second 'ทำซ้ำ 2 ครั้ง' block containing an 'ถ้า อยู่ที่ดอกไม้' (If at flower) block with a 'เก็บน้ำหวาน' (Collect nectar) block inside a 'ทำ (do)' block, and finally a 'ทำซ้ำ ??? ครั้ง' (Repeat ??? times) block.

กล่องเครื่องมือ

พื้นที่ทำงาน: 8 / 8 บล็อก

- ไปข้างหน้า
- หันขวา ๙๐
- หันซ้าย ๙๐
- เก็บน้ำหวาน
- ผลิตน้ำผึ้ง
- ถ้า อยู่ที่ดอกไม้
- ทำ (do)
- ทำซ้ำ ??? ครั้ง
- ทำ

```
เมื่อเรียกใช้งาน
ทำซ้ำ 2 ครั้ง
  ไปข้างหน้า
  หันขวา ๙๐
  ไปข้างหน้า
ทำซ้ำ 2 ครั้ง
  ถ้า อยู่ที่ดอกไม้
    ทำ (do)
      เก็บน้ำหวาน
```

ถ้าโปรแกรมที่เขียนขึ้นทำงานไม่ถูกต้อง อาจตรวจสอบการทำงานของโปรแกรม โดยให้ฝั่งทำงานทีละคำสั่ง โดยคลิกเมาส์ที่ปุ่ม **เป็นขั้นตอน** ดังภาพ



นักเรียนพิจารณาที่ละคำสั่งจนจบโปรแกรม หากพบตำแหน่งที่ผิดพลาดให้เขียนโปรแกรมใหม่ ซึ่งในกรณีนี้พบว่าในการทำซ้ำ 2 ครั้งสุดท้าย ผีงไม้ได้เดินไปข้างหน้า ดังนั้นจะต้องแก้ไขโดยนำบล็อกคำสั่ง **ไปข้างหน้า** มาไว้ภายในบล็อกคำสั่งทำซ้ำนั่นเอง

มีเมฆมาเพิ่มอีกแล้ว!
เช็ครูได้เมฆทุกๆ ก่อน ว่ามันซ่อนดอกไม้ไว้หรือเปล่า ก่อนที่คุณจะทำการเก็บน้ำหวานจากเกสร
จำไว้ว่าเมฆทุกๆ ก่อน ไม่ได้ซ่อนของอย่างเดียวกัน

กล่องเครื่องมือ พื้นที่ทำงาน: 8 / 8 บล็อก เริ่มเล่นใหม่ แสดงโค้ด

เมื่อเรียกใช้งาน
ทำซ้ำ 2 ครั้ง
ทำ ไปข้างหน้า
หันขวา 0°
ทำซ้ำ 2 ครั้ง
ทำ ไปข้างหน้า
ถ้า อยู่ที่ดอกไม้
ทำ (do) เก็บน้ำหวาน

← โปรแกรมที่ถูกต้อง นำบล็อกคำสั่ง “ไปข้างหน้า” มาวางในบล็อกคำสั่งทำซ้ำ 2 ครั้ง

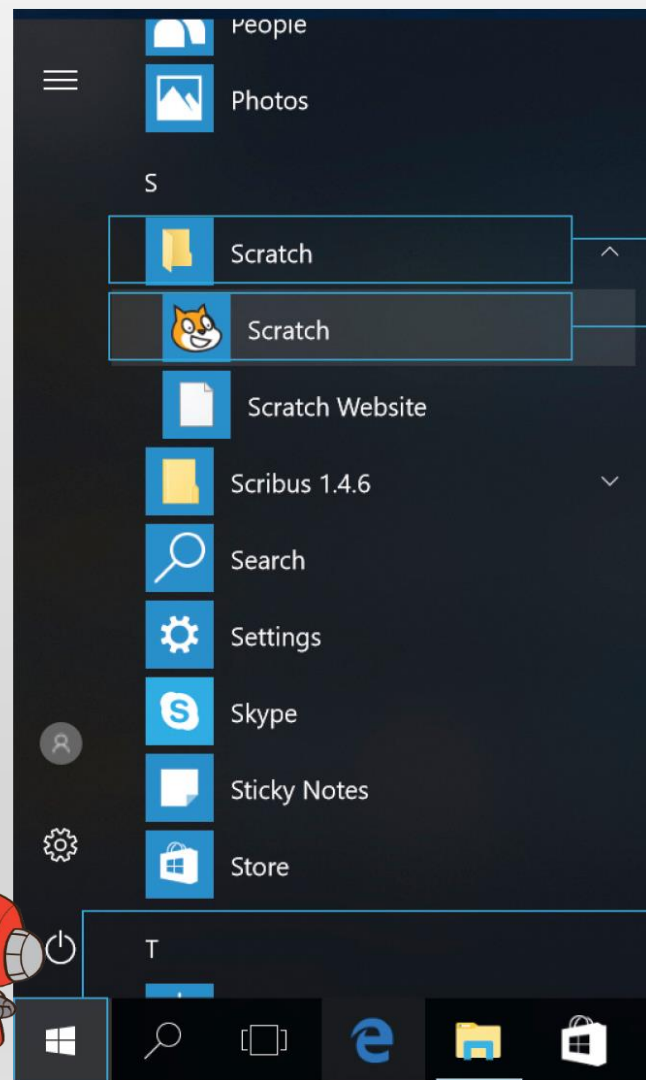
การตรวจสอบหาข้อผิดพลาดของโปรแกรมต้องใช้ประสบการณ์ แต่เราอาจฝึกสิ่งเหล่านี้ได้โดยการช่วยหาข้อผิดพลาดของโปรแกรมของผู้อื่น ซึ่งจะเป็นการพัฒนาทักษะในการหาสาเหตุของปัญหาได้ดียิ่งขึ้น



การเขียนโปรแกรมด้วย Scratch

การเขียนโปรแกรมเบื้องต้นมีเครื่องมือสำหรับเขียนโปรแกรมมากมาย โปรแกรมสแครช (Scratch) เป็นอีกเครื่องมือหนึ่งที่น่าสนใจเขียนโปรแกรมได้ง่าย โดยการออกแบบอัลกอริทึม แล้วนำบล็อกคำสั่งที่เกี่ยวข้องมาวางเพื่อให้โปรแกรมทำงาน เมื่อมีซอฟต์แวร์ติดตั้งอยู่ในคอมพิวเตอร์เราสามารถเปิดขึ้นมาได้ดังรูป

อีกวิธีหนึ่งในการเปิดใช้โปรแกรม Scratch ทำได้โดยเข้าไปที่เว็บไซต์ <https://scratch.mit.edu> โดยลักษณะของโปรแกรมแบบติดตั้งในเครื่องและแบบ online จะมีตำแหน่งของคำสั่งและตำแหน่งหน้าจอต่างกันเล็กน้อย



- ← 1. คลิกเมาส์ที่ Start
- ← 2. เลือกโฟลเดอร์ Scratch
- ← 3. เลือกโปรแกรม Scratch

เมื่อเปิดโปรแกรม Scratch ขึ้นมา จะเห็นว่ากลุ่มของคำสั่ง พื้นที่เขียนสคริปต์และเสียง และฉากสำหรับแสดงตัวละคร ดังนี้

10. พื้นที่การแสดงผลการทำงานของโปรแกรมที่มีขนาดที่เปลี่ยนแปลง

1. แลปเมนูเครื่องมือ

7. เวที

2. แลปเครื่องมือเวที

9. แลปเมนูแสดงข้อมูลสคริปต์ คอสตูม และเสียงของตัวละครหรือเวที

Scratch

Untitled

4. กลุ่มบล็อกคำสั่ง

สคริปต์ คอสตูม เสียง

การเคลื่อนที่

รูปร่าง

เสียง

ปากกา

ข้อมูล

เหตุการณ์

ควบคุม

การรับรู้

โอเปอร์เรเตอร์

บล็อกเพิ่มเติม

เลือก 10 ก้าว

หมุน 15 องศา

หมุน 15 องศา

ชี้ไปที่ทิศทาง 90

ชี้ไปทาง ตัวชี้เมาส์

ไปยัง x: 0 y: 0

ไปยัง ตัวชี้เมาส์

ร่อน 1 วินาที ไปที่ x: 0 y: 0

เปลี่ยน x ทีละ 10

ตั้งค่า x เป็น 0

เปลี่ยน y ทีละ 10

ตั้งค่า y เป็น 0

ถ้าชนขอบให้สะท้อนกลับ

ค้นรูปแบบการหมุน ชั่วคราว

x: -79 y: -142

ตัวละคร

ตัวละครใหม่

ฉาก

1 ฉากหลัง

New backdrop:

Sprite1

3. ข้อมูลของตัวละครที่ถูกเลือก

6. พื้นที่ทำงาน

8. รายการตัวละครและเวทีที่ใช้ในโปรเจกต์ปัจจุบัน

5. บล็อกในกลุ่มที่เลือก

บล็อกคำสั่งแต่ละคำสั่งจะรวมกลุ่มอยู่ในแถบโคดบล็อก (code blocks palette) ซึ่งมีคำสั่งต่าง ๆ อยู่มากมาย รวมกันเป็นหมวด ถ้านำเมาส์ไปคลิกที่เมนูในหมวดคำสั่ง โปรแกรมจะแสดงคำสั่งในหมวดนั้นออกมา เช่น ถ้าคลิกเมาส์ที่กลุ่มคำสั่งเสียง โปรแกรมจะแสดงคำสั่งออกมา ดังภาพ

The screenshot shows the Scratch interface with the 'Sound' block palette open. The palette is divided into two columns: 'Sound' (เสียง) and 'Sound Effects' (เสียงเอฟเฟกต์). The 'Sound' column contains blocks for playing a sound file, playing a sound file for a duration, stopping all sounds, playing a sound file for a duration and then stopping, playing a sound file for a duration and then fading out, setting the volume, and setting the tempo. The 'Sound Effects' column contains blocks for playing a sound effect, playing a sound effect for a duration, and playing a sound effect for a duration and then fading out. The 'Sound' block palette is highlighted with a blue box, and a yellow callout box with an arrow points to it, containing the text '3. โปรแกรมแสดงคำสั่งต่าง ๆ ออกมา'. The main stage area shows the Scratch cat sprite and a yellow callout box with an arrow pointing to the 'Sound' block palette, containing the text '1. คลิกเมาส์เลือกคำสั่งสคริปต์' and '2. คลิกเมาส์เลือกเสียง'. The bottom left corner shows the 'Sprites' area with the Scratch cat sprite selected.

Scratch v461

Untitled

สคริปต์ คอสตูม เสียง

การเคลื่อนที่ รูปร่าง เสียง ปากกา ข้อมูล เหตุการณ์ ความคม การรับรู้ โอเพอร์เรเตอร์ บล็อกเพิ่มเติม

เล่นเสียง meow

เล่นเสียง meow จนจบ

หยุดเสียงทั้งหมด

เล่น กลอง 1 เป็น 0.25 จังหวะ

หยุด 0.25 จังหวะ

เล่น โน้ต 60 เป็น 0.5 จังหวะ

ตั้งค่าเครื่องดนตรีเป็น 1

ปรับระดับเสียงทีละ -10

ตั้งค่าระดับเสียงเป็น 100 %

ระดับเสียง

ปรับความเร็วทีละ 20

ตั้งค่าความเร็วไปเป็น 60 bpm

จังหวะ

x: 240 y: 95

ตัวละคร ตัวละครใหม่

จาก 1 ฉากหลัง

New backdrop:

Sprite1

1. คลิกเมาส์เลือกคำสั่งสคริปต์

2. คลิกเมาส์เลือกเสียง

3. โปรแกรมแสดงคำสั่งต่าง ๆ ออกมา

The screenshot shows the Scratch interface with a script area containing the following blocks:

- Play sound 'meow' (dropdown)
- Play sound 'meow' when clicked (dropdown)
- Stop all sounds
- Play sound '1' for 0.25 seconds (dropdown)

A dropdown menu is open for the '1' sound block, listing 18 sound options:

- (1) กลองเล็ก
- (2) กลองใหญ่
- (3) ไซดัลติก
- (4) ฉาบส่ง
- (5) โอเพน ไฮ-แฮท
- (6) โคลไฮแฮท
- (7) แทมโบรีน
- (8) เสียงปรบมือ
- (9) เคลฟาส
- (10) Wood Block
- (11) กระดิ่งวัว
- (12) สามเหลี่ยม
- (13) บองโก
- (14) คองกา
- (15) คาบาซา
- (16) กีโร
- (17) ไวบราสแลป
- (18) โอเพน ครีคา

Two callout boxes provide instructions:

1. คลิกเมาส์ตรงนี้ → (Click the mouse here)
- ← 2. โปรแกรมแสดงเสียงกลองต่าง ๆ ออกมา (The program will display different drum sounds)

ในโปรแกรมได้กำหนดเสียงต่างๆ ให้
 เลือกใช้ได้อย่างง่ายและถ้าคลิกเมาส์
 เลือกเสียงกลอง โปรแกรมจะมีเสียงกลอง
 ลักษณะต่าง ๆ ให้เลือกอีกด้วย
 สามารถใช้เมาส์คลิกหรือป้อนข้อมูลเป็น
 ตัวเลข ดังภาพ

ถ้าคลิกเมาส์เลือกกลุ่มคำสั่งควบคุม โปรแกรมจะแสดงคำสั่งควบคุมออกมา ซึ่งใช้ควบคุมการทำงานของโปรแกรมและตัวละคร เช่น ให้ทำงานเมื่อมีการคลิกเมาส์ ให้ทำงานซ้ำ ๆ ให้หน่วงเวลารอ ดึงภาพ

The screenshot shows the Scratch interface with several Thai annotations:

- 1. คลิกเมาส์เลือกคำสั่งสคริปต์**: Points to the 'Scripts' category in the right-hand menu.
- 2. คลิกเมาส์เลือกกลุ่มคำสั่งควบคุม**: Points to the 'Control' sub-category in the right-hand menu.
- 3. กลุ่มบล็อกคำสั่งควบคุม**: Points to a block in the script area, specifically the 'wait 1 second' block.

The Scratch interface includes a top navigation bar with 'Scratch' logo and menu options like 'เพิ่ม' (Add), 'แก้ไข' (Edit), 'เคล็ดคลับ' (Clubs), and 'เกี่ยวกับ' (About). The main workspace shows a 'Sprite1' (Scratch cat) on a stage with coordinates x: 240, y: -1. The right-hand menu is categorized into 'สคริปต์' (Scripts), 'คอสมตูม' (Costumes), and 'เสียง' (Sounds). The 'Scripts' menu includes categories like 'เหตุการณ์' (Events), 'ควบคุม' (Control), 'การรับรู้' (Sensing), 'โอเปอเรเตอร์' (Operators), and 'บล็อกเพิ่มเติม' (More blocks). The 'Control' sub-menu is highlighted, showing blocks like 'รอ 1 วินาที' (wait 1 second), 'ทำซ้ำ 10' (repeat 10), 'วนซ้ำตลอด' (repeat forever), 'ถ้า...แล้ว' (if...then), 'ถ้า...แล้ว...มีฉะนั้น' (if...then...else), 'รอกจนกระทั่ง' (wait until), 'ทำซ้ำจน' (repeat until), 'หยุด ทั้งหมด' (stop all), 'เมื่อฉันเริ่ม ในฐานะโคลน' (when green flag clicked), 'สร้างตัว โคลนของ ตัวฉันเอง' (clone myself), and 'ลบ โคลนตัวนี้' (delete this clone).

วัตถุประสงค์


เขียนโปรแกรมให้ตัวละครเคลื่อนที่ตามที่ต้องการได้

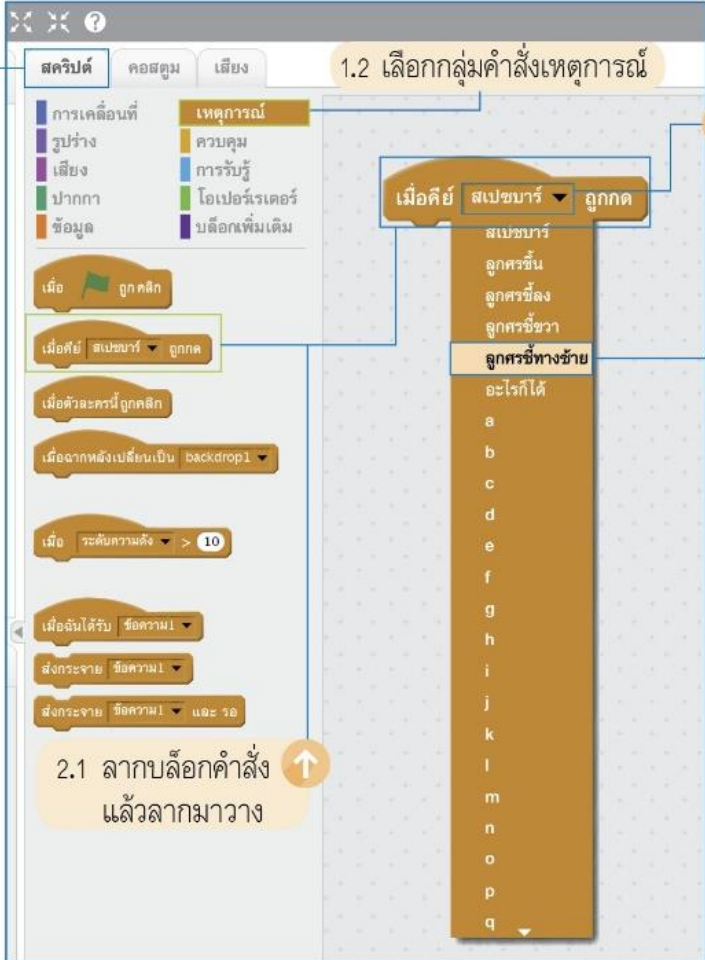
วัสดุอุปกรณ์

คอมพิวเตอร์ 1 เครื่อง

วิธีปฏิบัติ

นักเรียนเขียนโปรแกรมควบคุมให้ตัวละครเคลื่อนที่โดยเมื่อกดคีย์ลูกศรที่เป็นพิมพ์ชี้ไปทางซ้ายแมวจะเคลื่อนที่ไปทางซ้าย เมื่อกดคีย์ลูกศรชี้ไปทางขวาแมวจะเคลื่อนที่ไปทางขวา โดยปฏิบัติตามขั้นตอน ดังนี้

1. เลือกกลุ่มคำสั่งสคริปต์ แล้วเลือกคำสั่งเหตุการณ์
2. เลือกบล็อกคำสั่งสคริปต์แล้วเลือกคำสั่ง  แล้วลากมาวางบนพื้นที่เขียนสคริปต์
3. เลือกคีย์ลูกศรชี้ทางซ้าย



The screenshot shows the Scratch interface with the following annotations:

- 1.1** คลิกเมาส์เลือกคำสั่งสคริปต์ (Click mouse to select script command)
- 1.2** เลือกกลุ่มคำสั่งเหตุการณ์ (Select event command group)
- 2.1** ลากบล็อกคำสั่งแล้วลากมาวาง (Drag the command block and place it)
- 3.1** คลิกเมาส์เลือกที่เครื่องหมายนี้ (Click mouse to select this symbol)
- 3.2** เลือกคำสั่งลูกศรชี้ทางซ้าย (Select left arrow command)

วิธีปฏิบัติ (ต่อ)

- เลือกกลุ่มคำสั่งการเคลื่อนที่ แล้วนำบล็อกคำสั่ง **เคลื่อน 10 ก้าว** มาวางต่อ กำหนดให้เคลื่อนที่ไปทางซ้าย โดยกำหนดค่าเป็น -100 ดังภาพ
- เขียนคำสั่งต่อไปโดยนำบล็อกคำสั่ง **เมื่อคีย์ สเปซบาร์ ถูกกด** มาวางอีกครั้ง แล้วกำหนดเป็นคีย์ชี้ทางขวา หรือ right arrow แล้วนำบล็อกคำสั่ง **เคลื่อน -100 ก้าว** มาวางต่อโดยกำหนดการเคลื่อนที่เป็น 100 ดังภาพ
- จากโปรแกรมจะพบว่าเมื่อคลิกคำสั่งสองชุด ให้ทดลองรันโปรแกรม แล้วสังเกตการทำงานของโปรแกรม

จากการเขียนโปรแกรมจะพบว่าข้อมูลนำเข้าของโปรแกรม คือ การกดคีย์ลูกศร ซึ่งเป็นไปได้ 2 คีย์ โดยมีการทำงานแตกต่างกัน และตัวละครจะเคลื่อนที่ตามข้อมูลนำเข้า หรือคีย์ที่กด

4.1 ลากบล็อกคำสั่งเคลื่อนมาวาง

4.2 กำหนดค่าเป็น -100

5.1 นำบล็อกคำสั่งเมื่อคีย์...ถูกกดมาวาง

5.2 กำหนดคีย์ชี้ไปทางขวา

5.3 นำบล็อกคำสั่งเคลื่อนมาวาง

5.4 กำหนดค่าเป็น 100

วัตถุประสงค์

เขียนโปรแกรมให้แมวเคลื่อนที่ได้โดยเปลี่ยนรูปร่างไปที่ละแบบได้

วัสดุอุปกรณ์

คอมพิวเตอร์ 1 เครื่อง

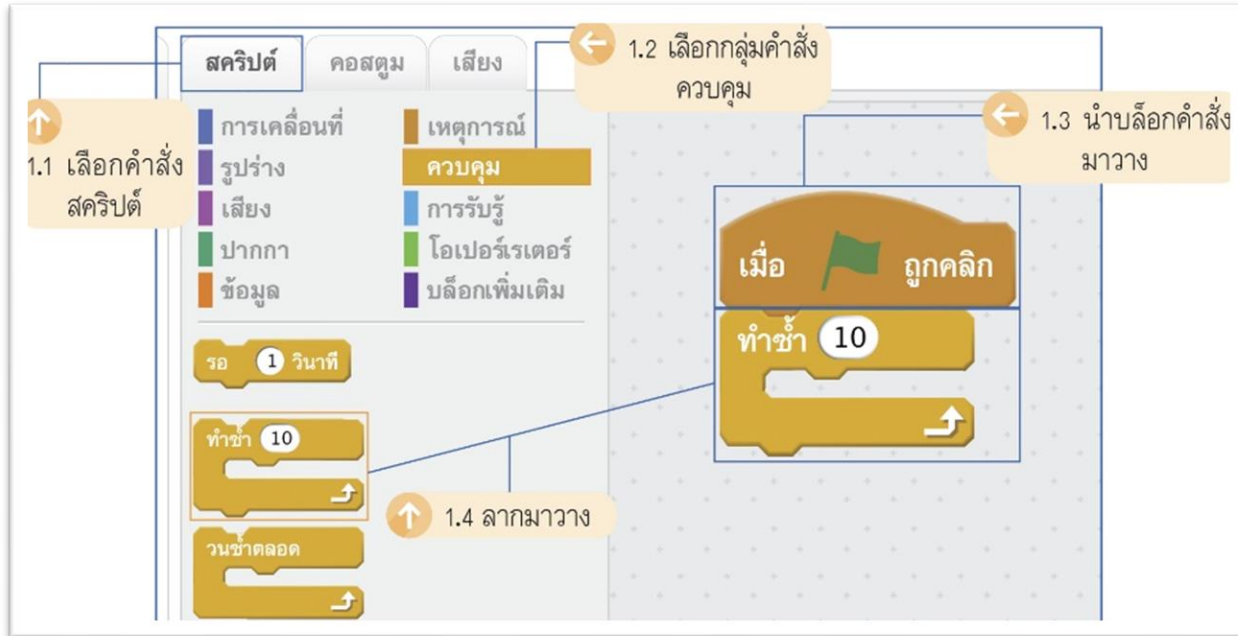
วิธีปฏิบัติ

กิจกรรมนี้เป็นการออกแบบให้ตัวละครเคลื่อนที่ได้เหมือนจริงมากขึ้น หากเราพิจารณาการเคลื่อนที่ของแมว โดยสั่งให้เคลื่อนที่ไปข้างหน้า 200 หน่วย แมวจะเปลี่ยนตำแหน่งจากตำแหน่งแรกไปยังตำแหน่งสุดท้ายทันที แต่ถ้าให้แมวค่อย ๆ เดินออกไปเป็นระยะ ๆ จนครบ 200 หน่วย การเคลื่อนที่จะดูสวยงามมากขึ้น หากระหว่างการเคลื่อนที่ไปแมวเปลี่ยนรูปร่างไปด้วยจะทำให้ดูมีชีวิตมากขึ้นเช่นกัน

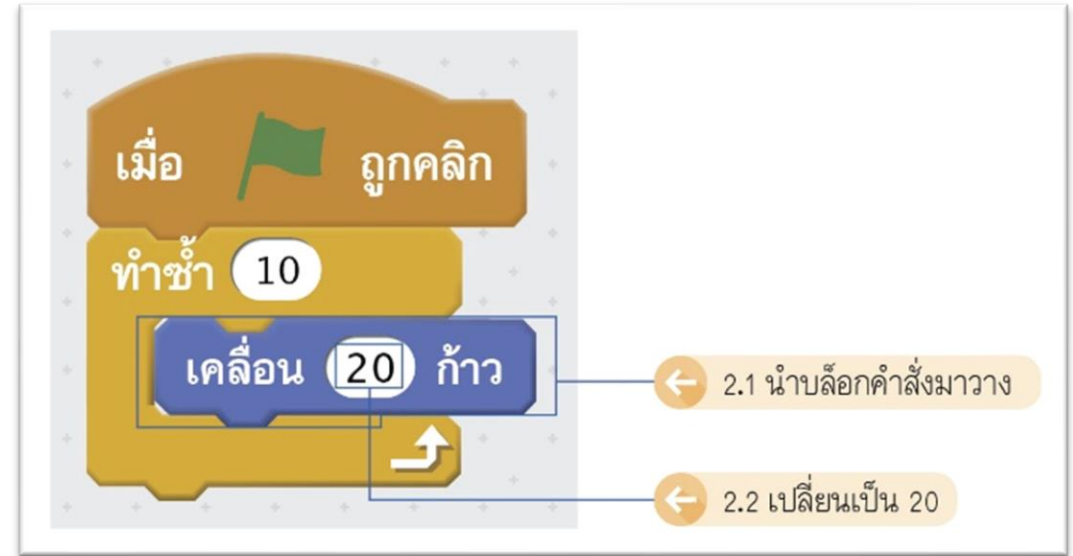
วิธีปฏิบัติ (ต่อ)

1. เมื่อเปิดโปรแกรม Scratch ขึ้นมา จะแสดงตัวละครภาพแมว จากนั้นเลือกกลุ่มคำสั่งสคริปต์ แล้วเลือกกลุ่มคำสั่งควบคุม จากนั้นนำบล็อกคำสั่ง **เมื่อ ถูกคลิก** มาวาง แล้วนำบล็อกคำสั่ง

ทำซ้ำ 10 มาวาง ดังภาพ

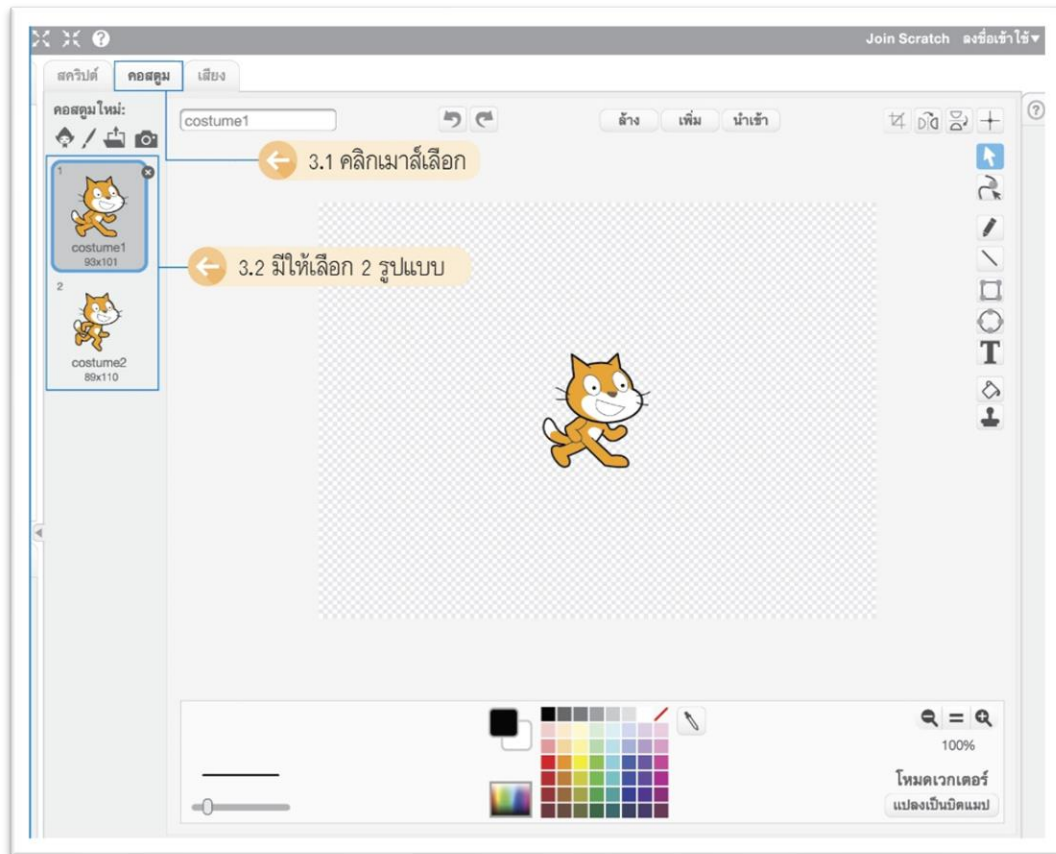


2. ต้องการให้แมวเคลื่อนที่ไป 200 หน่วย แต่ให้ทำซ้ำ 10 ครั้ง แต่แต่ละครั้งเคลื่อนที่ไปเป็นระยะ 20 หน่วย ดังนั้นนำบล็อกคำสั่ง **เคลื่อน 10 ก้าว** มาวางในรูปการทำซ้ำ แล้วแก้ไขเป็น 20 ก้าว ดังภาพ

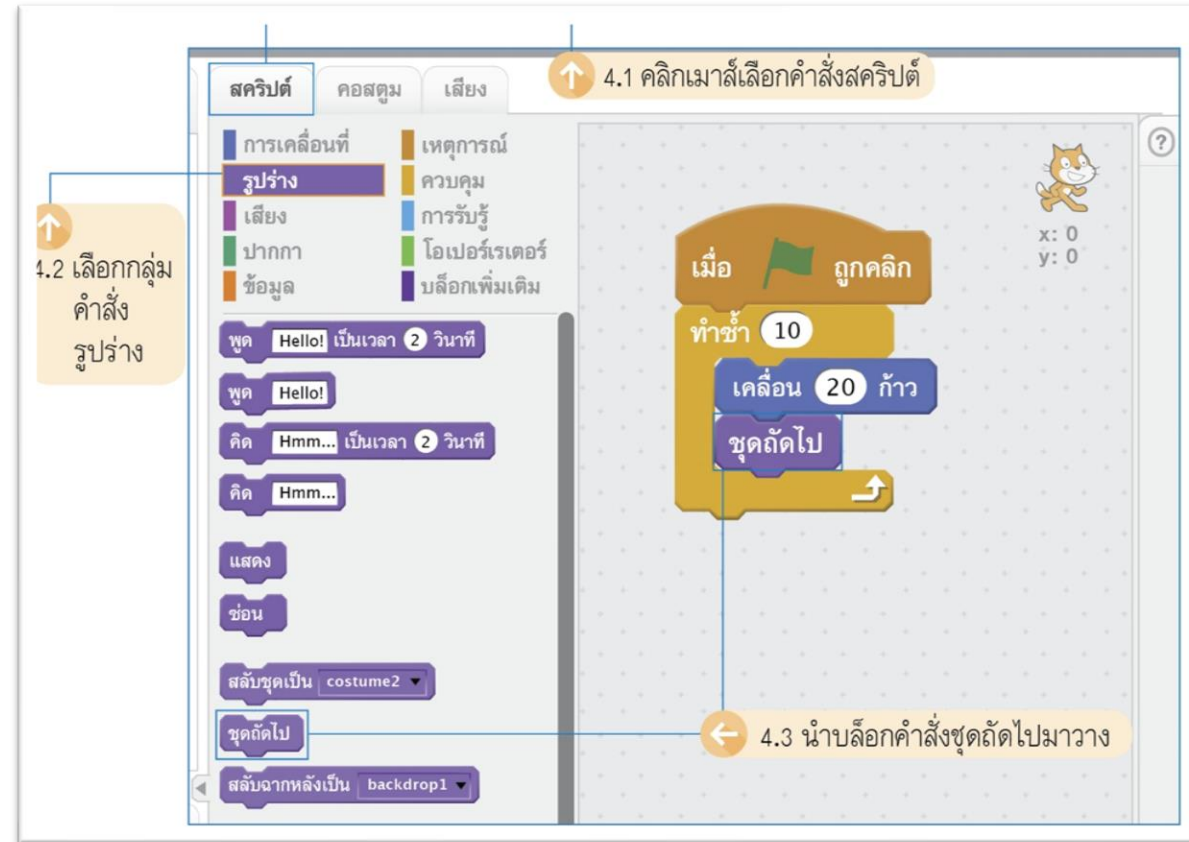


วิธีปฏิบัติ (ต่อ)

- คลิกเมาส์ที่คอสตุ่ม โปรแกรมจะแสดงลักษณะของตัวละครตัวนี้ โดยแสดงให้เป็น 2 รูปแบบ ถ้าหากใช้เมาส์คลิกเลือกรูปแบบที่สองและรูปแบบที่หนึ่ง จะเห็นลักษณะของแมวแตกต่างกันไป นอกจากนี้ยังสามารถปรับแต่งตัวละครเป็นรูปแบบต่าง ๆ ได้อีกด้วย



- เขียนโปรแกรมต่อโดยเลือกที่การเขียนโปรแกรมให้สคริปต์เลือกกลุ่มคำสั่งรูปร่าง แล้วนำบล็อกคำสั่ง **ชุดถัดไป** มาวางเพื่อเปลี่ยนเป็นรูปร่างของตัวละครถัดไป ดังภาพ



วิธีปฏิบัติ (ต่อ)

5. นำบล็อกคำสั่ง **รอ 1 วินาที** มาวาง แล้วเปลี่ยนเวลาเป็น 0.5 วินาที เพื่อให้แมวเคลื่อนที่แล้วเปลี่ยนรูปร่าง จากนั้นหยุดเป็นเวลาครึ่งวินาที ดังภาพ



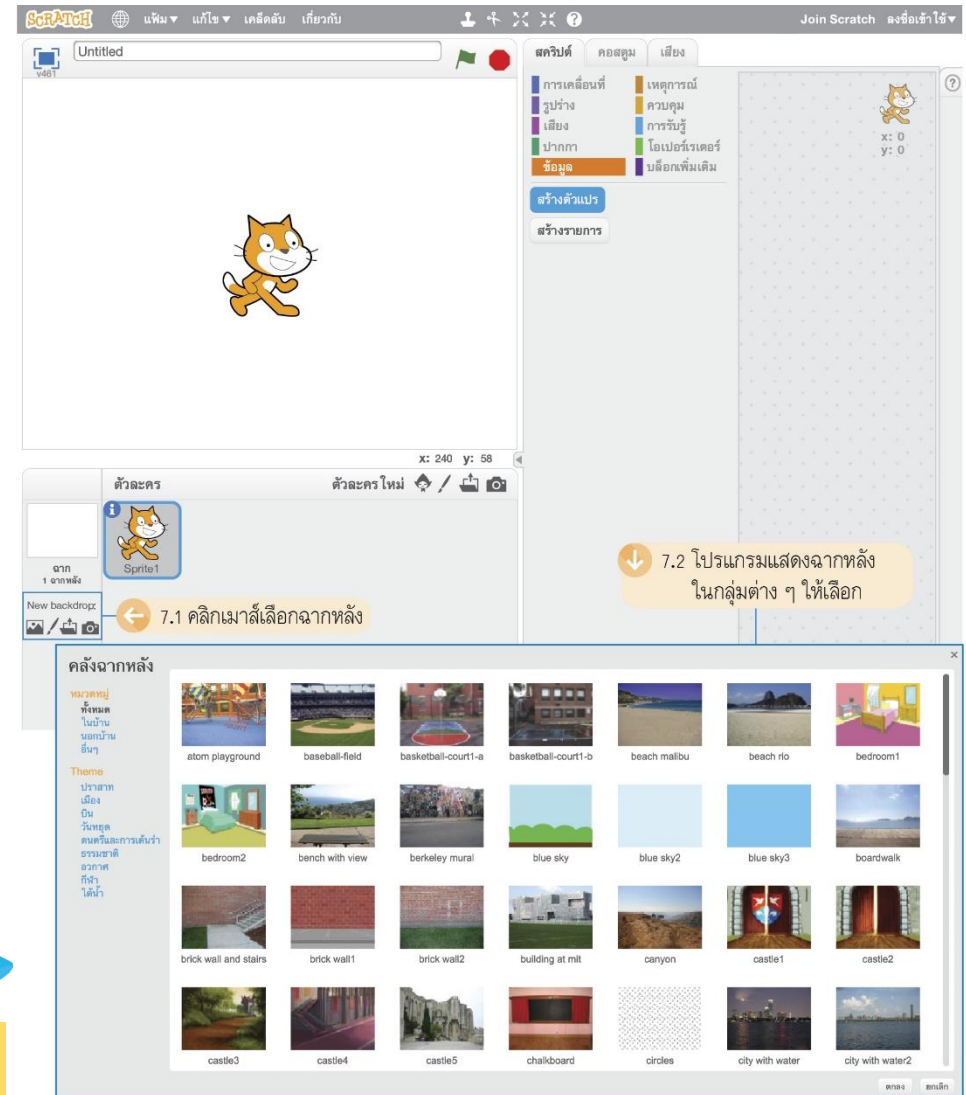
6. ทดลองรันโปรแกรมจะพบว่าแมวเคลื่อนที่อย่างสวยงาม

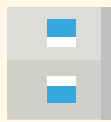
ถ้านักเรียนต้องการให้แมวเคลื่อนที่แบบสวยงาม โดยเคลื่อนที่ไปทางซ้ายเมื่อกดคีย์ลูกศรซ้ายไปทางซ้าย และเคลื่อนที่ไปทางขวาเมื่อกดคีย์ลูกศรขวา นักเรียนจะเขียนโปรแกรมอย่างไร



คำถามสำคัญ

7. การใช้โปรแกรม Scratch เราสามารถสร้างฉากหลังให้กับตัวละครได้ โดยคลิกเมาส์เลือกที่ New backdrop โปรแกรมจะแสดงฉากหลัง ให้เลือกใช้มากมาย ดังภาพ



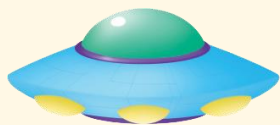


ตัวแปร

การเขียนโปรแกรมคอมพิวเตอร์ให้ประมวลผลข้อมูลมักมีการประกาศตัวแปรสำหรับเก็บข้อมูลที่รับเข้าไปประมวลผล หรือเก็บข้อมูลที่ได้จากการประมวลผล

ตัวอย่าง ถ้าเราต้องการให้คอมพิวเตอร์รับตัวเลขสองค่าจากแป้นพิมพ์เข้ามาบวกกัน แล้วแสดงผลลัพธ์ค่าแรกที่ได้รับเข้าไปอาจเก็บไว้ใน x ค่าที่สองเก็บไว้ใน y จากนั้นนำทั้งสองค่ามาบวกกันแล้วเก็บไว้ในตัวแปรชื่อ sum เพื่อใช้สำหรับแสดงผล เราสามารถเขียนอัลกอริทึมเป็นข้อความ ได้ดังนี้

ทำไมจึงกำหนดตัวแปรในการเขียนโปรแกรม

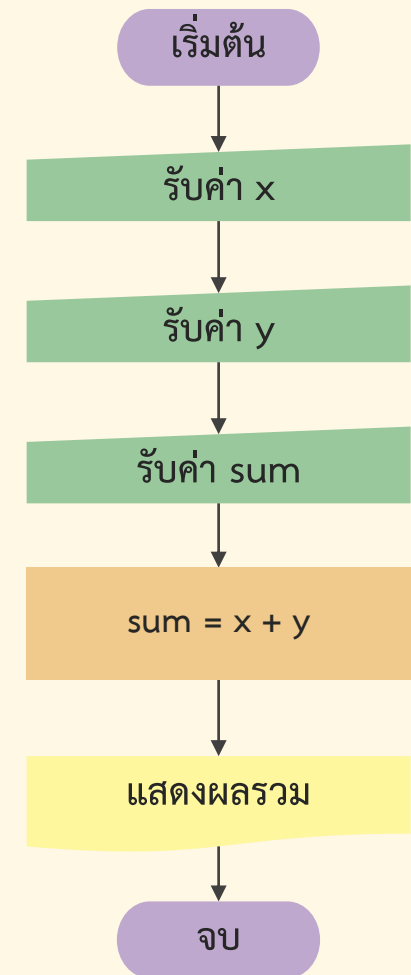


คำถามสำคัญ

เริ่มต้น

1. ประกาศตัวแปร x , y และ sum
2. รับค่าแรกเก็บใน x
3. รับค่าที่สองเก็บใน y
4. นำ x บวกกับ y แล้วเก็บไว้ใน sum
5. แสดงผล sum

จบ



วัตถุประสงค์

เขียนโปรแกรมสร้างเครื่องคิดเลขอย่างง่ายได้

วัสดุอุปกรณ์

คอมพิวเตอร์ 1 เครื่อง

วิธีปฏิบัติ

การเขียนโปรแกรมลักษณะนี้จะต้องประกาศตัวแปรขึ้นมา 3 ตัว แล้วนำมาประมวลผล โดยทำได้ดังขั้นตอนต่อไปนี้

1. เปิดโปรแกรม Scratch เลือกชุดคำสั่งสคริปต์ แล้วเลือกข้อมูล
2. ประกาศตัวแปร โดยคลิกเมาส์ที่ปุ่ม **สร้างตัวแปร**
3. โปรแกรมจะแสดงหน้าต่าง ให้สร้างตัวแปรใหม่ ให้ตั้งชื่อเป็น x แล้วคลิกเมาส์ที่ปุ่ม **ตกลง**

1.1 เลือกคำสั่งสคริปต์

1.2 เลือกกลุ่มคำสั่งข้อมูล

2.1 คลิกเมาส์ที่ปุ่มสร้างตัวแปร

3.1 พิมพ์ชื่อตัวแปร x

3.2 คลิกเมาส์ที่ปุ่มตกลง

ตัวแปรใหม่

ชื่อตัวแปร: x

สำหรับตัวละครทั้งหมด สำหรับตัวละครนี้เท่านั้น

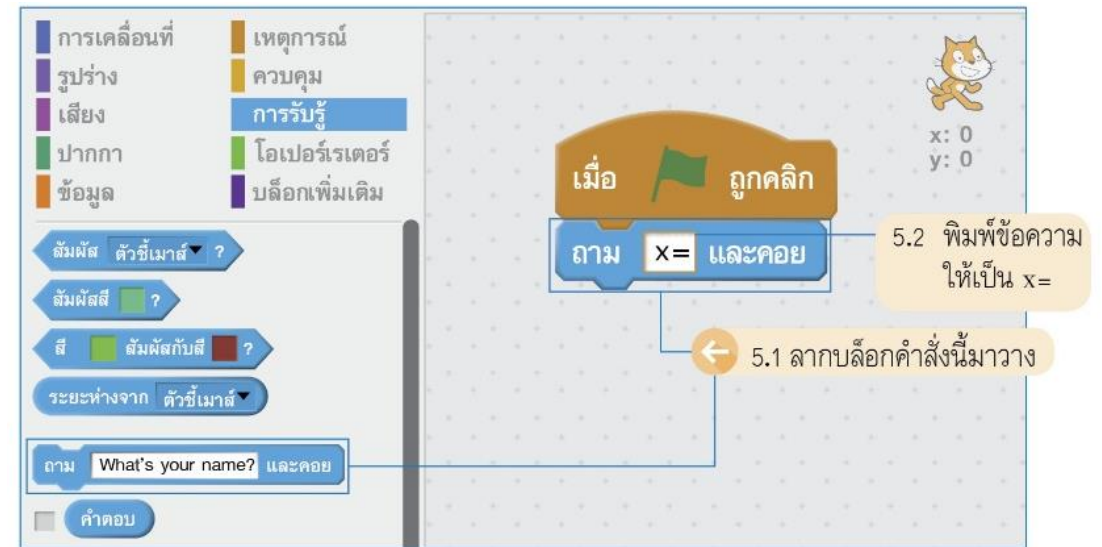
ตกลง ยกเลิก

วิธีปฏิบัติ (ต่อ)

4. ทำซ้ำ โดยประกาศตัวแปรชื่อ y และ sum จากนั้นโปรแกรมจะแสดงรายชื่อตัวแปรทั้งหมด และมีค่าเริ่มต้นเป็น 0 พร้อมทั้งแสดงค่าของตัวแปรไว้มุมบนซ้ายของโปรแกรม ดังภาพ



5. การรับข้อมูลทางแป้นพิมพ์ สามารถใช้บล็อกคำสั่ง **ถาม** **What's your name? และคอย** มาใช้งานได้ เมื่อมีการพิมพ์ข้อมูลเข้าไป ข้อมูลจะเก็บในตัวแปร คำตอบเสมอ ลากบล็อกคำสั่ง **ถาม** **What's your name? และคอย** มาวาง ดังภาพ แล้วพิมพ์ข้อความให้โปรแกรมถามเป็น $x =$



วิธีปฏิบัติ (ต่อ)

6. เมื่อโปรแกรมทำงานจะแสดงกล่องข้อความให้ป้อนข้อมูล และข้อมูลจะเก็บในตัวแปร คำตอบ ดังนั้นขั้นตอนต่อไปจะต้องเขียนโปรแกรมเพื่อกำหนดค่านี้ให้กับตัวแปร x โดยเลือกบล็อกคำสั่ง `set sum to 0` ซึ่งอยู่ในกลุ่มคำสั่งของข้อมูล แล้วคลิกเมาส์เปลี่ยนให้เป็นตัวแปร x ดังภาพ

6.1 เลือกคำสั่งสคริปต์

6.2 เลือกกลุ่มคำสั่งข้อมูล

6.3 ลากบล็อกคำสั่งกำหนดข้อมูลมาวาง

6.4 เปลี่ยนตัวแปรให้เป็น x

7. กำหนดค่า x ให้เป็น คำตอบ ซึ่งเป็นคำตอบของการถามด้วยคำสั่ง `ถาม What's your name? และคอย` โดยเลือกกลุ่มคำสั่ง การรับรู้ แล้วนำบล็อกคำสั่ง `คำตอบ` มาวาง ดังภาพ

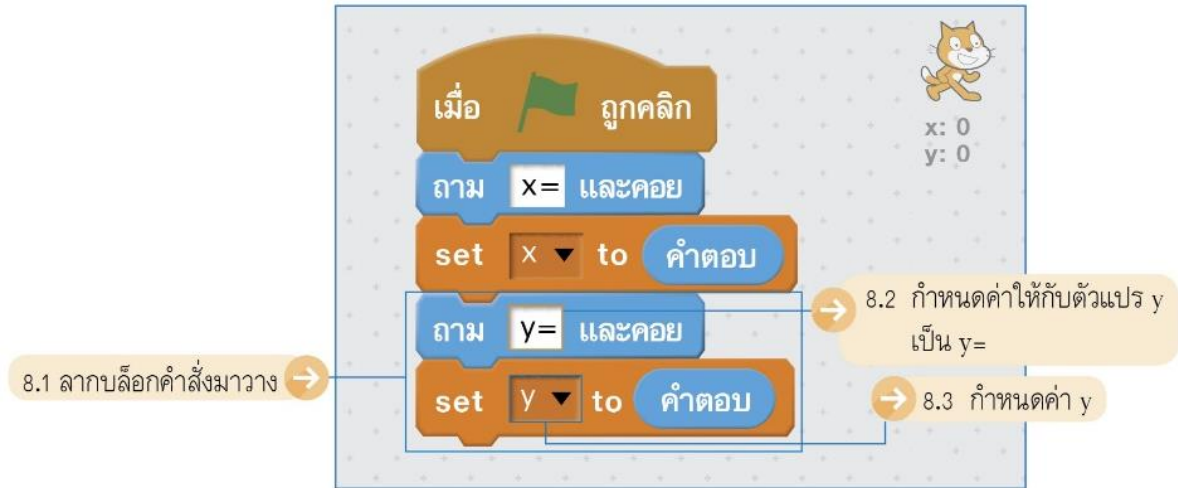
7.1 เลือกคำสั่งสคริปต์


7.2 เลือกกลุ่มคำสั่งการรับรู้

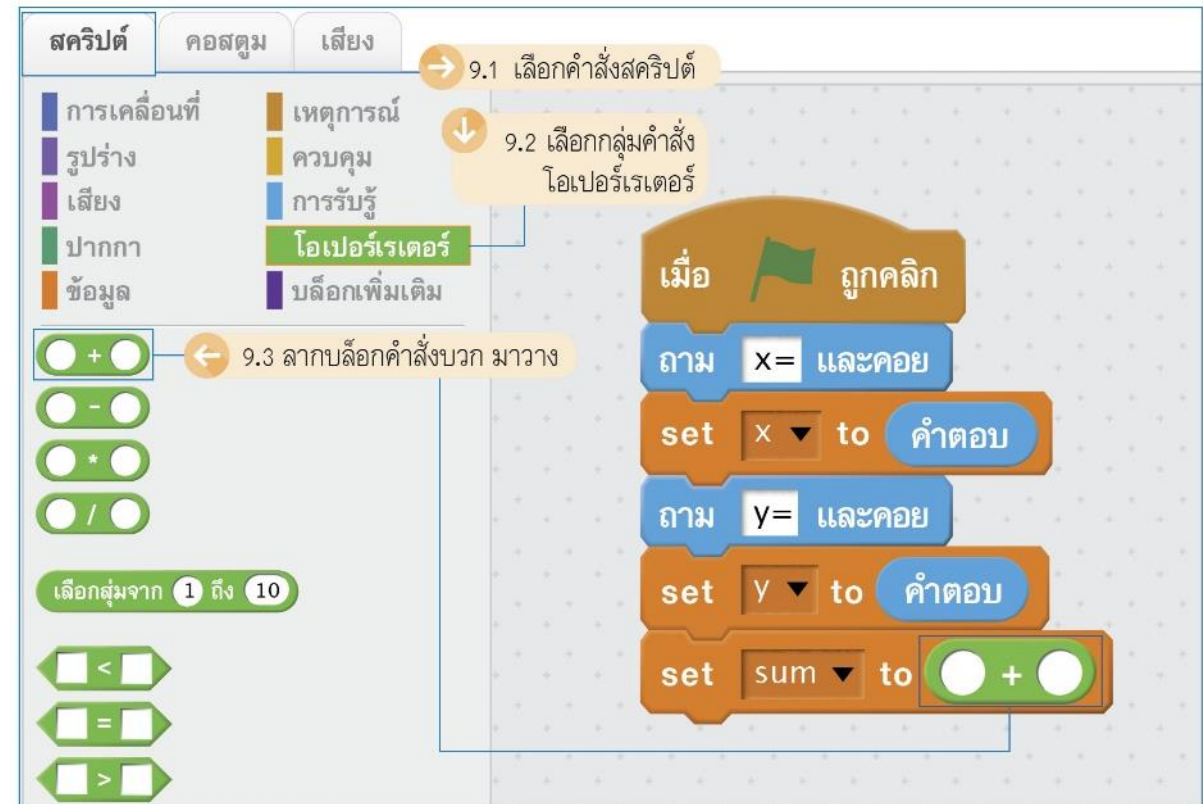
7.3 ลากมาวาง

วิธีปฏิบัติ (ต่อ)

8. นำบล็อกคำสั่ง **ถาม x= และคอย** และ **set x to คำตอบ** มาวาง จากนั้น กำหนดค่าให้กับตัวแปร y เป็น y = และ y ดังภาพ

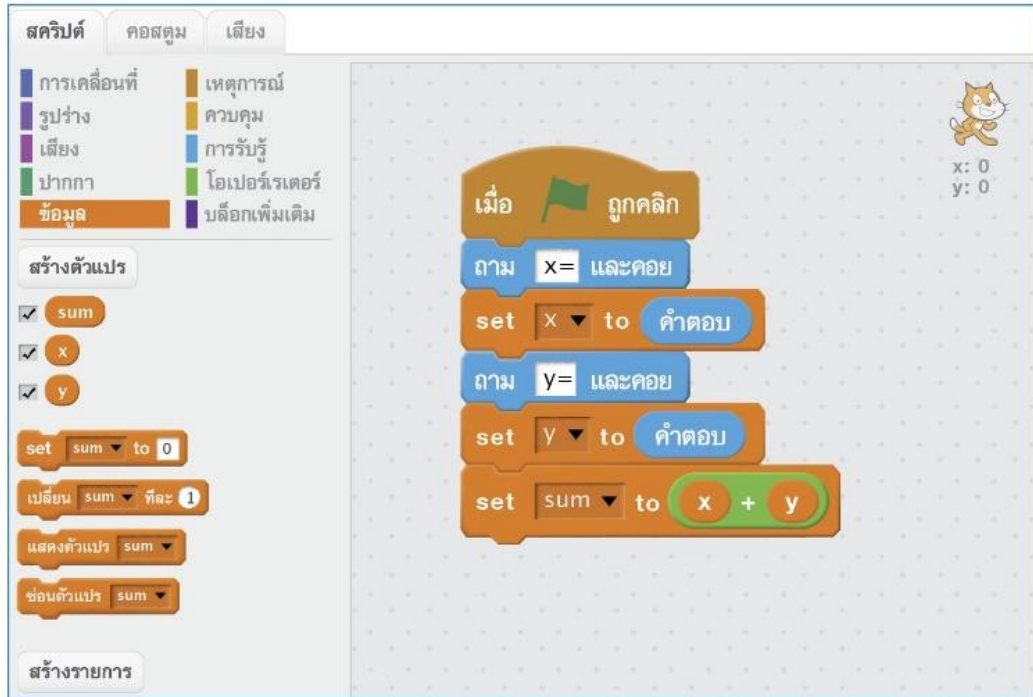


9. กำหนดค่าให้กับตัวแปร sum โดยนำข้อมูลของตัวแปร x กับตัวแปร y มาบวกกัน จากนั้นเลือกคำสั่งตัวดำเนินการในกลุ่มโอเปอเรเตอร์ แล้วเลือกบล็อกคำสั่ง  มาวาง ดังภาพ



วิธีปฏิบัติ (ต่อ)

10. กำหนดให้บวกค่าในตัวแปร x และ y โดยเลือกกลุ่มคำสั่งข้อมูล แล้วนำตัวแปร x และ y มาวางในบล็อกคำสั่ง ดังภาพ



11. ทดลองรันโปรแกรม จะพบว่า แมวจะแสดงข้อความ $x =$ พร้อมทั้งมีกล่องสำหรับป้อนข้อมูล แล้วทดลองป้อนค่า 5 ลงไป แล้วกดปุ่ม ที่เป็นพิมพ์ ดังภาพ



วิธีปฏิบัติ (ต่อ)

12. ป้อนค่า y เป็น 6 เมื่อโปรแกรมทำงานจะนำ 5 และ 6 มาบวกกัน แล้วเก็บผลลัพธ์ไว้ใน sum โดยโปรแกรมจะแสดงข้อมูลออกมา ดังภาพ



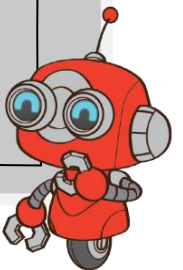
The screenshot shows the Scratch interface with a project titled "Untitled". The variable monitor area displays three variables: x with a value of 5, y with a value of 6, and sum with a value of 11. Two callout boxes with arrows point to the y and sum variables.

Variable	Value
x	5
y	6
sum	11

← 12.1 ป้อนข้อมูลเป็น 6

← 12.2 โปรแกรมแสดงผลลัพธ์

จากกิจกรรมที่ผ่านมาจะพบว่าเราสามารถเขียนโปรแกรมให้ประมวลผลทางคณิตศาสตร์ได้โดยง่าย นอกจากนี้เรายังนำไปประยุกต์กับการเขียนโปรแกรมคำนวณด้านต่าง ๆ ได้อีกด้วย



วัตถุประสงค์

เขียนโปรแกรมสร้างรูปเรขาคณิตได้

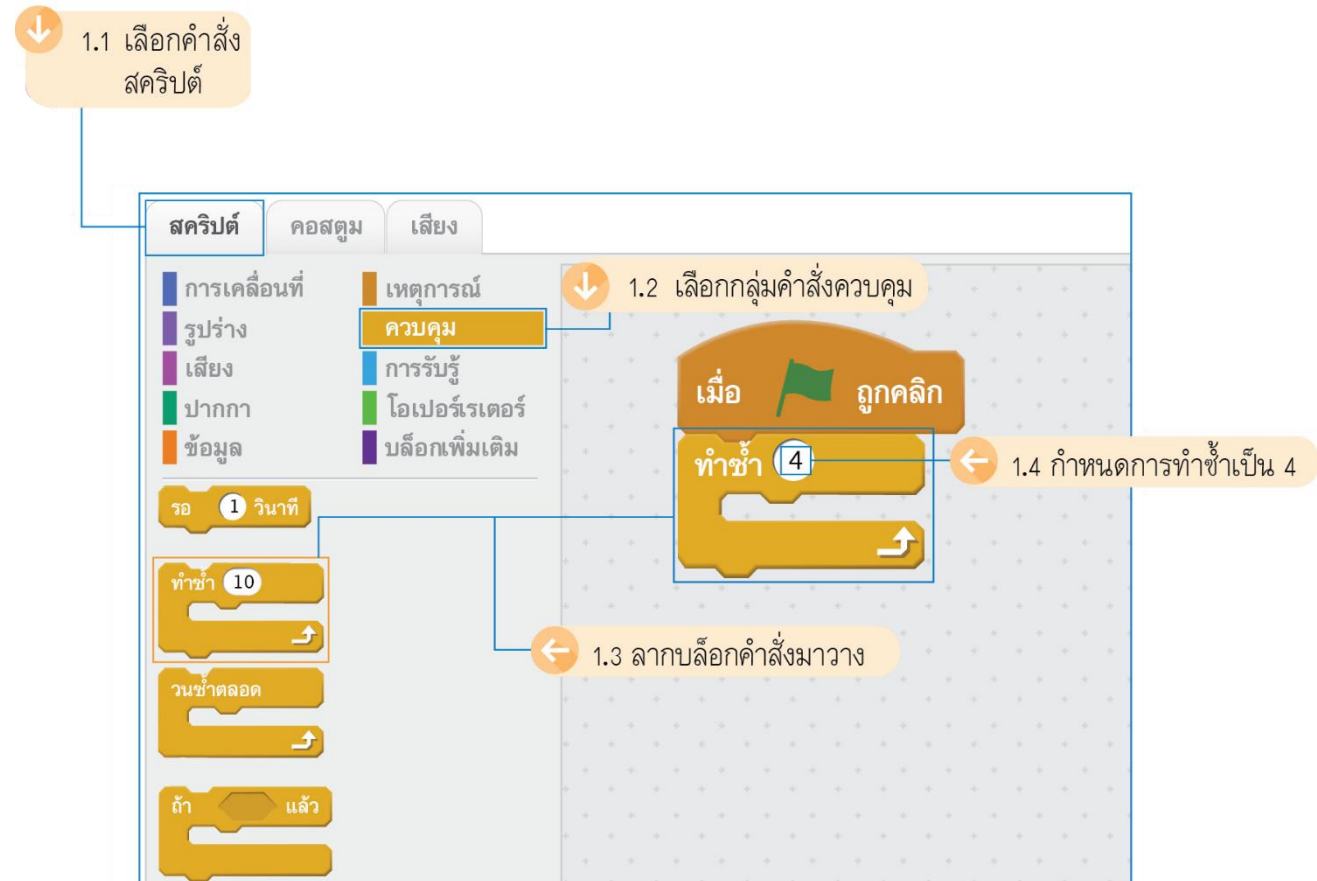
วิธีปฏิบัติ

โปรแกรม Scratch สามารถวาดภาพได้โดยการให้โปรแกรมวางปากกา ดังนั้นเราสามารถให้โปรแกรมวาดรูปร่างต่าง ๆ ตามที่เรากำหนดได้ เช่น ถ้าหากต้องการให้วาดรูปสี่เหลี่ยมจัตุรัส อาจให้โปรแกรมเคลื่อนที่ปากกาไปเป็นระยะทางเท่ากับขนาดของรูปสี่เหลี่ยม จากนั้นเลี้ยวเป็นมุม 90 องศา แล้วทำซ้ำ 4 ครั้ง ดังนี้

- นำคำสั่งทำซ้ำมาวาง แล้วกำหนดให้ทำซ้ำ 4 ครั้ง

วัสดุอุปกรณ์

คอมพิวเตอร์ 1 เครื่อง

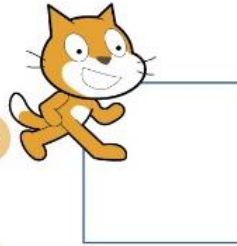


วิธีปฏิบัติ (ต่อ)

2. นำบล็อกคำสั่ง **จรดปากกา** มาวางในลูป แล้วตามด้วยการเคลื่อนที่เป็นระยะ 100 จากนั้นเลี้ยวเป็นมุม 90 องศา แล้วหน่วงเวลา 1 วินาที ดังภาพ และทดลองให้โปรแกรมทำงาน

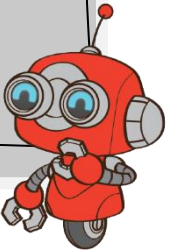


2.5 ผลการทดสอบโปรแกรม →



- ← 2.1 นำบล็อกคำสั่งจรดปากกามาวาง
- ← 2.2 นำบล็อกคำสั่งเคลื่อนมาวาง แล้วกำหนดการเคลื่อนที่เป็น 100
- ← 2.3 นำบล็อกคำสั่งหมุนมาวาง แล้วกำหนดมุมเป็น 90 องศา
- ← 2.4 นำบล็อกคำสั่งรอมาวาง แล้วหน่วงเวลาเป็น 1 วินาที

ถ้าเรามีความรู้ด้านการเขียนโปรแกรม เราสามารถนำไปประยุกต์กับการแก้ปัญหาต่าง ๆ ได้มากมาย โดยการเขียนโปรแกรมเพื่อแก้ปัญหา สำหรับการเขียนโปรแกรมที่มีเงื่อนไข ผู้ออกแบบโปรแกรมจะต้องเขียนอัลกอริทึมที่มีการตรวจสอบเงื่อนไขที่ครอบคลุมทุกกรณีเพื่อให้ได้ผลลัพธ์ ที่ถูกต้องตามต้องการ



ถ้านักเรียนต้องการให้ตัวละครลากเส้นเป็นสามเหลี่ยม นักเรียนจะออกแบบอัลกอริทึมสำหรับเขียนโปรแกรมอย่างไร

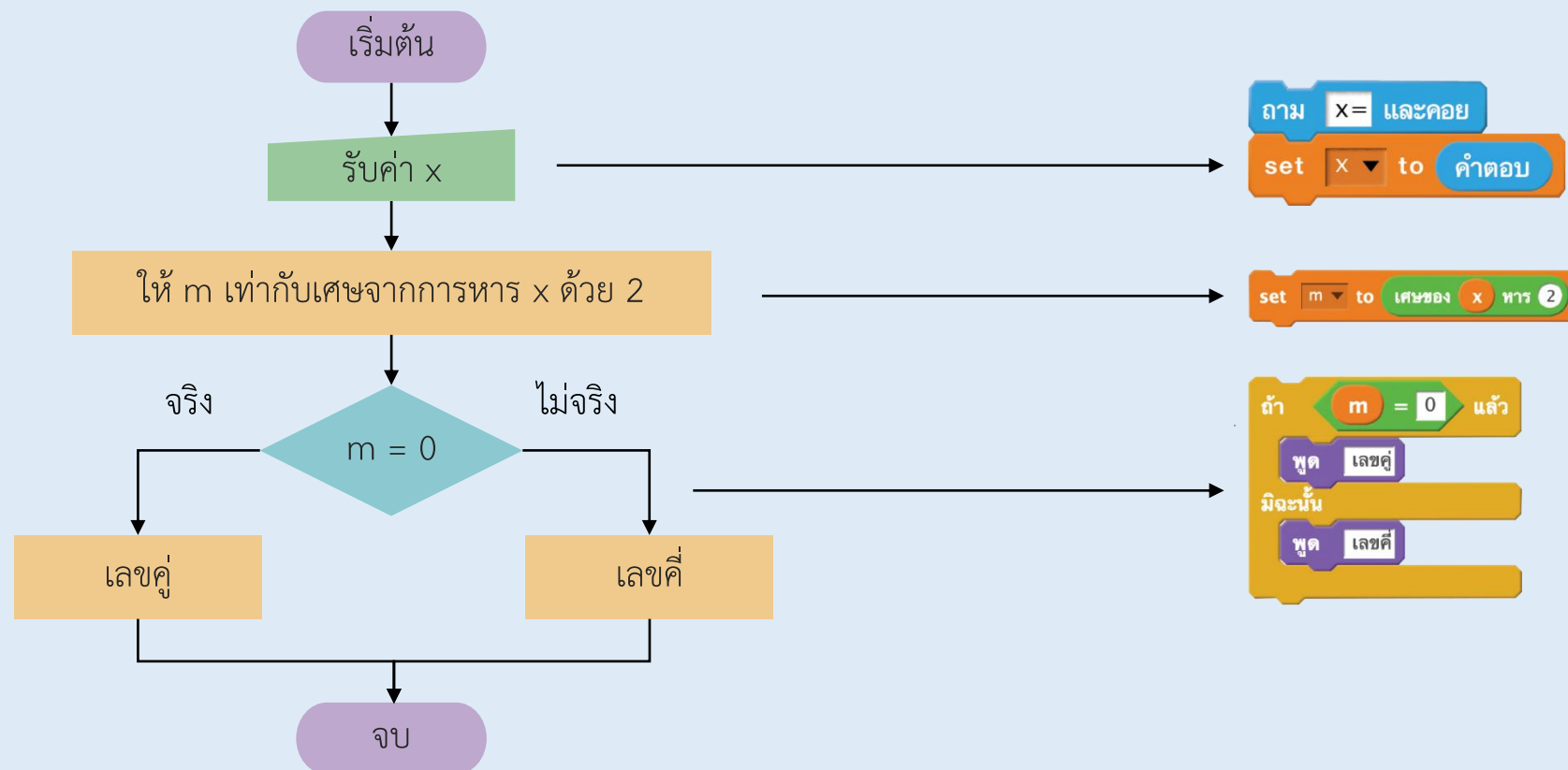


คำถามสำคัญ



การเขียนโปรแกรมตรวจสอบเลขคู่และเลขคี่

ถ้าต้องการเขียนโปรแกรมเพื่อตรวจสอบค่าตัวเลขที่รับเข้าทางแป้นพิมพ์ว่า ตัวเลขนั้นเป็นเลขคู่หรือเลขคี่ โปรแกรมลักษณะนี้สามารถใช้ Scratch เขียนได้เช่นกัน หากพิจารณาเงื่อนไขข้อแตกต่างของเลขคู่และเลขคี่แล้ว จะพบว่าเลขคู่เมื่อนำมาหารด้วย 2 แล้วจะหารได้ลงตัว หรือหารแล้วมีเศษจากการหารเป็นศูนย์นั่นเอง โดยเราสามารถเขียนอัลกอริทึมเทียบกับคำสั่งของ Scratch ได้ดังนี้



วัตถุประสงค์

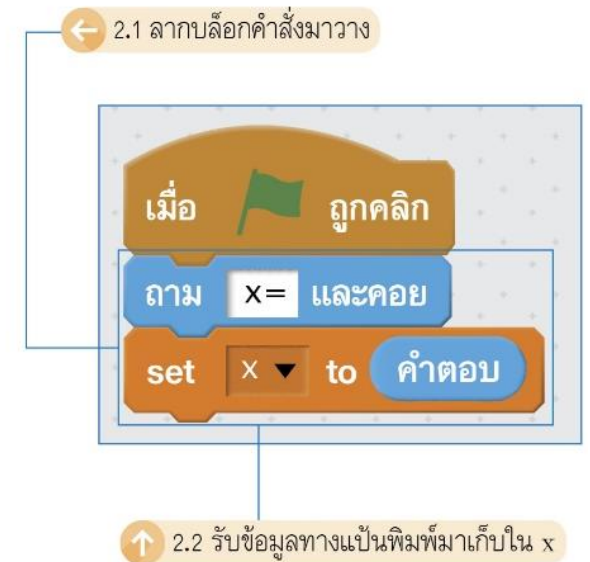
เขียนโปรแกรมตรวจสอบเลขคู่และเลขคี่ได้

วัสดุอุปกรณ์

คอมพิวเตอร์ 1 เครื่อง

วิธีปฏิบัติ

1. เปิดโปรแกรม Scratch พร้อมประกาศตัวแปรขึ้นมา 2 ตัว โดยตัวแปร x ใช้เก็บตัวเลขที่รับเข้าไป ตัวแปร m เก็บเศษที่ได้จากการหารด้วย 2
2. เขียนโปรแกรมรับค่ามาเก็บในตัวแปร x โดยลากบล็อกคำสั่งมาวาง ดังภาพ



วิธีปฏิบัติ (ต่อ)

3. กำหนดให้ m เก็บเศษที่ได้จากการหาร x ด้วย 2 โดยการหาเศษที่ได้จากการหารจะใช้บล็อกคำสั่ง **เศษของ** **หาร** ซึ่งอยู่ในกลุ่มคำสั่งโอเปอร์เรเตอร์ ดังภาพ

3.1 เลือกคำสั่งสคริปต์

3.2 เลือกกลุ่มคำสั่งโอเปอร์เรเตอร์

3.3 ลากบล็อกคำสั่ง Set คำตอบ แล้วกำหนดค่าเป็น m

3.4 นำบล็อกคำสั่งเศษของหาร มาวาง

4. นำตัวแปร x มาวาง แล้วเติม 2 ลงไป ดังภาพ

4.1 นำตัวแปร x มาวาง

4.2 กำหนดค่าเศษของหาร เป็น 2

วิธีปฏิบัติ (ต่อ)

5. นำคำสั่งเงื่อนไข มาวาง โดยเลือกคำสั่ง



The screenshot shows the Scratch script editor for step 5.1. The title bar says '5.1 เลือกคำสั่งสคริปต์'. The left sidebar shows the 'สคริปต์' (Scripts) category selected. The main workspace contains a script starting with 'เมื่อ ธง ถูกคลิก' (When green flag clicked), followed by 'ถาม x= และคอย' (Ask x= and wait), 'set x to คำตอบ' (Set x to answer), and 'set m to เศษของ x หาร 2' (Set m to remainder of x divided by 2). Below these are the 'ถ้า...แล้ว' and 'มิฉะนั้น' blocks. A yellow callout box with an arrow points to the 'set m to...' block, labeled '5.2 เลือกกลุ่มคำสั่งควบคุม' (5.2 Select control command group). Another yellow callout box with an arrow points to the 'ถ้า...แล้ว' block, labeled '5.3 คำสั่งเงื่อนไขเลือกทำมาวาง' (5.3 Place conditional command).

6. นำตัวดำเนินการเท่ากับ มาวาง แล้วนำตัวแปร m กับเลข 0 มาใส่เพื่อตรวจสอบว่าเศษจากการหาร เป็นศูนย์หรือไม่ ดังภาพ

The screenshot shows the Scratch script editor for step 6.1. The title bar says '6.1 เลือกคำสั่งสคริปต์'. The left sidebar shows the 'ข้อมูล' (Data) category selected. The main workspace contains a script starting with 'เมื่อ ธง ถูกคลิก' (When green flag clicked), followed by 'ถาม x= และคอย' (Ask x= and wait), 'set x to คำตอบ' (Set x to answer), and 'set m to เศษของ x หาร 2' (Set m to remainder of x divided by 2). Below these are the 'ถ้า...แล้ว' and 'มิฉะนั้น' blocks. A yellow callout box with an arrow points to the 'set m to...' block, labeled '6.2 เลือกกลุ่มคำสั่งข้อมูล' (6.2 Select data command group). Another yellow callout box with an arrow points to the 'ถ้า...แล้ว' block, labeled '6.3 นำบล็อกคำสั่งตัวดำเนินการเท่ากับ มาวางแล้วกำหนดค่าเป็น m = และ 0' (6.3 Place the equals operator block and set the value to m = and 0). The 'ถ้า...แล้ว' block contains the code 'm = 0'.

วิธีปฏิบัติ (ต่อ)

7. ถ้าเงื่อนไขเป็นจริง ให้ตอบว่าเป็นเลขคู่ ถ้าเงื่อนไขเป็นเท็จ ให้ตอบว่าเป็นเลขคี่ โดยนำบล็อกคำสั่ง **พูด Hello!** มาวาง แล้วเปลี่ยนข้อความเป็นเลขคู่ และเลขคี่ ดังภาพ

7.1 เลือกคำสั่งสคริปต์

7.2 เลือกกลุ่มคำสั่งรูปร่าง

7.3 ลากบล็อกคำสั่งพูด มาวาง 2 บล็อก แล้วกำหนดบล็อกแรกเป็นเลขคู่ กำหนดบล็อกที่ 2 เป็นเลขคี่

8. ทดลองรันโปรแกรมแล้วป้อนค่าตัวเลขที่ต้องการตรวจสอบเข้าไป โปรแกรมจะแสดงผลข้อมูลออกมาดังภาพ

8.1 ป้อนค่า 24

8.2 โปรแกรมแสดงผล



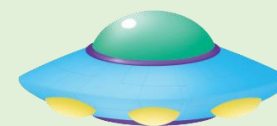
การเขียนโปรแกรมคำนวณความสมส่วนของร่างกาย

วิธีการคำนวณน้ำหนักที่เหมาะสมมีอยู่หลายวิธี แต่วิธีการหนึ่งที่คำนวณได้ง่าย ๆ จะคำนวณได้จาก 22.9 คูณกับความสูงเป็นเมตรยกกำลังสอง

$$\text{น้ำหนักที่เหมาะสม} = 22.9 \times (\text{ความสูงเป็นเมตร})^2$$

ถ้าน้ำหนักของเรามากกว่าค่าที่เหมาะสมแสดงว่าเราเริ่มอ้วน ถ้ามากกว่ามากแสดงว่าอ้วนมาก ถ้าน้อยกว่าแสดงว่าผอม และถ้าน้อยกว่ามากแสดงว่าผอมมากนั่นเอง

นักเรียนคำนวณหาค่าความสมส่วนของร่างกายเราอย่างไร



คำถามสำคัญ

วัตถุประสงค์

เขียนโปรแกรมคำนวณความสมส่วนของร่างกายได้

วัสดุอุปกรณ์

คอมพิวเตอร์ 1 เครื่อง

วิธีปฏิบัติ

กิจกรรมนี้จะเขียนโปรแกรมให้ป้อนค่าความสูงในหน่วยของเซนติเมตรแล้วให้คอมพิวเตอร์แจ้งน้ำหนักที่เหมาะสมออกมา โดยเราสามารถออกแบบโปรแกรมได้ ดังนี้

เริ่มต้น

1. รับค่าความสูงเป็นเซนติเมตรมาเก็บใน H
2. น้ำหนักที่เหมาะสมเท่ากับ $22.9 \times (H/100) \times (H/100)$
3. แสดงน้ำหนักที่เหมาะสม

จบ

วิธีปฏิบัติ (ต่อ)

1. เปิดโปรแกรม Scratch แล้วประกาศตัวแปรขึ้นมา 3 ตัว โดยตัวแปร B เป็นค่าน้ำหนักที่เหมาะสมที่คำนวณได้ ตัวแปร H เก็บความสูงเป็นเซนติเมตร ตัวแปร M เป็นน้ำหนักเป็นกิโลกรัม

The screenshot shows the Scratch 'Scripts' menu with the following annotations:

- 1.1 เลือกคำสั่งสคริปต์ (Select script command)
- 1.2 เลือกกลุ่มคำสั่งข้อมูล (Select data command group)
- 1.3 ประกาศตัวแปร 3 ตัว (Declare 3 variables)

The 'สร้างตัวแปร' (Create variable) section shows three variables: B, H, and M, all checked. Below the menu, the following code blocks are visible:

- set M to 0
- เปลี่ยน M ทีละ 1 (Change M by 1)
- แสดงตัวแปร M (Show variable M)
- ซ่อนตัวแปร M (Hide variable M)

2. เขียนโปรแกรมให้ถามว่าคุณสูงเท่าไร แล้วนำมาเก็บในตัวแปร H

The screenshot shows the Scratch 'When Clicked' block with the following annotations:

- 2.1 นำบล็อกคำสั่งถามและคอย มาวางแล้วเปลี่ยนข้อความ เป็น "คุณสูงเท่าไร" (Place the ask and wait block and change the text to "How tall are you")
- 2.2 ลากบล็อกคำสั่ง Set to มาวางแล้ว กำหนดเป็น H (Drag the Set to block and set it to H)
- 2.3 ลากบล็อกคำสั่งคำตอบมาวาง (Drag the answer block)

The code block structure is: 'เมื่อ ถูกคลิก' (When clicked) -> 'ถาม คุณสูงเท่าไร และคอย' (Ask "How tall are you" and wait) -> 'set H to คำตอบ' (Set H to the answer).

วิธีปฏิบัติ (ต่อ)

3. คำนวณค่าน้ำหนักที่เหมาะสมมาเก็บในตัวแปร B แต่การคำนวณต้องนำ 22.9 คูณกับ (H/100) คูณ (H/100) ดังนั้นนำตัวดำเนินการคูณมากำหนดค่าให้ B แล้วกำหนดตัวตั้งเป็น 22.9 ดังภาพ

The screenshot shows the Scratch code editor with a script block containing the following steps:

- 3.1 เลือกคำสั่งสคริปต์ (Select script command)
- 3.2 เลือกกลุ่มคำสั่งโอเพอร์เรเตอร์ (Select operator command group)
- 3.3 ลากบล็อกคำสั่ง Set to มาวาง (Drag the Set to block)
- 3.4 กำหนดตัวแปรเป็น B (Set variable to B)
- 3.5 ลากบล็อกคำสั่ง ตัวดำเนินการมาวาง (Drag the operator block)
- 3.6 กำหนดให้ B เท่ากับ 22.9 คูณกับค่าที่จะตามมา (Set B to 22.9 multiplied by the next value)

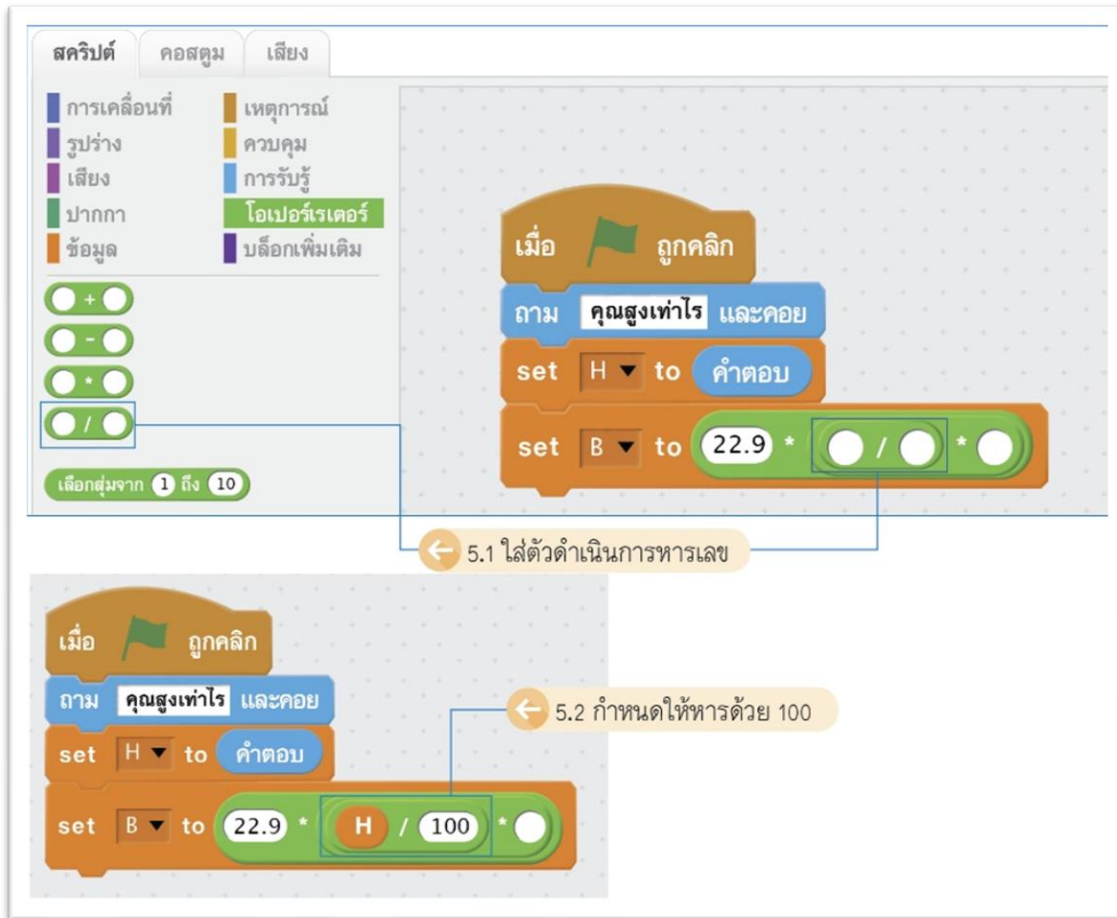
The code block is: `เมื่อ ถูกคลิก` → `ถาม คุณสูงเท่าไร และคอย` → `set H to คำตอบ` → `set B to 22.9 * []`

4. นำตัวดำเนินการคูณมาวางเพิ่มอีกหนึ่งตัว เนื่องจากการคำนวณต้องคูณตัวเลข 2 ครั้ง ดังภาพ

The close-up shows the `set B to` block with the value `22.9 * [] * []`. A callout points to the second multiplication operator, labeled "4.1 เพิ่มตัวดำเนินการคูณ" (Add another multiplication operator).

วิธีปฏิบัติ (ต่อ)

5. คำนวณค่าความสูงหารด้วย 100 โดยใส่ตัวดำเนินการหารลงไป แล้วกำหนดให้หารด้วย 100 ดังภาพ




The image shows two Scratch code snippets for step 5. The top snippet is a 'when green flag clicked' event with the following blocks: 'ask question "คุณสูงเท่าไร" and wait', 'set H to answer', and 'set B to 22.9 * (/ *)'. A callout bubble points to the division block with the text '5.1 ใส่ตัวดำเนินการหารเลข'. The bottom snippet is another 'when green flag clicked' event with the following blocks: 'ask question "คุณสูงเท่าไร" and wait', 'set H to answer', and 'set B to 22.9 * (H / 100 *)'. A callout bubble points to the division block with the text '5.2 กำหนดให้หารด้วย 100'.

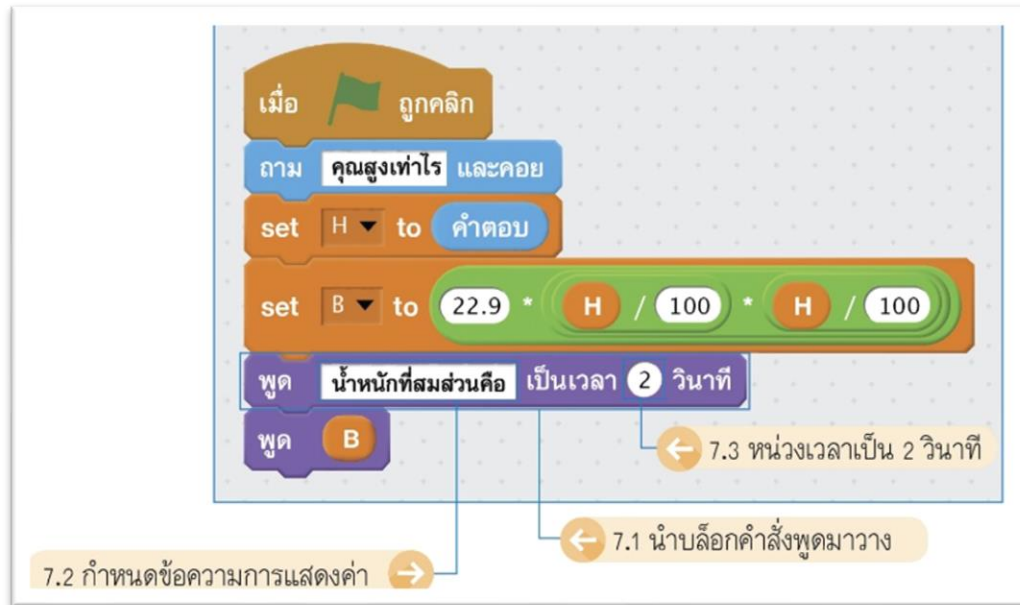
6. ใส่ตัวดำเนินการหารเพื่อกำหนดให้คุณด้วยความสูงแล้วหารด้วย 100 อีกครั้ง ดังภาพ



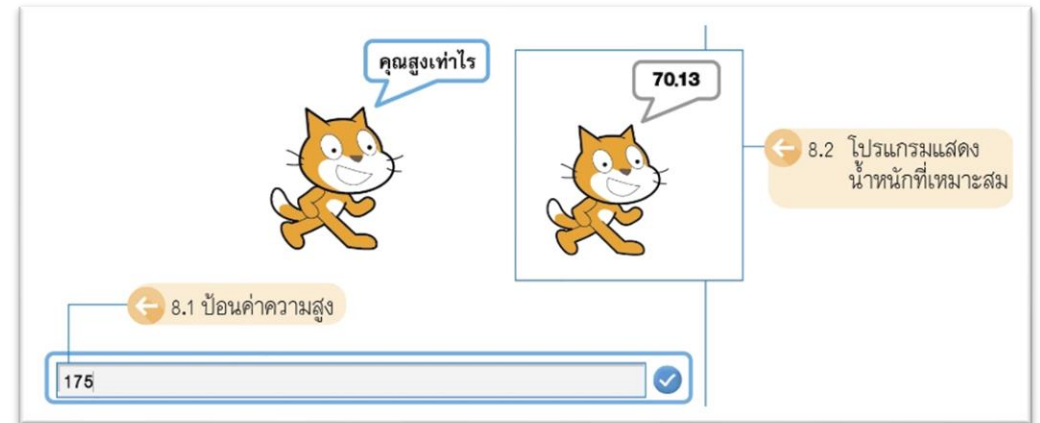
The image shows a Scratch code snippet for step 6. It starts with a 'when green flag clicked' event, followed by 'ask question "คุณสูงเท่าไร" and wait', 'set H to answer', and 'set B to 22.9 * (H / 100 * H / 100)'. A callout bubble points to the second division block with the text '6.2 ใส่ตัวดำเนินการแล้วกำหนดค่า H เป็น 100'.

วิธีปฏิบัติ (ต่อ)

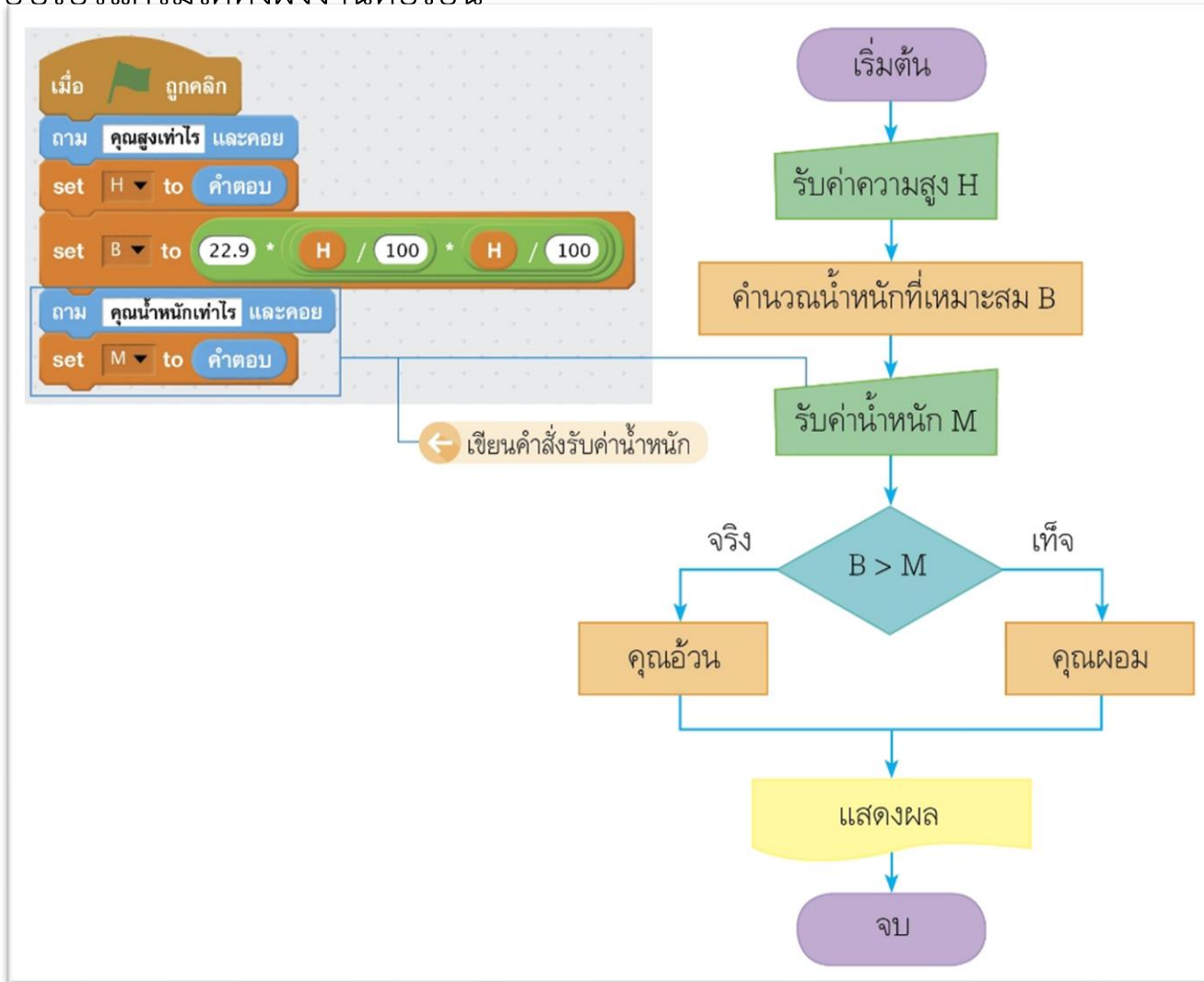
7. เขียนโปรแกรมให้แสดงค่าตัวแปร B ที่คำนวณได้ โดยนำบล็อกคำสั่ง  มาวาง โดยให้แสดงค่าน้ำหนักที่สมส่วนคือแล้วหน่วงเวลา 2 วินาที จากนั้นแสดงค่าในตัวแปร B



8. ทดลองรันโปรแกรม จะพบว่าแมวจะแสดงข้อความว่า “คุณสูงเท่าไร” จากนั้นทดลองป้อนค่า 175 ลงไป จะพบว่าโปรแกรมจะแจ้งว่าน้ำหนักที่เหมาะสมคือ 70.13 ดังภาพ



จากกิจกรรมเป็นการเขียนโปรแกรมให้คำนวณอย่างง่าย เราสามารถพัฒนาโปรแกรมต่อ โดยให้คอมพิวเตอร์ถามว่าเราน้ำหนักเท่าไร เมื่อป้อนน้ำหนักลงไปแล้ว แจ้งว่าเราอ้วนหรือเราผอม โดยออกแบบโปรแกรมได้ดังผังงานต่อไปนี้



สำหรับการตรวจสอบเงื่อนไขสามารถนำ
บล็อกคำสั่ง

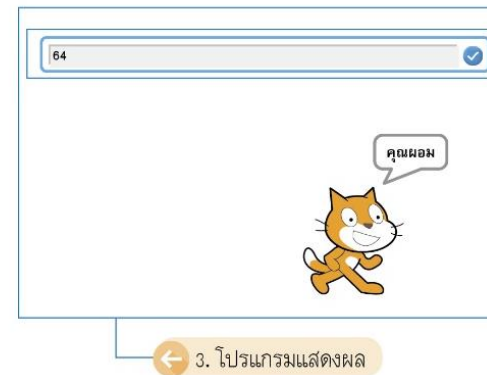


มาใช้งานได้ ดังภาพ

```
เมื่อ ถูกคลิก
ถาม คุณสูงเท่าไร และคอย
set H to คำตอบ
set B to 22.9 * H / 100 * H / 100
ถาม คุณน้ำหนักเท่าไร และคอย
set M to คำตอบ
ถ้า B > M แล้ว
พูด คุณอ้วน
มิฉะนั้น
พูด คุณผอม
```

← บล็อกคำสั่งตรวจสอบเงื่อนไข ถ้า B > M แจ้งว่า “คุณอ้วน” ถ้าไม่จริงแจ้งว่า “คุณผอม”

เมื่อทดลองรันโปรแกรมแล้วป้อนความสูงเป็น 175
ป้อนน้ำหนักเป็น 64 โปรแกรมจะแจ้งว่า “คุณผอม”





แบบพัฒนาทักษะในการทำข้อสอบปรนัยเพื่อประเมินผลตัวชี้วัด



หน่วยการเรียนรู้ที่ 2 การเขียนโปรแกรมเบื้องต้น

1. เครื่องพิมพ์เป็นอุปกรณ์ที่ทำหน้าที่เกี่ยวกับอะไร

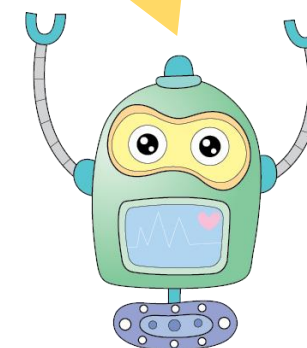
1) หน่วยรับเข้า

2) หน่วยส่งออก

3) หน่วยจัดเก็บข้อมูล

4) หน่วยประมวลผล

ตอบ 2) เพราะเครื่องพิมพ์
มีหน้าที่พิมพ์ข้อมูลออกมา
บนกระดาษ จึงเป็นการทำงาน
ของหน่วยส่งออก



2. คอมพิวเตอร์จะทำงานตามที่เราต้องการได้ ต้องมีอะไรเป็นตัวสั่งการ

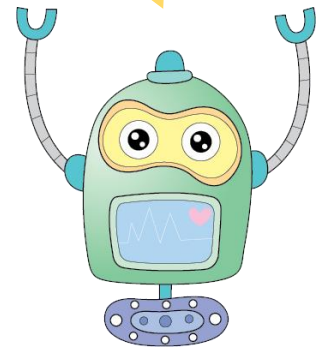
1) เสียง

2) จอภาพ

3) โปรแกรม

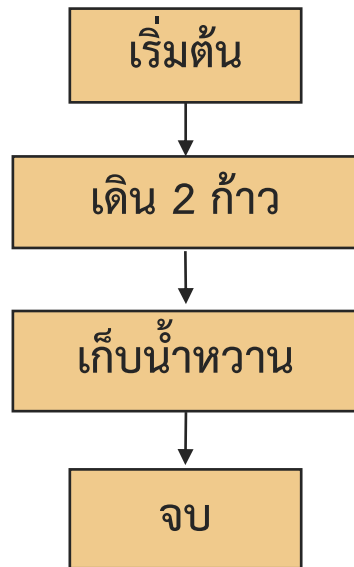
4) หน่วยประมวลผลกลาง

ตอบ 3) เพราะการที่คอมพิวเตอร์
จะทำงานตามที่เราต้องการได้
ต้องมีโปรแกรมเป็นตัวสั่งการ

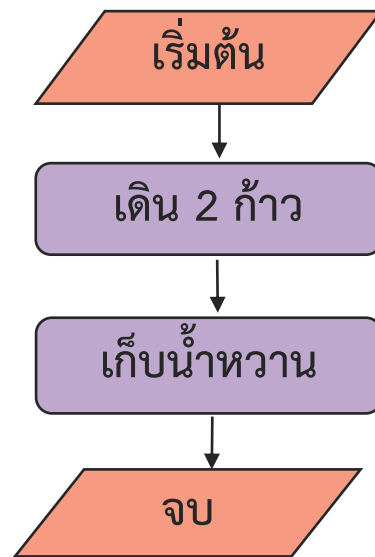


3. การเขียนอัลกอริทึมให้ฝั่งเดินไปเก็บน้ำหวาน ข้อใดใช้สัญลักษณ์ถูกต้อง

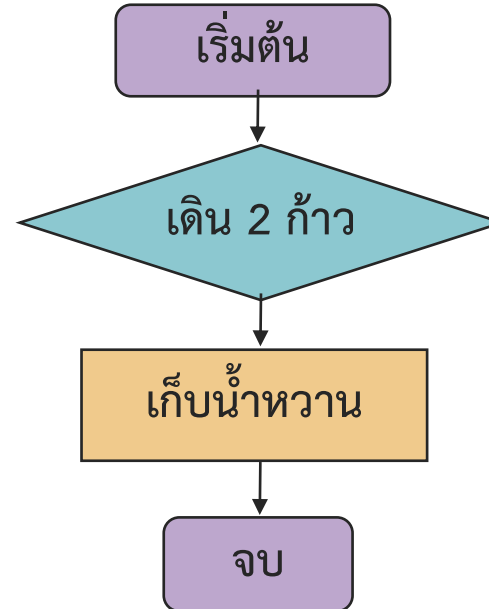
1)



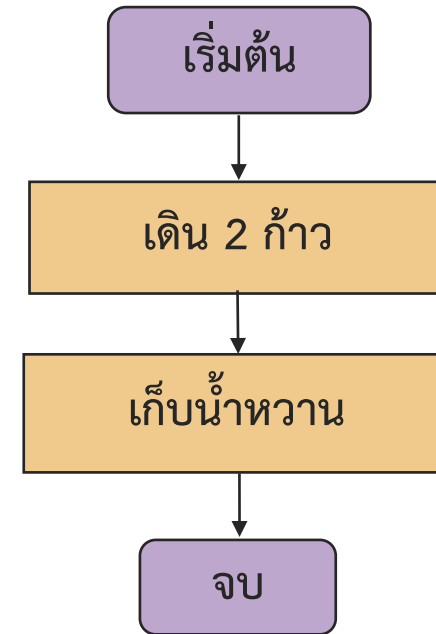
2)



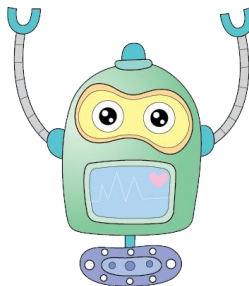
3)



4)



ตอบ 4) เพราะเป็นการใช้สัญลักษณ์
อัลกอริทึมที่ถูกต้อง



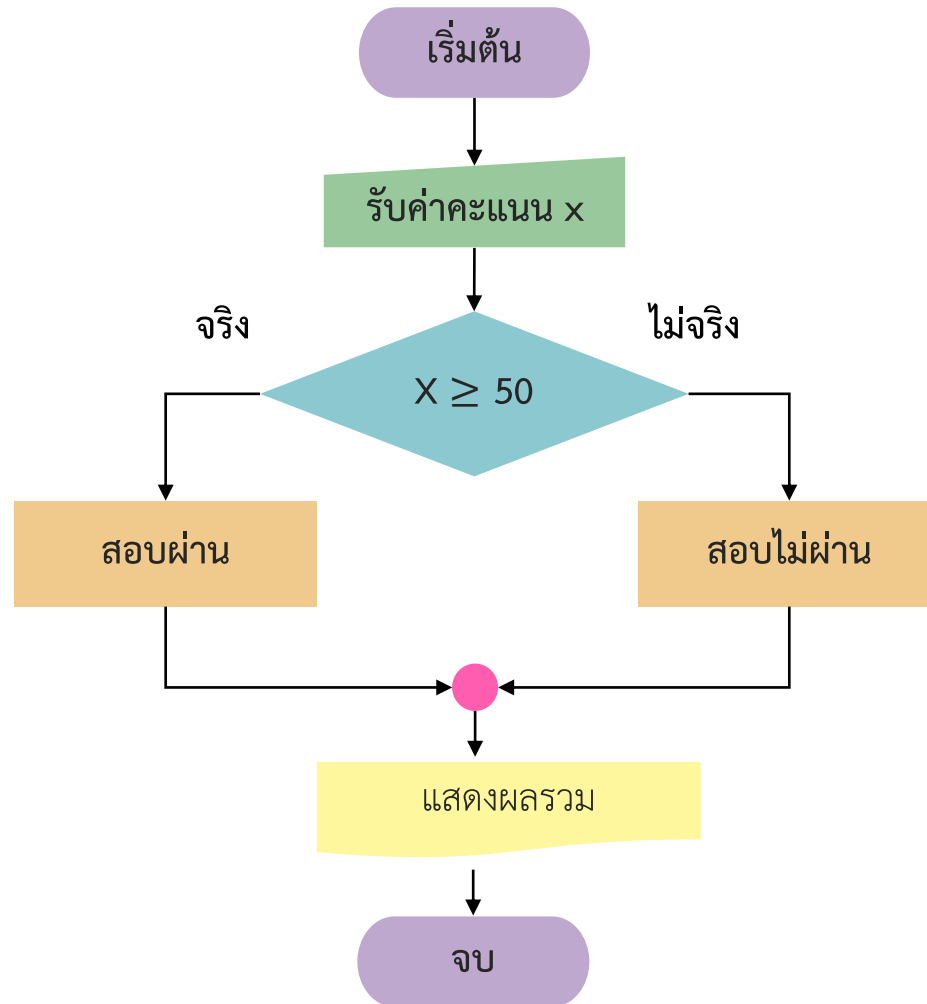
4. นักเรียนพิจารณาอัลกอริทึมว่าคะแนนสอบข้อใดสอบผ่านทั้งหมด

1) 49 50 54

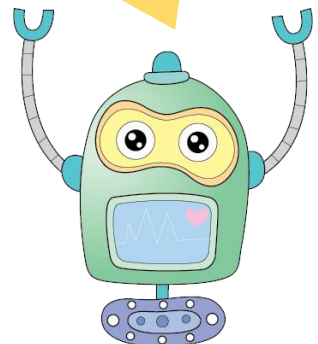
2) 50 38 60

3) 50 52 61

4) 49 55 70



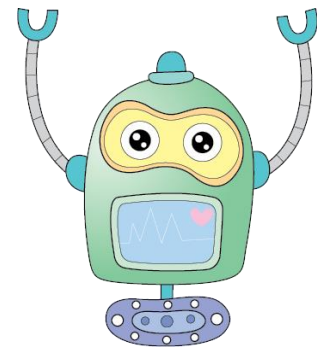
ตอบ 3) เพราะเงื่อนไข
ในอัลกอริทึมกำหนดให้คะแนน
ที่ได้ต้องมากกว่า หรือเท่ากับ
50 จึงจะสอบผ่าน ดังนั้น 50
52 61 จึงผ่านเงื่อนไขสอบผ่าน



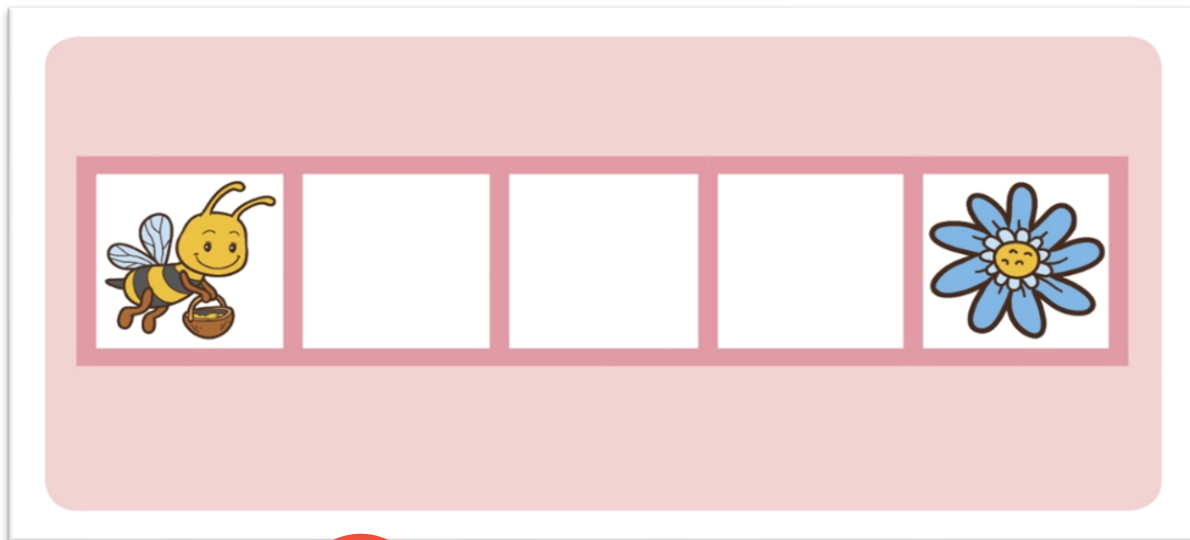
5. ถ้าต้องการฝึกเขียนโปรแกรม นักเรียนจะเข้าเว็บไซต์ใด

- 1) www.google.co.th
- 2) www.youtube.com
- 3) www.wikipedia.org
- 4) <https://code.org>

ตอบ 4) เพราะ <https://code.org>
เป็นเว็บไซต์สำหรับการเขียน
โปรแกรมเบื้องต้นจนถึงระดับยาก



6. ข้อใดเป็นบล็อกคำสั่งให้ผึ้งเดินไปเก็บน้ำหวาน



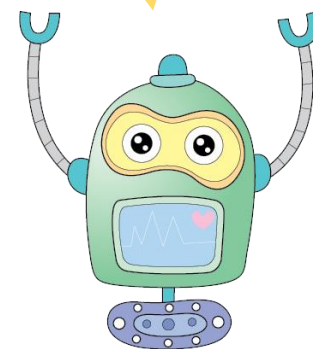
ตอบ 2) เพราะจากภาพ
เมื่อนับช่องเพื่อไปเก็บน้ำหวานแล้ว
จะได้ 4 ช่อง ดังนั้นการทำซ้ำ
จึงเป็น 4 ครั้ง

1)

2)

3)

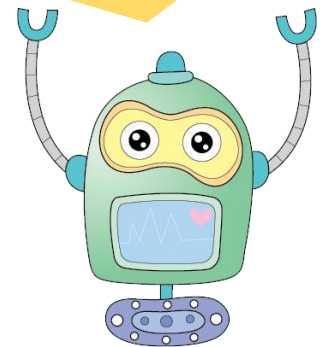
4)



7. ข้อใดเป็นการตรวจสอบข้อผิดพลาดของโปรแกรม

- 1) ตรวจสอบคำสั่งแรกเพียงคำสั่งเดียว
- 2) ตรวจสอบการทำงานคำสั่งสุดท้าย
- 3) ตรวจสอบทีละคำสั่งจากบนลงล่าง
- 4) ตรวจสอบทีละคำสั่งจากล่างขึ้นบน

ตอบ 3) เพราะการตรวจสอบ
ข้อผิดพลาดจากการเขียน
โปรแกรมจะตรวจสอบ
จากคำสั่งบนสุดลงมาล่างสุด
เพื่อให้แก้ไขได้อย่างเป็น
ลำดับขั้นตอน



8. ข้อใดเป็นบล็อกคำสั่งที่อยู่ในกลุ่มคำสั่งการเคลื่อนที่

1)



2)



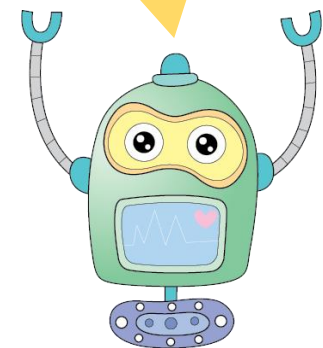
3)



4)



ตอบ 1) เพราะจากบล็อกคำสั่งเป็นการสั่งให้ตัวละครเคลื่อนที่โดยหมุนขวา 15 องศา ดังนั้นจึงอยู่ในกลุ่มคำสั่งการเคลื่อนที่



9. การเขียนโปรแกรม Scratch เพื่อให้ตัวละครเคลื่อนที่เหมือนจริง ควรเลือกใช้บล็อกคำสั่งใดต่อไปนี้

1)



2)



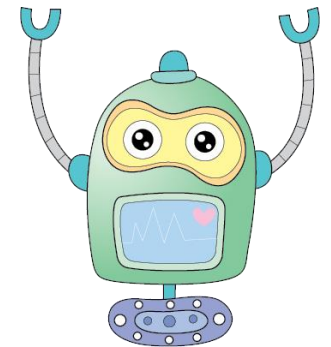
3)



4)



ตอบ 3) เพราะการสร้าง
ตัวละครเคลื่อนที่เหมือนจริง
ต้องมีการทำซ้ำ และต้องมีการ
รอเวลา



10. การเขียนโปรแกรมบวกเลข ซึ่งประกาศตัวแปร a b และ sum ขึ้นมา โดยรับค่าแรกเก็บใน a รับค่าที่สองเก็บใน b แล้วนำ a บวกกับ b เก็บผลบวกไว้ใน sum โปรแกรมจะแสดงผลที่ตัวแปรใด

1) แสดงผล a

2) แสดงผล b

3) แสดงผล sum

4) แสดงผล x และ y

ตอบ 3) เพราะ sum เก็บค่าผลบวกของ a และ b ไว้ ดังนั้นการแสดงผล ต้องแสดงที่ตัวแปร sum

