# บทที่ 5
# โครงสร้างการควบคุมโปรแกรมแบบทางเลือก
# Selection Control Structure

**บทนำ**

การแก้ปัญหาในชีวิตประจำวันและในการเขียนโปรแกรมมิได้มีลักษณะเป็นเส้นตรงเพียงอย่างเดียว หลายสถานการณ์จำเป็นต้องอาศัยการตัดสินใจภายใต้เงื่อนไขที่แตกต่างกัน เช่น หากฝนตกให้กางร่ม หากคะแนนผ่านเกณฑ์ให้แสดงผลว่าผ่าน แต่หากไม่ผ่านให้แสดงผลว่าไม่ผ่าน หรือหากจำนวนเงินไม่เพียงพอให้ยุติการซื้อสินค้า สถานการณ์ลักษณะนี้สะท้อนให้เห็นว่า การแก้ปัญหาหลายประเภทต้องอาศัยการเลือกทางตามเงื่อนไขที่กำหนดไว้ มิใช่เพียงการดำเนินการตามลำดับขั้นตอนแบบเรียงลำดับเท่านั้น

ในทางวิทยาการคำนวณ แนวคิดดังกล่าวได้รับการพัฒนาเป็น “โครงสร้างการควบคุมโปรแกรมแบบทางเลือก” ซึ่งเป็นหนึ่งในโครงสร้างพื้นฐานสำคัญของการเขียนโปรแกรม นอกเหนือจากโครงสร้างแบบเรียงลำดับและโครงสร้างแบบทำซ้ำ โครงสร้างแบบทางเลือกช่วยให้โปรแกรมสามารถตรวจสอบเงื่อนไข และเลือกปฏิบัติคำสั่งที่แตกต่างกันตามผลของเงื่อนไขนั้น ส่งผลให้โปรแกรมมีความยืดหยุ่นและสามารถตอบสนองต่อข้อมูลหรือสถานการณ์ได้อย่างเหมาะสม

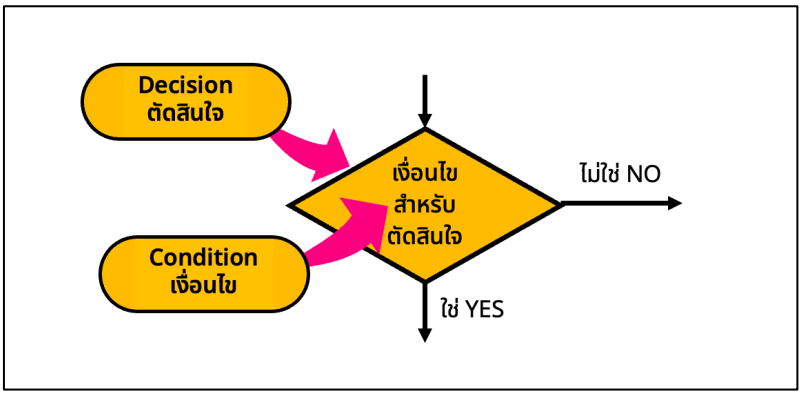
สำหรับนักศึกษาครู ความเข้าใจเรื่องโครงสร้างแบบทางเลือกมีความสำคัญอย่างมาก เพราะช่วยพัฒนาทักษะการคิดอย่างมีเหตุผล การตัดสินใจอย่างเป็นระบบ และการเชื่อมโยงเงื่อนไขกับผลลัพธ์ ซึ่งเป็นพื้นฐานทั้งของการเขียนโปรแกรมและของการจัดการเรียนรู้ในชีวิตจริง อีกทั้งครูยังจำเป็นต้องสามารถออกแบบกิจกรรมการเรียนรู้ที่ช่วยให้ผู้เรียนระดับประถมศึกษาและมัธยมศึกษาเข้าใจเรื่องเงื่อนไขและการตัดสินใจได้อย่างเป็นรูปธรรม ผ่านสถานการณ์ใกล้ตัว เกม กิจกรรมจำลอง และการใช้โปรแกรมช่วยสอน เช่น Flowgorithm

บทนี้มุ่งอธิบายหลักการตัดสินใจและเงื่อนไขในการแก้ปัญหา ความหมายและหลักการของโครงสร้างการควบคุมแบบทางเลือก การออกแบบผังงานแบบทางเลือกด้วยคำสั่งเงื่อนไข การใช้โปรแกรม Flowgorithm ในการเขียนและทดสอบผังงานแบบทางเลือก ตลอดจนการจัดทำแผนการจัดการเรียนรู้และกิจกรรมการสอนเรื่องโครงสร้างแบบทางเลือก เพื่อให้นักศึกษาครูมีความเข้าใจเชิงเนื้อหาและสามารถนำความรู้ไปใช้ในการจัดการเรียนรู้ได้อย่างมีประสิทธิภาพ

## 5.1 หลักการตัดสินใจและเงื่อนไขในการแก้ปัญหา

การตัดสินใจ หรือ Decision เป็นกระบวนการสำคัญที่มนุษย์ใช้ในการเลือกทางปฏิบัติภายใต้สถานการณ์ที่มีทางเลือกมากกว่าหนึ่งทาง โดยการตัดสินใจมักอาศัยเงื่อนไข หรือ เกณฑ์บางประการเป็นตัวกำหนดว่า ควรเลือกดำเนินการในรูปแบบใด ในบริบทของวิทยาการคำนวณ การตัดสินใจจึงหมายถึงกระบวนการตรวจสอบเงื่อนไขเพื่อกำหนดแนวทางการทำงานของโปรแกรมให้เหมาะสมกับข้อมูลที่ได้รับ

เงื่อนไข หรือ Condition คือ ข้อกำหนดหรือข้อความที่สามารถตรวจสอบได้ว่าเป็นจริงหรือเป็นเท็จ เช่น อายุมากกว่า 18 ปี คะแนนมากกว่าหรือเท่ากับ 50 อุณหภูมิสูงกว่า 37.5 องศาเซลเซียส หรือจำนวนเงินคงเหลือน้อยกว่าราคาสินค้า เงื่อนไขเหล่านี้มีบทบาทสำคัญต่อการตัดสินใจ เพราะผลลัพธ์ของการตรวจสอบเงื่อนไขจะเป็นตัวกำหนดว่าระบบหรือโปรแกรมควรดำเนินการอย่างไรต่อไป




ในชีวิตประจำวัน มนุษย์ใช้หลักการตัดสินใจและเงื่อนไขอยู่เสมอ เช่น หากไฟแดงให้หยุดรถ หากอากาศร้อนให้เปิดพัดลม หากฝนตกให้พกร่ม หรือหากทำการบ้านเสร็จจึงจะออกไปเล่นได้ ตัวอย่างเหล่านี้แสดงให้เห็นว่า การตัดสินใจไม่ใช่เรื่องไกลตัว แต่เป็นวิธีคิดพื้นฐานที่ใช้ในการดำเนินชีวิตประจำวันอย่างต่อเนื่อง

ในทางการเขียนโปรแกรม หลักการตัดสินใจมีความสำคัญอย่างมาก เพราะโปรแกรมจำนวนมากต้องตอบสนองต่อข้อมูลที่แตกต่างกัน ตัวอย่างเช่น โปรแกรมตรวจสอบผลการเรียนต้องตัดสินว่า “ผ่าน” หรือ “ไม่ผ่าน” ตามคะแนนที่ได้รับ โปรแกรมคำนวณส่วนลดต้องพิจารณาเงื่อนไขว่าซื้อครบจำนวนที่กำหนดหรือไม่ หรือโปรแกรมตรวจสอบเลขคู่เลขคี่ต้องตัดสินผลตามค่าที่ผู้ใช้ป้อนเข้ามา หากไม่มีแนวคิดเรื่องเงื่อนไข โปรแกรมจะไม่สามารถทำงานอย่างยืดหยุ่นได้

การเขียนเงื่อนไขในทางวิทยาการคำนวณมักอาศัยตัวดำเนินการเปรียบเทียบ เช่น มากกว่า (>), น้อยกว่า (<), มากกว่าหรือเท่ากับ (>=), น้อยกว่าหรือเท่ากับ (<=), เท่ากับ (= หรือ == ตามภาษาโปรแกรม), และไม่เท่ากับ (!= หรือ <>) รวมถึงตัวดำเนินการทางตรรกะ เช่น และ (AND), หรือ (OR), ไม่ (NOT) เพื่อใช้สร้างเงื่อนไขที่ซับซ้อนมากขึ้น

สำหรับผู้เรียน หรือ นักเรียน การทำความเข้าใจหลักการตัดสินใจและเงื่อนไขควรเริ่มจากสถานการณ์ใกล้ตัวที่เห็นภาพได้ง่าย เช่น การตัดสินว่าอากาศร้อนหรือไม่จากอุณหภูมิ การตัดสินผลสอบจากคะแนน หรือการเลือกกิจกรรมตามสภาพอากาศ จากนั้นจึงค่อยเชื่อมโยงไปสู่การเขียนข้อความเงื่อนไขอย่างเป็นระบบ และต่อยอดไปสู่การออกแบบผังงานและการเขียนโปรแกรม

สำหรับนักศึกษาครู ความเข้าใจเรื่องหลักการตัดสินใจมิได้มีความสำคัญเฉพาะต่อการเขียนโปรแกรม แต่ยังเป็นฐานในการออกแบบกิจกรรมการเรียนรู้ที่ฝึกให้ผู้เรียนคิดอย่างมีเหตุผล เช่น การให้ผู้เรียนวิเคราะห์สถานการณ์และสร้างเงื่อนไขด้วยตนเอง การเปรียบเทียบผลลัพธ์เมื่อเงื่อนไขต่างกัน หรือการอภิปรายเหตุผลว่าทำไมจึงเลือกแนวทางใดแนวทางหนึ่ง กิจกรรมลักษณะนี้จะช่วยส่งเสริมทั้งการคิดเชิงตรรกะและการสื่อสารความคิดอย่างเป็นระบบ

กล่าวโดยสรุป หลักการตัดสินใจและเงื่อนไขในการแก้ปัญหาเป็นพื้นฐานสำคัญของการเขียนโปรแกรมแบบทางเลือก เพราะช่วยให้ระบบสามารถเลือกการทำงานที่เหมาะสมกับสถานการณ์ได้อย่างมีเหตุผล ชัดเจน และตรวจสอบได้

## 5.2 โครงสร้างการควบคุมแบบทางเลือก (Selection Structure)

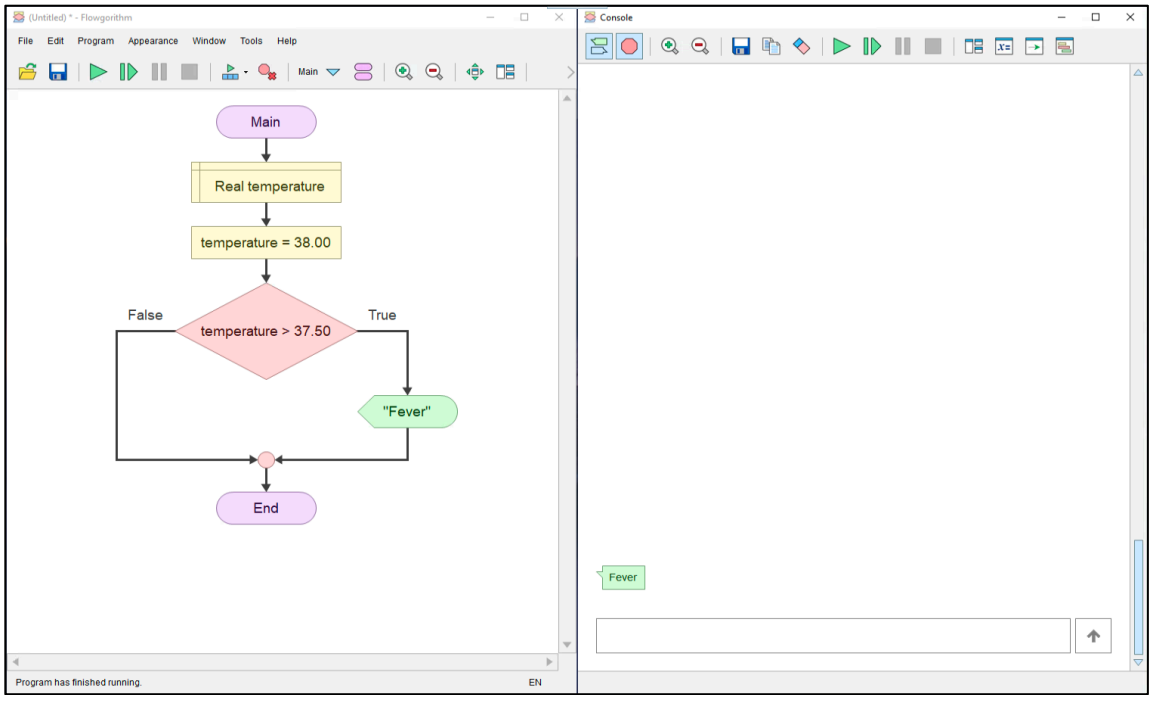
โครงสร้างการควบคุมแบบทางเลือก (Selection Structure) คือ รูปแบบการทำงานของโปรแกรมที่มีการตรวจสอบเงื่อนไขก่อน แล้วจึงเลือกดำเนินการตามผลของเงื่อนไขนั้น หากเงื่อนไขเป็นจริง โปรแกรมจะทำคำสั่งชุดหนึ่ง แต่หากเงื่อนไขเป็นเท็จ อาจทำอีกชุดคำสั่งหนึ่งหรือไม่ทำคำสั่งใดเลย ทั้งนี้ขึ้นอยู่กับรูปแบบของโครงสร้างที่ออกแบบไว้

โครงสร้างแบบทางเลือกเป็นหนึ่งในโครงสร้างพื้นฐานของการเขียนโปรแกรม มีบทบาทสำคัญในการทำให้โปรแกรมสามารถตอบสนองต่อความแตกต่างของข้อมูลนำเข้าได้ ตัวอย่างเช่น หากคะแนนมากกว่าหรือเท่ากับ 50 ให้แสดงผลว่า “ผ่าน” แต่หากน้อยกว่า 50 ให้แสดงผลว่า “ไม่ผ่าน” หรือหากจำนวนสินค้ามากกว่าหรือเท่ากับ 10 ชิ้น ให้คิดส่วนลดพิเศษ เป็นต้น

โดยทั่วไป โครงสร้างแบบทางเลือกสามารถแบ่งได้เป็น 2 ลักษณะสำคัญ ได้แก่

### 5.2.1 โครงสร้างทางเลือกแบบทางเดียว (Single Selection)

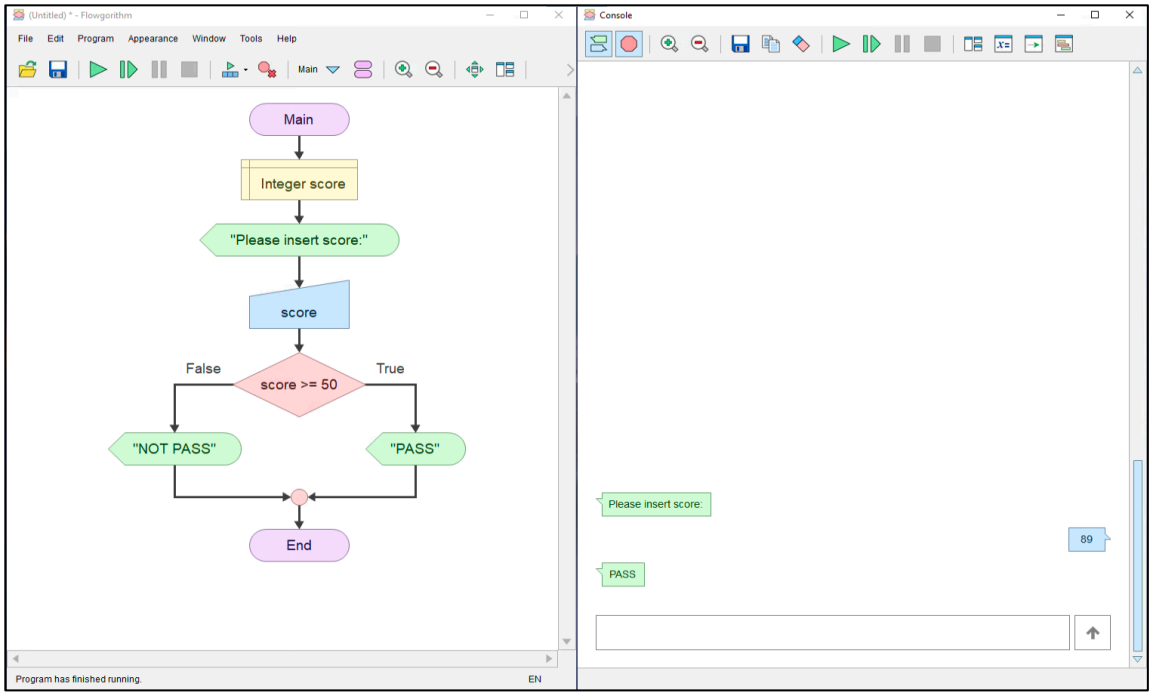
เป็นรูปแบบที่โปรแกรมตรวจสอบเงื่อนไข และจะดำเนินการก็ต่อเมื่อเงื่อนไขเป็นจริง แต่หากเงื่อนไขเป็นเท็จ โปรแกรมจะข้ามคำสั่งนั้นไปและดำเนินการต่อ เช่น หากอุณหภูมิสูงกว่า 37.5 ให้แสดงข้อความว่า “มีไข้” แต่หากไม่ถึงเกณฑ์ ก็ไม่ต้องแสดงข้อความดังกล่าว

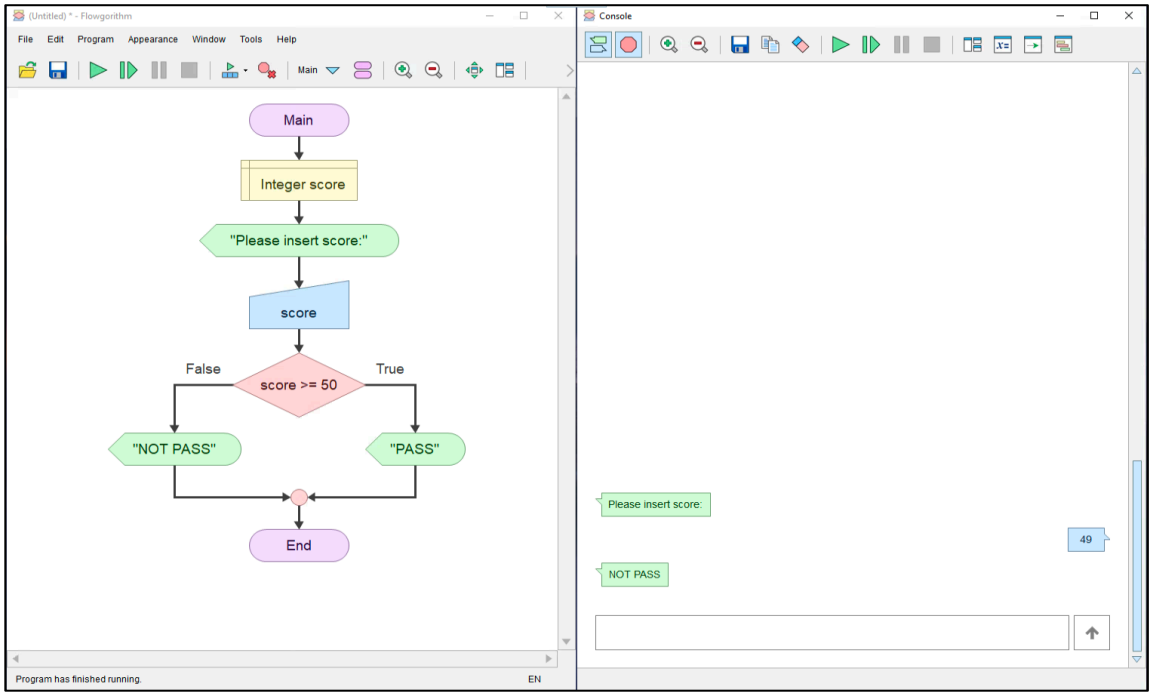



### 5.2.2 โครงสร้างทางเลือกแบบสองทาง (Double Selection)

เป็นรูปแบบที่โปรแกรมตรวจสอบเงื่อนไข และไม่ว่าผลจะเป็นจริงหรือเท็จ โปรแกรมจะมีทางเลือกให้ทำงานเสมอ เช่น หากคะแนนผ่านเกณฑ์ ให้แสดงผลว่า “ผ่าน” แต่หากไม่ผ่าน ให้แสดงผลว่า “ไม่ผ่าน” โครงสร้างลักษณะนี้พบได้บ่อยในการเขียนโปรแกรมเบื้องต้น

หลักการสำคัญของโครงสร้างแบบทางเลือก คือ การกำหนดเงื่อนไขให้ถูกต้อง ชัดเจน และสัมพันธ์กับผลลัพธ์ที่ต้องการ หากเงื่อนไขคลุมเครือหรือเขียนผิด โปรแกรมอาจตัดสินใจผิดและให้ผลลัพธ์ที่ไม่ถูกต้อง ตัวอย่างเช่น หากต้องการให้ผู้เรียนที่ได้คะแนนตั้งแต่ 50 ขึ้นไปถือว่าผ่าน แต่เขียนเงื่อนไขเป็น “คะแนนมากกว่า 50” ผู้เรียนที่ได้ 50 คะแนนจะถูกจัดเป็นไม่ผ่าน ทั้งที่ไม่ตรงตามเกณฑ์ที่ตั้งไว้


File Edit Program Appearance Window Tools Help
Main
Integer score
"Please insert score:"
score
False
score >= 50
True
"NOT PASS"
"PASS"
End
Program has finished running.
EN
Console
Please insert score:
89
PASS


File Edit Program Appearance Window Tools Help
Main
Integer score
"Please insert score:"
score
False
score >= 50
True
"NOT PASS"
"PASS"
End
Program has finished running.
EN
Console
Please insert score:
49
NOT PASS

การเรียนรู้โครงสร้างแบบทางเลือกจึงช่วยให้ผู้เรียนเข้าใจความสัมพันธ์ระหว่างเงื่อนไขกับผลลัพธ์ และเห็นความสำคัญของการใช้ตรรกะอย่างถูกต้อง นอกจากนี้ ยังช่วยพัฒนาทักษะการวิเคราะห์ปัญหา เพราะผู้เรียนต้องพิจารณาว่า ข้อมูลใดเป็นตัวกำหนดการตัดสินใจ และผลลัพธ์ใดควรเกิดขึ้นในแต่ละกรณี

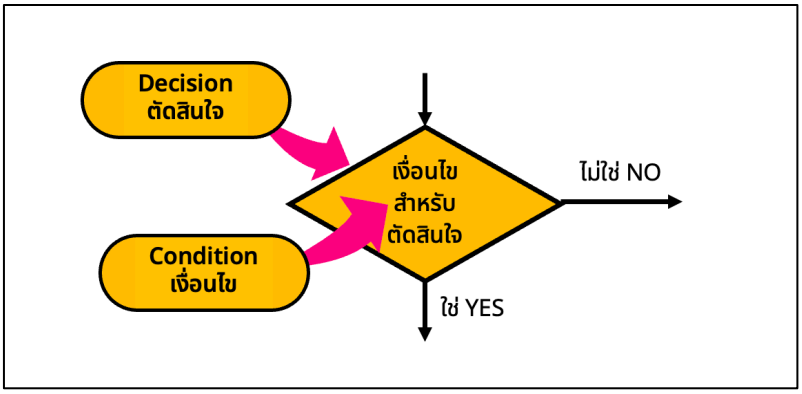
สำหรับนักศึกษาครู การสอนโครงสร้างแบบทางเลือกควรเริ่มจากสถานการณ์ใกล้ตัวก่อน เช่น การตัดสินใจพกร่มจากสภาพอากาศ การตัดสินใจใส่เสื้อกันหนาวจากอุณหภูมิ หรือการประเมินผลสอบจากคะแนน จากนั้นจึงเชื่อมโยงเข้าสู่การเขียนเงื่อนไขและการใช้ผังงาน วิธีการเช่นนี้จะช่วยให้ผู้เรียนเห็นภาพว่าโครงสร้างแบบทางเลือกมิใช่เรื่องซับซ้อน แต่เป็นการประยุกต์ใช้วิธีคิดที่พบได้ในชีวิตประจำวัน

ดังนั้น โครงสร้างการควบคุมแบบทางเลือกจึงเป็นกลไกสำคัญที่ช่วยให้โปรแกรมสามารถตัดสินใจและเลือกแนวทางการทำงานได้ตามเงื่อนไขที่กำหนด เป็นพื้นฐานสำคัญของการพัฒนาโปรแกรมที่ตอบสนองต่อสถานการณ์ได้อย่างเหมาะสม

## 5.3 การออกแบบผังงานแบบทางเลือกด้วยคำสั่งเงื่อนไข

ผังงานแบบทางเลือกเป็นการแสดงกระบวนการทำงานของโปรแกรมที่มีการตรวจสอบเงื่อนไขก่อนเลือกแนวทางการดำเนินงาน ผังงานลักษณะนี้ช่วยให้ผู้เรียนมองเห็นความสัมพันธ์ระหว่าง “การตัดสินใจ” กับ “ผลลัพธ์ของโปรแกรม” ได้อย่างชัดเจน และเป็นขั้นตอนสำคัญก่อนการเขียนโปรแกรมจริง

สัญลักษณ์สำคัญที่สุดในการออกแบบผังงานแบบทางเลือก คือ สัญลักษณ์การตัดสินใจ (Decision) ซึ่งมักแสดงเป็นรูปสี่เหลี่ยมข้าวหลามตัด ภายในสัญลักษณ์จะระบุเงื่อนไขที่ต้องการตรวจสอบ เช่น “คะแนน >= 50 ?” หรือ “จำนวน % 2 = 0 ?” จากนั้นจะมีเส้นทางแยกออกเป็นสองทาง โดยปกติมักกำกับว่า “จริง/เท็จ” หรือ “ใช่/ไม่ใช่” เพื่อบอกว่าหากผลของเงื่อนไขเป็นแบบใดโปรแกรมควรไปดำเนินการตามขั้นตอนใดต่อ



หลักการออกแบบผังงานแบบทางเลือกมีดังนี้ ประการแรก ต้องกำหนดปัญหาและวิเคราะห์ว่าจุดใดของกระบวนการต้องมีการตัดสินใจ ประการที่สอง ต้องเขียนเงื่อนไขให้ชัดเจนและตรวจสอบได้ ประการที่สาม ต้องกำหนดทางเลือกที่สอดคล้องกับผลของเงื่อนไข และประการที่สี่ ต้องเชื่อมโยงเส้นทางการทำงานให้ครบถ้วนจนถึงจุดสิ้นสุดของผังงาน

ตัวอย่างเช่น หากต้องการออกแบบผังงานเพื่อตรวจสอบผลการเรียนจากคะแนน อาจกำหนดขั้นตอนดังนี้

1. เริ่มต้น
2. รับค่าคะแนน
3. ตรวจสอบเงื่อนไขว่า คะแนน >= 50 หรือไม่
4. หากจริง ให้แสดงผลว่า “ผ่าน”
5. หากเท็จ ให้แสดงผลว่า “ไม่ผ่าน”
6. สิ้นสุด

ตัวอย่างอีกกรณีหนึ่ง เช่น การตรวจสอบเลขคู่เลขคี่ อาจออกแบบผังงานดังนี้

1. เริ่มต้น
2. รับค่าตัวเลข
3. ตรวจสอบว่า ตัวเลข mod 2 = 0 หรือไม่
4. หากจริง ให้แสดงผลว่า “เลขคู่”
5. หากเท็จ ให้แสดงผลว่า “เลขคี่”
6. สิ้นสุด

การฝึกออกแบบผังงานแบบทางเลือกมีประโยชน์อย่างมากต่อผู้เรียน เพราะช่วยให้เห็นลำดับความคิดอย่างเป็นระบบ เข้าใจการทำงานของเงื่อนไข และลดความผิดพลาดเมื่อต้องลงมือเขียนโปรแกรมจริง นอกจากนี้ ผังงานยังเป็นเครื่องมือที่ครูสามารถใช้ตรวจสอบความเข้าใจของผู้เรียนได้ว่าผู้เรียนสามารถเชื่อมโยงเงื่อนไขกับผลลัพธ์ได้ถูกต้องหรือไม่

สำหรับนักศึกษาครู การสอนเรื่องผังงานแบบทางเลือกควรใช้ตัวอย่างที่ใกล้ตัวและค่อย ๆ เพิ่มระดับความซับซ้อน เช่น เริ่มจากสถานการณ์ตัดสินใจแบบง่าย แล้วจึงพัฒนาไปสู่การใช้เงื่อนไขในโจทย์คำนวณ ครูอาจให้ผู้เรียนช่วยกันระบุเงื่อนไขจากสถานการณ์ที่กำหนด วาดผังงานร่วมกัน หรือแลกเปลี่ยนกันตรวจสอบความถูกต้องของผังงาน เพื่อส่งเสริมการเรียนรู้แบบมีส่วนร่วม

กล่าวโดยสรุป การออกแบบผังงานแบบทางเลือกด้วยคำสั่งเงื่อนไขเป็นกระบวนการสำคัญที่ช่วยให้ผู้เรียนเข้าใจการตัดสินใจของโปรแกรมอย่างเป็นรูปธรรม เป็นสะพานเชื่อมระหว่างการคิดเชิงตรรกะกับการเขียนโปรแกรมในระดับปฏิบัติ

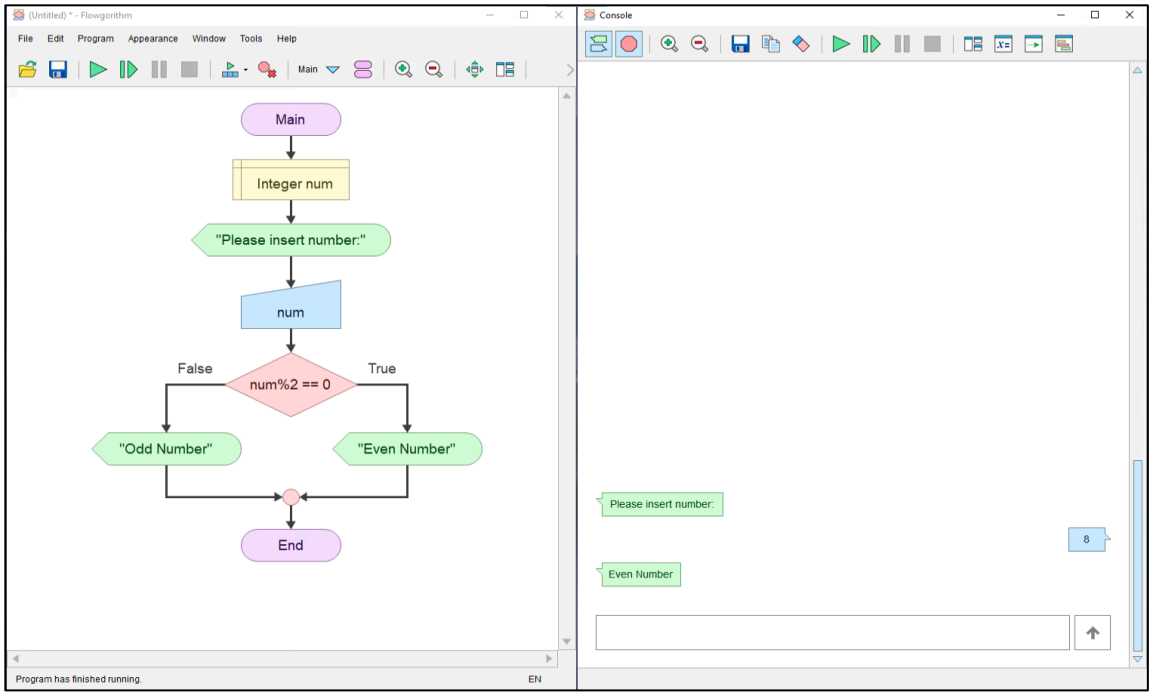
## 5.4 การใช้โปรแกรม Flowgorithm ในการเขียนและทดสอบผังงานแบบทางเลือก

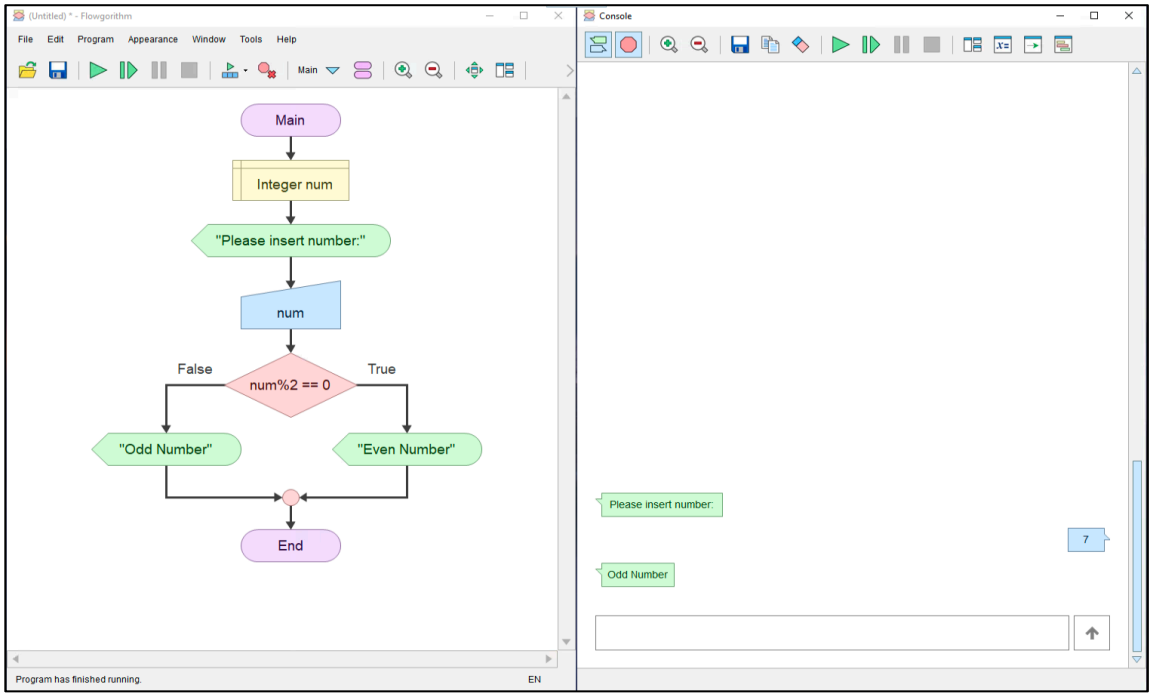
Flowgorithm เป็นโปรแกรมที่เหมาะสมอย่างยิ่งสำหรับการเรียนรู้โครงสร้างแบบทางเลือก เพราะช่วยให้ผู้เรียนสร้างผังงาน ตรวจสอบเงื่อนไข และทดสอบผลลัพธ์ได้อย่างเป็นรูปธรรม โดยไม่จำเป็นต้องเริ่มจากการเขียนโค้ดที่ซับซ้อน โปรแกรมนี้จึงช่วยลดภาระในการจดจำไวยากรณ์ของภาษาโปรแกรม และเปิดโอกาสให้ผู้เรียนมุ่งเน้นที่กระบวนการคิดเชิงอัลกอริทึมและตรรกะของการตัดสินใจ

ในการเขียนผังงานแบบทางเลือกด้วย Flowgorithm ผู้เรียนต้องเริ่มจากการวิเคราะห์โจทย์ก่อนว่า มีข้อมูลนำเข้าอะไร ต้องตรวจสอบเงื่อนไขใด และผลลัพธ์ของแต่ละกรณีควรเป็นอย่างไร จากนั้นจึงกำหนดตัวแปร เพิ่มสัญลักษณ์รับข้อมูล และใช้สัญลักษณ์การตัดสินใจเพื่อระบุเงื่อนไขที่ต้องการตรวจสอบ เมื่อกำหนดเงื่อนไขแล้ว จึงเพิ่มคำสั่งแสดงผลหรือคำสั่งประมวลผลตามเส้นทางของผลจริงและผลเท็จ

ตัวอย่างการใช้ Flowgorithm ในการตรวจสอบผลการเรียน อาจเริ่มจากกำหนดตัวแปร score จากนั้นเพิ่มสัญลักษณ์ Input เพื่อรับค่าคะแนน แล้วเพิ่มสัญลักษณ์ Decision โดยระบุเงื่อนไขว่า score >= 50 หากเงื่อนไขเป็นจริง ให้เพิ่มสัญลักษณ์ Output แสดงข้อความว่า “ผ่าน” แต่หากเงื่อนไขเป็นเท็จ ให้เพิ่มสัญลักษณ์ Output แสดงข้อความว่า “ไม่ผ่าน” แล้วจึงเชื่อมต่อไปยังจุดสิ้นสุดของผังงาน

ตัวอย่างอีกกรณีหนึ่ง เช่น การตรวจสอบเลขคู่เลขคี่ ผู้เรียนอาจกำหนดตัวแปร num รับค่าตัวเลขจากผู้ใช้ แล้วตรวจสอบเงื่อนไขว่า num % 2 = 0 หรือไม่ หากจริงให้แสดงผลว่า “เลขคู่” หรือ Even Number หากเท็จให้แสดงผลว่า “เลขคี่” หรือ Odd Number ตัวอย่างเหล่านี้ช่วยให้ผู้เรียนเห็นการทำงานของเงื่อนไขในเชิงรูปธรรมและสามารถเชื่อมโยงกับผลลัพธ์ที่เกิดขึ้นจริงได้


Main
Integer num
"Please insert number:"
num
False
num%2 == 0
True
"Odd Number"
"Even Number"
End
Program has finished running.
Console
Please insert number:
8
Even Number


Main
Integer num
"Please insert number:"
num
False
num%2 == 0
True
"Odd Number"
"Even Number"
End
Program has finished running.
Console
Please insert number:
7
Odd Number

จุดเด่นของ Flowgorithm คือ การสามารถ Run ผังงานและทดสอบค่าตัวอย่างได้ทันที ผู้เรียนจึงสามารถทดลองป้อนข้อมูลหลายรูปแบบ แล้วสังเกตผลลัพธ์ว่าเป็นไปตามที่คาดไว้หรือไม่ หากพบว่าผลลัพธ์ผิดพลาด ก็สามารถย้อนกลับไปตรวจสอบผังงานและแก้ไขเงื่อนไขหรือการเชื่อมต่อเส้นทางได้อย่างสะดวก กระบวนการเช่นนี้ช่วยส่งเสริมการเรียนรู้แบบทดลองและแก้ปัญหาอย่างมีเหตุผล

นอกจากนี้ Flowgorithm ยังช่วยให้ครูอธิบายข้อผิดพลาดที่พบบ่อยได้ชัดเจน เช่น การใช้ตัวดำเนินการเปรียบเทียบผิด การกำหนดเงื่อนไขไม่ครอบคลุม การสลับเส้นทางจริงกับเท็จ หรือการกำหนดผลลัพธ์ไม่สอดคล้องกับเงื่อนไข การให้ผู้เรียนเห็นและแก้ไขข้อผิดพลาดด้วยตนเองจะช่วยพัฒนาความเข้าใจได้ลึกซึ้งกว่าการรับฟังคำอธิบายเพียงอย่างเดียว

สำหรับนักศึกษาครู การใช้ Flowgorithm ควรเชื่อมโยงไปสู่การออกแบบการสอนด้วย เช่น การให้ผู้เรียนคาดการณ์ผลลัพธ์ก่อน Run การให้วิเคราะห์ผังงานที่ผิดและเสนอแนวทางแก้ไข หรือการให้ออกแบบผังงานจากสถานการณ์ที่กำหนดแล้วแลกเปลี่ยนกับเพื่อน กิจกรรมลักษณะนี้จะช่วยให้ครูฝึกทั้งทักษะด้านเนื้อหาและทักษะด้านวิธีสอนควบคู่กันไป

กล่าวโดยสรุป Flowgorithm เป็นเครื่องมือที่มีประสิทธิภาพสำหรับการเรียนรู้และการสอนโครงสร้างแบบทางเลือก เพราะช่วยให้ผู้เรียนสามารถสร้าง ทดสอบ และปรับปรุงผังงานได้อย่างเป็นรูปธรรม ส่งผลให้เข้าใจหลักการตัดสินใจและเงื่อนไขของโปรแกรมได้ชัดเจนยิ่งขึ้น

## 5.5 การจัดทำแผนการจัดการเรียนรู้และกิจกรรมการสอนเรื่องโครงสร้างแบบทางเลือก

การจัดการเรียนรู้เรื่องโครงสร้างการควบคุมแบบทางเลือกสำหรับผู้เรียนระดับประถมศึกษาและมัธยมศึกษา ควรมุ่งเน้นให้ผู้เรียนเข้าใจหลักการตัดสินใจจากสถานการณ์ที่ใกล้ตัวก่อน แล้วจึงเชื่อมโยงเข้าสู่การเขียนเงื่อนไข ผังงาน และการใช้โปรแกรม Flowgorithm ทั้งนี้ ครูควรออกแบบกิจกรรมให้เหมาะสมกับพัฒนาการของผู้เรียน เพื่อให้เกิดความเข้าใจอย่างเป็นขั้นตอนและไม่รู้สึกว่าเนื้อหายากเกินไป

การจัดทำแผนการจัดการเรียนรู้ควรเริ่มจากการกำหนดจุดประสงค์การเรียนรู้ให้ชัดเจน เช่น ผู้เรียนสามารถอธิบายความหมายของเงื่อนไขได้ ผู้เรียนสามารถเขียนผังงานแบบทางเลือกได้ หรือผู้เรียนสามารถใช้ Flowgorithm เพื่อสร้างและทดสอบผังงานแบบทางเลือกได้ จากนั้นจึงกำหนดสาระสำคัญ สื่อการเรียนรู้ กิจกรรมการเรียนรู้ และวิธีการประเมินผลให้สอดคล้องกับจุดประสงค์ดังกล่าว

สำหรับระดับประถมศึกษา กิจกรรมควรเน้นสถานการณ์ง่าย ๆ และเป็นรูปธรรม เช่น การตัดสินใจจากสภาพอากาศ การเลือกเสื้อผ้าตามฤดูกาล หรือการปฏิบัติตามกฎง่าย ๆ เช่น “ถ้าไฟแดงให้

หยุด ถ้าไฟเขียวให้ไป" ครูอาจใช้ภาพ บัตรคำ บทบาทสมมุติ หรือเกมเพื่อให้ผู้เรียนได้ฝึกคิดเงื่อนไขอย่างสนุกสนาน ก่อนที่จะค่อย ๆ แนะนำสัญลักษณ์ของผังงานอย่างง่าย

สำหรับระดับมัธยมศึกษา ผู้เรียนสามารถเรียนรู้ในระดับที่ซับซ้อนมากขึ้น เช่น การตรวจสอบเงื่อนไขจากโจทย์คำนวณ การออกแบบผังงานจากสถานการณ์หลายรูปแบบ การใช้โปรแกรม Flowgorithm เพื่อสร้างและ Run ผังงาน หรือการแก้ไขผังงานที่มีข้อผิดพลาด กิจกรรมลักษณะนี้จะช่วยให้ผู้เรียนพัฒนาทั้งทักษะการคิดเชิงตรรกะและทักษะการใช้เครื่องมือดิจิทัล

รูปแบบกิจกรรมที่เหมาะสมในการสอนเรื่องโครงสร้างแบบทางเลือก ได้แก่

1. การวิเคราะห์สถานการณ์และระบุเงื่อนไข
2. การเรียงลำดับผลลัพธ์ตามเงื่อนไขที่กำหนด
3. การเขียนข้อความเงื่อนไขจากสถานการณ์จริง
4. การวาดผังงานแบบทางเลือก
5. การใช้ Flowgorithm เพื่อสร้างและทดสอบผังงาน
6. การอภิปรายข้อผิดพลาดที่เกิดขึ้นและวิธีแก้ไข

ในการประเมินผล ครูควรใช้วิธีที่หลากหลายและประเมินตามสภาพจริง (Authentic Assessment) เช่น การสังเกตการมีส่วนร่วม การตรวจผังงาน การประเมินการใช้โปรแกรม Flowgorithm การตรวจใบงาน และการให้ผู้เรียนอธิบายเหตุผลของเงื่อนไขที่ตนออกแบบ นอกจากนี้อาจใช้การประเมินจากชิ้นงานหรือภารกิจ เช่น ให้ออกแบบผังงานเพื่อตรวจสอบส่วนลดสินค้า หรือให้จัดทำแผนการจัดการเรียนรู้ขนาดย่อมสำหรับสอนเรื่องเงื่อนไข

สำหรับนักศึกษาครู การฝึกจัดทำแผนการจัดการเรียนรู้เรื่องโครงสร้างแบบทางเลือกมีความสำคัญอย่างยิ่ง เพราะช่วยให้สามารถเชื่อมโยงความรู้ด้านเนื้อหากับศาสตร์การสอนได้อย่างเป็นระบบ นักศึกษาครูควรฝึกกำหนดจุดประสงค์ เลือกกิจกรรมที่เหมาะสมกับวัย ใช้สื่ออย่างเหมาะสม และออกแบบการประเมินที่สะท้อนทั้งความรู้ กระบวนการคิด และทักษะการปฏิบัติของผู้เรียน

กล่าวโดยสรุป การจัดทำแผนการจัดการเรียนรู้และกิจกรรมการสอนเรื่องโครงสร้างแบบทางเลือกควรยึดผู้เรียนเป็นสำคัญ เริ่มจากสถานการณ์ใกล้ตัว ใช้กิจกรรมที่กระตุ้นการคิดและการมีส่วนร่วม และค่อย ๆ เชื่อมโยงไปสู่การเขียนผังงานและการใช้โปรแกรม เพื่อให้ผู้เรียนเข้าใจแนวคิดเรื่องเงื่อนไขอย่างลึกซึ้งและนำไปประยุกต์ใช้ได้จริง

**บทสรุปประจำบทที่ 5**

โครงสร้างการควบคุมโปรแกรมแบบทางเลือกเป็นหนึ่งในโครงสร้างพื้นฐานสำคัญของวิทยาการคำนวณที่ช่วยให้โปรแกรมสามารถตัดสินใจและเลือกแนวทางการทำงานได้ตามเงื่อนไขที่กำหนด การทำความเข้าใจหลักการตัดสินใจและเงื่อนไขในการแก้ปัญหาจึงเป็นพื้นฐานสำคัญในการพัฒนาทักษะการคิดเชิงตรรกะและการออกแบบอัลกอริทึมอย่างเป็นระบบ

การออกแบบผังงานแบบทางเลือกช่วยให้ผู้เรียนมองเห็นความสัมพันธ์ระหว่างเงื่อนไขกับผลลัพธ์ได้อย่างชัดเจน และเมื่อใช้ร่วมกับโปรแกรม Flowgorithm ผู้เรียนจะสามารถสร้าง ทดสอบ และปรับปรุงผังงานได้อย่างเป็นรูปธรรม ส่งผลให้เข้าใจการทำงานของโปรแกรมได้ลึกซึ้งยิ่งขึ้น สำหรับนักศึกษาครู การเรียนรู้เรื่องโครงสร้างแบบทางเลือกควรเชื่อมโยงกับการออกแบบแผนการจัดการเรียนรู้และกิจกรรมการสอนที่เหมาะสมกับระดับของผู้เรียน โดยเริ่มจากสถานการณ์ใกล้ตัวและค่อย ๆ พัฒนาไปสู่การใช้ผังงานและโปรแกรม เพื่อให้ผู้เรียนเกิดทั้งความรู้ ทักษะ และเจตคติที่ดีต่อการเรียนวิทยาการคำนวณ

**คำถามท้ายบท**

1. จงอธิบายความหมายของเงื่อนไขและการตัดสินใจในการแก้ปัญหา พร้อมยกตัวอย่างจากชีวิตประจำวัน
2. โครงสร้างการควบคุมแบบทางเลือกคืออะไร และแตกต่างจากโครงสร้างแบบเรียงลำดับอย่างไร
3. โครงสร้างแบบทางเลือกสามารถแบ่งได้เป็นกี่ลักษณะ อะไรบ้าง
4. เพราะเหตุใดการเขียนเงื่อนไขให้ถูกต้องจึงมีความสำคัญต่อผลลัพธ์ของโปรแกรม
5. สัญลักษณ์ใดในผังงานที่ใช้แสดงการตัดสินใจ และมีหน้าที่อย่างไร
6. จงอธิบายขั้นตอนการออกแบบผังงานแบบทางเลือกจากโจทย์อย่างง่าย
7. โปรแกรม Flowgorithm มีประโยชน์อย่างไรต่อการเรียนรู้เรื่องโครงสร้างแบบทางเลือก
8. หากต้องการออกแบบผังงานเพื่อตรวจสอบว่าเลขที่ป้อนเข้ามาเป็นเลขคู่หรือเลขคี่ ท่านจะกำหนดเงื่อนไขอย่างไร
9. หากท่านเป็นครูระดับประถมศึกษา ท่านจะจัดกิจกรรมอย่างไรเพื่อให้ผู้เรียนเข้าใจเรื่องเงื่อนไขและการตัดสินใจ
10. หากท่านเป็นครูระดับมัธยมศึกษา ท่านจะออกแบบแผนการจัดการเรียนรู้เรื่องโครงสร้างแบบทางเลือกอย่างไรให้ผู้เรียนมีส่วนร่วมและเข้าใจได้จริง

**เอกสารอ้างอิง**

กระทรวงศึกษาธิการ. (2560). *ตัวชี้วัดและสาระการเรียนรู้แกนกลาง กลุ่มสาระการเรียนรู้วิทยาศาสตร์ (ฉบับปรับปรุง พ.ศ. 2560) ตามหลักสูตรแกนกลางการศึกษาขั้นพื้นฐาน พุทธศักราช 2551*. กรุงเทพมหานคร: ชุมนุมสหกรณ์การเกษตรแห่งประเทศไทย.

Gaddis, T. (2018). *Starting out with programming logic and design* (5th ed.). Boston, MA: Pearson.

Joyce, D. (2020). *Programming logic and design: Introductory* (9th ed.). Boston, MA: Cengage Learning.

Flowgorithm. (2024). *Flowgorithm documentation*. Retrieved from the official Flowgorithm documentation resources.

Wing, J. M. (2006). Computational thinking. *Communications of the ACM, 49*(3), 33–35.

Aho, A. V. (2012). Computation and computational thinking. *The Computer Journal, 55*(7), 832–835.