เอกสารประกอบการสอน รายวิชาวิทยาการคำนวณ

Computational Science

พัฒนาการคิดเชิงคำนวณ การเขียนโปรแกรม และการออกแบบโครงงานดิจิทัล
เพื่อสร้างสรรค์นวัตกรรมการเรียนรู้ในสถานศึกษา

ผู้ช่วยศาสตราจารย์ ดร.ณัฐภัทร แก้วรัตนภัทร์
มหาวิทยาลัยราชภัฏสวนสุนันทา

เอกสารประกอบการสอน
รายวิชาวิทยาการคำนวณ
Computational Science

ผู้ช่วยศาสตราจารย์ ดร.ณัฐภัทร แก้วรัตนภัทร์

สาขาวิชาเทคโนโลยีดิจิทัลเพื่อการศึกษา
คณะครุศาสตร์ มหาวิทยาลัยราชภัฏสวนสุนันทา

สงวนลิขสิทธิ์ ตามพระราชบัญญัติลิขสิทธิ์ พ.ศ. 2537

เอกสารประกอบการสอนรายวิชา DTI1306

วิทยาการคำนวณ

(Computational Science)

สงวนลิขสิทธิ์ ตามพระราชบัญญัติลิขสิทธิ์ พ.ศ. 2537
และที่แก้ไขเพิ่มเติม ห้ามคัดลอก ทำซ้ำ ดัดแปลง หรือนำส่วนหนึ่งส่วนใด
ในเอกสารประกอบการสอนฉบับนี้ไปใช้ หรือ เผยแพร่ต่อโดยไม่ได้รับอนุญาตจาก
นายณัฐภัทร แก้วรัตนภัทร์
การละเมิดลิขสิทธิ์ถือเป็นการกระทำที่มีความผิดทางกฎหมาย

จัดทำโดย
ผู้ช่วยศาสตราจารย์ ดร.ณัฐภัทร แก้วรัตนภัทร์
หลักสูตรครุศาสตรบัณฑิต สาขาวิชาเทคโนโลยีดิจิทัลเพื่อการศึกษา
คณะครุศาสตร์ มหาวิทยาลัยราชภัฏสวนสุนันทา
อีเมล nutthapat.ke@ssru.ac.th

สื่อการเรียนรู้ประกอบเอกสารคำสอน สามารถเข้าถึงได้ที่
https://ssrudlp.ssru.ac.th/teacher/nutthapat_kaew
ยินดีรับข้อคิดเห็นหรือข้อเสนอแนะ สามารถอีเมลได้ที่ nutthapat.ke@ssru.ac.th

คำนำ

เอกสารประกอบการสอนรายวิชาวิทยาการคำนวณ (Computational Science) นี้ได้เรียบเรียงขึ้นเพื่อใช้สำหรับการเรียนการสอนในหลักสูตรครุศาสตรบัณฑิต สาขาวิชาเทคโนโลยีดิจิทัลเพื่อการศึกษา คณะครุศาสตร์ มหาวิทยาลัยราชภัฏสวนสุนันทา มุ่งหวังให้ผู้เรียนมีความรู้ ความเข้าใจ และสามารถวิเคราะห์ปัญหาและออกแบบเพื่อแก้ไขปัญหาด้วยการคิดเชิงคำนวณ หรือ Computational Thinking รวมถึงสามารถจัดทำแผนการเรียนรู้และกิจกรรมสำหรับรายวิชาวิทยาการคำนวณสำหรับชั้นประถมและมัธยมศึกษาได้ สำหรับเนื้อหาสาระสำคัญในเอกสาร ประกอบด้วยพื้นฐานของวิทยาการคำนวณ เทคโนโลยีดิจิทัล การรู้เท่าทันดิจิทัล โครงสร้างการควบคุมโปรแกรมแบบเรียงลำดับ โครงสร้างการควบคุมโปรแกรมแบบทางเลือก โครงสร้างการควบคุมโปรแกรมแบบทำซ้ำ การโปรแกรมแบบบล็อกคำสั่ง การโปรแกรมด้วยไมโครบิต การโปรแกรมภาษาไพทอนเบื้องต้น และโครงงานทางด้านวิทยาการคำนวณเพื่อการศึกษา

ผู้เขียนหวังเป็นอย่างยิ่งว่าเอกสารนี้จะอำนวยประโยชน์ต่อผู้เรียนและผู้นำไปใช้อย่างสมควร ผู้เขียนน้อมรับความคิดเห็นและคำติชม เพื่อนำไปใช้ปรับปรุงให้ดียิ่งขึ้นในการพิมพ์ครั้งต่อไป ขอขอบพระคุณในความอนุเคราะห์มา ณ โอกาสนี้

ณัฐภัทร แก้วรัตนภัทร์
สาขาวิชาเทคโนโลยีดิจิทัลเพื่อการศึกษา คณะครุศาสตร์
มหาวิทยาลัยราชภัฏสวนสุนันทา

สงวนลิขสิทธิ์ ตามพระราชบัญญัติลิขสิทธิ์ พ.ศ. 2537

สารบัญ

หน้า

คำนำ

สารบัญ

สารบัญภาพ

สารบัญตาราง

แผนบริหารการสอนประจำวิชา

แผนบริหารการสอนประจำบทที่ 1

บทที่ 1 พื้นฐานของวิทยาการคำนวณ

1.1 ความหมาย ความสำคัญ และพัฒนาการของวิทยาการคำนวณ

1.2 องค์ประกอบสำคัญของวิทยาการคำนวณ

1.3 แนวคิดการคิดเชิงคำนวณ (Computational Thinking)

1.4 วิทยาการคำนวณกับการแก้ปัญหาในชีวิตประจำวันและการศึกษา

1.5 แนวทางการจัดการเรียนรู้วิทยาการคำนวณสำหรับผู้เรียนระดับประถมศึกษาและมัธยมศึกษา

บทสรุปประจำบทที่ 1

คำถามท้ายบท

เอกสารอ้างอิง

แผนบริหารการสอนประจำบทที่ 2

บทที่ 2 เทคโนโลยีดิจิทัล

2.1 ความหมายและลักษณะของเทคโนโลยีดิจิทัล

2.2 องค์ประกอบของระบบดิจิทัลและเทคโนโลยีสารสนเทศ

2.3 บทบาทของเทคโนโลยีดิจิทัลต่อการเรียนรู้และการดำรงชีวิต

2.4 การประยุกต์ใช้เทคโนโลยีดิจิทัลในบริบทการศึกษา

2.5 การออกแบบกิจกรรมการเรียนรู้โดยใช้เทคโนโลยีดิจิทัลอย่างเหมาะสม

สงวนลิขสิทธิ์ ตามพระราชบัญญัติลิขสิทธิ์ พ.ศ. 2537

บทสรุปประจำบทที่ 2

คำถามท้ายบท

เอกสารอ้างอิง

แผนบริหารการสอนประจำบทที่ 3

บทที่ 3 การรู้เท่าทันดิจิทัล

3.1 ความหมายและความสำคัญของการรู้เท่าทันดิจิทัล

3.2 ทักษะสำคัญของการรู้เท่าทันดิจิทัลในศตวรรษที่ 21

3.3 การใช้สื่อดิจิทัลอย่างปลอดภัย มีจริยธรรม และรับผิดชอบ

3.4 การวิเคราะห์ข้อมูล ข่าวสาร และสื่อดิจิทัลอย่างมีวิจารณญาณ

3.5 การออกแบบกิจกรรมส่งเสริมการรู้เท่าทันดิจิทัลสำหรับผู้เรียน

บทสรุปประจำบทที่ 3

คำถามท้ายบท

เอกสารอ้างอิง

แผนบริหารการสอนประจำบทที่ 4

บทที่ 4 โครงสร้างการควบคุมโปรแกรมแบบเรียงลำดับ

4.1 ความหมายของอัลกอริทึมและการแสดงลำดับขั้นตอนการทำงาน

4.2 หลักการของโครงสร้างการควบคุมแบบเรียงลำดับ (Sequential Structure)

4.3 สัญลักษณ์ผังงานเบื้องต้นและการออกแบบผังงานแบบเรียงลำดับ

4.4 การใช้โปรแกรม Flowgorithm ในการเขียนและทดสอบผังงานแบบเรียงลำดับ

4.5 การออกแบบกิจกรรมการเรียนรู้เพื่อสอนโครงสร้างแบบเรียงลำดับในระดับประถมศึกษาและมัธยมศึกษา

บทสรุปประจำบทที่ 4

คำถามท้ายบท

เอกสารอ้างอิง

แผนบริหารการสอนประจำบทที่ 5

บทที่ 5 โครงสร้างการควบคุมโปรแกรมแบบทางเลือก

5.1 หลักการตัดสินใจและเงื่อนไขในการแก้ปัญหา

5.2 โครงสร้างการควบคุมแบบทางเลือก (Selection Structure)

สงวนลิขสิทธิ์ ตามพระราชบัญญัติลิขสิทธิ์ พ.ศ. 2537

5.3 การออกแบบผังงานแบบทางเลือกด้วยคำสั่งเงื่อนไข

5.4 การใช้โปรแกรม Flowgorithm ในการเขียนและทดสอบผังงานแบบทางเลือก

5.5 การจัดทำแผนการจัดการเรียนรู้และกิจกรรมการสอนเรื่องโครงสร้างแบบทางเลือก

บทสรุปประจำบทที่ 5

คำถามท้ายบท

เอกสารอ้างอิง

แผนบริหารการสอนประจำบทที่ 6

บทที่ 6 โครงสร้างการควบคุมโปรแกรมแบบทำซ้ำ

6.1 ความหมายและลักษณะของโครงสร้างการควบคุมแบบทำซ้ำ (Iteration Structure)

6.2 ประเภทของการทำซ้ำและการประยุกต์ใช้ในการแก้ปัญหา

6.3 การออกแบบผังงานแบบทำซ้ำ

6.4 การใช้โปรแกรม Flowgorithm ในการเขียนและทดสอบผังงานแบบทำซ้ำ

6.5 การใช้ Trace Table เพื่อวิเคราะห์การทำงานของโครงสร้างแบบทำซ้ำ

6.6 การออกแบบกิจกรรมการเรียนรู้เพื่อพัฒนาความเข้าใจเรื่องการทำซ้ำสำหรับผู้เรียนระดับประถมศึกษาและมัธยมศึกษา

บทสรุปประจำบทที่ 6

คำถามท้ายบท

เอกสารอ้างอิง

แผนบริหารการสอนประจำบทที่ 7

บทที่ 7 การโปรแกรมแบบบล็อกคำสั่ง

7.1 แนวคิดพื้นฐานของการโปรแกรมแบบบล็อกคำสั่ง

7.2 ส่วนประกอบและเครื่องมือของโปรแกรมแบบบล็อกคำสั่ง

7.3 การออกแบบลำดับคำสั่ง เงื่อนไข และการทำซ้ำด้วยบล็อกคำสั่ง

7.4 การสร้างชิ้นงานหรือสื่อการเรียนรู้ด้วยการโปรแกรมแบบบล็อกคำสั่ง

7.5 แนวทางการจัดกิจกรรมการเรียนรู้ด้วยการโปรแกรมแบบบล็อกคำสั่งสำหรับนักเรียน

บทสรุปประจำบทที่ 7

คำถามท้ายบท

เอกสารอ้างอิง

สงวนลิขสิทธิ์ ตามพระราชบัญญัติลิขสิทธิ์ พ.ศ. 2537

แผนบริหารการสอนประจำบทที่ 8

บทที่ 8 การโปรแกรมด้วยไมโครบิต

8.1 ความรู้เบื้องต้นเกี่ยวกับไมโครบิตและระบบสมองกลฝังตัว

8.2 องค์ประกอบและการทำงานของบอร์ดไมโครบิต

8.3 การเขียนโปรแกรมควบคุมไมโครบิตเบื้องต้น

8.4 การประยุกต์ใช้ไมโครบิตในการสร้างชิ้นงานและนวัตกรรมทางการศึกษา

8.5 การออกแบบกิจกรรมการเรียนรู้โดยใช้ไมโครบิตสำหรับผู้เรียนในระดับการศึกษาขั้นพื้นฐาน

บทสรุปประจำบทที่ 8

คำถามท้ายบท

เอกสารอ้างอิง

แผนบริหารการสอนประจำบทที่ 9

บทที่ 9 การโปรแกรมภาษาไพทอนเบื้องต้น

9.1 ความรู้เบื้องต้นเกี่ยวกับภาษาไพทอน

9.2 ตัวแปร ชนิดข้อมูล และการรับ–แสดงผลข้อมูล

9.3 การใช้คำสั่งควบคุมแบบเรียงลำดับ ทางเลือก และทำซ้ำในภาษาไพทอน

9.4 การเขียนโปรแกรมแก้ปัญหาอย่างง่ายด้วยภาษาไพทอน

9.5 การออกแบบกิจกรรมการเรียนรู้เบื้องต้นสำหรับการสอนภาษาไพทอน

บทสรุปประจำบทที่ 9

คำถามท้ายบท

เอกสารอ้างอิง

แผนบริหารการสอนประจำบทที่ 10

บทที่ 10 โครงงานทางด้านวิทยาการคำนวณเพื่อการศึกษา

10.1 ความหมายและความสำคัญของโครงงานทางด้านวิทยาการคำนวณ

10.2 ประเภทและกระบวนการพัฒนาโครงงานทางด้านวิทยาการคำนวณเพื่อการศึกษา

10.3 การวิเคราะห์ปัญหา การกำหนดวัตถุประสงค์ และการออกแบบโครงงาน

10.4 การดำเนินงาน การทดสอบ และการนำเสนอผลงานโครงงาน

10.5 การออกแบบโครงงานวิทยาการคำนวณเพื่อใช้ในการจัดการเรียนรู้ในสถานศึกษา

สงวนลิขสิทธิ์ ตามพระราชบัญญัติลิขสิทธิ์ พ.ศ. 2537

บทสรุปประจำบทที่ 10

คำถามท้ายบท

เอกสารอ้างอิง

บรรณานุกรม

สงวนลิขสิทธิ์ ตามพระราชบัญญัติลิขสิทธิ์ พ.ศ. 2537

บทที่ 1
พื้นฐานของวิทยาการคำนวณ
Fundamental of Computational Thinking

บทนำ

ในโลกยุคดิจิทัลที่เทคโนโลยีเข้ามามีบทบาทต่อการดำเนินชีวิต การทำงาน การสื่อสาร และการเรียนรู้อย่างกว้างขวาง มนุษย์จำเป็นต้องมีความสามารถในการทำความเข้าใจปัญหา วิเคราะห์สถานการณ์อย่างเป็นระบบ และเลือกใช้วิธีการหรือเครื่องมือที่เหมาะสมเพื่อแก้ไขปัญหาอย่างมีประสิทธิภาพ ทักษะดังกล่าวมิได้จำกัดอยู่เพียงผู้ที่ศึกษาเฉพาะทางด้านคอมพิวเตอร์เท่านั้น แต่เป็นสมรรถนะพื้นฐานของพลเมืองในศตวรรษที่ 21 ที่ทุกคนควรได้รับการพัฒนา โดยเฉพาะผู้เรียนในระบบการศึกษาขั้นพื้นฐานและนักศึกษาครูซึ่งจะเป็นผู้มีบทบาทสำคัญในการออกแบบประสบการณ์การเรียนรู้ให้แก่ผู้เรียนรุ่นต่อไป

วิทยาการคำนวณเป็นศาสตร์ที่มุ่งเน้นการศึกษาแนวคิด หลักการ และกระบวนการในการแก้ปัญหาอย่างเป็นระบบ โดยอาศัยพื้นฐานของการคิดเชิงคำนวณ การออกแบบอัลกอริทึม การใช้ข้อมูลอย่างมีความหมาย และการประยุกต์ใช้เทคโนโลยีดิจิทัลอย่างเหมาะสม แม้ว่าหลายคนอาจเข้าใจว่าวิทยาการคำนวณหมายถึงการเขียนโปรแกรมคอมพิวเตอร์เพียงอย่างเดียว แต่แท้จริงแล้วศาสตร์นี้ครอบคลุมไปถึงการทำความเข้าใจปัญหา การแยกองค์ประกอบของปัญหา การมองหารูปแบบ การสร้างขั้นตอนวิธี และการประเมินผลลัพธ์อย่างมีเหตุผล ซึ่งสามารถนำไปใช้ได้ทั้งในห้องเรียน ชีวิตประจำวัน และการประกอบวิชาชีพ

สำหรับบริบทของการผลิตครู วิทยาการคำนวณมีความสำคัญอย่างยิ่ง เนื่องจากนักศึกษาครูจำเป็นต้องเข้าใจทั้งเนื้อหาสาระของศาสตร์นี้ และเข้าใจแนวทางการถ่ายทอดความรู้ให้สอดคล้องกับพัฒนาการของผู้เรียนในแต่ละช่วงวัย การจัดการเรียนรู้วิทยาการคำนวณจึงไม่ควรมุ่งเน้นเพียงการสอนให้ผู้เรียนใช้โปรแกรมหรือคำสั่งเท่านั้น หากแต่ควรส่งเสริมให้ผู้เรียนได้ฝึกคิด ฝึกสังเกต ทดลอง ลงมือปฏิบัติ และเชื่อมโยงความรู้กับสถานการณ์รอบตัว เพื่อพัฒนาความสามารถในการแก้ปัญหาอย่างสร้างสรรค์และมีเหตุผล

บทนี้จึงมุ่งนำเสนอพื้นฐานสำคัญของวิทยาการคำนวณ เริ่มจากความหมาย ความสำคัญ และพัฒนาการของศาสตร์ องค์ประกอบสำคัญของวิทยาการคำนวณ แนวคิดการคิดเชิงคำนวณ ความสัมพันธ์ระหว่างวิทยาการคำนวณกับการแก้ปัญหาในชีวิตประจำวันและการศึกษา ตลอดจนแนวทางการจัดการเรียนรู้วิทยาการคำนวณสำหรับผู้เรียนระดับประถมศึกษาและมัธยมศึกษา เพื่อวางรากฐานความเข้าใจที่ถูกต้องให้แก่นักศึกษาครู และเพื่อให้สามารถนำองค์ความรู้ดังกล่าวไปประยุกต์ใช้ในการออกแบบการเรียนรู้ได้อย่างเหมาะสมในอนาคต

1.1 ความหมาย ความสำคัญ และพัฒนาการของวิทยาการคำนวณ

วิทยาการคำนวณ (Computational Science หรือในบางบริบทเชื่อมโยงกับ Computer Science Education และ Computational Thinking) เป็นศาสตร์ที่ศึกษาเกี่ยวกับกระบวนการแก้ปัญหาอย่างเป็นระบบโดยอาศัยแนวคิดทางคอมพิวเตอร์ อัลกอริทึม ข้อมูล การประมวลผล และการใช้เทคโนโลยีดิจิทัลเป็นเครื่องมือสนับสนุน กล่าวโดยสรุป วิทยาการคำนวณไม่ใช่เพียงการสั่งงานคอมพิวเตอร์ แต่เป็นศาสตร์ว่าด้วย “วิธีคิด” เพื่อจัดการปัญหาอย่างมีขั้นตอนและตรวจสอบได้

ในแง่ความหมายเชิงการศึกษา วิทยาการคำนวณเป็นกระบวนการเรียนรู้ที่ช่วยให้ผู้เรียนเข้าใจว่าปัญหาหนึ่ง ๆ สามารถวิเคราะห์ แยกย่อย วางแผน ออกแบบวิธีการ และสร้างคำตอบได้อย่างเป็นระบบ ความสำคัญของศาสตร์นี้จึงอยู่ที่การพัฒนาความสามารถในการคิด ไม่ใช่เพียงการฝึกทักษะทางเทคนิคเท่านั้น ผู้เรียนที่ได้รับการพัฒนาด้านวิทยาการคำนวณจะสามารถคิดอย่างมีตรรกะ มองปัญหาอย่างเป็นองค์รวมและเป็นส่วนย่อย ตัดสินใจโดยอาศัยข้อมูล และสร้างแนวทางแก้ปัญหาที่เหมาะสมกับบริบท

วิทยาการคำนวณมีความสำคัญในหลายมิติ ประการแรก คือ ความสำคัญต่อผู้เรียนในฐานะพลเมืองดิจิทัล ผู้เรียนจำเป็นต้องเข้าใจโลกที่ขับเคลื่อนด้วยข้อมูล ระบบอัตโนมัติ ปัญญาประดิษฐ์ และเครือข่ายดิจิทัล ประการที่สอง คือ ความสำคัญต่อการเรียนรู้ในศตวรรษที่ 21 เนื่องจากศาสตร์นี้ช่วยพัฒนาทักษะการคิดวิเคราะห์ การคิดเชิงระบบ การคิดสร้างสรรค์ และการแก้ปัญหา ประการที่สาม คือ ความสำคัญต่อการประกอบอาชีพในอนาคต เพราะหลายวิชาชีพในปัจจุบันต้องอาศัยการทำงานร่วมกับเทคโนโลยี และประการที่สี่ คือ ความสำคัญต่อการพัฒนาการศึกษา เพราะช่วยให้ครูสามารถออกแบบกิจกรรมที่ส่งเสริมให้ผู้เรียนเรียนรู้จากการลงมือทำและการสร้างองค์ความรู้ด้วยตนเอง

หากพิจารณาด้านพัฒนาการของวิทยาการคำนวณ จะพบว่าศาสตร์นี้มีวิวัฒนาการต่อเนื่องจากแนวคิดทางคณิตศาสตร์ ตรรกศาสตร์ และวิทยาการคอมพิวเตอร์ ในระยะแรก มนุษย์ใช้วิธีคิดเชิงขั้นตอนในการคำนวณและแก้ปัญหา เช่น การหาผลลัพธ์ทางคณิตศาสตร์หรือการจัดระบบข้อมูล ต่อมาเมื่อ

คอมพิวเตอร์ได้รับการพัฒนา แนวคิดเรื่องอัลกอริทึม ภาษาโปรแกรม และระบบประมวลผลจึงมีความชัดเจนมากขึ้น หลังจากนั้น วิทยาการคำนวณได้ขยายตัวจากศาสตร์เฉพาะทางไปสู่ศาสตร์เพื่อการเรียนรู้ของคนทั่วไป โดยเฉพาะอย่างยิ่งเมื่อระบบการศึกษาหลายประเทศเริ่มบรรจุวิชาวิทยาการคำนวณ การเขียนโปรแกรม และการคิดเชิงคำนวณในหลักสูตรการศึกษาขั้นพื้นฐาน

ในบริบทประเทศไทย วิทยาการคำนวณได้รับความสำคัญเพิ่มขึ้นจากการปรับหลักสูตรแกนกลางการศึกษาขั้นพื้นฐาน โดยเน้นให้ผู้เรียนมีทักษะการคิดเชิงคำนวณ การใช้เทคโนโลยีอย่างรู้เท่าทัน และการสร้างนวัตกรรมเบื้องต้น ทั้งนี้ เป้าหมายสำคัญมิใช่เพื่อผลิตนักเขียนโปรแกรมเพียงอย่างเดียว แต่เพื่อพัฒนาผู้เรียนให้สามารถใช้กระบวนการคิดที่เป็นระบบในการเผชิญปัญหาในชีวิตจริง

ดังนั้น วิทยาการคำนวณจึงเป็นศาสตร์พื้นฐานที่มีบทบาทสำคัญอย่างยิ่งต่อการพัฒนาผู้เรียน ครู และระบบการศึกษา โดยเฉพาะสำหรับนักศึกษาครูที่ต้องมีความเข้าใจทั้งในระดับเนื้อหาและในระดับการออกแบบการเรียนรู้ เพื่อสร้างผู้เรียนที่พร้อมต่อการดำรงชีวิตในสังคมดิจิทัลอย่างมีคุณภาพ

1.2 องค์ประกอบสำคัญของวิทยาการคำนวณ

วิทยาการคำนวณเป็นศาสตร์ที่มีองค์ประกอบหลากหลายซึ่งเชื่อมโยงกันอย่างเป็นระบบ การทำความเข้าใจองค์ประกอบเหล่านี้จะช่วยให้ผู้เรียนมองเห็นภาพรวมของศาสตร์ได้ชัดเจนมากขึ้น และสามารถนำไปประยุกต์ใช้ในการเรียนรู้และการจัดการเรียนรู้ได้อย่างเหมาะสม โดยองค์ประกอบสำคัญของวิทยาการคำนวณอาจสรุปได้ดังนี้

1.2.1 การคิดเชิงคำนวณ

การคิดเชิงคำนวณเป็นแกนกลางสำคัญของวิทยาการคำนวณ เนื่องจากเป็นกระบวนการคิดที่ช่วยให้ผู้เรียนสามารถวิเคราะห์ปัญหาอย่างมีระบบ แยกส่วนปัญหาที่ซับซ้อนออกเป็นส่วนย่อย มองหารูปแบบร่วม สร้างนามธรรม และออกแบบขั้นตอนการแก้ปัญหาอย่างชัดเจน การคิดเชิงคำนวณจึงเป็นพื้นฐานของการเขียนโปรแกรม การออกแบบระบบ และการแก้ปัญหาในสถานการณ์ต่าง ๆ

1.2.2 ข้อมูลและสารสนเทศ

ข้อมูล (Data) เป็นข้อเท็จจริงที่ถูกรวบรวมไว้ ส่วนสารสนเทศ (Information) คือข้อมูลที่ผ่านการประมวลผลจนมีความหมาย วิทยาการคำนวณให้ความสำคัญกับการเก็บรวบรวมข้อมูล การจัดหมวดหมู่ การวิเคราะห์ การแปลความหมาย และการนำข้อมูลไปใช้ประกอบการตัดสินใจ ในยุคดิจิทัล ผู้เรียนจำเป็นต้องเข้าใจว่าข้อมูลมีบทบาทอย่างไรในระบบต่าง ๆ และควรใช้ข้อมูลอย่างมีวิจารณญาณ

1.2.3 อัลกอริทึม

อัลกอริทึม หรือ Algorithm คือ ลำดับขั้นตอนในการแก้ปัญหาหรือดำเนินงานอย่างเป็นระบบและชัดเจน เป็นองค์ประกอบสำคัญที่เชื่อมโยงระหว่างการคิดกับการปฏิบัติ เพราะเมื่อผู้เรียนเข้าใจปัญหาแล้ว จำเป็นต้องสามารถออกแบบวิธีการแก้ปัญหาเป็นลำดับขั้นตอนที่ผู้อื่นหรือเครื่องจักรสามารถปฏิบัติตามได้

1.2.4 การเขียนโปรแกรม

การเขียนโปรแกรม หรือ Programming เป็นกระบวนการนำแนวคิดหรืออัลกอริทึมมาแปลงเป็นคำสั่งที่คอมพิวเตอร์สามารถทำงานได้ องค์ประกอบนี้ช่วยให้ผู้เรียนได้เห็นผลลัพธ์ของแนวคิดเชิงนามธรรมในรูปของการทำงานจริง แม้วิทยาการคำนวณจะไม่เท่ากับการเขียนโปรแกรม แต่การเขียนโปรแกรมก็เป็นเครื่องมือสำคัญที่ช่วยพัฒนาความเข้าใจด้านตรรกะ ลำดับขั้นตอน และการตรวจสอบข้อผิดพลาด

1.2.5 เทคโนโลยีดิจิทัลและระบบคอมพิวเตอร์

องค์ประกอบนี้ครอบคลุมความเข้าใจเบื้องต้นเกี่ยวกับฮาร์ดแวร์ ซอฟต์แวร์ เครือข่าย การสื่อสารข้อมูล และระบบดิจิทัล ผู้เรียนควรตระหนักว่าเทคโนโลยีต่าง ๆ ทำงานอย่างไร มีข้อจำกัดและศักยภาพอย่างไร และสามารถเลือกใช้ได้อย่างเหมาะสมกับเป้าหมายของงานหรือการเรียนรู้

1.2.6 การตรวจสอบและปรับปรุงวิธีการแก้ปัญหา

ในวิทยาการคำนวณ การแก้ปัญหาไม่ได้สิ้นสุดเพียงการได้คำตอบ แต่รวมถึงการตรวจสอบความถูกต้อง ประสิทธิภาพ และความเหมาะสมของวิธีการที่ใช้ด้วย ผู้เรียนจึงควรฝึกทดสอบ เปรียบเทียบ ปรับปรุง และสะท้อนคิดเกี่ยวกับวิธีการแก้ปัญหาของตนอยู่เสมอ

เมื่อพิจารณาองค์ประกอบทั้งหมดร่วมกัน จะเห็นได้ว่าวิทยาการคำนวณเป็นศาสตร์ที่ผสมผสานการคิด การใช้ข้อมูล การออกแบบวิธีการ การใช้เครื่องมือ และการประเมินผลเข้าไว้ด้วยกันอย่างมีระบบ การจัดการเรียนรู้ในศาสตร์นี้จึงควรออกแบบให้ผู้เรียนได้เรียนรู้ครบทุกองค์ประกอบ มิใช่เน้นเพียงทักษะใดทักษะหนึ่งเพียงลำพัง

1.3 แนวคิดการคิดเชิงคำนวณ (Computational Thinking)

การคิดเชิงคำนวณเป็นแนวคิดสำคัญที่ได้รับการยอมรับอย่างกว้างขวางว่าเป็นหัวใจของวิทยาการคำนวณ เพราะเป็นกระบวนการคิดที่ช่วยให้มนุษย์สามารถแก้ปัญหาอย่างมีประสิทธิภาพ เป็นระบบ และสามารถนำแนวทางที่ได้ไปประยุกต์ใช้ซ้ำในสถานการณ์อื่นได้ การคิดเชิงคำนวณไม่จำเป็นต้องเกิดขึ้นเฉพาะขณะใช้คอมพิวเตอร์ แต่เป็นวิธีคิดที่มนุษย์สามารถใช้ได้ในชีวิตประจำวัน

โดยทั่วไป การคิดเชิงคำนวณประกอบด้วยแนวคิดหลัก 4 ประการ ได้แก่ การแยกย่อยปัญหา การหารูปแบบ การคิดเชิงนามธรรม และการออกแบบอัลกอริทึม

1.3.1 การแยกย่อยปัญหา (Decomposition)

การแยกย่อยปัญหา คือ การแบ่งปัญหาขนาดใหญ่หรือซับซ้อนออกเป็นปัญหาย่อยที่จัดการได้ง่ายขึ้น วิธีคิดนี้ช่วยให้ผู้เรียนไม่รู้สึกว่าปัญหายากเกินไป และสามารถจัดลำดับการจัดการได้อย่างเหมาะสม ตัวอย่างเช่น หากต้องการจัดกิจกรรมวันวิทยาศาสตร์ในโรงเรียน ผู้เรียนสามารถแยกงานออกเป็นการวางแผนสถานที่ การเตรียมอุปกรณ์ การแบ่งหน้าที่ และการประชาสัมพันธ์ เป็นต้น

1.3.2 การหารูปแบบ (Pattern Recognition)

เมื่อผู้เรียนแยกย่อยปัญหาแล้ว ขั้นต่อไปคือการมองหารูปแบบที่คล้ายคลึงกันในปัญหาหรือข้อมูล การสังเกตรูปแบบช่วยให้สามารถใช้วิธีการเดิมกับปัญหาใหม่ หรือสร้างแนวทางที่มีประสิทธิภาพมากขึ้น ตัวอย่างเช่น ในการสังเกตพฤติกรรมการส่งงานของนักเรียน อาจพบรูปแบบว่าการส่งงานล่าช้ามักเกิดขึ้นในสัปดาห์ที่มีหลายวิชากำหนดงานพร้อมกัน

1.3.3 การคิดเชิงนามธรรม (Abstraction)

การคิดเชิงนามธรรม คือ การคัดเลือกเฉพาะสาระสำคัญของปัญหา และละทิ้งรายละเอียดที่ไม่จำเป็น เพื่อให้มองเห็นแก่นของปัญหาได้ชัดเจนขึ้น เช่น หากต้องการออกแบบระบบยืมคืนหนังสือในห้องสมุด สิ่งสำคัญคือข้อมูลผู้ยืม รายการหนังสือ วันที่ยืม และวันที่คืน ส่วนรายละเอียดอื่นที่ไม่เกี่ยวกับกระบวนการหลักอาจยังไม่จำเป็นต้องพิจารณาในระยะแรก

1.3.4 การออกแบบอัลกอริทึม (Algorithm Design)

เมื่อเข้าใจปัญหาและสาระสำคัญแล้ว ผู้เรียนต้องสามารถออกแบบลำดับขั้นตอนในการแก้ปัญหาอย่างเป็นระบบ ชัดเจน และตรวจสอบได้ การออกแบบอัลกอริทึมเป็นผลลัพธ์ของการคิดเชิงคำนวณที่ทำให้แนวคิดสามารถนำไปปฏิบัติได้จริง ตัวอย่างเช่น การเขียนขั้นตอนการตัดสินผลการเรียนจากคะแนนรวม หรือการกำหนดลำดับขั้นตอนของการให้นักเรียนลงทะเบียนเข้าร่วมกิจกรรม

การคิดเชิงคำนวณมีประโยชน์อย่างมากต่อการเรียนรู้ เพราะช่วยให้ผู้เรียนพัฒนาความสามารถในการคิดอย่างเป็นเหตุเป็นผล ลดความสับสนเมื่อต้องเผชิญปัญหาซับซ้อน และเสริมสร้างนิสัยการทำงานอย่างเป็นระบบ นอกจากนี้ การคิดเชิงคำนวณยังสามารถบูรณาการกับวิชาอื่นได้ เช่น คณิตศาสตร์ วิทยาศาสตร์ ภาษา หรือสังคมศึกษา โดยใช้กระบวนการคิดเดียวกันในการสำรวจข้อมูล วิเคราะห์ปัญหา และสร้างแนวทางแก้ไข

สำหรับนักศึกษาครู การเข้าใจการคิดเชิงคำนวณมีความสำคัญอย่างยิ่ง เพราะจะช่วยให้สามารถออกแบบกิจกรรมที่ไม่เพียงสอนคำตอบ แต่สอน “วิธีคิด” ให้แก่ผู้เรียน เช่น การใช้เกม การใช้สถานการณ์จำลอง การใช้ผังงาน การใช้บัตรคำ หรือการให้ผู้เรียนวางแผนแก้ปัญหาจากสถานการณ์ใกล้ตัว ทั้งหมดนี้ล้วนเป็นแนวทางที่ช่วยให้การคิดเชิงคำนวณเกิดขึ้นอย่างเป็นรูปธรรมในชั้นเรียน

1.4 วิทยาการคำนวณกับการแก้ปัญหาในชีวิตประจำวันและการศึกษา

วิทยาการคำนวณมิได้เป็นศาสตร์ที่จำกัดอยู่ภายในห้องปฏิบัติการคอมพิวเตอร์หรือวิชาการเขียนโปรแกรมเท่านั้น แต่เป็นศาสตร์ที่มีบทบาทอย่างใกล้ชิดกับการดำเนินชีวิตในทุกวัน เนื่องจากมนุษย์ต้องเผชิญกับปัญหา การตัดสินใจ และการจัดการข้อมูลอยู่เสมอ หากบุคคลมีพื้นฐานวิทยาการคำนวณที่ดี ก็จะสามารถใช้กระบวนการคิดอย่างเป็นระบบเพื่อรับมือกับสถานการณ์ต่าง ๆ ได้อย่างเหมาะสม

ในชีวิตประจำวัน ตัวอย่างของการใช้วิทยาการคำนวณสามารถพบได้ในกิจกรรมง่าย ๆ เช่น การวางแผนเส้นทางไปยังสถานที่หนึ่งโดยเลือกเส้นทางที่ประหยัดเวลา การจัดลำดับงานที่ต้องทำในแต่ละวัน การเปรียบเทียบราคาสินค้าก่อนตัดสินใจซื้อ หรือการแบ่งขั้นตอนการทำอาหารให้ออกมาเป็นลำดับที่ชัดเจน กิจกรรมเหล่านี้ล้วนสะท้อนการคิดเชิงคำนวณ เพราะผู้ปฏิบัติต้องวิเคราะห์ข้อมูล แยกปัญหา มองรูปแบบ และตัดสินใจบนพื้นฐานของเหตุผล

ในด้านการศึกษา วิทยาการคำนวณช่วยให้ผู้เรียนพัฒนาทักษะที่จำเป็นต่อการเรียนรู้ในทุกกลุ่มสาระ เช่น ในวิชาคณิตศาสตร์ ผู้เรียนใช้การคิดอย่างเป็นลำดับขั้นในการแก้โจทย์ ในวิชาวิทยาศาสตร์ ผู้เรียนวิเคราะห์ตัวแปรและออกแบบการทดลอง ในวิชาภาษา ผู้เรียนจัดระบบความคิดก่อนการเขียนหรือการสื่อสาร และในวิชาสังคมศึกษา ผู้เรียนอาจวิเคราะห์ข้อมูลและเหตุปัจจัยของปัญหาสังคมอย่างมีระบบ ดังนั้น วิทยาการคำนวณจึงสามารถทำหน้าที่เป็นฐานคิดที่ช่วยยกระดับคุณภาพการเรียนรู้โดยรวม นอกจากนี้ วิทยาการคำนวณยังมีบทบาทต่อการออกแบบนวัตกรรมทางการศึกษา เช่น การสร้างสื่อการสอนเชิงโต้ตอบ การออกแบบเกมการเรียนรู้ การใช้ข้อมูลผู้เรียนเพื่อปรับกิจกรรมให้เหมาะสม และการพัฒนาชิ้นงานดิจิทัลที่ตอบสนองต่อปัญหาของผู้เรียนหรือสถานศึกษา การเรียนรู้วิทยาการคำนวณจึงมิได้

จบลงที่การรับรู้เนื้อหา แต่สามารถต่อยอดสู่การสร้างสรรค์ผลงานและการพัฒนาสภาพแวดล้อมทางการเรียนรู้ได้จริง

สำหรับนักศึกษาครู การมองเห็นความเชื่อมโยงระหว่างวิทยาการคำนวณกับชีวิตจริงเป็นสิ่งสำคัญ เพราะจะช่วยให้การจัดการเรียนรู้ไม่เป็นนามธรรมจนเกินไป ครูควรยกตัวอย่างสถานการณ์ใกล้ตัว เช่น การจัดตารางเรียน การเดินทางจากบ้านไปโรงเรียน การวางแผนใช้เงินค่าขนม หรือการจัดกิจกรรมภายในโรงเรียน เพื่อให้ผู้เรียนเข้าใจว่าวิทยาการคำนวณไม่ใช่เรื่องไกลตัว แต่เป็นเครื่องมือในการคิดและการใช้ชีวิตอย่างมีระบบ

กล่าวได้ว่า วิทยาการคำนวณเป็นทั้งศาสตร์แห่งการคิดและศาสตร์แห่งการปฏิบัติ ซึ่งมีคุณค่าอย่างยิ่งต่อการพัฒนาผู้เรียนให้เป็นบุคคลที่สามารถคิดเป็น แก้ปัญหาเป็น ใช้เทคโนโลยีเป็น และเรียนรู้ได้อย่างต่อเนื่องในโลกที่เปลี่ยนแปลงอย่างรวดเร็ว

1.5 แนวทางการจัดการเรียนรู้วิทยาการคำนวณสำหรับผู้เรียนระดับประถมศึกษาและมัธยมศึกษา

การจัดการเรียนรู้วิทยาการคำนวณสำหรับผู้เรียนระดับประถมศึกษาและมัธยมศึกษาควรคำนึงถึงพัฒนาการ ความพร้อม ประสบการณ์เดิม และบริบทของผู้เรียนเป็นสำคัญ เนื่องจากเป้าหมายของรายวิชานี้มิใช่เพียงให้ผู้เรียนเขียนโปรแกรมได้ แต่เพื่อพัฒนาวิธีคิด ทักษะการแก้ปัญหา การทำงานร่วมกัน และการประยุกต์ใช้เทคโนโลยีอย่างสร้างสรรค์และเหมาะสม

1.5.1 หลักการสำคัญในการจัดการเรียนรู้

ครูควรจัดกิจกรรมให้ผู้เรียนได้ลงมือปฏิบัติจริง ผ่านสถานการณ์หรือปัญหาที่ใกล้ตัว ใช้คำถามกระตุ้นการคิด เปิดโอกาสให้ผู้เรียนลองผิดลองถูก และส่งเสริมการสะท้อนคิดหลังการทำกิจกรรม การเรียนรู้วิทยาการคำนวณที่ดีจึงควรเป็นการเรียนรู้เชิงรุก (Active Learning) ที่ผู้เรียนมีส่วนร่วมมากกว่าการรับฟังคำอธิบายเพียงอย่างเดียว

1.5.2 แนวทางสำหรับระดับประถมศึกษา

ในระดับประถมศึกษา ควรเน้นกิจกรรมที่เป็นรูปธรรม เข้าใจง่าย และสัมพันธ์กับชีวิตประจำวัน เช่น เกมจัดลำดับขั้นตอน การใช้บัตรคำสั่ง การฝึกคิดผ่านนิทานหรือสถานการณ์ การเขียนผังงานอย่างง่าย และการเขียนโปรแกรมแบบบล็อกคำสั่ง กิจกรรมควรเน้นความสนุก การสังเกต การทดลอง และการสร้างความคุ้นเคยกับแนวคิดพื้นฐาน เช่น ลำดับ เหตุผล เงื่อนไข และรูปแบบ

1.5.3 แนวทางสำหรับระดับมัธยมศึกษา

ในระดับมัธยมศึกษา ผู้เรียนสามารถเรียนรู้แนวคิดที่ซับซ้อนมากขึ้น เช่น การออกแบบอัลกอริทึม การวิเคราะห์ปัญหาที่มีหลายเงื่อนไข การเขียนผังงานอย่างเป็นระบบ การเขียนโปรแกรมเบื้องต้น และการสร้างโครงงานขนาดเล็ก ครูควรออกแบบกิจกรรมที่ท้าทายมากขึ้น เปิดโอกาสให้ผู้เรียนทำงานเป็นทีม แก้ปัญหาเชิงสถานการณ์ และเชื่อมโยงกับปัญหาจริงในโรงเรียนหรือชุมชน

1.5.4 การใช้สื่อและเครื่องมือที่เหมาะสม

การจัดการเรียนรู้วิทยาการคำนวณสามารถใช้สื่อได้หลากหลาย ทั้งสื่อไม่ใช้คอมพิวเตอร์ (Unplugged Activities) และสื่อดิจิทัล เช่น โปรแกรมผังงาน โปรแกรมแบบบล็อกคำสั่ง ไมโครบิต หรือภาษาโปรแกรมเบื้องต้น การเลือกใช้สื่อควรพิจารณาจากจุดประสงค์การเรียนรู้และความพร้อมของผู้เรียน ไม่ควรให้เทคโนโลยีเป็นจุดสนใจหลักจนกลบกระบวนการคิด

1.5.5 การวัดและประเมินผล

การประเมินผลในวิทยาการคำนวณควรประเมินทั้งความรู้ กระบวนการคิด ทักษะการปฏิบัติ และคุณลักษณะในการทำงาน เช่น ความรับผิดชอบ ความร่วมมือ และความพยายามในการแก้ปัญหา รูปแบบการประเมินอาจประกอบด้วยการสังเกตพฤติกรรม การตรวจผลงาน การประเมินภารกิจ การใช้แบบสะท้อนคิด และการประเมินจากการนำเสนอผลงาน

1.5.6 บทบาทของครูในฐานะผู้ออกแบบการเรียนรู้

ครูในรายวิชาวิทยาการคำนวณควรทำหน้าที่เป็นผู้อำนวยความสะดวกในการเรียนรู้ (Facilitator) มากกว่าผู้ถ่ายทอดความรู้เพียงฝ่ายเดียว ครูต้องช่วยตั้งคำถาม กระตุ้นการคิด ให้ข้อเสนอแนะ และออกแบบประสบการณ์การเรียนรู้ที่เปิดโอกาสให้ผู้เรียนค้นพบแนวคิดด้วยตนเอง สำหรับนักศึกษาครู การฝึกออกแบบกิจกรรมการเรียนรู้วิทยาการคำนวณจึงเป็นทักษะสำคัญที่ต้องได้รับการพัฒนาอย่างจริงจัง

สรุปได้ว่า การจัดการเรียนรู้วิทยาการคำนวณที่มีคุณภาพต้องยึดผู้เรียนเป็นสำคัญ เชื่อมโยงกับชีวิตจริง ใช้กิจกรรมที่เหมาะสมกับวัย และเน้นการพัฒนากระบวนการคิดมากกว่าการท่องจำคำสั่งหรือขั้นตอนเพียงอย่างเดียว

บทสรุปประจำบทที่ 1

วิทยาการคำนวณเป็นศาสตร์พื้นฐานสำคัญที่มุ่งพัฒนาความสามารถในการคิด วิเคราะห์ วางแผน และแก้ปัญหาอย่างเป็นระบบ โดยมิได้จำกัดอยู่เพียงการเขียนโปรแกรมคอมพิวเตอร์ แต่ครอบคลุมถึงการใช้ข้อมูล การออกแบบอัลกอริทึม การทำความเข้าใจเทคโนโลยี และการประเมินผลวิธีการแก้ปัญหา แนวคิดสำคัญของศาสตร์นี้คือการคิดเชิงคำนวณ ซึ่งประกอบด้วยการแยกย่อยปัญหา การหารูปแบบ การคิดเชิงนามธรรม และการออกแบบอัลกอริทึม

นอกจากนี้ วิทยาการคำนวณยังมีความสัมพันธ์อย่างใกล้ชิดกับการดำเนินชีวิตและการจัดการศึกษา เพราะช่วยให้ผู้เรียนสามารถแก้ปัญหาในสถานการณ์จริงได้อย่างมีเหตุผล และสามารถนำไปประยุกต์ใช้ในรายวิชาอื่นและการสร้างนวัตกรรมทางการศึกษาได้ สำหรับนักศึกษาครู ความเข้าใจพื้นฐานของวิทยาการคำนวณเป็นสิ่งจำเป็นทั้งในฐานะผู้เรียนและในฐานะผู้ที่จะออกแบบการเรียนรู้ให้แก่ผู้เรียนระดับประถมศึกษาและมัธยมศึกษาในอนาคต การจัดการเรียนรู้วิทยาการคำนวณจึงควรเน้นกิจกรรมที่ส่งเสริมการคิด การลงมือปฏิบัติ การเชื่อมโยงกับชีวิตจริง และการประเมินผลอย่างหลากหลาย เพื่อพัฒนาผู้เรียนอย่างรอบด้าน

คำถามท้ายบท

1. จงอธิบายความหมายของวิทยาการคำนวณ และชี้ให้เห็นว่าวิทยาการคำนวณแตกต่างจากการเขียนโปรแกรมอย่างไร
2. เพราะเหตุใดวิทยาการคำนวณจึงมีความสำคัญต่อผู้เรียนในศตวรรษที่ 21
3. องค์ประกอบสำคัญของวิทยาการคำนวณมีอะไรบ้าง จงอธิบายโดยสังเขป
4. การคิดเชิงคำนวณคืออะไร และประกอบด้วยแนวคิดหลักอะไรบ้าง
5. จงยกตัวอย่างสถานการณ์ในชีวิตประจำวัน 1 สถานการณ์ และอธิบายว่ามีการใช้การคิดเชิงคำนวณอย่างไร
6. วิทยาการคำนวณสามารถบูรณาการกับรายวิชาอื่นในโรงเรียนได้อย่างไร
7. หากท่านเป็นครูผู้สอนระดับประถมศึกษา ท่านจะออกแบบกิจกรรมเพื่อส่งเสริมการคิดเชิงคำนวณอย่างไร

8. หากท่านเป็นครูผู้สอนระดับมัธยมศึกษา ท่านจะจัดกิจกรรมวิทยาการคำนวณให้เชื่อมโยงกับชีวิตจริงของผู้เรียนอย่างไร
9. เพราะเหตุใดการประเมินผลรายวิชาวิทยาการคำนวณจึงไม่ควรเน้นเฉพาะคำตอบสุดท้าย
10. จงสรุปบทบาทของครูในการจัดการเรียนรู้วิทยาการคำนวณ

เอกสารอ้างอิง

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

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.

Computer Science Teachers Association. (2017). *CSTA K-12 Computer Science Standards*. New York, NY: Author.

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 Annual Meeting of the American Educational Research Association*. Vancouver, Canada.

National Research Council. (2010). *Report of a workshop on the scope and nature of computational thinking*. Washington, DC: The National Academies Press.

บทที่ 2
เทคโนโลยีดิจิทัล
Digital Technology

บทนำ

เทคโนโลยีดิจิทัลได้เข้ามามีบทบาทอย่างลึกซึ้งต่อการดำเนินชีวิตของมนุษย์ในทุกมิติ ทั้งด้านการสื่อสาร การศึกษา การทำงาน เศรษฐกิจ สังคม และวัฒนธรรม การเปลี่ยนแปลงจากโลกแบบแอนะล็อกสู่โลกดิจิทัลทำให้รูปแบบการเข้าถึงข้อมูล การแลกเปลี่ยนความรู้ และการสร้างนวัตกรรมเกิดขึ้นอย่างรวดเร็วและต่อเนื่อง ผู้คนในสังคมปัจจุบันจึงจำเป็นต้องมีความรู้ ความเข้าใจ และทักษะในการใช้เทคโนโลยีดิจิทัลอย่างเหมาะสม เพื่อให้สามารถปรับตัวต่อการเปลี่ยนแปลง และใช้ประโยชน์จากเทคโนโลยีในการพัฒนาคุณภาพชีวิตได้อย่างมีประสิทธิภาพ

ในบริบททางการศึกษา เทคโนโลยีดิจิทัลมิได้เป็นเพียงเครื่องมือเสริมการจัดการเรียนรู้เท่านั้น แต่ยังเป็นกลไกสำคัญในการเปลี่ยนแปลงรูปแบบการเรียนรู้จากการรับสารแบบทางเดียวไปสู่การเรียนรู้ที่มีปฏิสัมพันธ์ ยืดหยุ่น เข้าถึงได้ทุกที่ทุกเวลา และตอบสนองต่อความแตกต่างระหว่างบุคคลได้มากขึ้น เทคโนโลยีดิจิทัลจึงมีความสำคัญทั้งในฐานะเครื่องมือสำหรับครู ผู้เรียน และผู้บริหารการศึกษา รวมถึงเป็นพื้นฐานในการพัฒนาทักษะแห่งอนาคตของผู้เรียน

สำหรับนักศึกษาครู ความเข้าใจเกี่ยวกับเทคโนโลยีดิจิทัลมีความจำเป็นอย่างยิ่ง เพราะครูในยุคปัจจุบันไม่อาจทำหน้าที่เพียงถ่ายทอดความรู้ตามตำรา แต่ต้องสามารถเลือกใช้ ออกแบบ และประยุกต์ใช้เทคโนโลยีเพื่อส่งเสริมการเรียนรู้ได้อย่างเหมาะสมกับบริบทของผู้เรียน โรงเรียน และสังคม นอกจากนี้ ครูยังต้องมีวิจารณญาณในการใช้เทคโนโลยี รู้เท่าทันการเปลี่ยนแปลง และสามารถปลูกฝังการใช้เทคโนโลยีอย่างสร้างสรรค์และรับผิดชอบให้แก่ผู้เรียน

บทนี้มุ่งอธิบายความหมายและลักษณะของเทคโนโลยีดิจิทัล องค์ประกอบของระบบดิจิทัลและเทคโนโลยีสารสนเทศ บทบาทของเทคโนโลยีดิจิทัลต่อการเรียนรู้และการดำรงชีวิต การประยุกต์ใช้เทคโนโลยีดิจิทัลในบริบทการศึกษา ตลอดจนแนวทางการออกแบบกิจกรรมการเรียนรู้โดยใช้เทคโนโลยีดิจิทัลอย่างเหมาะสม เพื่อให้นักศึกษาครูมีพื้นฐานความเข้าใจที่ถูกต้อง และสามารถนำองค์ความรู้ไปใช้ในการจัดการเรียนรู้ได้อย่างมีคุณภาพ

2.1 ความหมายและลักษณะของเทคโนโลยีดิจิทัล

เทคโนโลยีดิจิทัล หมายถึง เทคโนโลยีที่เกี่ยวข้องกับการสร้าง การจัดเก็บ การประมวลผล การสื่อสาร และการนำเสนอข้อมูลในรูปแบบดิจิทัล โดยข้อมูลดังกล่าวจะถูกแปลงให้อยู่ในรูปของสัญญาณตัวเลขหรือรหัสดิจิทัล ซึ่งคอมพิวเตอร์และอุปกรณ์อิเล็กทรอนิกส์สามารถประมวลผลได้อย่างรวดเร็วและมีประสิทธิภาพ เทคโนโลยีดิจิทัลจึงครอบคลุมทั้งฮาร์ดแวร์ ซอฟต์แวร์ เครือข่าย อินเทอร์เน็ต แอปพลิเคชัน ระบบคลาวด์ ปัญญาประดิษฐ์ และเครื่องมือดิจิทัลรูปแบบต่าง ๆ ที่ใช้ในชีวิตประจำวันและการทำงาน

คำว่า “ดิจิทัล” หรือ “Digital” สื่อถึงการแทนข้อมูลด้วยระบบเลขฐานสอง ได้แก่ 0 และ 1 ซึ่งเป็นพื้นฐานของการทำงานของคอมพิวเตอร์ เมื่อข้อมูลประเภทต่าง ๆ เช่น ข้อความ ภาพ เสียง หรือวิดีโอ ถูกแปลงเป็นข้อมูลดิจิทัลแล้ว จะสามารถจัดเก็บ ส่งต่อ แก้ไข และประมวลผลได้สะดวกยิ่งขึ้น ความสามารถนี้ทำให้เทคโนโลยีดิจิทัลแตกต่างจากเทคโนโลยีแบบเดิมที่มีข้อจำกัดด้านความเร็ว พื้นที่จัดเก็บ และความยืดหยุ่นในการใช้งาน

ลักษณะสำคัญของเทคโนโลยีดิจิทัลสามารถอธิบายได้หลายประการ ประการแรก คือ ความรวดเร็วในการประมวลผลและการสื่อสารข้อมูล ซึ่งช่วยให้ผู้ใช้เข้าถึงข้อมูลหรือส่งข้อมูลได้ในเวลาอันสั้น ประการที่สอง คือ ความสามารถในการจัดเก็บข้อมูลจำนวนมากในพื้นที่จำกัด ประการที่สาม คือ ความยืดหยุ่นในการใช้งาน เนื่องจากสามารถนำข้อมูลเดียวกันไปใช้ได้หลากหลายรูปแบบ เช่น อ่าน แก้ไข แชร์ หรือวิเคราะห์ต่อ ประการที่สี่ คือ การเชื่อมโยงเครือข่ายที่ทำให้เกิดการสื่อสารและการทำงานร่วมกันแบบไร้พรมแดน และประการที่ห้า คือ การพัฒนาอย่างต่อเนื่อง ซึ่งส่งผลให้ผู้ใช้ต้องเรียนรู้และปรับตัวอยู่เสมอ

อีกลักษณะหนึ่งที่สำคัญของเทคโนโลยีดิจิทัล คือ การทำให้กิจกรรมต่าง ๆ ในชีวิตประจำวันเกิดความสะดวกและมีประสิทธิภาพมากขึ้น เช่น การใช้สมาร์ตโฟนในการติดต่อสื่อสาร การใช้ระบบนำทางในการเดินทาง การซื้อขายสินค้าออนไลน์ การใช้แพลตฟอร์มการเรียนรู้ออนไลน์ และการใช้ระบบฐานข้อมูลเพื่อจัดการสารสนเทศ การเปลี่ยนแปลงเหล่านี้สะท้อนให้เห็นว่าเทคโนโลยีดิจิทัลมิได้เป็นเพียงเครื่องมือเท่านั้น แต่เป็นส่วนหนึ่งของวิถีชีวิตสมัยใหม่

อย่างไรก็ตาม แม้เทคโนโลยีดิจิทัลจะมีประโยชน์อย่างมาก แต่ก็มีข้อจำกัดและความท้าทาย เช่น ความเหลื่อมล้ำในการเข้าถึงเทคโนโลยี ความเสี่ยงด้านความเป็นส่วนตัว การแพร่กระจายของข้อมูลเท็จ การพึ่งพาเทคโนโลยีมากเกินไป และผลกระทบต่อสุขภาวะจากการใช้งานที่ไม่เหมาะสม ดังนั้น การทำ

ความเข้าใจความหมายและลักษณะของเทคโนโลยีดิจิทัลจึงควรครอบคลุมทั้งด้านประโยชน์ ข้อจำกัด และความรับผิดชอบในการใช้เทคโนโลยีอย่างสมดุล

สำหรับนักศึกษาครู การเข้าใจลักษณะของเทคโนโลยีดิจิทัลจะช่วยให้สามารถเลือกใช้เครื่องมือได้อย่างเหมาะสมกับวัตถุประสงค์ของการเรียนรู้ และสามารถส่งเสริมให้ผู้เรียนใช้เทคโนโลยีอย่างรู้เท่าทัน สร้างสรรค์ และมีจริยธรรม

2.2 องค์ประกอบของระบบดิจิทัลและเทคโนโลยีสารสนเทศ

ระบบดิจิทัลและเทคโนโลยีสารสนเทศเป็นโครงสร้างสำคัญที่ทำให้การทำงานของเทคโนโลยีดิจิทัลเกิดขึ้นได้อย่างเป็นระบบ การทำความเข้าใจองค์ประกอบของระบบเหล่านี้ช่วยให้ผู้เรียนมองเห็นภาพรวมของการทำงานทางเทคโนโลยี ไม่เพียงในมิติของการใช้งาน แต่รวมถึงมิติของการจัดการข้อมูล การประมวลผล และการสื่อสารสารสนเทศอย่างมีประสิทธิภาพ

โดยทั่วไป องค์ประกอบของระบบดิจิทัลและเทคโนโลยีสารสนเทศสามารถแบ่งออกเป็นองค์ประกอบสำคัญดังนี้

2.2.1 ฮาร์ดแวร์ (Hardware)

ฮาร์ดแวร์ คือ อุปกรณ์ที่จับต้องได้ของระบบคอมพิวเตอร์และระบบดิจิทัล เช่น คอมพิวเตอร์ตั้งโต๊ะ คอมพิวเตอร์พกพา สมาร์ตโฟน แท็บเล็ต เครื่องพิมพ์ จอภาพ อุปกรณ์จัดเก็บข้อมูล และอุปกรณ์เครือข่าย ฮาร์ดแวร์ทำหน้าที่รับข้อมูล ประมวลผล แสดงผล และจัดเก็บข้อมูลตามคำสั่งที่ได้รับจากผู้ใช้หรือระบบ

2.2.2 ซอฟต์แวร์ (Software)

ซอฟต์แวร์ คือ ชุดคำสั่งหรือโปรแกรมที่ควบคุมการทำงานของฮาร์ดแวร์ ซอฟต์แวร์แบ่งออกเป็นสองประเภทหลัก ได้แก่ ซอฟต์แวร์ระบบ เช่น ระบบปฏิบัติการ และซอฟต์แวร์ประยุกต์ เช่น โปรแกรมประมวลผลคำ โปรแกรมนำเสนอ โปรแกรมตารางคำนวณ หรือแอปพลิเคชันเพื่อการเรียนรู้ ซอฟต์แวร์มีบทบาทสำคัญในการทำให้ผู้ใช้สามารถสั่งงานอุปกรณ์และดำเนินกิจกรรมต่าง ๆ ได้อย่างสะดวก

2.2.3 ข้อมูล (Data)

ข้อมูลเป็นวัตถุดิบพื้นฐานของระบบสารสนเทศ อาจอยู่ในรูปข้อความ ตัวเลข ภาพ เสียง หรือวิดีโอ ข้อมูลที่ถูกรวบรวมอย่างเป็นระบบและผ่านกระบวนการประมวลผล จะกลายเป็นสารสนเทศที่มีความหมายและสามารถนำไปใช้ประโยชน์ในการวางแผน ตัดสินใจ หรือแก้ปัญหาได้อย่างเหมาะสม

2.2.4 บุคลากรหรือผู้ใช้ (People)

ระบบดิจิทัลจะไม่สามารถทำงานได้อย่างมีประสิทธิภาพหากขาดผู้ใช้งานที่มีความรู้และทักษะ บุคลากรในระบบอาจได้แก่ ผู้ใช้ทั่วไป ครู นักเรียน ผู้ดูแลระบบ นักพัฒนาซอฟต์แวร์ และผู้บริหาร ซึ่งแต่ละกลุ่มมีบทบาทแตกต่างกัน การพัฒนาความรู้และทักษะของผู้ใช้จึงเป็นปัจจัยสำคัญที่ส่งผลต่อความสำเร็จของการใช้เทคโนโลยีดิจิทัล

2.2.5 กระบวนการทำงาน (Process)

กระบวนการทำงาน หมายถึง ลำดับขั้นตอนหรือวิธีปฏิบัติที่กำหนดไว้เพื่อให้ระบบดำเนินงานอย่างถูกต้อง เช่น ขั้นตอนการบันทึกข้อมูล การตรวจสอบความถูกต้อง การประมวลผล การจัดเก็บ และการเผยแพร่สารสนเทศ ระบบที่ดีควรมีกระบวนการที่ชัดเจนและตรวจสอบได้

2.2.6 เครือข่ายและการสื่อสารข้อมูล (Network and Communication)

เครือข่ายเป็นองค์ประกอบที่ช่วยให้อุปกรณ์และระบบต่าง ๆ สามารถเชื่อมต่อถึงกัน แลกเปลี่ยนข้อมูล และทำงานร่วมกันได้ เช่น อินเทอร์เน็ต ระบบเครือข่ายภายในโรงเรียน หรือเครือข่ายไร้สาย การสื่อสารข้อมูลที่มีประสิทธิภาพทำให้เกิดการเรียนรู้แบบออนไลน์ การประชุมทางไกล การใช้ทรัพยากรร่วมกัน และการเข้าถึงแหล่งข้อมูลทั่วโลก

เมื่อองค์ประกอบทั้งหมดทำงานร่วมกันอย่างสอดคล้อง จะก่อให้เกิดระบบสารสนเทศที่มีประสิทธิภาพ เช่น ระบบทะเบียนนักศึกษา ระบบห้องสมุดดิจิทัล ระบบจัดการเรียนรู้ หรือระบบติดตามผลการเรียน การเข้าใจองค์ประกอบเหล่านี้จึงมีความสำคัญอย่างยิ่งต่อการใช้และการจัดการเทคโนโลยีในบริบทการศึกษา

2.3 บทบาทของเทคโนโลยีดิจิทัลต่อการเรียนรู้และการดำรงชีวิต

เทคโนโลยีดิจิทัลมีบทบาทอย่างมากต่อการเรียนรู้และการดำรงชีวิตของมนุษย์ในสังคมปัจจุบัน ทั้งในระดับบุคคล ครอบครัว ชุมชน และสังคมโดยรวม การเปลี่ยนแปลงดังกล่าวส่งผลให้รูปแบบการใช้ชีวิต การทำงาน และการเข้าถึงความรู้เปลี่ยนแปลงไปอย่างมีนัยสำคัญ

ในด้านการเรียนรู้ เทคโนโลยีดิจิทัลช่วยขยายโอกาสในการเข้าถึงแหล่งความรู้ที่หลากหลาย ผู้เรียนสามารถศึกษาจากเว็บไซต์ บทเรียนออนไลน์ วิดีโอเพื่อการศึกษา ห้องเรียนเสมือนจริง และฐานข้อมูลดิจิทัลได้ทุกที่ทุกเวลา เทคโนโลยีจึงช่วยลดข้อจำกัดด้านเวลาและสถานที่ อีกทั้งยังสนับสนุนการเรียนรู้ตามความแตกต่างระหว่างบุคคล เพราะผู้เรียนสามารถเลือกเนื้อหา ความเร็ว และรูปแบบการเรียนรู้ที่เหมาะสมกับตนเองได้มากขึ้น

นอกจากนี้ เทคโนโลยีดิจิทัลยังส่งเสริมการเรียนรู้แบบมีปฏิสัมพันธ์ เช่น การใช้เกมการศึกษา การตอบคำถามแบบทันที การเรียนรู้ผ่านสื่อมัลติมีเดีย การทำงานร่วมกันผ่านแพลตฟอร์มออนไลน์ และการใช้เครื่องมือสร้างสรรค์ผลงานดิจิทัล สิ่งเหล่านี้ช่วยกระตุ้นความสนใจของผู้เรียน และทำให้การเรียนรู้มีความหมายมากยิ่งขึ้น

ในด้านการดำรงชีวิต เทคโนโลยีดิจิทัลทำให้การติดต่อสื่อสารเป็นไปอย่างรวดเร็วและสะดวก ผู้คนสามารถส่งข้อความ สนทนาผ่านวิดีโอ ประชุมออนไลน์ และติดต่อกันข้ามพื้นที่ได้โดยง่าย เทคโนโลยียังมีบทบาทต่อการทำธุรกรรมทางการเงิน การซื้อสินค้าออนไลน์ การเข้าถึงบริการสาธารณะ การนำทาง การทำงานทางไกล และการรับข้อมูลข่าวสารแบบเรียลไทม์ สิ่งเหล่านี้ทำให้คุณภาพชีวิตหลายด้านได้รับการยกระดับ

อย่างไรก็ตาม บทบาทของเทคโนโลยีดิจิทัลมิได้มีเพียงด้านบวกเท่านั้น หากใช้อย่างไม่เหมาะสมอาจก่อให้เกิดผลกระทบ เช่น การเสพติดสื่อ การใช้เวลาหน้าจอมากเกินไป การละเมิดความเป็นส่วนตัว การถูกหลอกลวงทางออนไลน์ หรือการรับข้อมูลที่บิดเบือน ดังนั้น การใช้เทคโนโลยีดิจิทัลจึงควรควบคู่ไปกับการพัฒนาทักษะการรู้เท่าทันดิจิทัล การคิดอย่างมีวิจารณญาณ และจริยธรรมในการใช้สื่อ

สำหรับนักศึกษาครู การตระหนักถึงบทบาทของเทคโนโลยีดิจิทัลทั้งในด้านโอกาสและความท้าทายเป็นสิ่งจำเป็น เพราะจะช่วยให้สามารถชี้นำผู้เรียนให้ใช้เทคโนโลยีอย่างสมดุล สร้างสรรค์ และรับผิดชอบ พร้อมทั้งเลือกใช้เทคโนโลยีเพื่อสนับสนุนการเรียนรู้และการพัฒนาตนเองได้อย่างเหมาะสม

2.4 การประยุกต์ใช้เทคโนโลยีดิจิทัลในบริบทการศึกษา

การประยุกต์ใช้เทคโนโลยีดิจิทัลในบริบทการศึกษาเป็นแนวทางสำคัญในการยกระดับคุณภาพการเรียนรู้ให้สอดคล้องกับสังคมแห่งการเปลี่ยนแปลง เทคโนโลยีดิจิทัลช่วยให้กระบวนการจัดการศึกษาเกิดความยืดหยุ่น ทันสมัย และตอบสนองต่อความต้องการของผู้เรียนได้ดีขึ้น ทั้งในด้านการจัดการเรียนการสอน การวัดและประเมินผล การบริหารจัดการชั้นเรียน และการพัฒนาวิชาชีพครู

ประการแรก เทคโนโลยีดิจิทัลสามารถใช้ในการจัดการเรียนการสอนผ่านแพลตฟอร์มการเรียนรู้ออนไลน์ เช่น ระบบจัดการเรียนรู้ ห้องเรียนเสมือนจริง หรือแอปพลิเคชันเพื่อการศึกษา ซึ่งช่วยให้ครูสามารถเผยแพร่เนื้อหา มอบหมายงาน ติดตามความก้าวหน้าของผู้เรียน และสื่อสารกับผู้เรียนได้อย่างต่อเนื่อง

ประการที่สอง เทคโนโลยีดิจิทัลช่วยสร้างสื่อการเรียนรู้ที่หลากหลายและน่าสนใจ เช่น อินโฟกราฟิก วิดีโอ แบบจำลอง สื่อปฏิสัมพันธ์ เกมการศึกษา และแบบฝึกหัดออนไลน์ สื่อเหล่านี้ช่วยให้ผู้เรียนเข้าใจเนื้อหาได้ง่ายขึ้น โดยเฉพาะเนื้อหาที่เป็นนามธรรม หรือซับซ้อน

ประการที่สาม เทคโนโลยีดิจิทัลช่วยในการวัดและประเมินผล เช่น การใช้แบบทดสอบออนไลน์ การเก็บข้อมูลผลการเรียนแบบทันที การวิเคราะห์ผลการเรียนรายบุคคล และการให้ข้อเสนอแนะอย่างรวดเร็ว การประเมินผลด้วยเทคโนโลยีจึงช่วยเพิ่มประสิทธิภาพในการติดตามพัฒนาการของผู้เรียน

ประการที่สี่ เทคโนโลยีดิจิทัลสามารถประยุกต์ใช้ในการบริหารจัดการศึกษา เช่น การจัดตารางเรียน การบริหารข้อมูลนักเรียน การสื่อสารกับผู้ปกครอง การบริหารเอกสาร และการจัดทำรายงานข้อมูลทางการศึกษา ระบบดิจิทัลช่วยลดภาระงานซ้ำซ้อนและเพิ่มความแม่นยำในการจัดการข้อมูล

ประการที่ห้า เทคโนโลยีดิจิทัลมีบทบาทต่อการพัฒนาวิชาชีพครู ครูสามารถเรียนรู้ผ่านหลักสูตรออนไลน์ ชุมชนการเรียนรู้ทางวิชาชีพ สื่อดิจิทัล และเครือข่ายความร่วมมือ ทำให้สามารถพัฒนาความรู้และทักษะใหม่ ๆ ได้อย่างต่อเนื่อง

อย่างไรก็ตาม การประยุกต์ใช้เทคโนโลยีดิจิทัลในบริบทการศึกษาควรตั้งอยู่บนหลักความเหมาะสม ไม่ควรใช้เทคโนโลยีเพียงเพื่อความทันสมัย แต่ควรคำนึงถึงวัตถุประสงค์การเรียนรู้ ความพร้อมของผู้เรียน ความพร้อมของโรงเรียน และความคุ้มค่าในการใช้งาน ครูจึงต้องมีบทบาทสำคัญในการตัดสินใจเลือกใช้เทคโนโลยีอย่างมีเหตุผลและคำนึงถึงผลลัพธ์การเรียนรู้เป็นสำคัญ

2.5 การออกแบบกิจกรรมการเรียนรู้โดยใช้เทคโนโลยีดิจิทัลอย่างเหมาะสม

การออกแบบกิจกรรมการเรียนรู้โดยใช้เทคโนโลยีดิจิทัลอย่างเหมาะสมเป็นทักษะสำคัญของครูในยุคปัจจุบัน เนื่องจากเทคโนโลยีเพียงอย่างเดียวมิอาจทำให้การเรียนรู้เกิดคุณภาพได้ หากขาดการออกแบบกิจกรรมที่สอดคล้องกับวัตถุประสงค์ เนื้อหา และลักษณะของผู้เรียน การใช้เทคโนโลยีในการเรียนการสอนจึงควรมุ่งให้ผู้เรียนได้คิด ลงมือปฏิบัติ สื่อสาร ทำงานร่วมกัน และสะท้อนผลการเรียนรู้ของตนเอง

2.5.1 การกำหนดวัตถุประสงค์การเรียนรู้ให้ชัดเจน

ก่อนเลือกใช้เทคโนโลยี ครูต้องกำหนดก่อนว่าต้องการให้ผู้เรียนเกิดการเรียนรู้อะไร เช่น ต้องการให้เข้าใจเนื้อหา ฝึกทักษะ วิเคราะห์ปัญหา สร้างชิ้นงาน หรือสะท้อนความคิดเห็น เมื่อวัตถุประสงค์ชัดเจนแล้ว จึงค่อยเลือกเครื่องมือดิจิทัลที่เหมาะสม ไม่ควรเริ่มจากการเลือกเทคโนโลยีก่อนแล้วค่อยหาวัตถุประสงค์มารองรับ

2.5.2 การเลือกเทคโนโลยีให้เหมาะกับวัยและบริบทของผู้เรียน

ผู้เรียนแต่ละช่วงวัยมีความพร้อมและข้อจำกัดแตกต่างกัน ในระดับประถมศึกษา อาจใช้เทคโนโลยีที่เน้นความเรียบง่าย ภาพชัดเจน ปฏิสัมพันธ์สูง และไม่ซับซ้อนมากนัก ส่วนระดับมัธยมศึกษาอาจใช้เครื่องมือที่ซับซ้อนขึ้น เช่น โปรแกรมสร้างชิ้นงาน การสืบค้นข้อมูล หรือการทำงานร่วมกันออนไลน์ นอกจากนี้ ครูยังต้องคำนึงถึงบริบทของโรงเรียน เช่น จำนวนอุปกรณ์ ความพร้อมของอินเทอร์เน็ต และความสามารถในการเข้าถึงของผู้เรียน

2.5.3 การออกแบบกิจกรรมให้ผู้เรียนมีส่วนร่วม

กิจกรรมการเรียนรู้ที่ดีควรเปิดโอกาสให้ผู้เรียนมีบทบาทเป็นผู้ลงมือทำ ไม่ใช่เพียงผู้รับชมสื่อ ตัวอย่างเช่น การให้ผู้เรียนค้นคว้าข้อมูลจากแหล่งดิจิทัลแล้วสรุปเป็นอินโฟกราฟิก การทำแบบฝึกหัดโต้ตอบ การสร้างสื่อดิจิทัลอย่างง่าย การอภิปรายออนไลน์ หรือการทำโครงงานโดยใช้เครื่องมือดิจิทัลสนับสนุน การมีส่วนร่วมอย่างแท้จริงจะช่วยให้ผู้เรียนเกิดความเข้าใจลึกซึ้งและจดจำได้ดีขึ้น

2.5.4 การใช้เทคโนโลยีเพื่อสนับสนุนการคิด ไม่ใช่แทนที่การคิด

หลักสำคัญประการหนึ่งในการใช้เทคโนโลยีดิจิทัล คือ การใช้เทคโนโลยีเพื่อกระตุ้นและสนับสนุนกระบวนการคิดของผู้เรียน ไม่ใช่ใช้เพียงเพื่อให้กิจกรรมดูน่าสนใจหรือสะดวกขึ้นเท่านั้น ตัวอย่างเช่น การใช้แบบจำลองเพื่อให้ผู้เรียนวิเคราะห์ความสัมพันธ์ของตัวแปร การใช้แอปพลิเคชันร่วมกันระดมความคิด หรือการใช้สื่อดิจิทัลเป็นฐานในการอภิปรายและตั้งคำถาม

2.5.5 การประเมินผลและสะท้อนผลการเรียนรู้

การจัดกิจกรรมด้วยเทคโนโลยีดิจิทัลควรมีการประเมินผลที่สอดคล้องกับกิจกรรม เช่น ประเมินจากชิ้นงานดิจิทัล การมีส่วนร่วมในกิจกรรมออนไลน์ การตอบคำถาม การนำเสนอ หรือการเขียนสะท้อนคิด การประเมินผลควรครอบคลุมทั้งด้านความรู้ ทักษะ กระบวนการ และเจตคติ เพื่อให้เห็นพัฒนาการของผู้เรียนอย่างรอบด้าน

2.5.6 ตัวอย่างแนวทางสำหรับนักศึกษาครู

นักศึกษาครูอาจเริ่มต้นจากการออกแบบกิจกรรมอย่างง่าย เช่น การใช้แบบทดสอบออนไลน์เพื่อทบทวนบทเรียน การใช้วิดีโอสั้นนำเข้าสู่บทเรียน การให้ผู้เรียนค้นคว้าข้อมูลจากเว็บไซต์ที่ครูกำหนดแล้วสรุปสาระสำคัญเป็นแผนผังความคิด หรือการสร้างกิจกรรมกลุ่มที่ใช้เครื่องมือดิจิทัลในการรวบรวมความคิดเห็น วิธีการเหล่านี้จะช่วยให้นักศึกษาครูฝึกผสมผสานความรู้ด้านเนื้อหา วิธีสอน และเทคโนโลยีเข้าด้วยกันอย่างเหมาะสม

โดยสรุป การออกแบบกิจกรรมการเรียนรู้โดยใช้เทคโนโลยีดิจิทัลอย่างเหมาะสม จำเป็นต้องอาศัยทั้งความเข้าใจในผู้เรียน ความชัดเจนของจุดประสงค์ และการเลือกใช้เทคโนโลยีอย่างมีวิจารณญาณ เพื่อให้เทคโนโลยีทำหน้าที่เป็นเครื่องมือส่งเสริมการเรียนรู้ มิใช่เป็นเพียงสิ่งแปลกใหม่ที่แยกขาดจากสาระสำคัญของการศึกษา

บทสรุปประจำบทที่ 2

เทคโนโลยีดิจิทัลเป็นเทคโนโลยีที่เกี่ยวข้องกับการจัดเก็บ ประมวลผล สื่อสาร และนำเสนอข้อมูลในรูปแบบดิจิทัล ซึ่งมีบทบาทสำคัญอย่างยิ่งต่อการดำรงชีวิตและการเรียนรู้ในโลกปัจจุบัน ลักษณะเด่นของเทคโนโลยีดิจิทัล ได้แก่ ความรวดเร็ว ความยืดหยุ่น ความสามารถในการเชื่อมโยงเครือข่าย และศักยภาพในการพัฒนาอย่างต่อเนื่อง การทำความเข้าใจองค์ประกอบของระบบดิจิทัลและเทคโนโลยีสารสนเทศ เช่น ฮาร์ดแวร์ ซอฟต์แวร์ ข้อมูล บุคลากร กระบวนการ และเครือข่าย จะช่วยให้ผู้เรียนสามารถใช้เทคโนโลยีได้อย่างมีประสิทธิภาพและมีวิจารณญาณ

นอกจากนี้ เทคโนโลยีดิจิทัลยังมีบทบาทสำคัญต่อการขยายโอกาสทางการเรียนรู้ การสร้างสื่อและนวัตกรรมการศึกษา การวัดและประเมินผล ตลอดจนการบริหารจัดการทางการศึกษา อย่างไรก็ตาม การใช้เทคโนโลยีในบริบทการศึกษาควรคำนึงถึงความเหมาะสมกับวัตถุประสงค์ ผู้เรียน และบริบทแวดล้อมเป็นสำคัญ สำหรับนักศึกษาครู ความสามารถในการออกแบบกิจกรรมการเรียนรู้โดยใช้เทคโนโลยีดิจิทัลอย่างเหมาะสมจึงเป็นสมรรถนะจำเป็นที่ต้องได้รับการพัฒนา เพื่อให้สามารถจัดการเรียนรู้ที่มีคุณภาพ สอดคล้องกับการเปลี่ยนแปลงของสังคม และตอบสนองต่อความต้องการของผู้เรียนในศตวรรษที่ 21

คำถามท้ายบท

1. จงอธิบายความหมายของเทคโนโลยีดิจิทัล และยกตัวอย่างเทคโนโลยีดิจิทัลที่ใช้ในชีวิตประจำวัน
2. ลักษณะสำคัญของเทคโนโลยีดิจิทัลมีอะไรบ้าง และลักษณะเหล่านี้ส่งผลต่อการใช้ชีวิตอย่างไร
3. องค์ประกอบของระบบดิจิทัลและเทคโนโลยีสารสนเทศมีอะไรบ้าง จงอธิบายพอสังเขป
4. ข้อมูลและสารสนเทศแตกต่างกันอย่างไร พร้อมยกตัวอย่างประกอบ
5. เทคโนโลยีดิจิทัลมีบทบาทต่อการเรียนรู้อย่างไรในยุคปัจจุบัน
6. จงวิเคราะห์ทั้งประโยชน์และผลกระทบของเทคโนโลยีดิจิทัลต่อการดำรงชีวิต

7. ยกตัวอย่างการประยุกต์ใช้เทคโนโลยีดิจิทัลในบริบทการศึกษาอย่างน้อย 3 ตัวอย่าง
8. เพราะเหตุใดการใช้เทคโนโลยีดิจิทัลในการจัดการเรียนรู้จึงควรคำนึงถึงความเหมาะสมมากกว่าความทันสมัย
9. หากท่านเป็นครูผู้สอนระดับมัธยมศึกษา ท่านจะออกแบบกิจกรรมการเรียนรู้โดยใช้เทคโนโลยีดิจิทัลอย่างไรให้ผู้เรียนมีส่วนร่วม
10. จงอธิบายบทบาทของครูในการส่งเสริมให้ผู้เรียนใช้เทคโนโลยีดิจิทัลอย่างสร้างสรรค์และรับผิดชอบ

เอกสารอ้างอิง

กระทรวงดิจิทัลเพื่อเศรษฐกิจและสังคม. (ม.ป.ป.). *แนวคิดและการพัฒนาดิจิทัลเพื่อสังคมและเศรษฐกิจ*. กรุงเทพมหานคร: ผู้แต่ง.

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

Laudon, K. C., & Laudon, J. P. (2020). *Management information systems: Managing the digital firm* (16th ed.). Harlow: Pearson.

Selwyn, N. (2016). *Education and technology: Key issues and debates* (2nd ed.). London: Bloomsbury.

UNESCO. (2023). *Digital learning and transformation of education*. Paris: UNESCO.

OECD. (2021). *21st-century readers: Developing literacy skills in a digital world*. Paris: OECD Publishing.

บทที่ 3
การรู้เท่าทันดิจิทัล
Digital Literacy

บทนำ

ในโลกยุคดิจิทัล ข้อมูล ข่าวสาร และสื่อต่าง ๆ ถูกส่งผ่านเครือข่ายอินเทอร์เน็ตอย่างรวดเร็วและต่อเนื่อง ผู้คนสามารถเข้าถึงความรู้ ติดต่อสื่อสาร แลกเปลี่ยนความคิดเห็น และสร้างสรรค์เนื้อหาได้ด้วยอุปกรณ์ดิจิทัลหลากหลายรูปแบบ เช่น สมาร์ตโฟน แท็บเล็ต คอมพิวเตอร์ และแพลตฟอร์มสื่อสังคมออนไลน์ ความก้าวหน้าทางเทคโนโลยีเหล่านี้ก่อให้เกิดโอกาสอย่างมากต่อการเรียนรู้ การทำงาน และการดำรงชีวิต แต่ในขณะเดียวกันก็สร้างความท้าทายใหม่ที่ผู้ใช้ทุกคนจำเป็นต้องเผชิญ เช่น ข่าวปลอม การบิดเบือนข้อมูล การละเมิดความเป็นส่วนตัว การกลั่นแกล้งทางออนไลน์ และการใช้สื่ออย่างไม่เหมาะสม

การรู้เท่าทันดิจิทัลจึงกลายเป็นสมรรถนะสำคัญของพลเมืองในศตวรรษที่ 21 เพราะการใช้เทคโนโลยีดิจิทัลอย่างมีประสิทธิภาพไม่ได้หมายถึงเพียงความสามารถในการใช้งานอุปกรณ์หรือแอปพลิเคชันเท่านั้น แต่ยังรวมถึงความสามารถในการเข้าถึงข้อมูลอย่างมีเป้าหมาย วิเคราะห์และประเมินข้อมูลอย่างมีวิจารณญาณ ใช้สื่ออย่างปลอดภัย มีจริยธรรม และรับผิดชอบต่อผลกระทบที่เกิดขึ้นต่อตนเองและสังคม

ในมิติทางการศึกษา การรู้เท่าทันดิจิทัลมีความสำคัญอย่างยิ่งต่อผู้เรียนและครู ผู้เรียนจำเป็นต้องมีภูมิคุ้มกันทางความคิด สามารถเลือกใช้สื่ออย่างเหมาะสม แยกแยะข้อเท็จจริงออกจากความคิดเห็น และไม่ตกเป็นเหยื่อของข้อมูลที่บิดเบือน ขณะเดียวกัน ครูจำเป็นต้องมีความรู้ความเข้าใจในเรื่องนี้ เพื่อทำหน้าที่ทั้งเป็นผู้ใช้สื่ออย่างมีคุณภาพ และเป็นผู้ออกแบบกิจกรรมการเรียนรู้ที่ช่วยพัฒนาทักษะดังกล่าวให้แก่ผู้เรียน

บทนี้มุ่งอธิบายความหมายและความสำคัญของการรู้เท่าทันดิจิทัล ทักษะสำคัญของการรู้เท่าทันดิจิทัลในศตวรรษที่ 21 การใช้สื่อดิจิทัลอย่างปลอดภัย มีจริยธรรม และรับผิดชอบ การวิเคราะห์ข้อมูลข่าวสาร และสื่อดิจิทัลอย่างมีวิจารณญาณ ตลอดจนแนวทางการออกแบบกิจกรรมส่งเสริมการรู้เท่าทันดิจิทัลสำหรับผู้เรียน เพื่อให้นักศึกษาครูมีพื้นฐานความเข้าใจที่ถูกต้อง และสามารถนำไปประยุกต์ใช้ในการจัดการเรียนรู้ได้อย่างเหมาะสม

3.1 ความหมายและความสำคัญของการรู้เท่าทันดิจิทัล

การรู้เท่าทันดิจิทัล (Digital Literacy) หมายถึง ความสามารถในการเข้าถึง ใช้ วิเคราะห์ ประเมิน สร้างสรรค์ และสื่อสารข้อมูลผ่านเทคโนโลยีดิจิทัลอย่างมีประสิทธิภาพ เหมาะสม และรับผิดชอบ ความหมายของการรู้เท่าทันดิจิทัลจึงกว้างกว่าความสามารถในการใช้อุปกรณ์หรือโปรแกรมต่าง ๆ เพราะครอบคลุมทั้งทักษะเชิงเทคนิค ทักษะการคิด และจริยธรรมในการใช้สื่อ

ในอดีต การรู้หนังสือ หรือ Literacy อาจหมายถึงเพียงการอ่านออกเขียนได้ แต่ในโลกปัจจุบัน การดำรงชีวิตในสังคมที่ขับเคลื่อนด้วยข้อมูลจำเป็นต้องอาศัยความสามารถมากกว่านั้น บุคคลต้องสามารถเข้าถึงข้อมูลจากแหล่งต่าง ๆ ได้อย่างเหมาะสม รู้จักเลือกใช้เครื่องมือดิจิทัล วิเคราะห์คุณภาพของข้อมูล แยกแยะระหว่างข้อเท็จจริงกับความคิดเห็น และสามารถตัดสินใจบนพื้นฐานของเหตุผลและหลักฐาน การรู้เท่าทันดิจิทัลจึงเป็นพื้นฐานของการเป็นพลเมืองดิจิทัลที่มีคุณภาพ

ความสำคัญของการรู้เท่าทันดิจิทัลสามารถพิจารณาได้หลายประการ ประการแรก คือ ความสำคัญต่อการดำรงชีวิตประจำวัน เพราะผู้คนต้องใช้เทคโนโลยีในการสื่อสาร สืบค้นข้อมูล ทำธุรกรรมทางการเงิน ซื้อขายสินค้า และเข้าถึงบริการสาธารณะ ประการที่สอง คือ ความสำคัญต่อการเรียนรู้ เนื่องจากผู้เรียนในยุคปัจจุบันต้องศึกษาค้นคว้าจากสื่อดิจิทัลอยู่เสมอ หากขาดทักษะในการประเมินข้อมูล ก็อาจเชื่อข้อมูลที่คลาดเคลื่อนหรือไม่ถูกต้องได้ง่าย ประการที่สาม คือ ความสำคัญต่อความปลอดภัยและคุณธรรม เพราะการใช้สื่ออย่างไม่รู้เท่าทันอาจทำให้เกิดความเสี่ยง เช่น การเปิดเผยข้อมูลส่วนบุคคล การถูกหลอกลวงทางออนไลน์ หรือการเผยแพร่เนื้อหาที่ไม่เหมาะสม และประการที่สี่ คือ ความสำคัญต่อสังคมประชาธิปไตย เพราะพลเมืองที่รู้เท่าทันข้อมูลจะสามารถพิจารณาประเด็นสาธารณะอย่างมีเหตุผล ไม่ถูกชักจูงโดยข้อมูลที่บิดเบือนโดยง่าย

ในบริบทของผู้เรียนระดับการศึกษาขั้นพื้นฐาน การรู้เท่าทันดิจิทัลเป็นทักษะที่ควรได้รับการส่งเสริมตั้งแต่ระดับต้น เพื่อสร้างภูมิคุ้มกันทางความคิดและวางรากฐานการใช้เทคโนโลยีอย่างเหมาะสม ส่วนในบริบทของนักศึกษาครู การรู้เท่าทันดิจิทัลยิ่งมีความสำคัญมากขึ้น เพราะครูมิใช่เพียงผู้ใช้สื่อ แต่ยังเป็นผู้คัดเลือกสื่อ ออกแบบกิจกรรมการเรียนรู้ และเป็นแบบอย่างในการใช้เทคโนโลยีให้แก่ผู้เรียนอีกด้วย

กล่าวโดยสรุป การรู้เท่าทันดิจิทัลเป็นสมรรถนะที่จำเป็นต่อการดำรงชีวิต การเรียนรู้ และการอยู่ร่วมกันในสังคมดิจิทัล เพราะช่วยให้บุคคลสามารถใช้เทคโนโลยีอย่างมีประสิทธิภาพ ปลอดภัย มีเหตุผล และรับผิดชอบต่อผลกระทบที่เกิดขึ้น

3.2 ทักษะสำคัญของการรู้เท่าทันดิจิทัลในศตวรรษที่ 21

การรู้เท่าทันดิจิทัลในศตวรรษที่ 21 มิได้เป็นเพียงทักษะใดทักษะหนึ่ง แต่เป็นชุดของความสามารถหลายด้านที่เกื้อหนุนกันอย่างเป็นระบบ บุคคลที่มีการรู้เท่าทันดิจิทัลควรสามารถใช้เทคโนโลยีและสื่อได้อย่างเหมาะสม พร้อมทั้งคิด วิเคราะห์ และตัดสินใจได้อย่างมีคุณภาพ ทักษะสำคัญของการรู้เท่าทันดิจิทัลสามารถอธิบายได้ดังนี้

3.2.1 ทักษะการเข้าถึงข้อมูลดิจิทัล

ทักษะการเข้าถึงข้อมูลดิจิทัล หมายถึง ความสามารถในการค้นหา เข้าถึง และเลือกใช้ข้อมูลจากแหล่งดิจิทัลได้อย่างเหมาะสม ผู้เรียนควรรู้จักใช้คำค้น การเลือกใช้แหล่งข้อมูลที่สอดคล้องกับวัตถุประสงค์ และการเข้าถึงทรัพยากรดิจิทัลอย่างมีประสิทธิภาพ การเข้าถึงข้อมูลที่ดีเป็นจุดเริ่มต้นของการเรียนรู้ในโลกดิจิทัล

3.2.2 ทักษะการวิเคราะห์และประเมินข้อมูล

เมื่อเข้าถึงข้อมูลได้แล้ว ผู้เรียนต้องสามารถวิเคราะห์และประเมินคุณภาพของข้อมูล โดยพิจารณาความถูกต้อง ความน่าเชื่อถือ ความเป็นปัจจุบัน และเจตนาของผู้ส่งสาร ทักษะนี้ช่วยให้ผู้เรียนไม่หลงเชื่อข้อมูลโดยง่าย และสามารถตัดสินใจโดยอาศัยหลักฐานที่เหมาะสม

3.2.3 ทักษะการสื่อสารผ่านสื่อดิจิทัล

ผู้เรียนควรสามารถใช้เทคโนโลยีดิจิทัลในการสื่อสารอย่างสร้างสรรค์ เหมาะสม และรับผิดชอบทั้งในรูปแบบข้อความ ภาพ เสียง หรือวิดีโอ ทักษะการสื่อสารผ่านสื่อดิจิทัลยังรวมถึงมารยาทในการสื่อสาร การเคารพความคิดเห็นที่แตกต่าง และการเลือกใช้ช่องทางสื่อสารให้เหมาะกับสถานการณ์

3.2.4 ทักษะการสร้างสรรค์เนื้อหาดิจิทัล

ในโลกดิจิทัล ผู้ใช้ไม่ได้เป็นเพียงผู้รับสาร แต่ยังสามารถเป็นผู้สร้างสารได้ด้วย ดังนั้น ผู้เรียนควรมีความสามารถในการผลิตเนื้อหาดิจิทัล เช่น การเขียนบทความ การออกแบบอินโฟกราฟิก การสร้างสื่อวิดีโอ หรือการจัดทำสื่อการนำเสนออย่างมีคุณภาพ โดยต้องคำนึงถึงความถูกต้อง ความเหมาะสม และจริยธรรมในการใช้ข้อมูลประกอบ

3.2.5 ทักษะด้านความปลอดภัยดิจิทัล

ทักษะด้านความปลอดภัยดิจิทัลเกี่ยวข้องกับการปกป้องข้อมูลส่วนบุคคล การตั้งรหัสผ่านอย่างปลอดภัย การระมัดระวังภัยคุกคามทางไซเบอร์ การรู้เท่าทันกลโกงออนไลน์ และการตระหนักถึงผลกระทบจากร่องรอยดิจิทัลที่ตนสร้างไว้ การมีทักษะด้านความปลอดภัยดิจิทัลช่วยให้ผู้เรียนสามารถใช้เทคโนโลยีได้อย่างมั่นใจและปลอดภัยมากขึ้น

3.2.6 ทักษะด้านจริยธรรมและความรับผิดชอบ

การใช้เทคโนโลยีดิจิทัลควรตั้งอยู่บนพื้นฐานของจริยธรรม ผู้เรียนต้องเคารพสิทธิของผู้อื่น ไม่ละเมิดลิขสิทธิ์ ไม่เผยแพร่ข้อมูลเท็จ ไม่กลั่นแกล้งผู้อื่นทางออนไลน์ และรับผิดชอบต่อผลของการกระทำของตนในโลกดิจิทัล ทักษะนี้มีความสำคัญอย่างยิ่งต่อการอยู่ร่วมกันในสังคมออนไลน์อย่างสร้างสรรค์

3.2.7 ทักษะการปรับตัวและเรียนรู้อย่างต่อเนื่อง

เทคโนโลยีดิจิทัลเปลี่ยนแปลงอย่างรวดเร็ว ผู้เรียนจึงจำเป็นต้องมีความสามารถในการเรียนรู้สิ่งใหม่ ปรับตัวต่อเครื่องมือและแพลตฟอร์มใหม่ ๆ และพัฒนาตนเองอย่างต่อเนื่อง ทักษะนี้จะช่วยให้สามารถดำรงชีวิตและทำงานในโลกอนาคตได้อย่างยืดหยุ่น

จากที่กล่าวมาจะเห็นได้ว่า การรู้เท่าทันดิจิทัลเป็นความสามารถแบบบูรณาการที่เชื่อมโยงทั้งการใช้เครื่องมือ การคิดวิเคราะห์ ความปลอดภัย จริยธรรม และการสร้างสรรค์ ครูจึงควรจัดการเรียนรู้ที่พัฒนาทักษะเหล่านี้ไปพร้อมกัน ไม่ควรเน้นเพียงการใช้งานเทคโนโลยีในเชิงเทคนิคเท่านั้น

3.3 การใช้สื่อดิจิทัลอย่างปลอดภัย มีจริยธรรม และรับผิดชอบ

การใช้สื่อดิจิทัลเป็นส่วนหนึ่งของชีวิตประจำวันของผู้เรียนในยุคปัจจุบัน ไม่ว่าจะเป็นการสื่อสารกับเพื่อน การค้นหาข้อมูล การรับชมความบันเทิง หรือการแสดงความคิดเห็นบนสื่อสังคมออนไลน์ แม้ว่าสื่อดิจิทัลจะช่วยอำนวยความสะดวกและเปิดโอกาสในการเรียนรู้จำนวนมาก แต่หากใช้อย่างขาดความระมัดระวัง ก็อาจก่อให้เกิดผลกระทบทั้งต่อตนเองและผู้อื่น ดังนั้น การใช้สื่ออย่างปลอดภัย มีจริยธรรม และรับผิดชอบจึงเป็นพื้นฐานสำคัญของการเป็นพลเมืองดิจิทัล

ประเด็นแรก คือ ความปลอดภัยในการใช้สื่อดิจิทัล ผู้เรียนควรตระหนักถึงความสำคัญของข้อมูลส่วนบุคคล และหลีกเลี่ยงการเปิดเผยข้อมูลที่อาจก่อให้เกิดความเสี่ยง เช่น ที่อยู่ เบอร์โทรศัพท์ รหัสผ่าน หรือข้อมูลส่วนตัวอื่น ๆ ต่อสาธารณะ นอกจากนี้ ควรมีความระมัดระวังในการคลิกลิงก์ ดาวน์โหลดไฟล์ หรือรับข้อความจากแหล่งที่ไม่น่าเชื่อถือ เพราะอาจนำไปสู่การหลอกลวง การขโมยข้อมูล หรือการติดมัลแวร์ได้

ประเด็นที่สอง คือ จริยธรรมในการใช้สื่อดิจิทัล ผู้ใช้ควรตระหนักว่าการสื่อสารในโลกออนไลน์มีผลกระทบต่อความรู้สึกและสิทธิของผู้อื่นเช่นเดียวกับการสื่อสารในชีวิตจริง การใช้ถ้อยคำควรสุภาพ เคารพความคิดเห็นที่แตกต่าง ไม่ใช้สื่อเพื่อกลั่นแกล้ง ดูหมิ่น หรือทำให้ผู้อื่นเสียหาย การกลั่นแกล้งทางไซเบอร์เป็นปัญหาสำคัญที่อาจส่งผลกระทบต่อสภาพจิตใจของผู้ถูกกระทำอย่างรุนแรง จึงเป็นเรื่องที่ครูและผู้เรียนควรตระหนักร่วมกัน

ประเด็นที่สาม คือ ความรับผิดชอบต่อข้อมูลและเนื้อหาที่เผยแพร่ ผู้ใช้สื่อดิจิทัลควรตรวจสอบข้อมูลก่อนแชร์หรือส่งต่อ ไม่ควรเผยแพร่ข่าวลือ ข่าวปลอม หรือข้อมูลที่ยังไม่ได้รับการยืนยัน เพราะการกระทำดังกล่าวอาจสร้างความเข้าใจผิดและก่อให้เกิดผลเสียในวงกว้าง นอกจากนี้ ควรคำนึงถึงผลกระทบของเนื้อหาที่โพสต์หรือแสดงความคิดเห็นว่าอาจส่งผลต่อบุคคล กลุ่มคน หรือสังคมอย่างไร

อีกประเด็นหนึ่งที่สำคัญคือ การเคารพลิขสิทธิ์และสิทธิทางปัญญา ผู้เรียนควรเรียนรู้ว่า ภาพ ข้อความ วิดีโอ เพลง หรือผลงานต่าง ๆ บนโลกดิจิทัลล้วนมีเจ้าของ การนำไปใช้ควรอ้างอิงแหล่งที่มาอย่างเหมาะสม ไม่คัดลอก ดัดแปลง หรือเผยแพร่โดยไม่ได้รับอนุญาต การฝึกให้ผู้เรียนเคารพสิทธิของผู้อื่นตั้งแต่ระดับต้น จะช่วยสร้างพื้นฐานจริยธรรมในการใช้สื่อที่ยั่งยืน

ในบริบทของการศึกษา ครูควรเป็นแบบอย่างที่ดีในการใช้สื่อดิจิทัลอย่างปลอดภัยและรับผิดชอบ พร้อมทั้งจัดกิจกรรมที่ช่วยให้ผู้เรียนได้ฝึกคิดและตัดสินใจในประเด็นเหล่านี้ เช่น การวิเคราะห์กรณีศึกษาการละเมิดความเป็นส่วนตัว การอภิปรายเรื่องผลกระทบของการแชร์ข้อมูลโดยไม่ตรวจสอบ หรือการจำลองสถานการณ์เกี่ยวกับการสื่อสารที่เหมาะสมในโลกออนไลน์

กล่าวโดยสรุป การใช้สื่อดิจิทัลอย่างปลอดภัย มีจริยธรรม และรับผิดชอบ เป็นรากฐานสำคัญของการอยู่ร่วมกันในสังคมดิจิทัลอย่างสร้างสรรค์ และเป็นทักษะที่ครูควรส่งเสริมให้เกิดขึ้นกับผู้เรียนอย่างต่อเนื่อง

3.4 การวิเคราะห์ข้อมูล ข่าวสาร และสื่อดิจิทัลอย่างมีวิจารณญาณ

ในยุคที่ข้อมูลข่าวสารถูกผลิตและเผยแพร่อย่างรวดเร็วผ่านเครือข่ายดิจิทัล ผู้เรียนจำเป็นต้องมีความสามารถในการวิเคราะห์ข้อมูล ข่าวสาร และสื่อดิจิทัลอย่างมีวิจารณญาณ เพราะการรับสารโดยปราศจากการไตร่ตรองอาจทำให้เกิดความเข้าใจผิด ถูกชักจูง หรือเผยแพร่ข้อมูลที่ไม่ถูกต้องต่อไปได้

การวิเคราะห์อย่างมีวิจารณญาณเริ่มจากการพิจารณาแหล่งที่มาของข้อมูล ผู้เรียนควรตั้งคำถามว่าข้อมูลนั้นมาจากใคร เป็นหน่วยงาน องค์กร หรือบุคคลที่มีความน่าเชื่อถือหรือไม่ มีความเชี่ยวชาญในประเด็นนั้นเพียงใด และมีวัตถุประสงค์ใดในการนำเสนอข้อมูล ข้อมูลจากแหล่งที่น่าเชื่อถือมักมีหลักฐานอ้างอิง มีความชัดเจน และตรวจสอบย้อนกลับได้

นอกจากแหล่งที่มาแล้ว ผู้เรียนควรพิจารณาเนื้อหาและวิธีการนำเสนอด้วย เช่น ข้อมูลนั้นมีข้อเท็จจริงรองรับหรือไม่ ใช้ภาษาที่กระตุ้นอารมณ์เกินไปหรือไม่ มีการเลือกนำเสนอเพียงบางส่วนเพื่อชี้นำความคิดหรือไม่ รวมถึงภาพ วิดีโอ หรือกราฟิกประกอบนั้นมีความสอดคล้องกับเนื้อหาจริงหรือไม่ เพราะสื่อบางประเภทอาจใช้เทคนิคการนำเสนอเพื่อทำให้ผู้รับสารเชื่อโดยไม่ตั้งคำถาม

อีกแนวทางหนึ่งที่สำคัญ คือ การเปรียบเทียบข้อมูลจากหลายแหล่ง หากพบข้อมูลที่น่าสงสัย ผู้เรียนควรตรวจสอบจากแหล่งข้อมูลอื่นที่เชื่อถือได้ เช่น หน่วยงานทางวิชาการ สื่อมวลชนที่มีมาตรฐาน หรือเว็บไซต์ทางการของหน่วยงานที่เกี่ยวข้อง การเปรียบเทียบข้อมูลช่วยให้เห็นความสอดคล้องหรือความแตกต่าง และลดโอกาสในการสรุปอย่างผิดพลาดจากข้อมูลเพียงแหล่งเดียว

ผู้เรียนควรตระหนักด้วยว่า ยอดไลก์ ยอดแชร์ หรือยอดเข้าชม ไม่ได้เป็นตัวชี้วัดความถูกต้องของข้อมูลเสมอไป เนื้อหาบางอย่างอาจได้รับความนิยมเพราะสร้างอารมณ์ ความตื่นเต้น หรือความขัดแย้ง แต่ไม่ได้หมายความว่าเป็นข้อมูลจริง ดังนั้น การหยุดคิดก่อนเชื่อ และตรวจสอบก่อนแชร์ จึงเป็นหลักสำคัญของการใช้สื่ออย่างมีวิจารณญาณ

สำหรับนักศึกษาครู ทักษะการวิเคราะห์ข้อมูลและสื่อดิจิทัลอย่างมีวิจารณญาณมีความสำคัญเป็นพิเศษ เพราะครูทำหน้าที่คัดเลือกสื่อประกอบการเรียนการสอนและมีอิทธิพลต่อแนวคิดของผู้เรียน หากครูสามารถประเมินสื่อได้อย่างเหมาะสม ก็จะช่วยให้ผู้เรียนได้รับข้อมูลที่มีคุณภาพ และเรียนรู้วิธีคิดเชิงวิพากษ์จากการสังเกตและการฝึกปฏิบัติร่วมกับครู

กล่าวโดยสรุป การวิเคราะห์ข้อมูล ข่าวสาร และสื่อดิจิทัลอย่างมีวิจารณญาณ เป็นทักษะสำคัญที่ช่วยให้ผู้เรียนสามารถใช้เหตุผล หลักฐาน และการตรวจสอบอย่างรอบคอบในการตัดสินใจ ซึ่งเป็นพื้นฐานของการเป็นผู้ใช้สื่อที่มีคุณภาพในสังคมดิจิทัล

3.5 การออกแบบกิจกรรมส่งเสริมการรู้เท่าทันดิจิทัลสำหรับผู้เรียน

การส่งเสริมการรู้เท่าทันดิจิทัลให้แก่ผู้เรียนไม่ควรจำกัดอยู่เพียงการบรรยายหรือการให้ข้อควรระวังเท่านั้น แต่ควรออกแบบเป็นกิจกรรมการเรียนรู้ที่เปิดโอกาสให้ผู้เรียนได้คิด วิเคราะห์ ตัดสินใจ และสะท้อนผลจากสถานการณ์ที่ใกล้ตัว การออกแบบกิจกรรมจึงเป็นภารกิจสำคัญของครูในการพัฒนาผู้เรียนให้สามารถใช้สื่อดิจิทัลอย่างปลอดภัย มีวิจารณญาณ และมีจริยธรรม

หลักการสำคัญประการแรก คือ การใช้สถานการณ์จริงหรือกรณีตัวอย่างใกล้ตัวผู้เรียน เช่น ข่าวปลอมในสื่อสังคมออนไลน์ การแชร์ข้อมูลโดยไม่ตรวจสอบ การแสดงความคิดเห็นที่ไม่เหมาะสม การเปิดเผยข้อมูลส่วนบุคคล หรือการกลั่นแกล้งทางออนไลน์ เมื่อผู้เรียนเห็นว่าสิ่งที่เรียนเกี่ยวข้องกับชีวิตจริง ก็จะเกิดความตระหนักและมีแรงจูงใจในการเรียนรู้มากขึ้น

แนวทางหนึ่งที่เหมาะสม คือ การใช้กรณีศึกษา ครูอาจนำตัวอย่างข่าว ข้อความ หรือโพสต์จากสื่อดิจิทัลมาให้ผู้เรียนวิเคราะห์ โดยตั้งคำถาม เช่น ข้อมูลนี้น่าเชื่อถือหรือไม่ เพราะเหตุใด หากพบข้อมูล

เช่นนี้ควรปฏิบัติอย่างไร หรือหากเป็นผู้ผลิตสื่อควรคำนึงถึงประเด็นใดบ้าง กิจกรรมลักษณะนี้ช่วยให้ผู้เรียนพัฒนาทั้งการคิดวิเคราะห์และการตัดสินใจเชิงจริยธรรม

อีกแนวทางหนึ่ง คือ การอภิปรายและแลกเปลี่ยนความคิดเห็นในประเด็นดิจิทัล เช่น เสรีภาพในการแสดงความคิดเห็นกับความรับผิดชอบต่อสังคม ความสะดวกของเทคโนโลยีกับความเสี่ยงด้านความเป็นส่วนตัว หรือข้อดีและข้อจำกัดของปัญญาประดิษฐ์ในการเข้าถึงข้อมูล การอภิปรายช่วยให้ผู้เรียนเห็นความหลากหลายของมุมมอง และฝึกใช้เหตุผลในการสื่อสาร

ครูยังสามารถออกแบบกิจกรรมสร้างสรรค์ เช่น ให้ผู้เรียนจัดทำอินโฟกราฟิก คลิปวิดีโอสั้น โปสเตอร์ หรือคู่มือการใช้สื่ออย่างรู้เท่าทันสำหรับเพื่อนหรือชุมชนโรงเรียน กิจกรรมลักษณะนี้ช่วยให้ผู้เรียนสังเคราะห์ความรู้และแปลงความเข้าใจของตนออกมาเป็นชิ้นงานที่เป็นรูปธรรม

สำหรับระดับประถมศึกษา กิจกรรมควรเป็นรูปธรรม เรียบง่าย และใช้สถานการณ์ใกล้ตัว เช่น การแยกแยะข้อมูลจริงกับข้อมูลลวง การฝึกตั้งรหัสผ่านที่ปลอดภัย การสังเกตว่าข้อมูลใดควรหรือไม่ควรเปิดเผย หรือการเล่นบทบาทสมมุติเกี่ยวกับการสื่อสารอย่างสุภาพในโลกออนไลน์ ส่วนในระดับมัธยมศึกษา อาจใช้กิจกรรมที่ซับซ้อนขึ้น เช่น การเปรียบเทียบข่าวจากหลายแหล่ง การวิเคราะห์เจตนาของผู้ผลิตสื่อ การอภิปรายจริยธรรมดิจิทัล หรือการออกแบบรณรงค์เกี่ยวกับการใช้สื่ออย่างรับผิดชอบ

ในด้านการประเมินผล ครูควรใช้วิธีที่หลากหลาย เช่น การสังเกตการมีส่วนร่วม การประเมินการตัดสินใจจากกรณีศึกษา การตรวจชิ้นงาน การประเมินการนำเสนอ และการสะท้อนคิดของผู้เรียน การประเมินเช่นนี้จะช่วยให้เห็นทั้งความรู้ ทักษะ และเจตคติของผู้เรียนอย่างรอบด้าน

โดยสรุป การออกแบบกิจกรรมส่งเสริมการรู้เท่าทันดิจิทัลควรเป็นกระบวนการที่เชื่อมโยงกับชีวิตจริง เปิดโอกาสให้ผู้เรียนลงมือคิดและตัดสินใจด้วยตนเอง และมุ่งพัฒนาผู้เรียนให้เป็นผู้ใช้สื่อดิจิทัลที่ปลอดภัย มีวิจารณญาณ มีจริยธรรม และรับผิดชอบต่อสังคม

บทสรุปประจำบทที่ 3

การรู้เท่าทันดิจิทัลเป็นสมรรถนะสำคัญของผู้เรียนในศตวรรษที่ 21 เพราะโลกปัจจุบันเต็มไปด้วยข้อมูล ข่าวสาร และสื่อดิจิทัลที่เปลี่ยนแปลงอย่างรวดเร็ว การใช้เทคโนโลยีอย่างมีคุณภาพจึงต้องอาศัยทั้งความสามารถในการเข้าถึงข้อมูล การวิเคราะห์และประเมินข้อมูล การสื่อสารและสร้างสรรค์เนื้อหา ตลอดจนการใช้สื่ออย่างปลอดภัย มีจริยธรรม และรับผิดชอบ การรู้เท่าทันดิจิทัลจึงมิได้หมายถึงเพียงการใช้งานเครื่องมือได้ แต่หมายถึงการใช้เทคโนโลยีด้วยความเข้าใจ รอบคอบ และคำนึงถึงผลกระทบที่อาจเกิดขึ้น

การพัฒนาผู้เรียนให้มีการรู้เท่าทันดิจิทัลจำเป็นต้องอาศัยการจัดการเรียนรู้ที่เชื่อมโยงกับชีวิตจริง เปิดโอกาสให้ผู้เรียนได้ฝึกคิด วิเคราะห์ ตัดสินใจ และสร้างสรรค์ผลงานจากสถานการณ์ที่พบได้ในโลกดิจิทัล สำหรับนักศึกษาครู ความเข้าใจในเรื่องนี้มีความสำคัญทั้งในฐานะผู้ใช้สื่อและในฐานะผู้ออกแบบการเรียนรู้ ครูจึงควรมีบทบาทในการปลูกฝังผู้เรียนให้เป็นพลเมืองดิจิทัลที่มีคุณภาพ สามารถใช้สื่ออย่างปลอดภัย มีวิจารณญาณ และมีความรับผิดชอบต่อสังคม

คำถามท้ายบท

1. จงอธิบายความหมายของการรู้เท่าทันดิจิทัล และชี้ให้เห็นว่าแตกต่างจากการใช้เทคโนโลยีเป็นอย่างไร
2. เพราะเหตุใดการรู้เท่าทันดิจิทัลจึงมีความสำคัญต่อผู้เรียนในศตวรรษที่ 21
3. ทักษะสำคัญของการรู้เท่าทันดิจิทัลมีอะไรบ้าง จงอธิบายโดยสังเขป
4. จงยกตัวอย่างพฤติกรรมการใช้สื่อดิจิทัลอย่างปลอดภัย มีจริยธรรม และรับผิดชอบ
5. การเปิดเผยข้อมูลส่วนบุคคลในโลกออนไลน์อาจก่อให้เกิดผลกระทบอย่างไรบ้าง
6. การวิเคราะห์ข้อมูล ข่าวสาร และสื่อดิจิทัลอย่างมีวิจารณญาณควรพิจารณาประเด็นใดบ้าง
7. เพราะเหตุใดจึงไม่ควรเชื่อข้อมูลจากแหล่งเดียว หรือพิจารณาความจริงจากยอดการแชร์เพียงอย่างเดียว
8. หากท่านเป็นครูระดับประถมศึกษา ท่านจะออกแบบกิจกรรมเพื่อส่งเสริมการรู้เท่าทันดิจิทัลอย่างไร
9. หากท่านเป็นครูระดับมัธยมศึกษา ท่านจะใช้กรณีศึกษาเรื่องข่าวปลอมเพื่อพัฒนาทักษะของผู้เรียนอย่างไร
10. จงอธิบายบทบาทของครูในการพัฒนาผู้เรียนให้เป็นพลเมืองดิจิทัลที่มีคุณภาพ

เอกสารอ้างอิง

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

Hobbs, R. (2017). *Create to learn: Introduction to digital literacy*. Hoboken, NJ: Wiley.

Livingstone, S. (2014). Developing social media literacy: How children learn to interpret risky opportunities on social network sites. *Communications, 39*(3), 283–303.

Ng, W. (2012). Can we teach digital natives digital literacy? *Computers & Education, 59*(3), 1065–1078.

Ribble, M. (2015). *Digital citizenship in schools* (3rd ed.). Eugene, OR: International Society for Technology in Education.

UNESCO. (2018). *A global framework of reference on digital literacy skills for indicator 4.4.2*. Paris: UNESCO.

บทที่ 4
โครงสร้างการควบคุมโปรแกรมแบบเรียงลำดับ
Sequential Control Structure

บทนำ

การแก้ปัญหาอย่างเป็นระบบเป็นหัวใจสำคัญของวิทยาการคำนวณ เพราะไม่ว่าปัญหาจะอยู่ในรูปของการคำนวณ การตัดสินใจ หรือการพัฒนาชิ้นงานดิจิทัล ผู้เรียนจำเป็นต้องสามารถลำดับความคิดและกำหนดขั้นตอนการดำเนินงานได้อย่างชัดเจน หนึ่งในพื้นฐานสำคัญของการเขียนโปรแกรมและการออกแบบวิธีการแก้ปัญหา คือ ความเข้าใจเกี่ยวกับโครงสร้างการควบคุมโปรแกรม ซึ่งเป็นรูปแบบการกำหนดลำดับการทำงานของคำสั่งในโปรแกรม โดยโครงสร้างพื้นฐานของการควบคุมโปรแกรมโดยทั่วไปประกอบด้วย 3 รูปแบบสำคัญ ได้แก่ โครงสร้างแบบเรียงลำดับ โครงสร้างแบบทางเลือก และโครงสร้างแบบทำซ้ำ

ในบรรดาโครงสร้างพื้นฐานทั้งสามรูปแบบ โครงสร้างการควบคุมแบบเรียงลำดับถือเป็นจุดเริ่มต้นที่สำคัญที่สุด เนื่องจากเป็นรูปแบบการทำงานที่ง่ายและตรงไปตรงมา กล่าวคือ คำสั่งจะถูกดำเนินการจากบนลงล่างหรือจากต้นไปปลายตามลำดับที่กำหนดไว้โดยไม่มีการข้าม การเลือกเงื่อนไขหรือการวนซ้ำ ความเข้าใจในโครงสร้างนี้จึงเป็นรากฐานสำคัญสำหรับการเรียนรู้การเขียนอัลกอริทึม การสร้างผังงาน และการเขียนโปรแกรมในระดับต่อไป

สำหรับผู้เรียนในสาขาครุศาสตร์ โดยเฉพาะนักศึกษาครูที่ต้องเรียนรายวิชาวิทยาการคำนวณ ความเข้าใจเรื่องโครงสร้างแบบเรียงลำดับมิได้มีคุณค่าเฉพาะในด้านการเขียนโปรแกรมเท่านั้น แต่ยังมีความสำคัญในฐานะเครื่องมือในการฝึกคิดอย่างมีขั้นตอน และเป็นพื้นฐานสำหรับการออกแบบกิจกรรมการเรียนรู้ให้แก่ผู้เรียนระดับประถมศึกษาและมัธยมศึกษา ครูที่เข้าใจโครงสร้างแบบเรียงลำดับอย่างแท้จริงจะสามารถอธิบายเรื่องอัลกอริทึมและผังงานให้ผู้เรียนเข้าใจได้ง่าย โดยเชื่อมโยงกับสถานการณ์ในชีวิตจริง เช่น การแปรงฟัน การทำอาหาร การเดินทางไปโรงเรียน หรือการจัดกิจวัตรประจำวัน

ในบทนี้จะกล่าวถึงความหมายของอัลกอริทึมและการแสดงลำดับขั้นตอนการทำงาน หลักการของโครงสร้างการควบคุมแบบเรียงลำดับ สัญลักษณ์ผังงานเบื้องต้นและการออกแบบผังงานแบบเรียงลำดับ การใช้โปรแกรม Flowgorithm เพื่อเขียนและทดสอบผังงานแบบเรียงลำดับ ตลอดจน

แนวทางการออกแบบกิจกรรมการเรียนรู้เพื่อสอนโครงสร้างแบบเรียงลำดับในระดับประถมศึกษาและมัธยมศึกษา เพื่อให้นักศึกษาครูสามารถนำความรู้ไปประยุกต์ใช้ได้ทั้งในเชิงวิชาการและเชิงวิชาชีพครู

4.1 ความหมายของอัลกอริทึมและการแสดงลำดับขั้นตอนการทำงาน

อัลกอริทึม (Algorithm) หมายถึง ลำดับขั้นตอนหรือวิธีการแก้ปัญหาที่ถูกกำหนดไว้อย่างชัดเจน เป็นระบบ และสามารถปฏิบัติตามได้เพื่อให้ได้ผลลัพธ์ตามต้องการ อัลกอริทึมจึงเป็นเสมือนแผนปฏิบัติการที่บอกว่า หากต้องการแก้ปัญหาหนึ่ง ๆ ควรทำอะไร อย่างไร และตามลำดับขั้นตอนใด

ในบริบทของวิทยาการคำนวณ อัลกอริทึมมีบทบาทสำคัญอย่างยิ่ง เพราะเป็นพื้นฐานของการเขียนผังงาน การพัฒนาโปรแกรม และการออกแบบระบบต่าง ๆ ก่อนที่ผู้เรียนจะเขียนคำสั่งด้วยภาษาโปรแกรมได้ จำเป็นต้องสามารถคิดและเรียบเรียงขั้นตอนการทำงานให้เป็นระบบเสียก่อน หากขาดความเข้าใจในอัลกอริทึม การเขียนโปรแกรมอาจกลายเป็นเพียงการจำคำสั่งโดยไม่เข้าใจโครงสร้างการทำงานอย่างแท้จริง

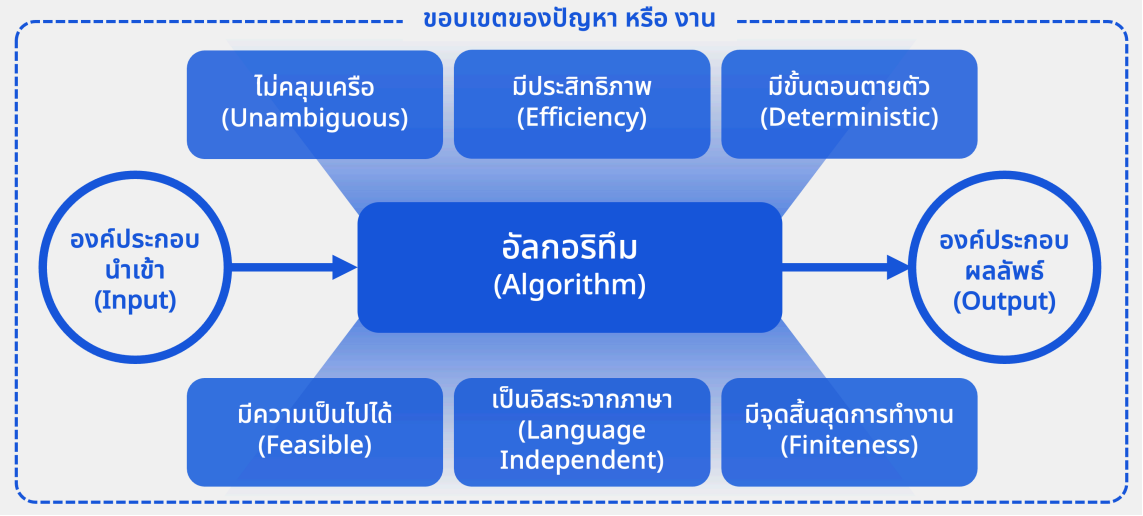
ลักษณะสำคัญของอัลกอริทึมที่ดีควรประกอบด้วยประเด็นต่อไปนี้ ประการแรก คือ มีความชัดเจน กล่าวคือ แต่ละขั้นตอนต้องระบุอย่างชัดเจนว่าให้ทำอะไร ประการที่สอง คือ มีลำดับขั้นตอนที่ถูกต้อง ไม่สับสนหรือวกวน ประการที่สาม คือ สามารถปฏิบัติได้จริง หมายความว่าผู้ปฏิบัติหรือเครื่องคอมพิวเตอร์สามารถดำเนินการตามขั้นตอนนั้นได้ และประการที่สี่ คือ มีจุดเริ่มต้นและจุดสิ้นสุดที่แน่นอน

อัลกอริทึมสามารถแสดงได้หลายรูปแบบ เช่น การบรรยายเป็นภาษาธรรมชาติ การเขียนเป็นรหัสลำลอง (Pseudocode) และการเขียนเป็นผังงาน (Flowchart) การบรรยายเป็นภาษาธรรมชาติเป็นวิธีที่เข้าใจง่าย เหมาะกับผู้เริ่มต้น เช่น การเขียนขั้นตอนการชงกาแฟว่า 1) ต้มน้ำ 2) ใส่กาแฟ 3) เติมน้ำร้อน 4) คนให้เข้ากัน 5) เสิร์ฟ เป็นต้น อย่างไรก็ตาม หากขั้นตอนมีความซับซ้อนมากขึ้น การใช้ผังงานหรือรหัสลำลองจะช่วยให้เห็นลำดับความคิดได้ชัดเจนกว่า

การแสดงลำดับขั้นตอนการทำงานเป็นทักษะสำคัญสำหรับผู้เรียน เพราะช่วยฝึกให้คิดอย่างมีระบบและสามารถถ่ายทอดแนวคิดให้ผู้อื่นเข้าใจได้ ตัวอย่างในชีวิตประจำวัน เช่น ขั้นตอนการล้างมือ การแต่งกาย การส่งงานออนไลน์ หรือการเปิดใช้งานแอปพลิเคชัน ล้วนสามารถอธิบายในลักษณะของอัลกอริทึมได้ทั้งสิ้น

ลักษณะของอัลกอริทึมที่ดีที่ประกอบด้วยคุณสมบัติ ดังนี้

1. **ไม่คลุมเครือ (Unambiguous)** อัลกอริทึมต้องไม่มีความกำหนดที่ไม่ชัดเจน และต้องมีชุดคำสั่งที่เข้าใจง่ายและไม่ทำให้เกิดความสับสนในการประมวลผล
2. **มีประสิทธิภาพ (Efficiency)** อัลกอริทึมควรมีประสิทธิภาพในการทำงาน คือต้องให้ผลลัพธ์ที่ถูกต้องและใช้เวลาน้อยที่สุดในการแก้ปัญหา
3. **มีขั้นตอนตายตัว (Deterministic)** อัลกอริทึมต้องเป็น deterministic ทำหมายความว่าจะต้องให้ผลลัพธ์ที่เหมือนกันสำหรับข้อมูลนำเข้าที่เหมือนกันทุกครั้งที่รัน
4. **มีความเป็นไปได้ (Feasible)** อัลกอริทึมต้องเป็นไปได้ในการนำไปใช้งานจริง สามารถทำงานบนเครื่องคอมพิวเตอร์หรือสภาพแวดล้อมที่เหมาะสม
5. **เป็นอิสระจากภาษา (Language Independent)** อัลกอริทึมควรไม่เกี่ยวข้องกับภาษาเฉพาะเสียเพียงอย่างใด ๆ และสามารถนำไปใช้งานในหลายภาษาโปรแกรมได้
6. **มีจุดสิ้นสุดการทำงาน (Finiteness)** อัลกอริทึมต้องสามารถสิ้นสุดการทำงานหลังจากทำขั้นตอนทุกขั้นตอนเสร็จสมบูรณ์ และไม่ต้องเข้าสู่วงจรไม่สิ้นสุด (Infinity Loop)



สำหรับนักศึกษาครู การเข้าใจอัลกอริทึมไม่ควรหยุดอยู่ที่การเขียนลำดับขั้นตอนเพียงอย่างเดียว แต่ควรสามารถเชื่อมโยงไปสู่การออกแบบกิจกรรมการเรียนรู้ เช่น ให้ผู้เรียนลองเขียนขั้นตอนจากกิจกรรมใกล้ตัว ทดลองปฏิบัติตามขั้นตอนของเพื่อน และตรวจสอบว่ามีจุดใดที่สื่อสารไม่ชัดเจน กิจกรรม

ลักษณะนี้จะช่วยให้ผู้เรียนตระหนักว่าอัลกอริทึมที่ดีต้องชัดเจน ถูกต้อง และสื่อสารได้อย่างมีประสิทธิภาพ

กล่าวโดยสรุป อัลกอริทึมเป็นรากฐานของการคิดเชิงคำนวณและการเขียนโปรแกรม เพราะช่วยให้ผู้เรียนสามารถวางแผนและลำดับขั้นตอนการแก้ปัญหาอย่างเป็นระบบ ซึ่งเป็นสมรรถนะพื้นฐานที่จำเป็นต่อการเรียนรู้วิทยาการคำนวณในระดับที่สูงขึ้น

4.2 หลักการของโครงสร้างการควบคุมแบบเรียงลำดับ (Sequential Structure)

โครงสร้างการควบคุมแบบเรียงลำดับ (Sequential Structure) คือ รูปแบบการทำงานของโปรแกรมที่คำสั่งแต่ละคำสั่งจะถูกดำเนินการตามลำดับจากต้นไปปลาย หรือจากบนลงล่างตามที่กำหนดไว้ โดยไม่มีการข้ามขั้นตอน ไม่มีการตรวจสอบเงื่อนไขเพื่อเลือกทางใดทางหนึ่ง และไม่มีการทำซ้ำหลายรอบ โครงสร้างนี้จึงเป็นรูปแบบที่ง่ายที่สุดและเป็นพื้นฐานของการเขียนโปรแกรมทุกประเภท

หลักการสำคัญของโครงสร้างแบบเรียงลำดับ คือ การให้ความสำคัญกับ “ลำดับ” ของคำสั่ง เพราะหากจัดลำดับผิด ผลลัพธ์ที่ได้อาจไม่ถูกต้อง ตัวอย่างเช่น หากต้องการคำนวณพื้นที่สี่เหลี่ยมผืนผ้าจำเป็นต้องรับค่าความกว้างและความยาวก่อน จากนั้นจึงนำมาคูณกัน และสุดท้ายจึงแสดงผลลัพธ์ หากสลับลำดับ เช่น ให้แสดงผลก่อนรับข้อมูล ผลลัพธ์ย่อมไม่สมบูรณ์หรือไม่ถูกต้อง

ในทางปฏิบัติ โครงสร้างแบบเรียงลำดับมักใช้กับงานที่มีขั้นตอนตายตัว เช่น การรับข้อมูล การคำนวณ และการแสดงผล เช่น โปรแกรมคำนวณอายุ โปรแกรมหาค่าเฉลี่ยคะแนน หรือโปรแกรมแปลงหน่วยอุณหภูมิ ขั้นตอนการทำงานมักเริ่มจากกำหนดตัวแปร รับค่าจากผู้ใช้ ประมวลผลตามสูตรที่กำหนด แล้วจึงแสดงผลออกมา

ตัวอย่างง่าย ๆ ของโครงสร้างแบบเรียงลำดับ เช่น การคำนวณผลรวมของตัวเลขสองจำนวน อาจมีลำดับขั้นตอนดังนี้

1. เริ่มต้นการทำงาน
2. รับค่าตัวเลขจำนวนที่ 1
3. รับค่าตัวเลขจำนวนที่ 2
4. นำค่าทั้งสองมาบวกกัน
5. แสดงผลลัพธ์
6. สิ้นสุดการทำงาน

จากตัวอย่างจะเห็นได้ว่า ทุกขั้นตอนต้องเกิดขึ้นตามลำดับ หากละขั้นตอนใดไปหรือสลับลำดับผลลัพธ์ของโปรแกรมก็อาจไม่เป็นไปตามที่ต้องการ ดังนั้น ผู้เรียนจึงควรฝึกมองเห็นความสัมพันธ์ระหว่างข้อมูลนำเข้า (Input) กระบวนการประมวลผล (Process) และผลลัพธ์ (Output) ซึ่งเป็นแกนสำคัญของโครงสร้างแบบเรียงลำดับ

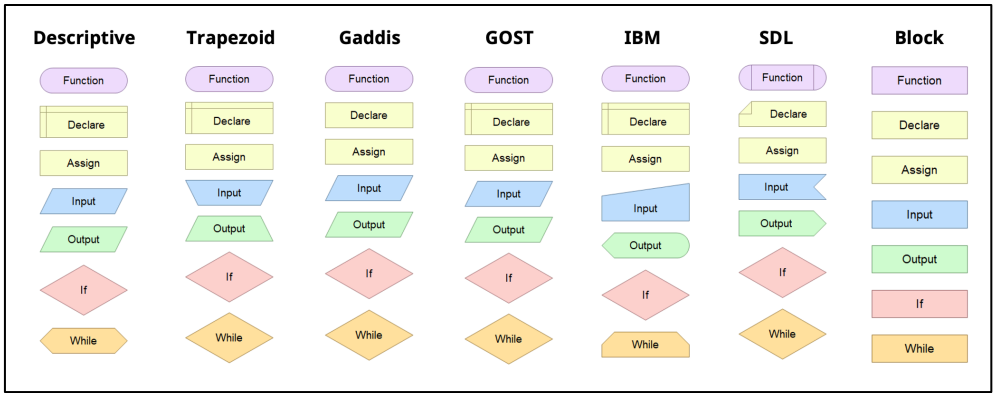
การเรียนรู้โครงสร้างแบบเรียงลำดับยังช่วยปูพื้นฐานให้ผู้เรียนเข้าใจแนวคิดการออกแบบโปรแกรมในระดับสูงขึ้น เนื่องจากโครงสร้างแบบทางเลือกและแบบทำซ้ำต่างก็อาศัยการวางลำดับขั้นตอนเป็นฐาน หากผู้เรียนยังไม่เข้าใจการจัดลำดับคำสั่งอย่างถูกต้อง การเรียนโครงสร้างที่ซับซ้อนขึ้นย่อมเกิดความสับสนได้ง่าย

สำหรับนักศึกษาครู ควรตระหนักว่าการสอนโครงสร้างแบบเรียงลำดับไม่จำเป็นต้องเริ่มจากภาษาโปรแกรมเสมอไป แต่สามารถเริ่มจากกิจกรรมที่ผู้เรียนคุ้นเคย เช่น การเรียงลำดับภาพ การจัดลำดับขั้นตอนกิจกรรม การอธิบายวิธีทำงานบ้าน หรือการให้เพื่อนทำตามคำสั่งทีละขั้น สิ่งเหล่านี้ช่วยให้ผู้เรียนเห็นภาพของการทำงานแบบเรียงลำดับอย่างเป็นรูปธรรม ก่อนที่จะเชื่อมโยงไปสู่การเขียนผังงานหรือโปรแกรม

ดังนั้น โครงสร้างการควบคุมแบบเรียงลำดับจึงเป็นรากฐานสำคัญของการเขียนโปรแกรมและการคิดเชิงระบบ เพราะช่วยให้ผู้เรียนเข้าใจว่าการทำงานของโปรแกรมต้องอาศัยคำสั่งที่ดำเนินไปอย่างเป็นลำดับและมีความสัมพันธ์กันอย่างชัดเจน

4.3 สัญลักษณ์ผังงานเบื้องต้นและการออกแบบผังงานแบบเรียงลำดับ

ผังงาน (Flowchart) คือ เครื่องมือที่ใช้แสดงลำดับขั้นตอนการทำงานของอัลกอริทึมหรือกระบวนการต่าง ๆ ด้วยสัญลักษณ์มาตรฐานและเส้นลูกศร เพื่อช่วยให้เห็นภาพรวมของการทำงานได้อย่างชัดเจน ผังงานมีประโยชน์อย่างมากต่อการเรียนวิทยาการคำนวณ เพราะช่วยให้ผู้เรียนสามารถถ่ายทอดความคิดเชิงขั้นตอนออกมาในรูปแบบที่เป็นระบบ เข้าใจง่าย และตรวจสอบได้



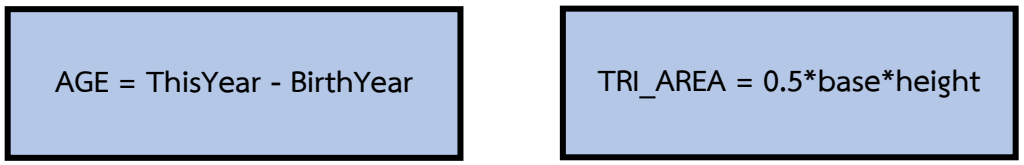
การออกแบบผังงานแบบเรียงลำดับจำเป็นต้องรู้จักสัญลักษณ์พื้นฐานที่ใช้บ่อยในงานเขียนผังงาน ได้แก่

4.3.1 สัญลักษณ์เริ่มต้นและสิ้นสุด (Terminator)

ใช้แทนจุดเริ่มต้นและจุดสิ้นสุดของการทำงาน มักมีลักษณะเป็นรูปวงรีหรือรูปมุมมน ในผังงานแบบเรียงลำดับทุกผังงานควรมีจุดเริ่มต้นและจุดสิ้นสุดอย่างชัดเจน

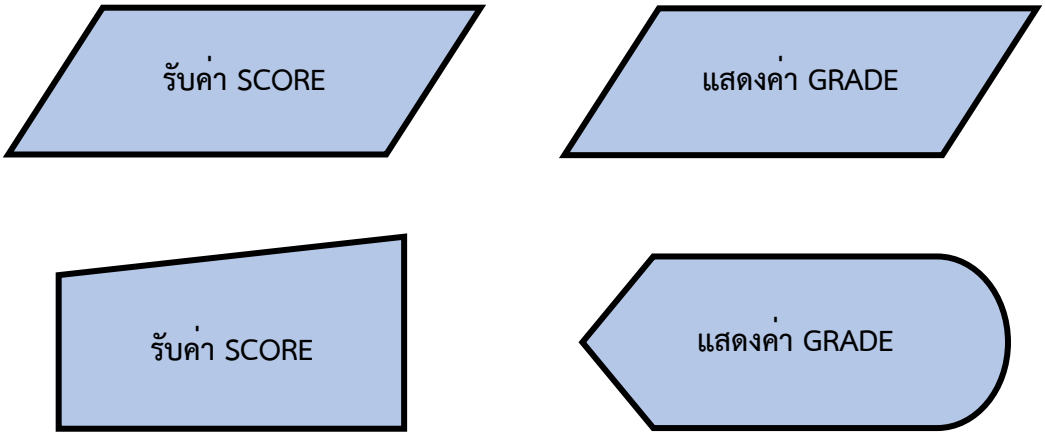
4.3.2 สัญลักษณ์กระบวนการ (Process)

ใช้แทนขั้นตอนการทำงานหรือการประมวลผล เช่น การคำนวณ การกำหนดค่าให้ตัวแปร หรือการดำเนินการใด ๆ ภายในโปรแกรม สัญลักษณ์นี้มักเป็นรูปสี่เหลี่ยมผืนผ้า



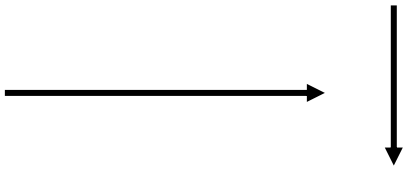
4.3.3 สัญลักษณ์ข้อมูลเข้าและข้อมูลออก (Input/Output)

ใช้แทนการรับข้อมูลจากผู้ใช้ หรือการแสดงผลลัพธ์จากโปรแกรม เช่น การรับค่าตัวเลข การแสดงผลข้อความ หรือการพิมพ์ผลลัพธ์ออกทางหน้าจอ สัญลักษณ์นี้มักเป็นรูปสี่เหลี่ยมด้านขนาน



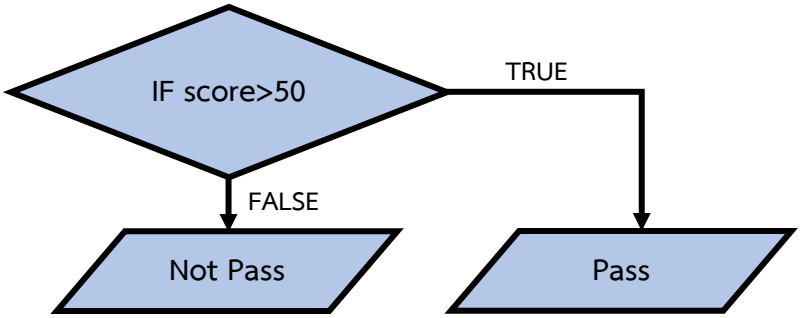
4.3.4 เส้นแสดงทิศทางการไหล (Flowline)

ใช้ลูกศรเชื่อมสัญลักษณ์ต่าง ๆ เพื่อแสดงลำดับการดำเนินงานของโปรแกรมจากขั้นตอนหนึ่งไปยังอีกขั้นตอนหนึ่ง เส้นลูกศรช่วยให้ผู้เรียนเห็นว่าขั้นตอนใดต้องทำก่อนหรือหลัง



4.3.5 สัญลักษณ์การตัดสินใจ (Decision)

ใช้แทนการตัดสินใจตามเงื่อนไข เพื่อเลือกเส้นทางที่จะทำงานต่อไป เช่น ถ้าคะแนนตั้งแต่ 80 ขึ้นไปได้เกรด A หรือ ถ้าคะแนนมากกว่า 50 แสดงว่า ผ่าน และถ้าต่ำกว่า 50 แสดงผลว่า ตก



สำหรับผังงานแบบเรียงลำดับ การออกแบบมีหลักการไม่ซับซ้อน กล่าวคือ ควรวางขั้นตอนจากบนลงล่างอย่างชัดเจน แต่ละคำสั่งควรเชื่อมต่อกันตามลำดับ ไม่มีทางแยกและไม่มีการวนกลับ ข้อความในแต่ละสัญลักษณ์ควรกระชับ ชัดเจน และสื่อความหมายตรงกับการทำงานจริง

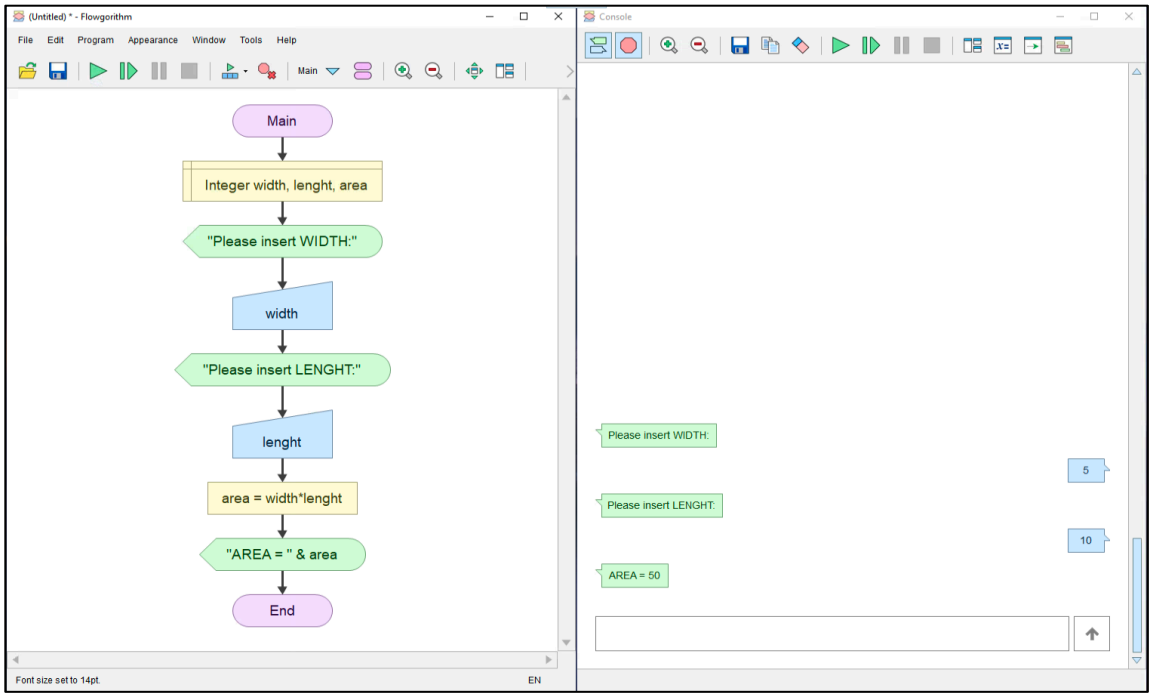
ตัวอย่างการออกแบบผังงานแบบเรียงลำดับสำหรับการคำนวณพื้นที่สี่เหลี่ยมผืนผ้า อาจประกอบด้วยขั้นตอนดังนี้

1. เริ่มต้น
2. รับค่าความกว้าง (width)
3. รับค่าความยาว (length)
4. คำนวณพื้นที่ = ความกว้าง × ความยาว

   area = width * length

5. แสดงผลพื้นที่ (area)
6. สิ้นสุด

เมื่อนำขั้นตอนดังกล่าวไปเขียนเป็นผังงาน ผู้เรียนจะเห็นภาพการทำงานอย่างชัดเจนว่าโปรแกรมรับข้อมูลก่อน แล้วจึงประมวลผล และสุดท้ายจึงแสดงผลลัพธ์ ซึ่งสอดคล้องกับหลักการของโครงสร้างแบบเรียงลำดับ



การฝึกออกแบบผังงานมีประโยชน์ต่อผู้เรียนหลายด้าน ได้แก่ ช่วยฝึกการคิดอย่างมีลำดับ ช่วยลดความผิดพลาดก่อนลงมือเขียนโปรแกรมจริง และช่วยให้ครูสามารถตรวจสอบความเข้าใจของผู้เรียนได้ง่ายขึ้น หากผังงานมีลำดับผิดพลาดหรือขาดขั้นตอนบางส่วน ครูจะสามารถให้ข้อเสนอแนะได้ตั้งแต่ต้น สำหรับนักศึกษาครู การใช้ผังงานเป็นเครื่องมือในการจัดการเรียนรู้มีความเหมาะสมอย่างยิ่ง เพราะช่วยให้ผู้เรียนในระดับประถมศึกษาและมัธยมศึกษาเข้าใจหลักการทำงานของโปรแกรมโดยไม่ต้องเริ่มจากโค้ดซึ่งอาจมีความซับซ้อนเกินไป ครูสามารถใช้ผังงานร่วมกับภาพ กิจกรรมการ์ดลำดับขั้น หรือสถานการณ์ใกล้ตัว เพื่อสร้างความเข้าใจเรื่องอัลกอริทึมอย่างเป็นรูปธรรม

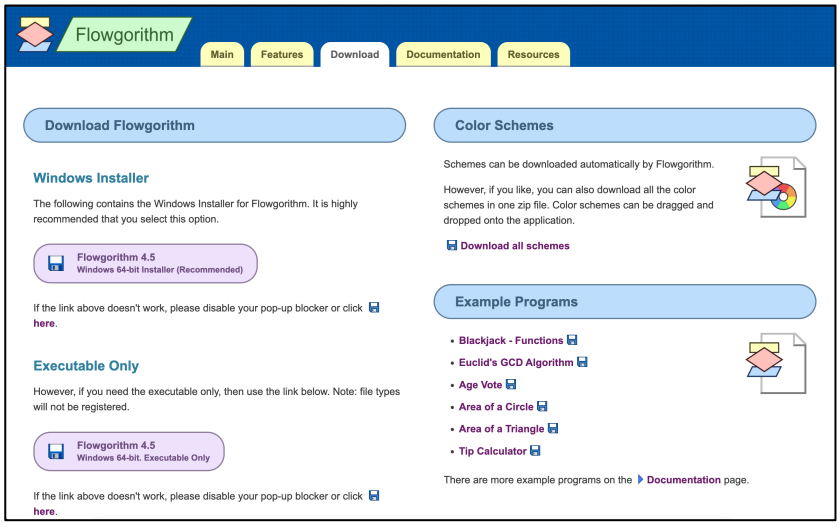
กล่าวโดยสรุป ผังงานเป็นเครื่องมือสำคัญในการแสดงลำดับขั้นตอนการทำงานของอัลกอริทึม โดยเฉพาะโครงสร้างแบบเรียงลำดับ ซึ่งช่วยให้ผู้เรียนสามารถเห็นภาพการทำงานของโปรแกรมอย่างชัดเจน เป็นระบบ และพร้อมต่อยอดไปสู่การเขียนโปรแกรมจริงในลำดับถัดไป

4.4 การใช้โปรแกรม Flowgorithm ในการเขียนและทดสอบผังงานแบบเรียงลำดับ

Flowgorithm เป็นโปรแกรมที่ออกแบบมาเพื่อช่วยให้ผู้เรียนสามารถสร้างผังงานและทดสอบการทำงานของอัลกอริทึมได้อย่างสะดวก โดยเฉพาะสำหรับผู้เริ่มต้นเรียนเขียนโปรแกรม โปรแกรมนี้มีจุดเด่นคือ ใช้การออกแบบผังงานแทนการพิมพ์โค้ดโดยตรง ทำให้ผู้เรียนมองเห็นลำดับขั้นตอนการทำงานได้อย่างชัดเจน และสามารถเข้าใจความสัมพันธ์ระหว่างการรับข้อมูล การประมวลผล และการแสดงผลได้ดียิ่งขึ้น

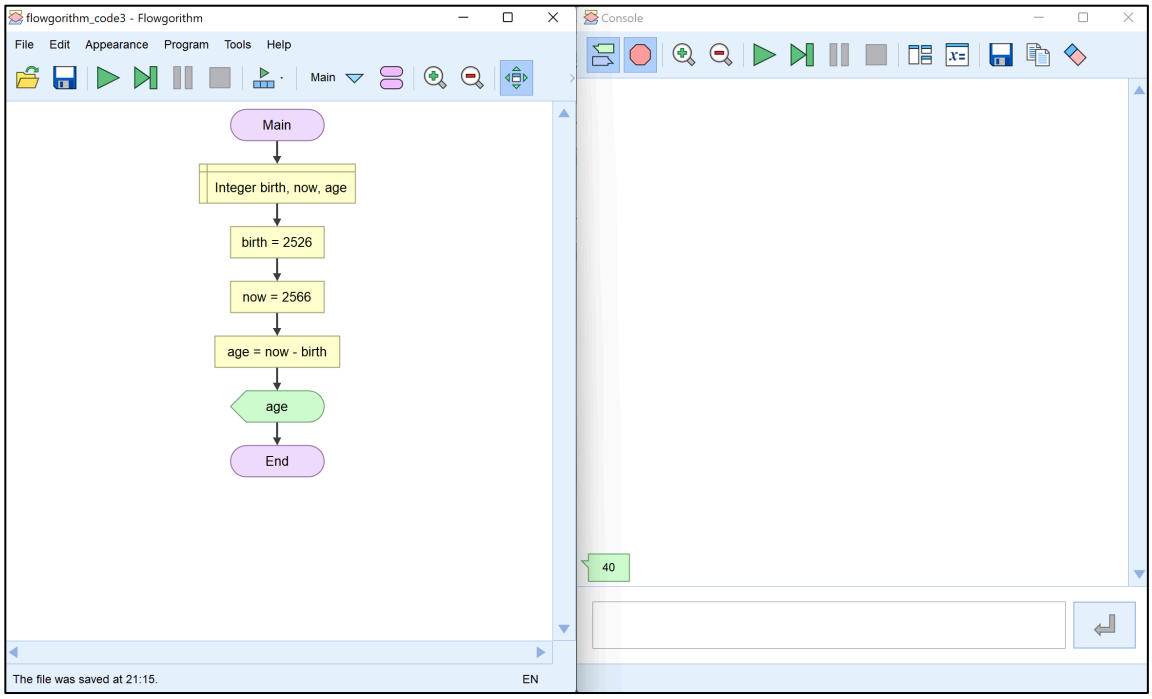
การใช้ Flowgorithm มีความเหมาะสมอย่างมากในรายวิชาวิทยาการคำนวณสำหรับนักศึกษาครู เพราะเป็นเครื่องมือที่ช่วยเชื่อมโยงระหว่างแนวคิดอัลกอริทึมกับการปฏิบัติจริง ผู้เรียนสามารถทดลองสร้างผังงาน แก้ไขขั้นตอน ตรวจสอบผลลัพธ์ และเรียนรู้จากข้อผิดพลาดได้ทันที จึงเป็นสื่อการเรียนรู้ที่ส่งเสริมการคิดเชิงระบบและการเรียนรู้แบบลงมือปฏิบัติ

การใช้งานโปรแกรม Flowgorithm สามารถดาวน์โหลดโปรแกรมเพื่อติดตั้งได้ที่เว็บไซต์ http://www.flowgorithm.org/download/index.html โดยควรเลือกรุ่นของโปรแกรมที่เป็นรุ่นล่าสุด



4.4.1 ส่วนประกอบเบื้องต้นของโปรแกรม Flowgorithm

เมื่อเปิดโปรแกรม Flowgorithm ผู้เรียนจะพบพื้นที่สำหรับวางผังงาน แถบเครื่องมือสำหรับเพิ่มสัญลักษณ์ เช่น การรับข้อมูล การแสดงผล และการประมวลผล รวมถึงคำสั่งสำหรับรันหรือทดสอบผังงาน โปรแกรมจะแสดงจุดเริ่มต้นและสิ้นสุดไว้ให้เบื้องต้น เพื่อให้ผู้เรียนสามารถเพิ่มเติมขั้นตอนการทำงานระหว่างกลางได้ตามต้องการ

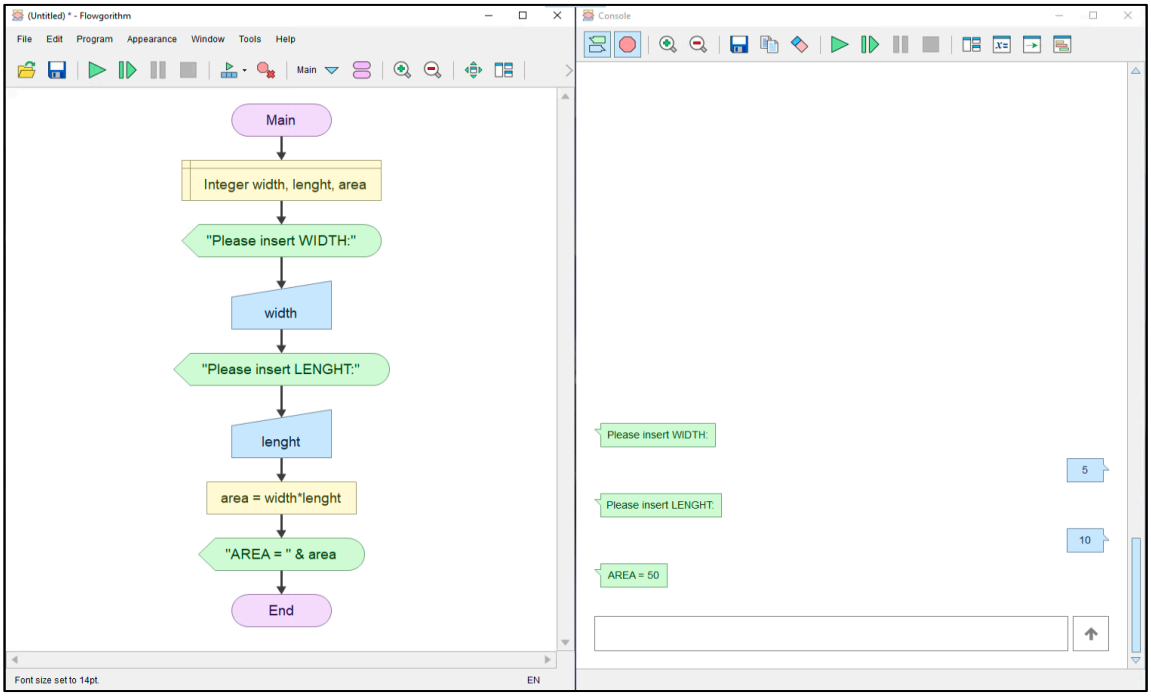


4.4.2 ขั้นตอนการสร้างผังงานแบบเรียงลำดับ

การสร้างผังงานแบบเรียงลำดับใน Flowgorithm โดยทั่วไปประกอบด้วยขั้นตอนดังนี้

1. กำหนดปัญหาหรือโจทย์ที่ต้องการแก้
2. วิเคราะห์ข้อมูลนำเข้า กระบวนการ และผลลัพธ์
3. เพิ่มสัญลักษณ์รับข้อมูล หากโจทย์ต้องรับค่าจากผู้ใช้
4. เพิ่มสัญลักษณ์กระบวนการ เพื่อคำนวณหรือกำหนดค่าตัวแปร
5. เพิ่มสัญลักษณ์แสดงผล เพื่อสื่อสารผลลัพธ์ออกทางหน้าจอ
6. ตรวจสอบความถูกต้องของลำดับขั้นตอนก่อนทดสอบการทำงาน

ตัวอย่างเช่น หากต้องการเขียนผังงานเพื่อคำนวณพื้นที่สี่เหลี่ยมผืนผ้า ผู้เรียนอาจกำหนดตัวแปร width, length และ area จากนั้นสร้างผังงานดังนี้ รับค่าความกว้าง รับค่าความยาว คำนวณ area = width × length และแสดงค่า area



4.4.3 การทดสอบหรือ Run ผังงาน

จุดเด่นสำคัญของ Flowgorithm คือ ผู้เรียนสามารถทดสอบการทำงานของผังงานได้ทันที โดยโปรแกรมจะทำงานตามลำดับขั้นตอนที่ออกแบบไว้ หากมีการรับข้อมูล โปรแกรมจะเปิดช่องให้ผู้ใช้ป้อนค่า และเมื่อถึงขั้นตอนการแสดงผล ก็จะแสดงคำตอบตามกระบวนการที่กำหนดไว้ การทดสอบเช่นนี้ช่วยให้ผู้เรียนเห็นว่าผังงานที่ตนเขียนสามารถนำไปสู่ผลลัพธ์จริงได้หรือไม่

4.4.4 การตรวจสอบข้อผิดพลาดและการปรับปรุงผังงาน

เมื่อผู้เรียน Run ผังงาน อาจพบข้อผิดพลาดหลายลักษณะ เช่น ลำดับขั้นตอนไม่ถูกต้อง ลืมกำหนดตัวแปร ใช้สูตรคำนวณผิด หรือแสดงผลผิดจากที่คาดไว้ การใช้ Flowgorithm ช่วยให้ผู้เรียนตรวจสอบข้อผิดพลาดได้ง่ายขึ้น เพราะสามารถมองย้อนกลับไปยังผังงานทีละขั้นตอน และปรับปรุงแก้ไขได้อย่างเป็นระบบ

4.4.5 ประโยชน์ของ Flowgorithm ต่อการจัดการเรียนรู้

Flowgorithm มีประโยชน์หลายประการต่อการเรียนการสอน ได้แก่ ช่วยลดความซับซ้อนของการเริ่มต้นเรียนเขียนโปรแกรม ช่วยให้ผู้เรียนเห็นภาพกระบวนการทำงานอย่างชัดเจน ช่วยส่งเสริมการเรียนรู้จากการทดลองและการตรวจสอบผล และช่วยเตรียมความพร้อมก่อนเข้าสู่การเขียนโค้ดด้วยภาษาโปรแกรมจริง นอกจากนี้ ยังเป็นเครื่องมือที่เหมาะสมสำหรับครูในการอธิบายอัลกอริทึมให้ผู้เรียนเข้าใจได้ง่ายขึ้น

สำหรับนักศึกษาครู การเรียนรู้การใช้ Flowgorithm ควรครอบคลุมทั้งการใช้งานเครื่องมือและการพิจารณาว่าจะนำไปออกแบบกิจกรรมการเรียนรู้สำหรับผู้เรียนอย่างไร เช่น การให้ผู้เรียนช่วยกันวิเคราะห์โจทย์ก่อนสร้างผังงาน การให้คาดการณ์ผลลัพธ์ก่อน Run หรือการให้อภิปรายข้อผิดพลาดที่เกิดขึ้นจากผังงานของตนและเพื่อน กิจกรรมเช่นนี้จะช่วยพัฒนาทั้งความรู้ด้านวิทยาการคำนวณและทักษะการอธิบายของผู้เรียน

กล่าวโดยสรุป Flowgorithm เป็นเครื่องมือที่มีประสิทธิภาพสำหรับการเรียนรู้และการสอนโครงสร้างแบบเรียงลำดับ เพราะช่วยให้ผู้เรียนสามารถแปลงอัลกอริทึมเป็นผังงานที่มองเห็นได้ ทดสอบผลลัพธ์ได้ และเรียนรู้จากข้อผิดพลาดได้อย่างเป็นรูปธรรม

4.5 การออกแบบกิจกรรมการเรียนรู้เพื่อสอนโครงสร้างแบบเรียงลำดับในระดับประถมศึกษาและมัธยมศึกษา

การจัดการเรียนรู้เรื่องโครงสร้างการควบคุมแบบเรียงลำดับสำหรับผู้เรียนระดับประถมศึกษาและมัธยมศึกษา ควรคำนึงถึงพัฒนาการและประสบการณ์เดิมของผู้เรียนเป็นสำคัญ แม้เนื้อหาเรื่องอัลกอริทึมและผังงานจะเป็นพื้นฐานทางวิทยาการคำนวณ แต่หากครูนำเสนอในรูปแบบที่เป็นนามธรรมมากเกินไป ผู้เรียนอาจรู้สึกว่าเนื้อหายากและห่างไกลจากชีวิตจริง ดังนั้น การออกแบบกิจกรรมการเรียนรู้จึงควรเริ่มจากสิ่งที่เป็นรูปธรรม ใกล้ตัว และเปิดโอกาสให้ผู้เรียนได้ลงมือปฏิบัติจริง

4.5.1 หลักการออกแบบกิจกรรม

ครูควรเริ่มต้นจากสถานการณ์หรือกิจกรรมในชีวิตประจำวัน เช่น การล้างมือ การทำแซนด์วิช การแต่งกายไปโรงเรียน หรือการปลูกต้นไม้ เพื่อให้ผู้เรียนเห็นว่ากิจกรรมต่าง ๆ ล้วนมีลำดับขั้นตอน หากสลับขั้นตอนอาจทำให้ผลลัพธ์ผิดพลาดได้ แนวทางนี้ช่วยให้ผู้เรียนเข้าใจแนวคิดของโครงสร้างแบบเรียงลำดับก่อนที่จะเชื่อมโยงไปสู่ผังงานและโปรแกรม

4.5.2 แนวทางสำหรับระดับประถมศึกษา

ในระดับประถมศึกษา ควรใช้กิจกรรมที่เน้นภาพ การลงมือทำ และความสนุกสนาน เช่น

1. กิจกรรมเรียงภาพลำดับขั้นตอน
2. การเขียนขั้นตอนจากกิจวัตรประจำวัน
3. เกมสั่งเพื่อนทำตามคำสั่งทีละขั้น
4. การใช้บัตรคำสั่งแทนลำดับกิจกรรม
5. การวาดผังงานอย่างง่ายจากสถานการณ์ใกล้ตัว

กิจกรรมเหล่านี้ช่วยให้ผู้เรียนเข้าใจว่า การทำงานอย่างเป็นลำดับเป็นพื้นฐานของทั้งชีวิตประจำวันและการเขียนโปรแกรม

4.5.3 แนวทางสำหรับระดับมัธยมศึกษา

ในระดับมัธยมศึกษา ผู้เรียนสามารถเรียนรู้ในระดับที่เป็นนามธรรมมากขึ้น ครูอาจออกแบบกิจกรรม เช่น

1. วิเคราะห์โจทย์แล้วระบุ Input, Process, Output
2. เขียนอัลกอริทึมเป็นภาษาธรรมชาติหรือรหัสลำลอง
3. เขียนผังงานแบบเรียงลำดับจากโจทย์คำนวณอย่างง่าย
4. ใช้โปรแกรม Flowgorithm เพื่อสร้างและ Run ผังงาน
5. ให้ผู้เรียนปรับปรุงผังงานจากกรณีตัวอย่างที่ผิดพลาด

กิจกรรมระดับนี้ควรมุ่งให้ผู้เรียนเข้าใจความสัมพันธ์ระหว่างการคิด การออกแบบผังงาน และผลลัพธ์ของโปรแกรมอย่างชัดเจน

4.5.4 การบูรณาการกับการจัดทำแผนการเรียนรู้

สำหรับนักศึกษาครู การออกแบบกิจกรรมไม่ควรหยุดเพียงการเลือกกิจกรรม แต่ควรเชื่อมโยงกับการจัดทำแผนการจัดการเรียนรู้ เช่น การกำหนดจุดประสงค์การเรียนรู้ การเลือกสื่อและอุปกรณ์ การกำหนดขั้นตอนกิจกรรม การวัดและประเมินผล และการสะท้อนผลการเรียนรู้ของผู้เรียน ครูควรออกแบบให้ผู้เรียนได้ทั้งความรู้ ทักษะ และเจตคติที่ดีต่อการเรียนวิทยาการคำนวณ

4.5.5 การวัดและประเมินผล

การประเมินผลการเรียนรู้เรื่องโครงสร้างแบบเรียงลำดับควรครอบคลุมทั้งความเข้าใจและการปฏิบัติ เช่น การตรวจลำดับขั้นตอนที่ผู้เรียนเขียน การประเมินผังงานที่ออกแบบ การสังเกตการทำกิจกรรม การตรวจผลลัพธ์จากการ Run ผังงาน และการให้ผู้เรียนอธิบายเหตุผลของลำดับขั้นตอนที่ตน

กำหนด การประเมินลักษณะนี้จะช่วยให้ครูเห็นพัฒนาการของผู้เรียนได้ชัดเจนกว่าการดูคำตอบสุดท้ายเพียงอย่างเดียว

4.5.6 บทบาทของครู

ครูควรทำหน้าที่เป็นผู้อำนวยความสะดวกในการเรียนรู้ หรือที่เรียกว่า Facilitator โดยใช้คำถามกระตุ้นการคิด เช่น “ถ้าสลับขั้นตอนนี้จะเกิดอะไรขึ้น” หรือ “เหตุใดต้องรับข้อมูลก่อนคำนวณ” คำถามลักษณะนี้ช่วยให้ผู้เรียนไม่ได้เพียงจำลำดับขั้นตอน แต่เข้าใจเหตุผลเบื้องหลังการจัดลำดับของโปรแกรมด้วย

โดยสรุป การสอนโครงสร้างแบบเรียงลำดับควรมุ่งให้ผู้เรียนเข้าใจผ่านกิจกรรมที่เป็นรูปธรรมและค่อย ๆ เชื่อมไปสู่การใช้ผังงานและโปรแกรม Flowgorithm ทั้งนี้ นักศึกษาครูควรได้รับการฝึกฝนให้สามารถออกแบบกิจกรรมการเรียนรู้ที่เหมาะสมกับระดับชั้นของผู้เรียน เพื่อให้การเรียนวิทยาการคำนวณเป็นเรื่องเข้าใจได้ สนุก และมีความหมายต่อชีวิต

บทสรุปประจำบทที่ 4

โครงสร้างการควบคุมโปรแกรมแบบเรียงลำดับเป็นพื้นฐานสำคัญของการเรียนรู้วิทยาการคำนวณและการเขียนโปรแกรม เพราะเป็นรูปแบบการทำงานที่คำสั่งต่าง ๆ ถูกดำเนินไปตามลำดับอย่างชัดเจนจากต้นจนจบ การทำความเข้าใจอัลกอริทึมและการแสดงลำดับขั้นตอนการทำงานจึงเป็นจุดเริ่มต้นที่สำคัญสำหรับผู้เรียนในการพัฒนาทักษะการคิดเชิงระบบและการแก้ปัญหาอย่างมีขั้นตอน

ผังงานเป็นเครื่องมือที่ช่วยให้ผู้เรียนมองเห็นโครงสร้างการทำงานของอัลกอริทึมได้อย่างเป็นรูปธรรม โดยเฉพาะเมื่อใช้ร่วมกับโปรแกรม Flowgorithm ซึ่งช่วยให้สามารถสร้างและทดสอบผังงานได้ทันที ผู้เรียนจึงสามารถเชื่อมโยงระหว่างแนวคิด อัลกอริทึม ผังงาน และผลลัพธ์ของโปรแกรมได้อย่างชัดเจน นอกจากนี้ สำหรับนักศึกษาครู การเรียนรู้เรื่องโครงสร้างแบบเรียงลำดับยังควรเชื่อมโยงไปสู่การออกแบบกิจกรรมการเรียนรู้ที่เหมาะสมกับผู้เรียนในระดับประถมศึกษาและมัธยมศึกษา โดยเน้นการใช้สถานการณ์ใกล้ตัว การลงมือปฏิบัติ และการวัดผลที่ครอบคลุมทั้งความรู้และกระบวนการคิด

คำถามท้ายบท

1. จงอธิบายความหมายของอัลกอริทึม และเหตุใดอัลกอริทึมจึงมีความสำคัญต่อการเขียนโปรแกรม
2. อัลกอริทึมที่ดีควรมีลักษณะสำคัญอย่างไรบ้าง
3. จงอธิบายหลักการของโครงสร้างการควบคุมแบบเรียงลำดับ พร้อมยกตัวอย่าง
4. เพราะเหตุใดลำดับของคำสั่งจึงมีความสำคัญต่อการทำงานของโปรแกรมแบบเรียงลำดับ
5. สัญลักษณ์พื้นฐานของผังงานมีอะไรบ้าง และแต่ละสัญลักษณ์มีหน้าที่อย่างไร
6. จงอธิบายประโยชน์ของการใช้ผังงานก่อนการเขียนโปรแกรม
7. โปรแกรม Flowgorithm มีประโยชน์อย่างไรต่อการเรียนรู้เรื่องอัลกอริทึมและผังงาน
8. หากให้ท่านออกแบบผังงานเพื่อคำนวณผลรวมของตัวเลข 2 จำนวน ท่านจะกำหนดลำดับขั้นตอนอย่างไร
9. หากท่านเป็นครูระดับประถมศึกษา ท่านจะจัดกิจกรรมอย่างไรเพื่อให้ผู้เรียนเข้าใจโครงสร้างแบบเรียงลำดับ
10. หากท่านเป็นครูระดับมัธยมศึกษา ท่านจะใช้ Flowgorithm เพื่อส่งเสริมการเรียนรู้ของผู้เรียนอย่างไร

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.

Rapaport, W. J. (2005). Algorithmic thinking, computational thinking, and the humanities. *Journal of Computing Sciences in Colleges, 20*(5), 1–9.

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.

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

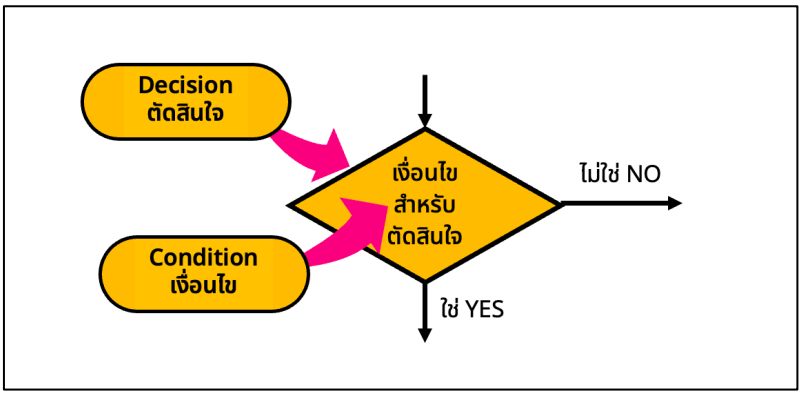
สำหรับนักศึกษาครู ความเข้าใจเรื่องโครงสร้างแบบทางเลือกมีความสำคัญอย่างมาก เพราะช่วยพัฒนาทักษะการคิดอย่างมีเหตุผล การตัดสินใจอย่างเป็นระบบ และการเชื่อมโยงเงื่อนไขกับผลลัพธ์ ซึ่งเป็นพื้นฐานทั้งของการเขียนโปรแกรมและของการจัดการเรียนรู้ในชีวิตจริง อีกทั้งครูยังจำเป็นต้องสามารถออกแบบกิจกรรมการเรียนรู้ที่ช่วยให้ผู้เรียนระดับประถมศึกษาและมัธยมศึกษาเข้าใจเรื่องเงื่อนไขและการตัดสินใจได้อย่างเป็นรูปธรรม 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)

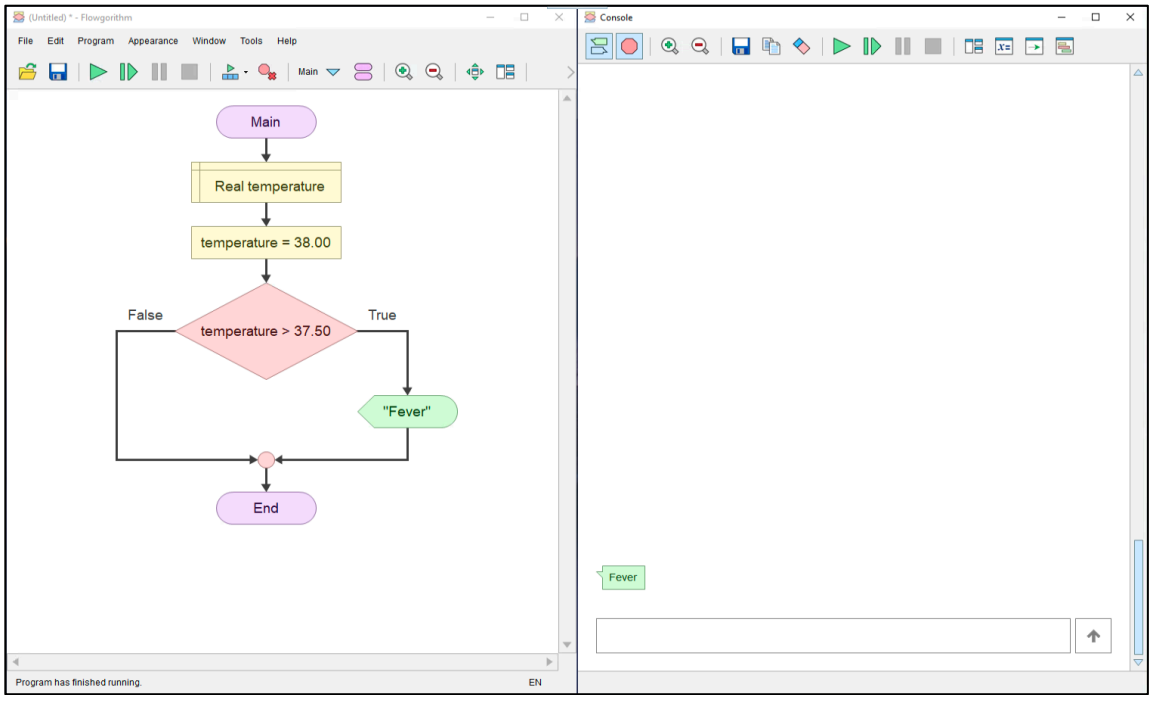
โครงสร้างการควบคุมแบบทางเลือก (Selection Structure) คือ รูปแบบการทำงานของโปรแกรมที่มีการตรวจสอบเงื่อนไขก่อน แล้วจึงเลือกดำเนินการตามผลของเงื่อนไขนั้น หากเงื่อนไขเป็นจริง โปรแกรมจะทำคำสั่งชุดหนึ่ง แต่หากเงื่อนไขเป็นเท็จ อาจทำอีกชุดคำสั่งหนึ่งหรือไม่ทำคำสั่งใดเลย ทั้งนี้ขึ้นอยู่กับรูปแบบของโครงสร้างที่ออกแบบไว้

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)

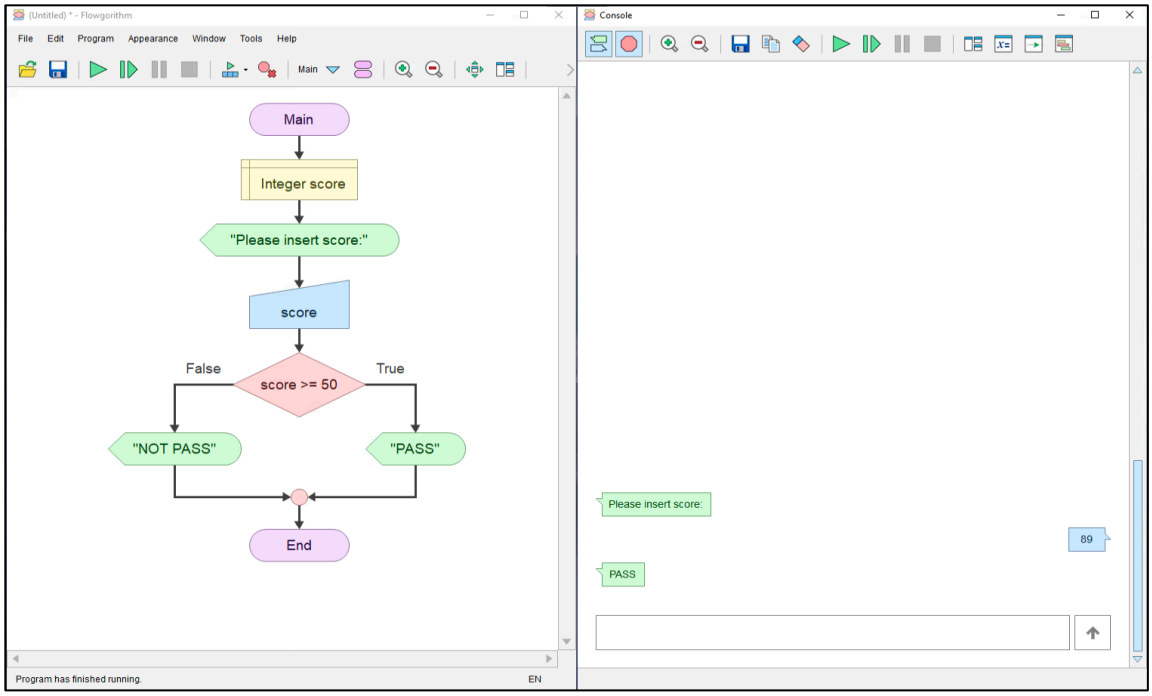
เป็นรูปแบบที่โปรแกรมตรวจสอบเงื่อนไข และจะดำเนินการก็ต่อเมื่อเงื่อนไขเป็นจริง แต่หากเงื่อนไขเป็นเท็จ โปรแกรมจะข้ามคำสั่งนั้นไปและดำเนินการต่อ เช่น หากอุณหภูมิสูงกว่า 37.5 ให้แสดงข้อความว่า “มีไข้” แต่หากไม่ถึงเกณฑ์ ก็ไม่ต้องแสดงข้อความดังกล่าว

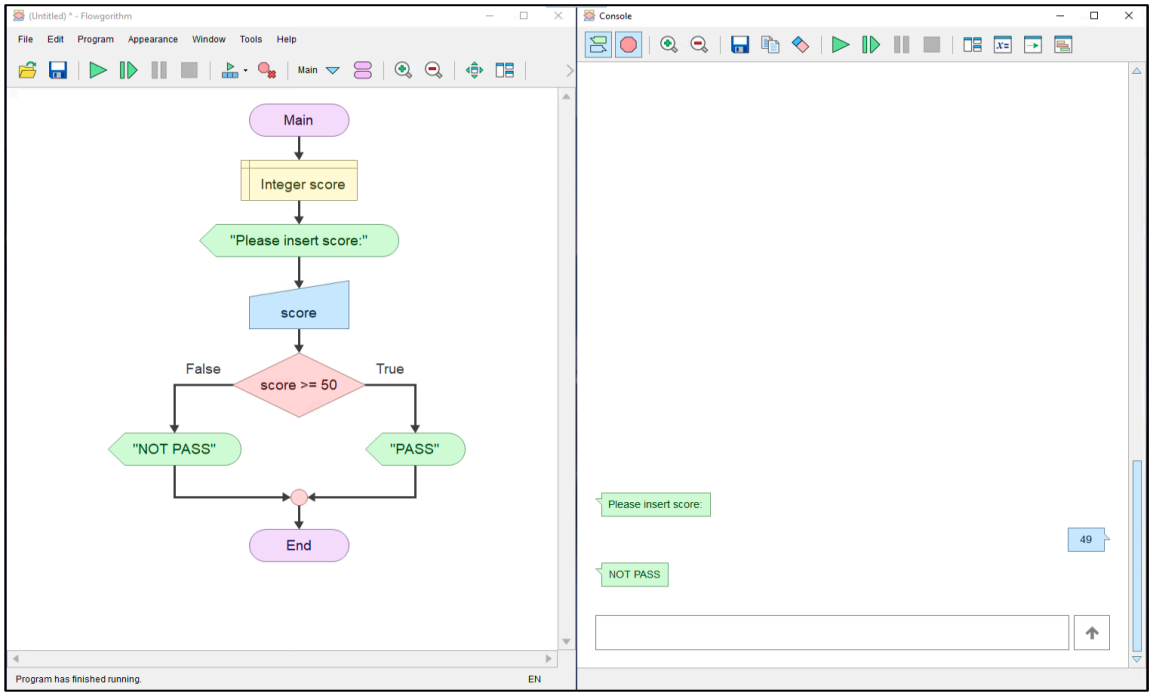


5.2.2 โครงสร้างทางเลือกแบบสองทาง (Double Selection)

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

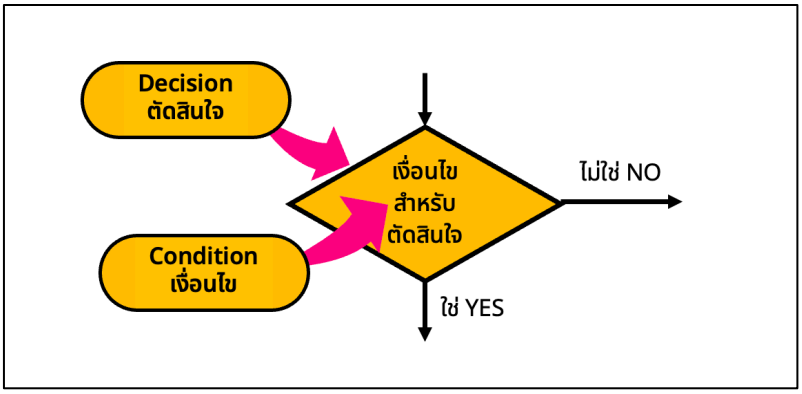
สำหรับนักศึกษาครู การสอนโครงสร้างแบบทางเลือกควรเริ่มจากสถานการณ์ใกล้ตัวก่อน เช่น การตัดสินใจพกร่มจากสภาพอากาศ การตัดสินใจใส่เสื้อกันหนาวจากอุณหภูมิ หรือการประเมินผลสอบจากคะแนน จากนั้นจึงเชื่อมโยงเข้าสู่การเขียนเงื่อนไขและการใช้ผังงาน วิธีการเช่นนี้จะช่วยให้ผู้เรียนเห็นภาพว่าโครงสร้างแบบทางเลือกมิใช่เรื่องซับซ้อน 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

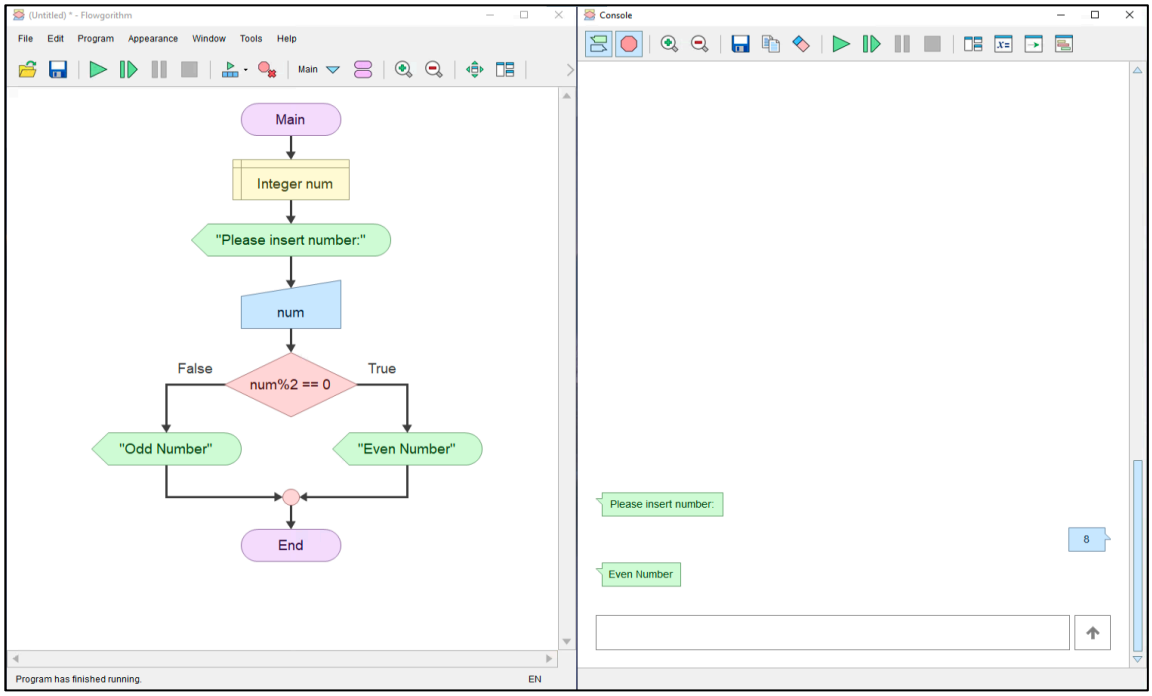
5.4 การใช้โปรแกรม Flowgorithm ในการเขียนและทดสอบผังงานแบบทางเลือก

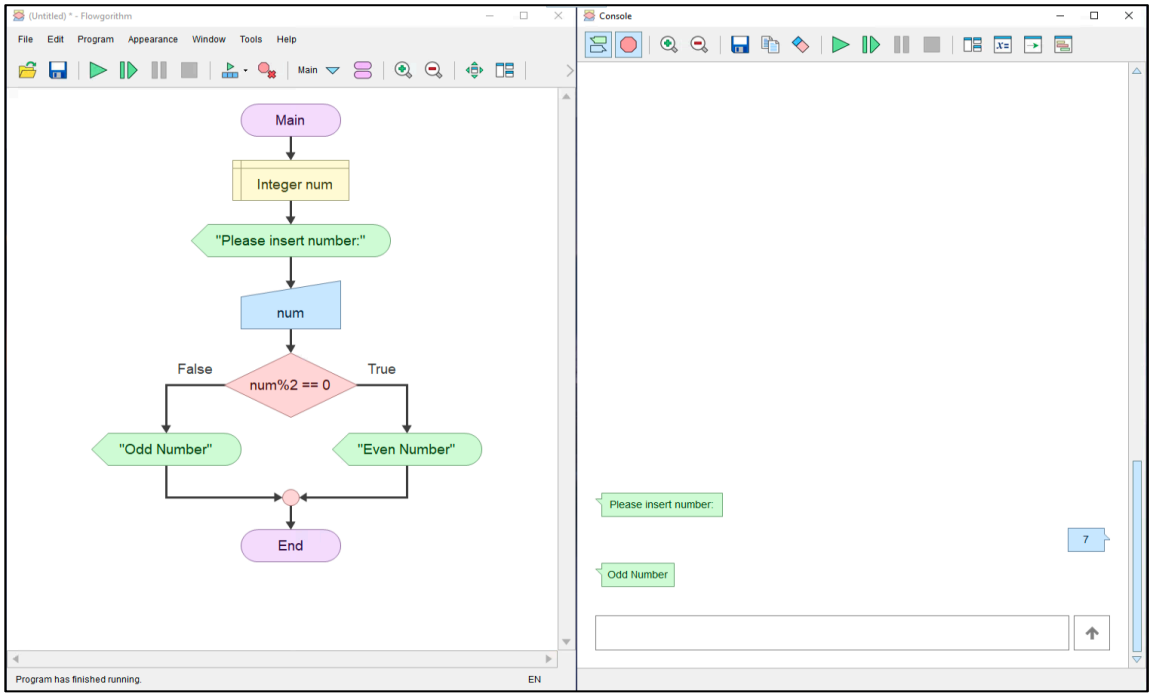
Flowgorithm เป็นโปรแกรมที่เหมาะสมอย่างยิ่งสำหรับการเรียนรู้โครงสร้างแบบทางเลือก เพราะช่วยให้ผู้เรียนสร้างผังงาน ตรวจสอบเงื่อนไข และทดสอบผลลัพธ์ได้อย่างเป็นรูปธรรม โดยไม่จำเป็นต้องเริ่มจากการเขียนโค้ดที่ซับซ้อน โปรแกรมนี้จึงช่วยลดภาระในการจดจำไวยากรณ์ของภาษาโปรแกรม และเปิดโอกาสให้ผู้เรียนมุ่งเน้นที่กระบวนการคิดเชิงอัลกอริทึมและตรรกะของการตัดสินใจ

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.

บทที่ 6
โครงสร้างการควบคุมโปรแกรมแบบทำซ้ำ
Iteration Control Structure

บทนำ

ในการแก้ปัญหาทางวิทยาการคำนวณ มีหลายสถานการณ์ที่คำสั่งบางอย่างจำเป็นต้องถูกดำเนินซ้ำมากกว่าหนึ่งครั้ง เช่น การแสดงตัวเลขตั้งแต่ 1 ถึง 10 การคำนวณผลรวมของคะแนนหลายค่า การตรวจสอบข้อมูลทีละรายการ หรือการทำงานซ้ำจนกว่าจะเป็นไปตามเงื่อนไขที่กำหนด หากเขียนคำสั่งเหล่านี้แบบเรียงลำดับทีละบรรทัดทั้งหมด จะทำให้โปรแกรมยาว ซ้ำซ้อน และยากต่อการจัดการ ดังนั้นจึงจำเป็นต้องมีโครงสร้างการควบคุมที่ช่วยให้โปรแกรมสามารถดำเนินคำสั่งเดิมซ้ำได้อย่างเป็นระบบ ซึ่งก็คือ “โครงสร้างการควบคุมแบบทำซ้ำ”

โครงสร้างการควบคุมแบบทำซ้ำ (Iteration Structure หรือ Loop Structure) เป็นหนึ่งในโครงสร้างพื้นฐานสำคัญของการเขียนโปรแกรม ร่วมกับโครงสร้างแบบเรียงลำดับและโครงสร้างแบบทางเลือก โครงสร้างนี้ช่วยให้โปรแกรมทำงานซ้ำภายใต้จำนวนรอบหรือเงื่อนไขที่กำหนดไว้อย่างชัดเจน ส่งผลให้การเขียนโปรแกรมมีประสิทธิภาพมากขึ้น ลดความซ้ำซ้อนของคำสั่ง และช่วยให้การแก้ปัญหาที่มีลักษณะเกิดขึ้นซ้ำทำได้อย่างเหมาะสม

สำหรับนักศึกษาครู ความเข้าใจเรื่องการทำซ้ำมีความสำคัญทั้งในเชิงเนื้อหาวิทยาการคำนวณและในเชิงวิธีสอน เพราะการทำซ้ำมิใช่เพียงกลไกของโปรแกรมเท่านั้น แต่ยังเป็นวิธีคิดที่เกี่ยวข้องกับการมองเห็นรูปแบบและความเป็นระบบในกิจกรรมต่าง ๆ ของชีวิตประจำวัน เช่น การเดินขึ้นบันไดทีละขั้น การกวาดห้องจนสะอาด การตอบคำถามหลายข้อ หรือการทบทวนบทเรียนจนเข้าใจ ครูที่เข้าใจหลักการของการทำซ้ำอย่างชัดเจนจะสามารถออกแบบกิจกรรมที่ช่วยให้ผู้เรียนเห็นภาพของการทำงานแบบวนซ้ำได้อย่างเป็นรูปธรรม และสามารถเชื่อมโยงจากสถานการณ์ใกล้ตัวไปสู่ผังงานและการเขียนโปรแกรมได้อย่างมีประสิทธิภาพ

บทนี้มุ่งอธิบายความหมายและลักษณะของโครงสร้างการควบคุมแบบทำซ้ำ ประเภทของการทำซ้ำและการประยุกต์ใช้ในการแก้ปัญหา การออกแบบผังงานแบบทำซ้ำ การใช้โปรแกรม Flowgorithm ในการเขียนและทดสอบผังงานแบบทำซ้ำ ตลอดจนการออกแบบกิจกรรมการเรียนรู้เพื่อ

พัฒนาความเข้าใจเรื่องการทำซ้ำสำหรับผู้เรียนระดับประถมศึกษาและมัธยมศึกษา เพื่อให้นักศึกษาครูสามารถนำความรู้ไปใช้ได้ทั้งในด้านวิชาการและการจัดการเรียนรู้ในอนาคต

6.1 ความหมายและลักษณะของโครงสร้างการควบคุมแบบทำซ้ำ (Iteration Structure)

โครงสร้างการควบคุมแบบทำซ้ำ (Iteration Structure) หมายถึง รูปแบบการทำงานของโปรแกรมที่สั่งให้ชุดคำสั่งหนึ่งหรือหลายคำสั่งถูกดำเนินการซ้ำหลายครั้งตามจำนวนรอบหรือภายใต้เงื่อนไขที่กำหนดไว้ จนกว่าจะครบตามเงื่อนไขของการวนซ้ำหรือสิ้นสุดตามที่ออกแบบไว้ โครงสร้างนี้ช่วยลดความซ้ำซ้อนของการเขียนคำสั่ง และทำให้โปรแกรมสามารถจัดการกับงานที่ต้องทำซ้ำได้อย่างมีประสิทธิภาพ

สาระสำคัญของโครงสร้างแบบทำซ้ำอยู่ที่แนวคิดเรื่อง “การวนกลับ” ไปทำคำสั่งเดิมอีกครั้ง โดยแต่ละรอบของการทำงานจะต้องมีการตรวจสอบเงื่อนไขบางประการ เช่น ทำซ้ำจำนวน 10 ครั้ง ทำซ้ำตราบใดที่ค่าคะแนนยังไม่ครบ หรือทำซ้ำจนกว่าผู้ใช้จะป้อนค่าที่ถูกต้อง โครงสร้างการทำซ้ำจึงมีความสัมพันธ์อย่างใกล้ชิดกับแนวคิดเรื่องเงื่อนไข การตรวจสอบค่า และการควบคุมลำดับการทำงานของโปรแกรม

ลักษณะสำคัญของโครงสร้างแบบทำซ้ำสามารถสรุปได้ดังนี้ ประการแรก คือ มีชุดคำสั่งที่ต้องการให้ทำงานซ้ำ ประการที่สอง คือ มีตัวควบคุมการทำซ้ำ เช่น ตัวแปรนับรอบหรือเงื่อนไขในการตรวจสอบ ประการที่สาม คือ มีการกำหนดจุดเริ่มต้นและจุดสิ้นสุดของการวนซ้ำอย่างชัดเจน และประการที่สี่ คือ ต้องมีเงื่อนไขที่ทำให้การวนซ้ำสิ้นสุด มิฉะนั้นอาจเกิดการวนซ้ำไม่รู้จบ ซึ่งเรียกว่า วนซ้ำไม่สิ้นสุด (Infinite Loop)

ในชีวิตประจำวัน แนวคิดของการทำซ้ำสามารถพบได้ในกิจกรรมทั่วไป เช่น การเดินขึ้นบันไดทีละขั้นจนถึงชั้นเรียน การล้างจานทีละใบจนหมด การทบทวนคำศัพท์วันละหลายคำ หรือการวิ่งรอบสนามตามจำนวนรอบที่กำหนด กิจกรรมเหล่านี้ล้วนมีลักษณะร่วมกันคือ มีการทำสิ่งเดิมซ้ำหลายครั้งภายใต้จุดมุ่งหมายหรือเงื่อนไขบางประการ จึงสามารถใช้เป็นพื้นฐานในการอธิบายแนวคิดเรื่องการทำซ้ำในเชิงโปรแกรมได้อย่างดี

ในทางการเขียนโปรแกรม โครงสร้างแบบทำซ้ำมีประโยชน์มากเมื่อโปรแกรมต้องจัดการกับข้อมูลจำนวนมาก หรือเมื่อต้องทำงานลักษณะเดียวกันหลายครั้ง เช่น การแสดงลำดับตัวเลข การหาผลรวมของจำนวนหลายค่า การตรวจคำตอบหลายข้อ หรือการประมวลผลข้อมูลทีละรายการ หากไม่ใช้

โครงสร้างแบบทำซ้ำ ผู้เขียนโปรแกรมอาจต้องเขียนคำสั่งเดิมซ้ำหลายบรรทัด ซึ่งไม่เหมาะสมและเพิ่มโอกาสเกิดข้อผิดพลาด

สำหรับนักศึกษาครู การเข้าใจโครงสร้างแบบทำซ้ำควรครอบคลุมทั้งในมิติของตรรกะการเขียนโปรแกรมและการนำไปออกแบบการเรียนรู้ ครูควรสามารถอธิบายให้ผู้เรียนเห็นว่า การทำซ้ำไม่ได้เป็นเรื่องของโค้ดเพียงอย่างเดียว แต่เป็นรูปแบบการคิดที่เกี่ยวข้องกับกิจกรรมซ้ำ ๆ ในชีวิตจริง และสามารถใช้กิจกรรมเชิงปฏิบัติช่วยสร้างความเข้าใจให้แก่ผู้เรียนก่อนที่จะนำเข้าสู่ผังงานและโปรแกรม

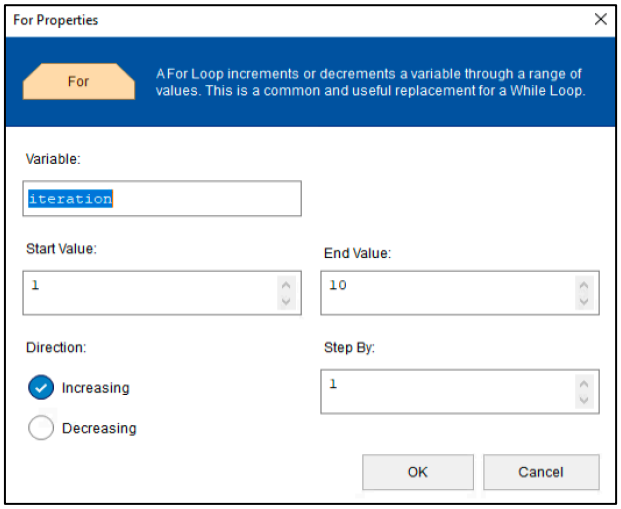
กล่าวโดยสรุป โครงสร้างการควบคุมแบบทำซ้ำเป็นกลไกสำคัญที่ช่วยให้โปรแกรมทำงานซ้ำอย่างเป็นระบบตามจำนวนรอบหรือเงื่อนไขที่กำหนด เป็นพื้นฐานสำคัญของการเขียนโปรแกรมที่มีประสิทธิภาพ และเป็นแนวคิดที่ครูสามารถนำไปเชื่อมโยงกับประสบการณ์ของผู้เรียนได้อย่างกว้างขวาง

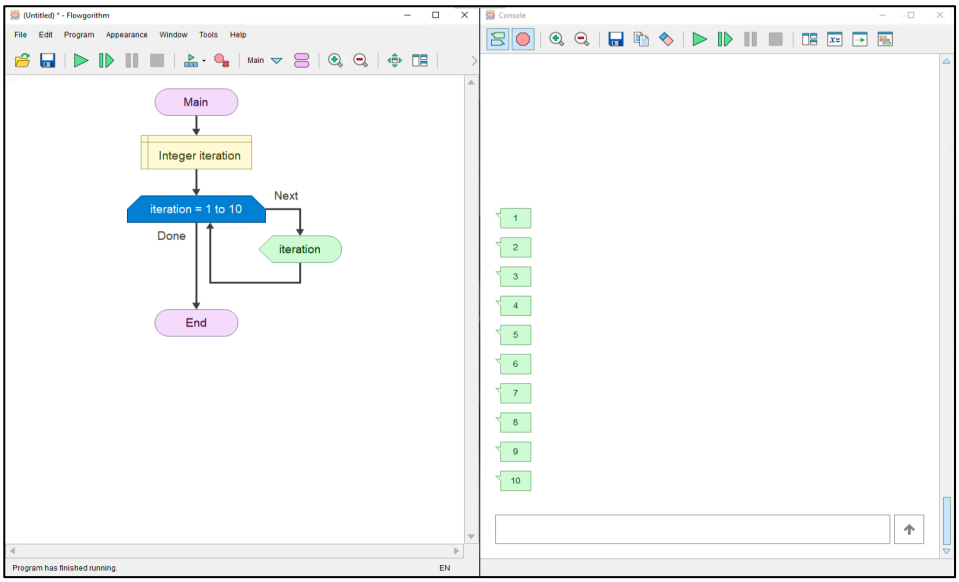
6.2 ประเภทของการทำซ้ำและการประยุกต์ใช้ในการแก้ปัญหา

โครงสร้างการทำซ้ำในทางวิทยาการคำนวณสามารถจำแนกได้หลายลักษณะตามวิธีการควบคุมการวนซ้ำ โดยการเข้าใจประเภทของการทำซ้ำจะช่วยให้ผู้เรียนเลือกใช้แนวทางที่เหมาะสมกับลักษณะของปัญหาและออกแบบอัลกอริทึมได้อย่างมีประสิทธิภาพ โดยทั่วไป ประเภทของการทำซ้ำที่สำคัญสามารถอธิบายได้ดังนี้

6.2.1 การทำซ้ำตามจำนวนรอบที่กำหนด (For)

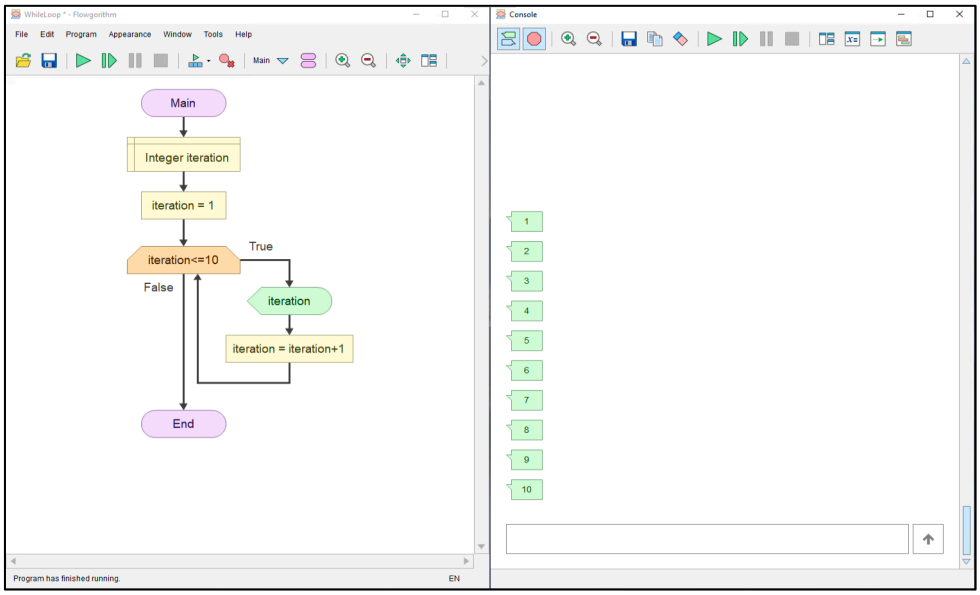
การทำซ้ำประเภทนี้เกิดขึ้นเมื่อผู้เขียนโปรแกรมทราบล่วงหน้าว่าต้องการให้ชุดคำสั่งทำงานกี่ครั้ง เช่น แสดงตัวเลข 1 ถึง 10 พิมพ์ชื่อนักเรียน 5 ครั้ง หรือคำนวณผลรวมของคะแนน 10 ค่า การทำซ้ำลักษณะนี้มักใช้ตัวแปรนับรอบเพื่อควบคุมจำนวนครั้งของการทำงาน และเหมาะกับปัญหาที่มีจำนวนรอบแน่นอน





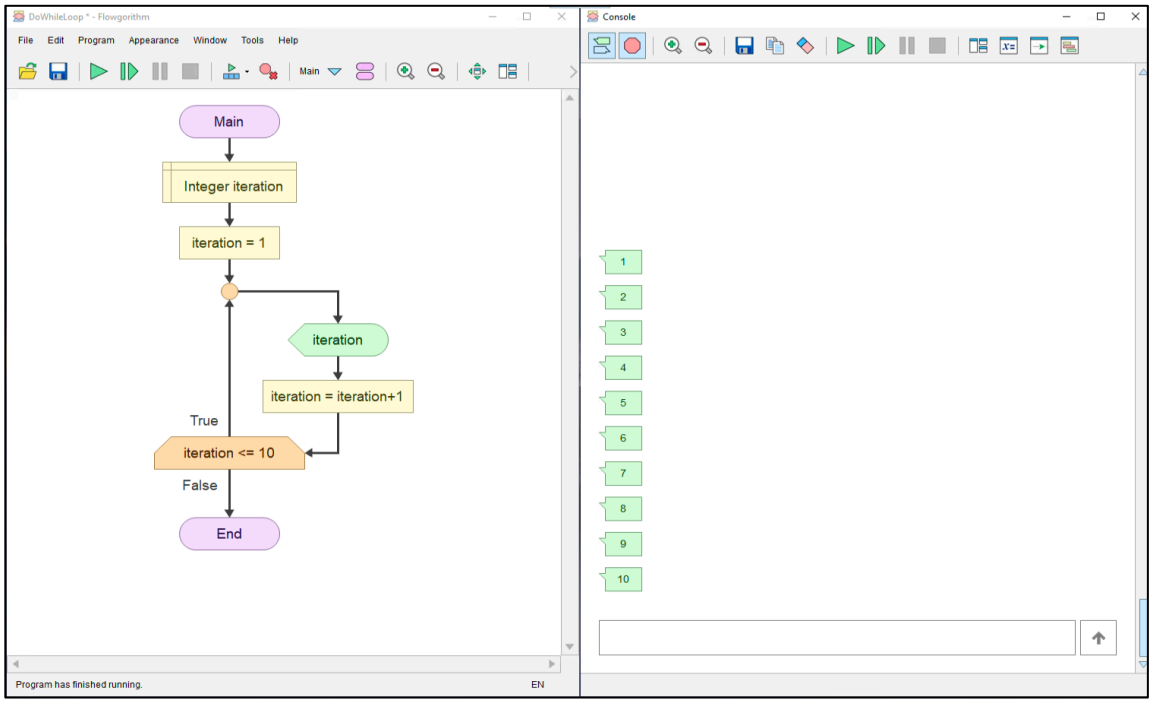
6.2.2 การทำซ้ำตามเงื่อนไขก่อนทำงาน (While)

การทำซ้ำประเภทนี้จะตรวจสอบเงื่อนไขก่อนทุกครั้ง หากเงื่อนไขเป็นจริงจึงจะทำชุดคำสั่งซ้ำต่อไป แต่หากเงื่อนไขเป็นเท็จจะออกจากการวนซ้ำทันที ตัวอย่างเช่น การแสดงตัวเลขตั้งแต่ 1 จนถึง 10 โดยทำซ้ำตราบใดที่ค่าตัวแปรยังไม่เกิน 10 หรือการรับข้อมูลจากผู้ใช้ไปเรื่อย ๆ จนกว่าผู้ใช้จะป้อนค่าที่ถูกต้อง



6.2.3 การทำซ้ำตามเงื่อนไขหลังทำงาน (Do-While)

การทำซ้ำประเภทนี้จะให้ชุดคำสั่งทำงานก่อนหนึ่งรอบ แล้วจึงตรวจสอบเงื่อนไขว่าจะทำซ้ำต่อหรือไม่ ลักษณะนี้เหมาะกับสถานการณ์ที่ต้องการให้โปรแกรมทำงานอย่างน้อยหนึ่งครั้งก่อนเสมอ เช่น การแสดงเมนูให้ผู้ใช้เลือกก่อน แล้วจึงตรวจสอบว่าต้องการทำรายการต่อหรือไม่



การเลือกใช้ประเภทของการทำซ้ำควรพิจารณาจากลักษณะของปัญหา หากทราบจำนวนรอบแน่นอน การใช้การทำซ้ำตามจำนวนรอบ (For) จะเหมาะสมและเข้าใจง่าย แต่หากไม่ทราบจำนวนรอบล่วงหน้าและขึ้นอยู่กับเงื่อนไข เช่น การรับค่าจากผู้ใช้จนกว่าจะถูกต้อง การใช้การทำซ้ำตามเงื่อนไข (While) จะมีความยืดหยุ่นมากกว่า

ในทางปฏิบัติ การทำซ้ำถูกนำไปประยุกต์ใช้อย่างกว้างขวางในการแก้ปัญหาต่าง ๆ เช่น การคำนวณผลรวมและค่าเฉลี่ยของคะแนนหลายค่า การตรวจสอบข้อมูลทีละรายการ การแสดงแม่สูตรคูณ การหาค่ามากที่สุดหรือค่าน้อยที่สุดในชุดข้อมูล การสร้างแบบฝึกหัดที่มีข้อจำนวนมาก หรือการทำงานซ้ำในเกมและโปรแกรมเชิงโต้ตอบ ปัญหาเหล่านี้ล้วนต้องอาศัยการทำงานซ้ำอย่างเป็นระบบเพื่อให้โปรแกรมทำงานได้อย่างครบถ้วน

สำหรับผู้เรียน การทำความเข้าใจประเภทของการทำซ้ำจะช่วยให้มองเห็นความแตกต่างของปัญหา และสามารถเลือกใช้วิธีการที่เหมาะสมในการเขียนผังงานและโปรแกรมได้ นอกจากนี้ ยังช่วยพัฒนาทักษะการวิเคราะห์ปัญหา เพราะผู้เรียนต้องพิจารณาว่าปัญหานั้นควบคุมด้วยจำนวนรอบที่แน่นอน หรือด้วยเงื่อนไขที่อาจเปลี่ยนแปลงได้

สำหรับนักศึกษาครู ควรเชื่อมโยงแนวคิดเรื่องการทำซ้ำกับกิจกรรมในชีวิตประจำวันและการจัดกิจกรรมการเรียนรู้ เช่น การฝึกให้นักเรียนปรบมือ 5 ครั้ง การเดินไปข้างหน้าจนถึงเส้นที่กำหนด การตอบคำถามจนกว่าจะถูก หรือการทำกิจกรรมซ้ำภายใต้เงื่อนไขบางอย่าง กิจกรรมเช่นนี้จะช่วยให้ผู้เรียนเข้าใจประเภทของการทำซ้ำจากประสบการณ์ตรง ก่อนที่จะเรียนรู้ในรูปของผังงานหรือภาษาโปรแกรม

กล่าวโดยสรุป ประเภทของการทำซ้ำมีความแตกต่างกันตามวิธีการควบคุมรอบของการทำงาน การเข้าใจลักษณะเหล่านี้จะช่วยให้ผู้เรียนสามารถออกแบบวิธีแก้ปัญหาได้อย่างเหมาะสม และเป็นพื้นฐานสำคัญของการพัฒนาโปรแกรมที่มีประสิทธิภาพ

6.3 การออกแบบผังงานแบบทำซ้ำ

ผังงานแบบทำซ้ำเป็นการแสดงลำดับขั้นตอนการทำงานของโปรแกรมที่มีการวนกลับไปทำชุดคำสั่งเดิมซ้ำหลายครั้งตามจำนวนรอบหรือเงื่อนไขที่กำหนด การออกแบบผังงานลักษณะนี้ช่วยให้ผู้เรียนเห็นภาพของการวนซ้ำอย่างชัดเจน เข้าใจจุดเริ่มต้น จุดตรวจสอบเงื่อนไข ทิศทางการไหลของงาน และจุดสิ้นสุดของการทำซ้ำได้ดียิ่งขึ้น

สัญลักษณ์ที่ใช้ในผังงานแบบทำซ้ำยังคงอาศัยสัญลักษณ์พื้นฐานเช่นเดียวกับผังงานทั่วไป ได้แก่ สัญลักษณ์เริ่มต้นและสิ้นสุด สัญลักษณ์กระบวนการ สัญลักษณ์ข้อมูลเข้าและข้อมูลออก และสัญลักษณ์การตัดสินใจ อย่างไรก็ตาม สิ่งที่ทำให้ผังงานแบบทำซ้ำแตกต่างจากผังงานแบบเรียงลำดับและแบบทางเลือก คือ การมีเส้นทางย้อนกลับจากชุดคำสั่งไปยังจุดตรวจสอบเงื่อนไข หรือกลับไปยังจุดเริ่มต้นของรอบถัดไป

หลักการออกแบบผังงานแบบทำซ้ำมีประเด็นสำคัญดังนี้ ประการแรก ต้องกำหนดให้ชัดเจนว่าคำสั่งใดคือส่วนที่ต้องทำซ้ำ ประการที่สอง ต้องกำหนดเงื่อนไขหรือจำนวนรอบของการทำซ้ำให้แน่นอน ประการที่สาม ต้องมีการปรับค่าของตัวแปรควบคุมในแต่ละรอบ เช่น การเพิ่มค่า ลดค่า หรือเปลี่ยนสถานะของตัวแปร และประการที่สี่ ต้องออกแบบให้มีเงื่อนไขสิ้นสุดที่ชัดเจนเพื่อป้องกันการวนซ้ำไม่สิ้นสุด

ตัวอย่างผังงานแบบทำซ้ำที่เข้าใจง่าย คือ การแสดงตัวเลข 1 ถึง 5 โดยอาจออกแบบดังนี้

1. เริ่มต้น
2. กำหนดค่า i = 1
3. ตรวจสอบว่า i <= 5 หรือไม่
4. หากจริง ให้แสดงค่า i
5. เพิ่มค่า i = i + 1
6. กลับไปตรวจสอบเงื่อนไขอีกครั้ง
7. หากเท็จ ให้สิ้นสุดการทำงาน

อีกตัวอย่างหนึ่ง คือ การหาผลรวมของตัวเลข 1 ถึง 10 โดยอาจออกแบบให้มีตัวแปร sum สำหรับเก็บผลรวม และตัวแปร i สำหรับนับรอบ จากนั้นให้วนซ้ำเพิ่มค่า sum ด้วยค่า i ในแต่ละรอบ แล้วเพิ่มค่า i ทีละ 1 จนครบเงื่อนไข การออกแบบเช่นนี้ช่วยให้ผู้เรียนเห็นบทบาทของตัวแปรควบคุมและตัวแปรสะสมผลอย่างชัดเจน

การฝึกออกแบบผังงานแบบทำซ้ำมีประโยชน์หลายประการ ได้แก่ ช่วยพัฒนาการคิดอย่างเป็นระบบ ช่วยให้ผู้เรียนเห็นความสัมพันธ์ระหว่างเงื่อนไขกับการทำงานซ้ำ และช่วยเตรียมความพร้อมก่อนการเขียนโปรแกรมจริง นอกจากนี้ การออกแบบผังงานยังช่วยให้ครูตรวจสอบได้ง่ายว่าผู้เรียนเข้าใจจุดที่ต้องวนซ้ำ จุดที่ต้องเปลี่ยนค่าตัวแปร และจุดที่ควรหยุดการทำงานหรือไม่

สำหรับนักศึกษาครู การสอนผังงานแบบทำซ้ำควรค่อย ๆ พัฒนาจากสถานการณ์ที่เป็นรูปธรรม เช่น การเดินรอบสนาม 3 รอบ การปรบมือ 5 ครั้ง หรือการตอบคำถามจนกว่าจะถูก แล้วจึงเชื่อมโยงไปสู่การเขียนผังงานจากโจทย์คำนวณอย่างง่าย ครูอาจใช้กิจกรรมให้ผู้เรียนสวมบทบาทเป็น “ตัวแปร” และ “เงื่อนไข” เพื่อช่วยให้เห็นภาพการทำงานของการวนซ้ำอย่างสนุกและเข้าใจง่าย

กล่าวโดยสรุป การออกแบบผังงานแบบทำซ้ำเป็นกระบวนการสำคัญที่ช่วยให้ผู้เรียนเข้าใจการทำงานแบบวนซ้ำของโปรแกรมอย่างเป็นรูปธรรม และเป็นพื้นฐานสำคัญก่อนการพัฒนาไปสู่การเขียนโปรแกรมจริงด้วยภาษาโปรแกรมต่าง ๆ

6.4 การใช้โปรแกรม Flowgorithm ในการเขียนและทดสอบผังงานแบบทำซ้ำ

Flowgorithm เป็นโปรแกรมที่มีประโยชน์อย่างมากต่อการเรียนรู้โครงสร้างการควบคุมแบบทำซ้ำ เพราะช่วยให้ผู้เรียนสามารถสร้างผังงานที่มีการวนซ้ำ และทดสอบการทำงานของแต่ละรอบได้อย่างชัดเจน ผู้เรียนจึงไม่เพียงเห็นภาพรวมของผังงาน แต่ยังสามารถสังเกตลำดับการทำงานจริงของโปรแกรมและตรวจสอบค่าตัวแปรที่เปลี่ยนแปลงไปในแต่ละรอบได้อีกด้วย

ในการใช้ Flowgorithm เขียนผังงานแบบทำซ้ำ ผู้เรียนควรเริ่มจากการวิเคราะห์โจทย์ก่อนว่าปัญหานั้นต้องการให้ทำซ้ำกี่ครั้ง หรือทำซ้ำจนกว่าเงื่อนไขใดจะเป็นจริง จากนั้นจึงกำหนดตัวแปรที่จำเป็น เช่น ตัวแปรนับรอบ ตัวแปรเก็บผลรวม หรือตัวแปรสำหรับรับข้อมูล แล้วจึงเลือกใช้รูปแบบของการทำซ้ำที่เหมาะสมภายในโปรแกรม

ตัวอย่างเช่น หากต้องการสร้างผังงานเพื่อแสดงตัวเลข 1 ถึง 5 ผู้เรียนอาจกำหนดตัวแปร i และกำหนดค่าเริ่มต้น i = 1 จากนั้นใช้โครงสร้างทำซ้ำที่ตรวจสอบว่า i <= 5 หากเงื่อนไขเป็นจริง ให้แสดงค่า i แล้วเพิ่มค่า i ขึ้นทีละ 1 จากนั้นกลับไปตรวจสอบเงื่อนไขอีกครั้ง จนกระทั่ง i มีค่ามากกว่า 5 จึงสิ้นสุดการทำงาน

อีกตัวอย่างหนึ่ง คือ การหาผลรวมของตัวเลข 1 ถึง 10 ผู้เรียนอาจกำหนดตัวแปร i = 1 และ sum = 0 จากนั้นให้วนซ้ำตราบใดที่ i <= 10 โดยในแต่ละรอบให้คำนวณ sum = sum + i แล้วเพิ่มค่า i = i + 1 เมื่อครบเงื่อนไขแล้วจึงแสดงค่าของ sum ออกมา ตัวอย่างนี้ช่วยให้ผู้เรียนเข้าใจบทบาทของตัวแปรนับรอบและตัวแปรสะสมผลได้อย่างชัดเจน

จุดเด่นของ Flowgorithm คือ ผู้เรียนสามารถ Run ผังงานและสังเกตผลการทำงานทีละขั้นตอน หากผังงานมีข้อผิดพลาด เช่น ลืมเพิ่มค่าตัวแปรนับรอบ เขียนเงื่อนไขผิด หรือกำหนดค่าเริ่มต้นไม่เหมาะสม โปรแกรมจะช่วยให้ผู้เรียนเห็นพฤติกรรมที่ผิดปกติ เช่น วนซ้ำไม่สิ้นสุด หรือได้ผลลัพธ์ไม่ตรงตามคาดหมาย การสังเกตเช่นนี้ช่วยให้ผู้เรียนเรียนรู้จากข้อผิดพลาดได้อย่างเป็นรูปธรรม

นอกจากนี้ Flowgorithm ยังเอื้อต่อการอภิปรายในชั้นเรียน ครูสามารถใช้ผังงานตัวอย่างที่ตั้งใจออกแบบให้มีข้อผิดพลาด แล้วให้ผู้เรียนร่วมกันวิเคราะห์ว่าเหตุใดโปรแกรมจึงทำงานผิด เช่น เหตุใดจึงวนซ้ำไม่หยุด หรือเหตุใดจึงแสดงผลเพียงบางค่า กิจกรรมลักษณะนี้ช่วยพัฒนาทั้งทักษะการคิดวิเคราะห์และการอธิบายเหตุผลของผู้เรียน

สำหรับนักศึกษาครู การใช้ Flowgorithm ควรเชื่อมโยงไปสู่การออกแบบกิจกรรมการเรียนรู้ เช่น การให้ผู้เรียนทำนายผลลัพธ์ก่อน Run ผังงาน การให้เปรียบเทียบผลลัพธ์เมื่อเปลี่ยนค่าเริ่มต้นหรือเปลี่ยนเงื่อนไข หรือการให้ออกแบบผังงานเองจากโจทย์ที่กำหนดแล้วนำมาแลกเปลี่ยนกับเพื่อน กิจกรรมเหล่านี้จะช่วยให้ครูฝึกทั้งการใช้เทคโนโลยีและการจัดการเรียนรู้แบบส่งเสริมการคิด

กล่าวโดยสรุป การใช้โปรแกรม Flowgorithm ในการเขียนและทดสอบผังงานแบบทำซ้ำช่วยให้ผู้เรียนเข้าใจการวนซ้ำอย่างลึกซึ้งมากขึ้น เพราะสามารถเห็นทั้งโครงสร้างของผังงาน ลำดับการทำงานของโปรแกรม และผลจากการเปลี่ยนแปลงค่าตัวแปรในแต่ละรอบได้อย่างชัดเจน

6.5 การใช้ Trace Table เพื่อวิเคราะห์การทำงานของโครงสร้างแบบทำซ้ำ

Trace Table เป็นเครื่องมือสำคัญที่ใช้ในการติดตามและบันทึกค่าของตัวแปรระหว่างการทำงานของอัลกอริทึม ผังงาน หรือโปรแกรมในแต่ละขั้นตอนอย่างเป็นลำดับ โดยเฉพาะอย่างยิ่งในการเรียนรู้เรื่องโครงสร้างการควบคุมแบบทำซ้ำ ซึ่งผู้เรียนมักเกิดความสับสนเกี่ยวกับจำนวนรอบของการทำงาน การเปลี่ยนแปลงค่าของตัวแปร และเงื่อนไขที่ทำให้การทำซ้ำสิ้นสุด การใช้ Trace Table จึงช่วยทำให้กระบวนการที่มีลักษณะเป็นนามธรรมกลายเป็นสิ่งที่มองเห็นและตรวจสอบได้อย่างชัดเจน

ในบริบทของการเรียนวิทยาการคำนวณ Trace Table ทำหน้าที่เสมือนเครื่องมือช่วย “แกะรอย” การทำงานของโปรแกรม โดยแสดงให้เห็นว่าในแต่ละรอบของการทำซ้ำ ตัวแปรมีค่าเริ่มต้นอย่างไร มีการเปลี่ยนแปลงค่าเมื่อใด และส่งผลต่อผลลัพธ์สุดท้ายอย่างไร เครื่องมือนี้จึงมีประโยชน์อย่างมากต่อผู้เรียนที่กำลังเริ่มต้นทำความเข้าใจแนวคิดเรื่องลูป เพราะช่วยลดความซับซ้อนของการคิดหลายขั้นตอนพร้อมกัน และทำให้สามารถติดตามกระบวนการทำงานของโปรแกรมได้ทีละรอบอย่างเป็นระบบ

สำหรับนักศึกษาครู การใช้ Trace Table มีคุณค่าอย่างยิ่งทั้งในฐานะผู้เรียนและในฐานะผู้สอน กล่าวคือ ในฐานะผู้เรียน Trace Table ช่วยเสริมสร้างความเข้าใจเชิงตรรกะเกี่ยวกับการทำซ้ำได้อย่างลึกซึ้ง ส่วนในฐานะผู้สอน Trace Table เป็นสื่อการอธิบายที่มีประสิทธิภาพ สามารถนำไปใช้ช่วยให้นักเรียนระดับประถมศึกษาและมัธยมศึกษาเข้าใจการทำงานของลูปได้ง่ายขึ้น ลดความสับสน และลดภาระทางปัญญาในการประมวลผลข้อมูลหลายส่วนพร้อมกัน

6.5.1 ความหมายและความสำคัญของ Trace Table

Trace Table หมายถึง ตารางที่ใช้บันทึกค่าของตัวแปรที่เกี่ยวข้องกับการทำงานของโปรแกรมในแต่ละขั้นตอนหรือแต่ละรอบของการประมวลผล โดยตารางนี้มักประกอบด้วยคอลัมน์สำหรับลำดับรอบ เงื่อนไขที่ตรวจสอบ ค่าของตัวแปร และผลลัพธ์ที่เกิดขึ้น จุดมุ่งหมายสำคัญของ Trace Table คือเพื่อช่วยให้ผู้เรียนสามารถวิเคราะห์ลำดับการทำงานของโปรแกรมได้อย่างชัดเจนและตรวจสอบได้ว่าผลลัพธ์สุดท้ายเกิดขึ้นจากกระบวนการใด

ความสำคัญของ Trace Table ต่อการเรียนรู้โครงสร้างแบบทำซ้ำมีหลายประการ ประการแรก คือ ช่วยให้ผู้เรียนมองเห็นค่าของตัวแปรในแต่ละรอบ เช่น ตัวแปรนับรอบ ตัวแปรสะสมผล หรือค่าที่รับเข้ามา ประการที่สอง คือ ช่วยให้เห็นความสัมพันธ์ระหว่างเงื่อนไขการทำซ้ำกับการเปลี่ยนแปลงค่าของตัวแปร ประการที่สาม คือ ช่วยให้ตรวจสอบข้อผิดพลาดได้ง่าย เช่น การกำหนดค่าเริ่มต้นผิด การลืมเพิ่มค่าตัวแปรนับรอบ หรือการกำหนดเงื่อนไขหยุดไม่ถูกต้อง และประการที่สี่ คือ ช่วยให้ผู้เรียนเข้าใจผลลัพธ์สุดท้ายของโปรแกรมอย่างมีเหตุผล

ในแง่การจัดการเรียนรู้ Trace Table ยังมีประโยชน์อย่างมากในการลด Cognitive Load หรือภาระทางปัญญาของผู้เรียน เนื่องจากการทำงานของลูปมักต้องอาศัยการติดตามหลายองค์ประกอบพร้อมกัน ได้แก่ ค่าเริ่มต้นของตัวแปร เงื่อนไขของลูป การเปลี่ยนค่าของตัวแปรในแต่ละรอบ และผลลัพธ์สะสม หากผู้เรียนพยายามจินตนาการสิ่งเหล่านี้ทั้งหมดในใจพร้อมกัน อาจทำให้เกิดความสับสนได้ง่าย แต่เมื่อใช้ Trace Table ผู้เรียนจะสามารถติดตามข้อมูลทีละส่วนและทีละรอบอย่างเป็นขั้นตอน ส่งผลให้การทำความเข้าใจเกิดขึ้นได้ง่ายและเป็นระบบมากขึ้น

6.5.2 องค์ประกอบของ Trace Table

แม้ Trace Table จะสามารถออกแบบได้หลากหลายตามลักษณะของปัญหา แต่โดยทั่วไปองค์ประกอบพื้นฐานมักประกอบด้วยส่วนสำคัญดังนี้

1) ลำดับรอบ (Iteration / รอบที่) ใช้แสดงว่าข้อมูลในแถวนั้นเป็นการทำงานรอบใดของลูป เพื่อให้ผู้เรียนติดตามความต่อเนื่องของการทำซ้ำได้อย่างชัดเจน

2) เงื่อนไขการทำซ้ำ ใช้บันทึกผลของการตรวจสอบเงื่อนไขในแต่ละรอบว่าเป็นจริงหรือเท็จ ซึ่งช่วยให้เห็นว่าลูปยังคงทำงานต่อหรือสิ้นสุดลงในรอบใด

3) ค่าของตัวแปรสำคัญ เช่น ตัวแปรนับรอบ ตัวแปรสะสมผล ตัวแปรข้อมูลนำเข้า หรือค่าชั่วคราวต่าง ๆ การบันทึกค่านี้ช่วยให้ผู้เรียนเห็นว่าตัวแปรมีการเปลี่ยนแปลงอย่างไรในแต่ละรอบ

4) ผลลัพธ์หรือการแสดงผล ในบางกรณีอาจมีคอลัมน์สำหรับแสดงข้อความหรือค่าที่โปรแกรมแสดงออกมาในแต่ละรอบ เพื่อให้เห็นความสัมพันธ์ระหว่างกระบวนการภายในกับผลลัพธ์ภายนอก

องค์ประกอบของ Trace Table ควรเลือกให้เหมาะสมกับวัตถุประสงค์ของการวิเคราะห์ หากต้องการเน้นการเปลี่ยนแปลงของตัวแปรสะสมผล ก็ควรมีคอลัมน์สำหรับตัวแปรนั้นอย่างชัดเจน หากต้องการเน้นการตรวจสอบเงื่อนไข ก็ควรมีคอลัมน์แสดงผลการเปรียบเทียบในแต่ละรอบด้วย ทั้งนี้ การออกแบบตารางที่เหมาะสมจะช่วยให้ผู้เรียนมองเห็นกระบวนการทำงานของลูปได้ง่ายขึ้น

6.5.3 การใช้ Trace Table กับการทำซ้ำตามจำนวนรอบ

การทำซ้ำตามจำนวนรอบที่กำหนดเป็นกรณีที่เหมาะสมอย่างยิ่งสำหรับการเริ่มต้นใช้ Trace Table เพราะมีลักษณะตรงไปตรงมาและช่วยให้ผู้เรียนเห็นรูปแบบการเปลี่ยนแปลงของตัวแปรได้ชัดเจน ตัวอย่างเช่น การแสดงตัวเลข 1 ถึง 5 อาจกำหนดขั้นตอนการทำงานดังนี้

1. กำหนดค่าเริ่มต้นให้ i = 1
2. ตรวจสอบว่า i <= 5 หรือไม่
3. หากจริง ให้แสดงค่า i
4. เพิ่มค่า i = i + 1
5. กลับไปตรวจสอบเงื่อนไขอีกครั้ง

Trace Table สำหรับกรณีนี้อาจเขียนได้ดังนี้

รอบที่	i ก่อนตรวจสอบ	เงื่อนไข i <= 5	แสดงผล	i หลังเพิ่มค่า
1	1	จริง	1	2
2	2	จริง	2	3
3	3	จริง	3	4
4	4	จริง	4	5
5	5	จริง	5	6
6	6	เท็จ	-	-

จากตารางจะเห็นได้อย่างชัดเจนว่า ตัวแปร i เริ่มต้นที่ 1 และเพิ่มค่าทีละ 1 ในแต่ละรอบ จนกระทั่งมีค่าเป็น 6 ซึ่งทำให้เงื่อนไข i <= 5 เป็นเท็จ ลูปจึงหยุดทำงาน การใช้ Trace Table ลักษณะนี้ช่วยให้ผู้เรียนเข้าใจว่า แม้โปรแกรมจะแสดงผลเพียง 1 ถึง 5 แต่การตรวจสอบเงื่อนไขยังคงเกิดขึ้นอีกรอบหนึ่งก่อนสิ้นสุดการทำงาน

อีกตัวอย่างหนึ่ง เช่น การหาผลรวมของตัวเลข 1 ถึง 4 โดยมีขั้นตอนดังนี้

1. กำหนดค่า i = 1 และ sum = 0
2. ตรวจสอบว่า i <= 4 หรือไม่
3. หากจริง ให้คำนวณ sum = sum + i
4. เพิ่มค่า i = i + 1
5. กลับไปตรวจสอบเงื่อนไข

Trace Table อาจแสดงได้ดังนี้

รอบที่	i	sum ก่อนบวก	sum หลังบวก	i หลังเพิ่มค่า	เงื่อนไข i <= 4
1	1	0	1	2	จริง
2	2	1	3	3	จริง
3	3	3	6	4	จริง
4	4	6	10	5	จริง
5	5	10	10	-	เท็จ

ตารางนี้ช่วยให้ผู้เรียนมองเห็นบทบาทของตัวแปร 2 ประเภทอย่างชัดเจน คือ ตัวแปร i ที่ทำหน้าที่นับรอบ และตัวแปร sum ที่ทำหน้าที่สะสมผล เมื่อเห็นการเปลี่ยนแปลงค่าทีละรอบ ผู้เรียนจะเข้าใจได้ง่ายขึ้นว่าผลรวมสุดท้ายเกิดขึ้นจากการบวกสะสมอย่างไร

6.5.4 การใช้ Trace Table กับการทำซ้ำตามเงื่อนไข

นอกจากการทำซ้ำตามจำนวนรอบที่แน่นอนแล้ว Trace Table ยังมีประโยชน์อย่างมากกับการทำซ้ำตามเงื่อนไข ซึ่งมักมีความซับซ้อนกว่า เพราะจำนวนรอบอาจไม่สามารถทราบล่วงหน้าได้อย่างแน่นอน ตัวอย่างเช่น การรับค่าคะแนนจากผู้ใช้ไปเรื่อย ๆ จนกว่าจะป้อนค่าที่อยู่ในช่วง 0 ถึง 100

สมมติขั้นตอนการทำงานเป็นดังนี้

1. รับค่าคะแนน
2. ขณะที่คะแนนน้อยกว่า 0 หรือมากกว่า 100 ให้รับค่าคะแนนใหม่
3. เมื่อได้คะแนนที่ถูกต้องแล้ว ให้แสดงข้อความว่า “คะแนนถูกต้อง”

หากผู้ใช้ป้อนค่า 120, -5, 75 ตามลำดับ Trace Table อาจเขียนได้ดังนี้

รอบที่	คะแนนที่รับ	เงื่อนไข คะแนน < 0 หรือ > 100	การทำงาน
1	120	จริง	รับค่าใหม่
2	-5	จริง	รับค่าใหม่
3	75	เท็จ	แสดงผลว่าคะแนนถูกต้อง

ตัวอย่างนี้ช่วยให้ผู้เรียนเข้าใจว่า การทำซ้ำตามเงื่อนไขไม่ได้ขึ้นอยู่กับจำนวนรอบที่ตายตัว แต่ขึ้นอยู่กับผลของข้อมูลที่ได้รับในแต่ละรอบ การใช้ Trace Table จึงช่วยให้มองเห็นจุดสิ้นสุดของลูปได้อย่างเป็นรูปธรรม และช่วยลดความสับสนเกี่ยวกับคำถามที่ว่า “ลูปนี้จะหยุดเมื่อใด”

6.5.5 การใช้ Trace Table เพื่อตรวจสอบข้อผิดพลาดของผังงานและโปรแกรม

ประโยชน์สำคัญอีกประการหนึ่งของ Trace Table คือ การใช้เป็นเครื่องมือวิเคราะห์ข้อผิดพลาดของผังงานและโปรแกรม โดยเฉพาะข้อผิดพลาดที่เกี่ยวข้องกับการทำซ้ำ ซึ่งมักเกิดขึ้นได้บ่อยในผู้เรียนระดับเริ่มต้น เช่น

1. กำหนดค่าเริ่มต้นของตัวแปรไม่ถูกต้อง
2. เขียนเงื่อนไขการทำซ้ำผิด
3. ลืมเพิ่มหรือลดค่าตัวแปรควบคุม
4. บวกสะสมผลผิดลำดับ
5. ทำให้เกิดการวนซ้ำไม่สิ้นสุด

ตัวอย่างเช่น หากผู้เรียนออกแบบลูปแสดงตัวเลข 1 ถึง 5 แต่ลืมเพิ่มค่า i = i + 1 เมื่อสร้าง Trace Table จะพบว่า i มีค่าเป็น 1 ตลอดทุกแถว และเงื่อนไข i <= 5 จะเป็นจริงเสมอ ส่งผลให้โปรแกรมวนซ้ำไม่สิ้นสุด การใช้ตารางจึงช่วยให้ผู้เรียนเห็นข้อผิดพลาดอย่างชัดเจนกว่าการมองจากผังงานหรือโค้ดเพียงอย่างเดียว

ในมุมของครู Trace Table ยังใช้เป็นเครื่องมือวินิจฉัยความเข้าใจของผู้เรียนได้ดี หากผู้เรียนเติมตารางผิด ครูจะสามารถวิเคราะห์ได้ว่าผู้เรียนสับสนที่จุดใด เช่น ไม่เข้าใจว่าต้องตรวจสอบเงื่อนไขก่อนหรือหลังทำงาน ไม่เข้าใจการเปลี่ยนค่าในแต่ละรอบ หรือไม่เข้าใจบทบาทของตัวแปรสะสมผล การวิเคราะห์เช่นนี้จะช่วยให้ครูวางแผนการสอนเสริมได้ตรงจุดมากขึ้น

6.5.6 แนวทางการใช้ Trace Table เพื่อลด Cognitive Load ของผู้เรียน

แนวคิดเรื่อง Cognitive Load หรือภาระทางปัญญา มีความสำคัญอย่างยิ่งต่อการสอนวิทยาการคำนวณ เพราะผู้เรียนมักต้องประมวลผลหลายองค์ประกอบพร้อมกัน เช่น ค่าเริ่มต้นของตัวแปร การเปลี่ยนค่าในแต่ละรอบ เงื่อนไขของลูป และผลลัพธ์ของโปรแกรม หากครูอธิบายทั้งหมดพร้อมกันโดยไม่มีเครื่องมือช่วย ผู้เรียนอาจเกิดความสับสนและไม่สามารถสร้างความเข้าใจที่มั่นคงได้

Trace Table ช่วยลดภาระทางปัญญาได้ด้วยการแยกกระบวนการทำงานของโปรแกรมออกเป็นหน่วยย่อยที่ตรวจสอบได้ทีละรอบ ผู้เรียนไม่จำเป็นต้องจดจำทุกอย่างไว้ในความคิดพร้อมกัน แต่สามารถมองตารางทีละแถวและติดตามการเปลี่ยนแปลงค่าของตัวแปรอย่างเป็นลำดับ วิธีการนี้ช่วยให้ผู้เรียนมุ่งความสนใจไปยังสาระสำคัญของแต่ละช่วงการทำงาน และค่อย ๆ สร้างความเข้าใจภาพรวมจากรายละเอียดทีละส่วน

สำหรับนักศึกษาครู แนวทางการใช้ Trace Table เพื่อลด Cognitive Load อาจดำเนินการได้ดังนี้

1) เริ่มจากตัวอย่างง่ายก่อน ควรเริ่มด้วยลูปที่มีตัวแปรไม่มาก เช่น การแสดงตัวเลข 1 ถึง 5 ก่อน แล้วจึงค่อยเพิ่มความซับซ้อน เช่น การหาผลรวม ค่าเฉลี่ย หรือการรับข้อมูลหลายค่า

2) ใช้ตารางที่มีคอลัมน์เท่าที่จำเป็น ไม่ควรใส่ตัวแปรมากเกินไปในช่วงเริ่มต้น เพราะจะทำให้ผู้เรียนสับสน ควรเลือกเฉพาะตัวแปรที่สำคัญต่อการทำความเข้าใจโจทย์นั้น

3) ให้ผู้เรียนติดตามทีละรอบ ครูควรพาผู้เรียนวิเคราะห์ทีละแถว โดยเน้นคำถาม เช่น “รอบนี้ตัวแปรมีค่าเท่าไร” “เงื่อนไขเป็นจริงหรือเท็จ” “หลังทำคำสั่งแล้วค่าเปลี่ยนเป็นเท่าไร” วิธีนี้ช่วยให้ผู้เรียนไม่ข้ามขั้นตอน

4) เชื่อมโยงกับผังงานและการ Run โปรแกรม หลังจากเติม Trace Table แล้ว ควรให้ผู้เรียนเปรียบเทียบกับผังงานหรือผลจากการ Run ใน Flowgorithm เพื่อให้เห็นว่าตารางไม่ใช่สิ่งแยกขาดจากโปรแกรม แต่เป็นเครื่องมืออธิบายการทำงานของโปรแกรมอย่างเป็นระบบ

5) ใช้ Trace Table เป็นสะพานจากรูปธรรมสู่นามธรรม ในระดับประถมศึกษาและมัธยมศึกษาตอนต้น ครูอาจเริ่มจากกิจกรรมที่เป็นรูปธรรม เช่น การเดินทีละก้าวหรือการนับลูกบอล แล้วจึงสร้างตารางติดตามค่าหรือจำนวนในแต่ละรอบ เพื่อค่อย ๆ เชื่อมโยงไปสู่แนวคิดเรื่องตัวแปรและลูป

กล่าวโดยสรุป Trace Table เป็นเครื่องมือที่มีคุณค่าอย่างยิ่งในการสอนโครงสร้างการควบคุมแบบทำซ้ำ เพราะช่วยให้ผู้เรียนติดตามการเปลี่ยนแปลงค่าของตัวแปรในแต่ละรอบได้อย่างชัดเจน ช่วยตรวจสอบข้อผิดพลาดของผังงานและโปรแกรม และช่วยลด Cognitive Load ในกระบวนการเรียนรู้ สำหรับนักศึกษาครู การเข้าใจและสามารถใช้ Trace Table ได้อย่างเหมาะสม จะเป็นพื้นฐานสำคัญในการอธิบายเรื่องการทำซ้ำให้แก่ผู้เรียนได้ง่ายขึ้น เป็นระบบขึ้น และมีประสิทธิภาพมากขึ้น

6.6 การออกแบบกิจกรรมการเรียนรู้เพื่อพัฒนาความเข้าใจเรื่องการทำซ้ำสำหรับผู้เรียนระดับประถมศึกษาและมัธยมศึกษา

การจัดการเรียนรู้เรื่องโครงสร้างแบบทำซ้ำสำหรับผู้เรียนระดับประถมศึกษาและมัธยมศึกษาควรยึดหลักการสำคัญคือ ทำให้ผู้เรียนเห็นว่าการทำซ้ำเป็นสิ่งที่เกิดขึ้นได้ในชีวิตจริง และสามารถเข้าใจได้ผ่านกิจกรรมที่เป็นรูปธรรม ก่อนที่จะพัฒนาไปสู่การใช้ผังงานและโปรแกรม การออกแบบกิจกรรมจึงควรเน้นความใกล้ตัว ความชัดเจน และการมีส่วนร่วมของผู้เรียน

สำหรับระดับประถมศึกษา กิจกรรมควรมุ่งให้ผู้เรียนสังเกตและปฏิบัติการทำซ้ำจากสถานการณ์ง่าย ๆ เช่น การปรบมือ 5 ครั้ง การกระโดด 3 ครั้ง การเดินตามคำสั่งซ้ำ ๆ หรือการวาดรูปแบบที่ซ้ำกัน ครูอาจใช้บัตรคำสั่ง เกม หรือบทบาทสมมุติเพื่อให้ผู้เรียนเข้าใจว่าคำสั่งเดิมสามารถทำซ้ำได้หลายครั้งโดยไม่ต้องเขียนใหม่ทุกครั้ง จากนั้นจึงเชื่อมโยงไปสู่การวาดภาพลำดับขั้นตอนหรือผังงานอย่างง่าย

สำหรับระดับมัธยมศึกษา ผู้เรียนสามารถเรียนรู้ในระดับที่ซับซ้อนมากขึ้น ครูอาจใช้โจทย์เกี่ยวกับการแสดงลำดับตัวเลข การคำนวณผลรวม การสร้างแม่สูตรคูณ หรือการประมวลผลข้อมูลหลายค่า เพื่อให้ผู้เรียนฝึกวิเคราะห์ว่าปัญหาใดเหมาะกับการใช้โครงสร้างแบบทำซ้ำ จากนั้นให้เขียนอัลกอริทึม วาดผังงาน และทดลองสร้างผังงานใน Flowgorithm

แนวทางการออกแบบกิจกรรมที่เหมาะสมอาจประกอบด้วย

1. การเชื่อมโยงกับสถานการณ์จริง เช่น การวิ่งรอบสนาม การเก็บลูกบอลใส่กล่องทีละลูก หรือการนับจำนวนวัตถุ
2. การใช้กิจกรรม Unplugged เช่น เกมลำดับคำสั่ง เกมการ์ด หรือบทบาทสมมุติเพื่อสาธิตการวนซ้ำ
3. การใช้ผังงานและภาพประกอบเพื่อช่วยให้ผู้เรียนเห็นโครงสร้างของการทำซ้ำ
4. การใช้ Flowgorithm เพื่อให้ผู้เรียนทดลองสร้างและ Run ผังงานด้วยตนเอง
5. การอภิปรายข้อผิดพลาดจากการออกแบบการวนซ้ำ เช่น ลืมปรับค่าตัวแปรหรือตั้งเงื่อนไขผิด

ในการจัดกิจกรรม ครูควรใช้คำถามกระตุ้นการคิด เช่น “เราจะรู้ได้อย่างไรว่าควรหยุดทำซ้ำเมื่อใด” “ถ้าไม่เพิ่มค่าตัวแปรในแต่ละรอบจะเกิดอะไรขึ้น” หรือ “สถานการณ์นี้ควรใช้การทำซ้ำตามจำนวนรอบหรือการทำซ้ำตามเงื่อนไข” คำถามลักษณะนี้จะช่วยให้ผู้เรียนไม่เพียงทำตามตัวอย่าง แต่สามารถเข้าใจเหตุผลของโครงสร้างการทำซ้ำได้อย่างแท้จริง

ในด้านการประเมินผล ครูควรประเมินทั้งความเข้าใจเชิงแนวคิดและทักษะการปฏิบัติ เช่น การตรวจผังงาน การสังเกตการมีส่วนร่วมในกิจกรรม การประเมินผลจากการ Run ผังงานใน Flowgorithm การให้ผู้เรียนอธิบายลำดับการทำงานของโปรแกรม หรือการให้สร้างชิ้นงานหรือใบงานที่สะท้อนความเข้าใจเรื่องการทำซ้ำอย่างเป็นระบบ

สำหรับนักศึกษาครู การออกแบบกิจกรรมเรื่องการทำซ้ำควรเชื่อมโยงไปสู่การจัดทำแผนการจัดการเรียนรู้ โดยกำหนดจุดประสงค์การเรียนรู้ให้ชัดเจน เลือกสื่อและกิจกรรมที่เหมาะกับวัย วาง

ขั้นตอนการจัดการเรียนรู้ให้เป็นลำดับ และกำหนดการประเมินผลที่ครอบคลุมทั้งด้านความรู้ ทักษะ และเจตคติ ทั้งนี้เพื่อให้ผู้เรียนเกิดความเข้าใจเรื่องการทำซ้ำอย่างลึกซึ้งและสามารถนำไปใช้ได้จริง

กล่าวโดยสรุป การออกแบบกิจกรรมการเรียนรู้เพื่อพัฒนาความเข้าใจเรื่องการทำซ้ำควรเริ่มจากประสบการณ์ใกล้ตัว ใช้กิจกรรมที่ผู้เรียนได้ลงมือปฏิบัติ และค่อย ๆ เชื่อมโยงไปสู่การใช้ผังงานและโปรแกรม เพื่อให้ผู้เรียนมองเห็นว่าการทำซ้ำเป็นทั้งแนวคิดพื้นฐานของวิทยาการคำนวณและเครื่องมือสำคัญในการแก้ปัญหาอย่างเป็นระบบ

บทสรุปประจำบทที่ 6

โครงสร้างการควบคุมโปรแกรมแบบทำซ้ำเป็นหนึ่งในโครงสร้างพื้นฐานสำคัญของวิทยาการคำนวณที่ช่วยให้โปรแกรมสามารถดำเนินคำสั่งเดิมซ้ำหลายครั้งตามจำนวนรอบหรือเงื่อนไขที่กำหนดได้อย่างมีประสิทธิภาพ การเข้าใจความหมาย ลักษณะ และประเภทของการทำซ้ำจึงเป็นพื้นฐานสำคัญในการพัฒนาทักษะการคิดเชิงตรรกะ การมองเห็นรูปแบบของปัญหา และการออกแบบวิธีการแก้ปัญหาอย่างเป็นระบบ

การออกแบบผังงานแบบทำซ้ำช่วยให้ผู้เรียนเห็นภาพการทำงานของการวนซ้ำได้อย่างชัดเจน โดยเฉพาะเมื่อใช้ร่วมกับโปรแกรม Flowgorithm ซึ่งช่วยให้ผู้เรียนสามารถสร้าง ทดสอบ และปรับปรุงผังงานได้อย่างเป็นรูปธรรม นอกจากนี้ สำหรับนักศึกษาครู ความเข้าใจเรื่องการทำซ้ำควรเชื่อมโยงไปสู่การออกแบบกิจกรรมการเรียนรู้ที่เหมาะสมกับระดับของผู้เรียน โดยเริ่มจากสถานการณ์ใกล้ตัว กิจกรรมเชิงปฏิบัติ และการใช้สื่อที่ช่วยให้ผู้เรียนค่อย ๆ พัฒนาไปสู่การเข้าใจผังงานและโปรแกรมอย่างมีความหมาย

คำถามท้ายบท

1. จงอธิบายความหมายของโครงสร้างการควบคุมแบบทำซ้ำ และเหตุใดจึงมีความสำคัญต่อการเขียนโปรแกรม
2. โครงสร้างแบบทำซ้ำมีลักษณะสำคัญอะไรบ้าง
3. ประเภทของการทำซ้ำมีอะไรบ้าง และแต่ละประเภทเหมาะกับปัญหาแบบใด
4. เพราะเหตุใดการกำหนดเงื่อนไขสิ้นสุดของการทำซ้ำจึงมีความสำคัญ
5. วนซ้ำไม่สิ้นสุดคืออะไร และอาจเกิดจากสาเหตุใดบ้าง
6. จงอธิบายหลักการออกแบบผังงานแบบทำซ้ำจากโจทย์อย่างง่าย

7. โปรแกรม Flowgorithm มีประโยชน์อย่างไรต่อการเรียนรู้เรื่องการทำซ้ำ
8. หากต้องการออกแบบผังงานเพื่อแสดงตัวเลข 1 ถึง 10 ท่านจะกำหนดตัวแปรและเงื่อนไขอย่างไร
9. หากท่านเป็นครูระดับประถมศึกษา ท่านจะจัดกิจกรรมอย่างไรเพื่อให้ผู้เรียนเข้าใจแนวคิดเรื่องการทำซ้ำ
10. หากท่านเป็นครูระดับมัธยมศึกษา ท่านจะใช้ Flowgorithm เพื่อส่งเสริมความเข้าใจเรื่องการทำซ้ำของผู้เรียนอย่างไร

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.

บทที่ 7
การโปรแกรมแบบบล็อกคำสั่ง
Block-Based Programming

บทนำ

การพัฒนาทักษะด้านวิทยาการคำนวณและการคิดเชิงคำนวณสำหรับผู้เรียนในศตวรรษที่ 21 มิได้เริ่มต้นจากการเขียนโค้ดที่ซับซ้อนเสมอไป แต่สามารถเริ่มต้นจากการเรียนรู้แนวคิดพื้นฐานของการเขียนโปรแกรมผ่านเครื่องมือที่เหมาะสมกับวัยและระดับประสบการณ์ของผู้เรียน หนึ่งในแนวทางที่ได้รับความนิยมอย่างกว้างขวางในระดับการศึกษาขั้นพื้นฐาน คือ การโปรแกรมแบบบล็อกคำสั่ง (Block-Based Programming) ซึ่งช่วยให้ผู้เรียนเข้าใจหลักการสำคัญของการเขียนโปรแกรม เช่น ลำดับคำสั่ง เงื่อนไข การทำซ้ำ ตัวแปร เหตุการณ์ และการโต้ตอบ โดยไม่ต้องเผชิญกับความซับซ้อนของไวยากรณ์ภาษาโปรแกรมแบบข้อความในระยะแรกเริ่ม

การโปรแกรมแบบบล็อกคำสั่งมีจุดเด่นสำคัญคือ การใช้ชิ้นส่วนคำสั่งในลักษณะของบล็อกที่ผู้เรียนสามารถลาก วาง และเชื่อมต่อกันได้เหมือนการต่อชิ้นส่วนจิ๊กซอว์ วิธีการนี้ช่วยลดข้อผิดพลาดด้านการพิมพ์คำสั่ง ลดภาระทางปัญญาที่เกี่ยวข้องกับโครงสร้างไวยากรณ์ และทำให้ผู้เรียนสามารถมุ่งความสนใจไปยังตรรกะการคิดและการออกแบบขั้นตอนการทำงานได้มากขึ้น การเรียนรู้ด้วยวิธีนี้จึงเหมาะอย่างยิ่งสำหรับผู้เริ่มต้น และเป็นพื้นฐานสำคัญก่อนการพัฒนาไปสู่การเขียนโปรแกรมในรูปแบบที่ซับซ้อนขึ้น

ในบริบทของเอกสารประกอบการสอนรายวิชาวิทยาการคำนวณสำหรับนักศึกษาครู ผู้เขียนเลือกใช้ **Scratch ของ MIT** เป็นเครื่องมือหลักในการอธิบายและฝึกปฏิบัติการโปรแกรมแบบบล็อกคำสั่ง เนื่องจาก Scratch เป็นแพลตฟอร์มที่ได้รับการยอมรับอย่างกว้างขวางในระดับนานาชาติ ใช้งานง่าย มีสภาพแวดล้อมที่เป็นมิตรต่อผู้เรียน สนับสนุนการสร้างงานหลากหลายรูปแบบ เช่น นิทาน เกม แอนิเมชัน สื่อการสอน และกิจกรรมเชิงโต้ตอบ อีกทั้งยังเหมาะกับการนำไปใช้ในโรงเรียนทั้งระดับประถมศึกษาและมัธยมศึกษา

สำหรับนักศึกษาครู ความรู้เรื่องการโปรแกรมแบบบล็อกคำสั่งมีความสำคัญในสองมิติสำคัญ มิติแรก คือ การเข้าใจหลักการเขียนโปรแกรมเบื้องต้นด้วยตนเอง เพื่อพัฒนาความคิดเชิงตรรกะและการออกแบบอัลกอริทึม มิติที่สอง คือ ความสามารถในการนำ Scratch ไปประยุกต์ใช้ในการออกแบบ

กิจกรรมการเรียนรู้และสร้างสื่อการเรียนรู้ที่เหมาะสมกับผู้เรียนในโรงเรียน การเข้าใจทั้งเนื้อหาและวิธีสอนจึงเป็นสิ่งจำเป็นอย่างยิ่งสำหรับนักศึกษาครูในยุคดิจิทัล

บทนี้มุ่งอธิบายแนวคิดพื้นฐานของการโปรแกรมแบบบล็อกคำสั่ง ส่วนประกอบและเครื่องมือของโปรแกรม Scratch การออกแบบลำดับคำสั่ง เงื่อนไข และการทำซ้ำด้วยบล็อกคำสั่ง การสร้างชิ้นงานหรือสื่อการเรียนรู้ด้วย Scratch ตลอดจนแนวทางการจัดกิจกรรมการเรียนรู้ด้วยการโปรแกรมแบบบล็อกคำสั่งสำหรับนักเรียน เพื่อให้นักศึกษาครูสามารถนำความรู้ไปใช้ได้อย่างมีประสิทธิภาพทั้งในเชิงวิชาการและในเชิงวิชาชีพครู

7.1 แนวคิดพื้นฐานของการโปรแกรมแบบบล็อกคำสั่ง

การโปรแกรมแบบบล็อกคำสั่ง คือ รูปแบบการเขียนโปรแกรมที่ใช้บล็อกคำสั่งสำเร็จรูปแทนการพิมพ์โค้ดด้วยข้อความ โดยผู้ใช้สามารถลากและวางบล็อกคำสั่งมาต่อกันเป็นลำดับเพื่อสั่งงานให้โปรแกรมทำงานตามที่ต้องการ วิธีการนี้ช่วยให้ผู้เรียนมองเห็นโครงสร้างของโปรแกรมได้ง่ายขึ้น ลดปัญหาการเขียนคำสั่งผิดรูปแบบ และส่งเสริมให้มุ่งเน้นที่การคิดเชิงตรรกะและกระบวนการแก้ปัญหาเป็นสำคัญ

แนวคิดสำคัญของการโปรแกรมแบบบล็อกคำสั่งอยู่ที่การทำให้ผู้เรียนเข้าใจว่า โปรแกรมคอมพิวเตอร์เกิดจากการกำหนดชุดคำสั่งอย่างเป็นลำดับและมีเหตุผล โดยแต่ละบล็อกคำสั่งทำหน้าที่เฉพาะ เช่น การเคลื่อนที่ การเปลี่ยนรูปลักษณ์ การรับข้อมูล การตัดสินใจ หรือการทำซ้ำ เมื่อบล็อกเหล่านี้ถูกเชื่อมต่อกันอย่างถูกต้อง จะเกิดเป็นลำดับการทำงานที่มีความหมายและสามารถสร้างผลลัพธ์ได้ตามที่ออกแบบไว้

Scratch ของ MIT เป็นตัวอย่างสำคัญของเครื่องมือสำหรับการโปรแกรมแบบบล็อกคำสั่งที่ได้รับความนิยมอย่างแพร่หลาย โดยเฉพาะในการศึกษาระดับประถมศึกษาและมัธยมศึกษา เนื่องจากมีสภาพแวดล้อมที่ใช้งานง่าย รองรับการเรียนรู้ผ่านการลงมือปฏิบัติ และส่งเสริมความคิดสร้างสรรค์ ผู้เรียนสามารถสร้างเรื่องเล่าแบบโต้ตอบ เกมอย่างง่าย แอนิเมชัน สื่อการสอน หรือโครงงานสร้างสรรค์ได้ตามจินตนาการของตน

ข้อดีสำคัญของการโปรแกรมแบบบล็อกคำสั่ง คือ ช่วยลดความกังวลของผู้เรียนที่เพิ่งเริ่มต้นเรียนเขียนโปรแกรม ช่วยลดภาระทางปัญญา (Cognitive Load) ของผู้เรียนลงได้ เพราะไม่ต้องเผชิญกับไวยากรณ์ (Syntax) ของภาษาโปรแกรมที่ซับซ้อน เช่น เครื่องหมาย วงเล็บ หรือรูปแบบคำสั่งที่ต้องพิมพ์

อย่างเคร่งครัด ผู้เรียนจึงสามารถมุ่งเน้นไปที่การคิดลำดับขั้นตอน การเชื่อมโยงเหตุและผล และการทดลองแก้ปัญหาด้วยวิธีต่าง ๆ ได้มากขึ้น

อีกประเด็นหนึ่งที่สำคัญคือ การโปรแกรมแบบบล็อกคำสั่งส่งเสริมการเรียนรู้เชิงสร้างสรรค์ ผู้เรียนมิได้เพียงทำตามคำสั่ง แต่สามารถทดลอง ปรับเปลี่ยน และสร้างชิ้นงานของตนเองได้ การเรียนรู้ในลักษณะนี้ช่วยส่งเสริมแรงจูงใจภายใน ความภูมิใจในผลงาน และความเข้าใจอย่างเป็นรูปธรรมเกี่ยวกับแนวคิดด้านวิทยาการคำนวณ

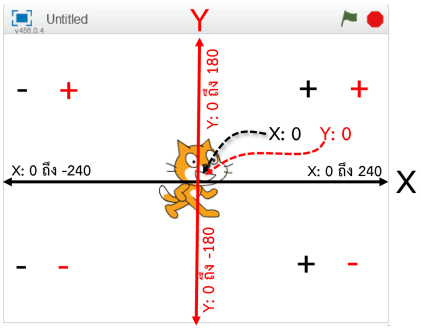
สำหรับนักศึกษาครู การเข้าใจแนวคิดพื้นฐานของการโปรแกรมแบบบล็อกคำสั่งมีความสำคัญอย่างยิ่ง เพราะเป็นฐานในการออกแบบกิจกรรมการเรียนรู้ที่เหมาะกับวัยของผู้เรียน และช่วยให้สามารถอธิบายแนวคิดการเขียนโปรแกรมในรูปแบบที่เข้าถึงง่าย โดยเฉพาะเมื่อต้องทำงานกับนักเรียนที่ยังไม่มีพื้นฐานการเขียนโค้ดมาก่อน

กล่าวโดยสรุป การโปรแกรมแบบบล็อกคำสั่งเป็นแนวทางสำคัญในการเริ่มต้นเรียนรู้การเขียนโปรแกรม เพราะช่วยให้ผู้เรียนเข้าใจหลักการของคำสั่งและการทำงานของโปรแกรมอย่างเป็นระบบ เป็นรูปธรรม และเอื้อต่อการพัฒนาทักษะการคิดเชิงคำนวณในระยะยาว

7.2 ส่วนประกอบและเครื่องมือของโปรแกรมแบบบล็อกคำสั่ง

ในการเรียนรู้การโปรแกรมแบบบล็อกคำสั่งด้วย Scratch ของ MIT ผู้เรียนจำเป็นต้องเข้าใจส่วนประกอบสำคัญของโปรแกรมและหน้าที่ของเครื่องมือแต่ละส่วน เนื่องจากการรู้จักสภาพแวดล้อมของโปรแกรมจะช่วยให้สามารถออกแบบและพัฒนาชิ้นงานได้อย่างมีประสิทธิภาพ และลดความสับสนในระหว่างการใช้งาน

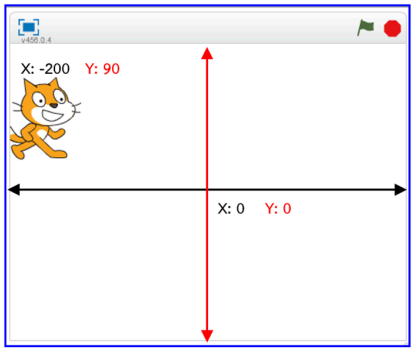
Scratch ประกอบด้วยพื้นที่ทำงานหลักหลายส่วนที่เชื่อมโยงกันอย่างเป็นระบบ ส่วนแรก คือ **เวที (Stage)** ซึ่งเป็นพื้นที่สำหรับแสดงผลการทำงานของโครงการ ไม่ว่าจะเป็นการเคลื่อนไหวของตัวละคร การเปลี่ยนฉาก หรือการโต้ตอบกับผู้ใช้ เวทีจึงเปรียบเสมือนพื้นที่แสดงผลลัพธ์ของโปรแกรมที่ผู้เรียนออกแบบไว้



ส่วนที่สอง คือ **ตัวละคร (Sprite)** ซึ่งเป็นวัตถุหรือองค์ประกอบที่สามารถถูกสั่งให้ทำงานได้ ตัวละครอาจเป็นคน สัตว์ สิ่งของ หรือสัญลักษณ์ต่าง ๆ ใน Scratch ผู้เรียนสามารถเลือกตัวละครจากคลังของโปรแกรม วาดขึ้นเอง หรืออัปโหลดภาพจากภายนอกได้ ตัวละครแต่ละตัวสามารถมีคำสั่ง การเคลื่อนไหว และพฤติกรรมเฉพาะของตนเอง

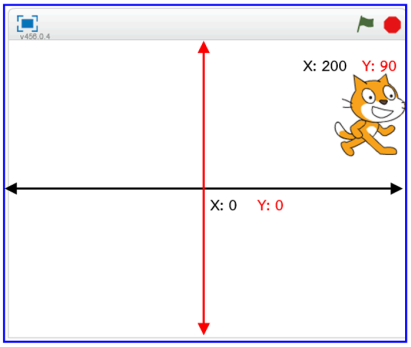
ทำให้ Sprite1 อยู่ทางซ้าย และ ด้านบน ของฉากละคร

เพราะ X มีค่าทางลบ และ Y มีค่าทางบวก

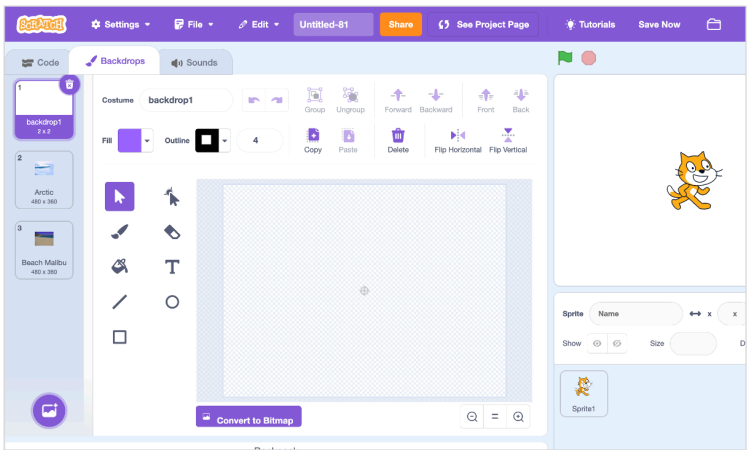


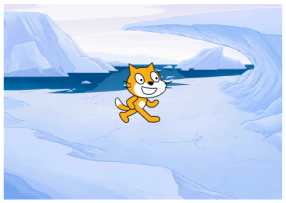
ทำให้ Sprite1 อยู่ ทางขวา และ ด้านบน ของฉากละคร

เพราะ X และ Y มีค่าทางบวก

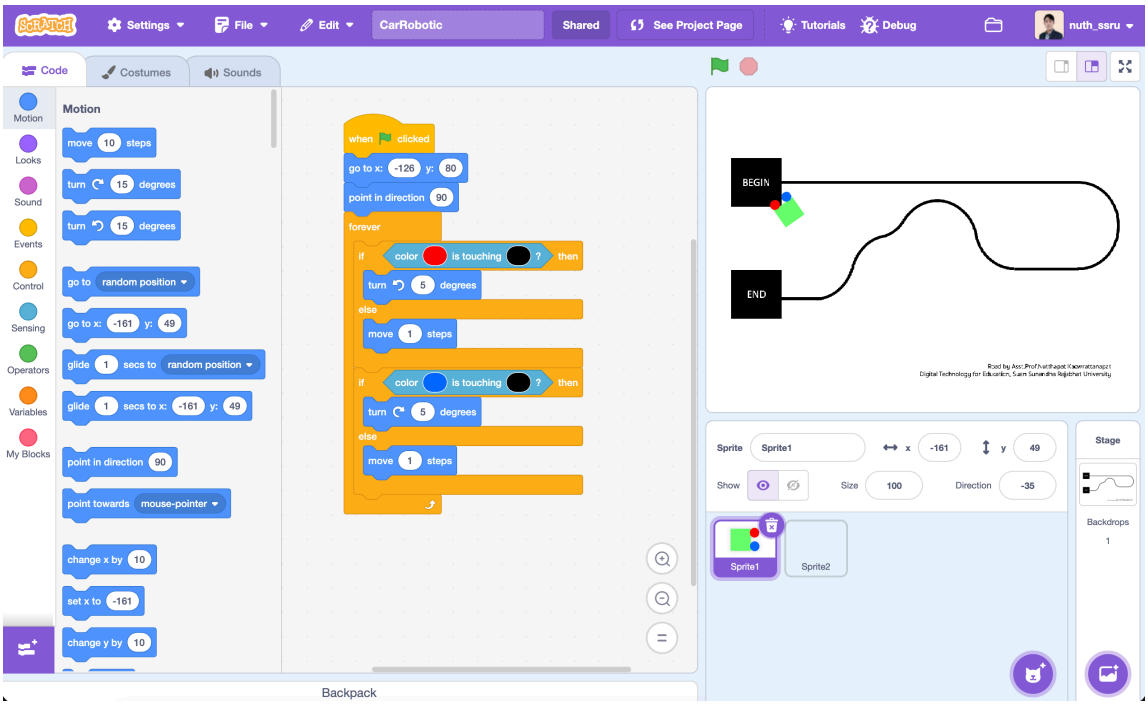


ส่วนที่สาม คือ **ฉากหลัง (Backdrop)** ซึ่งใช้กำหนดสภาพแวดล้อมหรือบริบทของโครงการ เช่น ห้องเรียน สนามเด็กเล่น เมือง หรืออวกาศ ฉากหลังมีบทบาทสำคัญต่อการสื่อความหมายและบรรยากาศของชิ้นงาน โดยเฉพาะเมื่อใช้ Scratch เพื่อสร้างนิทาน เกม หรือสื่อการเรียนรู้





ส่วนที่สี่ คือ **พื้นที่บล็อกคำสั่ง (Blocks Palette)** ซึ่งเป็นแหล่งรวบรวมบล็อกคำสั่งประเภทต่าง ๆ ที่ผู้เรียนสามารถเลือกมาใช้งาน บล็อกคำสั่งใน Scratch ถูกจัดเป็นหมวดหมู่ตามหน้าที่ เช่น การเคลื่อนไหว (Motion) รูปลักษณ์ (Looks) เสียง (Sound) เหตุการณ์ (Events) การควบคุม (Control) การรับรู้ (Sensing) ตัวดำเนินการ (Operators) ตัวแปร (Variables) และบล็อกของฉัน (My Blocks) การจัดหมวดหมู่เช่นนี้ช่วยให้ผู้เรียนเลือกใช้คำสั่งได้สะดวกและเข้าใจบทบาทของแต่ละประเภทได้ง่ายขึ้น



ส่วนที่ห้า คือ **พื้นที่เขียนโปรแกรม (Scripts Area)** ซึ่งเป็นบริเวณที่ผู้เรียนลากบล็อกคำสั่งมาต่อกันเพื่อสร้างลำดับการทำงานของตัวละครหรือวัตถุต่าง ๆ พื้นที่นี้ถือเป็นหัวใจสำคัญของการสร้างโปรแกรม เพราะเป็นจุดที่แนวคิดของผู้เรียนถูกแปลงเป็นโครงสร้างคำสั่งที่สามารถทำงานได้จริง

นอกจากส่วนประกอบหลักแล้ว Scratch ยังมีเครื่องมือเสริมอื่น ๆ เช่น เครื่องมือแก้ไขรูปลักษณ์ของตัวละคร เครื่องมือแก้ไขเสียง ระบบแชร์โครงการผ่านชุมชนออนไลน์ และเครื่องมือสำหรับทดสอบการทำงานของโปรแกรม สิ่งเหล่านี้ช่วยให้ Scratch เป็นมากกว่าเครื่องมือสำหรับเขียนโปรแกรม แต่เป็นแพลตฟอร์มสำหรับการสร้างสรรค์ชิ้นงานดิจิทัลอย่างครบวงจร

สำหรับนักศึกษาครู การเข้าใจส่วนประกอบและเครื่องมือของ Scratch มีความสำคัญในเชิงการสอนอย่างมาก เพราะช่วยให้สามารถอธิบายโครงสร้างของโปรแกรมแก่ผู้เรียนได้อย่างเป็นระบบ และ

สามารถเลือกใช้คุณลักษณะของ Scratch ให้เหมาะกับวัตถุประสงค์ของบทเรียน เช่น หากต้องการเน้นเรื่องเหตุการณ์และการตอบสนอง อาจใช้บล็อก Events และ Sensing มากขึ้น หากต้องการให้ผู้เรียนสร้างเกมอย่างง่าย ก็อาจเน้นการใช้ Motion, Control และ Variables ควบคู่กัน

กล่าวโดยสรุป การเข้าใจส่วนประกอบและเครื่องมือของ Scratch เป็นพื้นฐานสำคัญของการเรียนรู้การโปรแกรมแบบบล็อกคำสั่ง เพราะช่วยให้ผู้เรียนสามารถใช้งานโปรแกรมได้อย่างคล่องตัว มองเห็นความสัมพันธ์ระหว่างส่วนต่าง ๆ ของระบบ และต่อยอดไปสู่การสร้างชิ้นงานที่มีความหมายได้อย่างมีประสิทธิภาพ

7.3 การออกแบบลำดับคำสั่ง เงื่อนไข และการทำซ้ำด้วยบล็อกคำสั่ง

การเขียนโปรแกรมด้วย Scratch แม้จะอยู่ในรูปของบล็อกคำสั่ง แต่ยังคงอาศัยหลักการพื้นฐานของการเขียนโปรแกรมเช่นเดียวกับภาษาโปรแกรมรูปแบบอื่น กล่าวคือ ผู้เรียนต้องสามารถออกแบบลำดับคำสั่ง กำหนดเงื่อนไข และใช้การทำซ้ำได้อย่างเหมาะสม แนวคิดทั้งสามนี้ถือเป็นหัวใจสำคัญของการสร้างโปรแกรมที่สามารถทำงานได้อย่างเป็นระบบและตอบสนองต่อสถานการณ์ต่าง ๆ ได้ตามต้องการ

7.3.1 การออกแบบลำดับคำสั่ง

ลำดับคำสั่ง คือ การกำหนดขั้นตอนการทำงานของโปรแกรมจากต้นไปปลายอย่างเป็นลำดับ เช่น เมื่อต้องการให้ตัวละครเคลื่อนที่และพูดข้อความ ผู้เรียนต้องวางบล็อกคำสั่งในลำดับที่ถูกต้อง เช่น เริ่มจากเหตุการณ์เมื่อกดธงเขียว จากนั้นให้ตัวละครเคลื่อนที่ แล้วจึงแสดงข้อความ หากสลับลำดับคำสั่ง ผลลัพธ์ที่ได้อาจแตกต่างจากที่ตั้งใจไว้ การฝึกออกแบบลำดับคำสั่งจึงช่วยพัฒนาทักษะการคิดอย่างเป็นขั้นตอนและความเข้าใจเรื่องอัลกอริทึม

7.3.2 การใช้เงื่อนไข

เงื่อนไขช่วยให้โปรแกรมสามารถตัดสินใจเลือกทำงานตามสถานการณ์ได้ เช่น หากตัวละครแตะขอบจอให้เด้งกลับ หากตอบคำถามถูกให้เพิ่มคะแนน หรือหากค่าตัวแปรถึงเกณฑ์ให้เปลี่ยนฉาก ใน Scratch การใช้เงื่อนไขมักอาศัยบล็อกคำสั่งในหมวด Control ร่วมกับบล็อกในหมวด Operators หรือ Sensing การเรียนรู้เรื่องเงื่อนไขช่วยให้ผู้เรียนเข้าใจว่าการทำงานของโปรแกรมไม่ได้เป็นเพียงการทำตามลำดับอย่างเดียว แต่สามารถตอบสนองต่อข้อมูลและเหตุการณ์ได้ด้วย

7.3.3 การใช้การทำซ้ำ

การทำซ้ำช่วยให้โปรแกรมสามารถทำคำสั่งเดิมหลายครั้งโดยไม่ต้องวางบล็อกคำสั่งซ้ำ ๆ เช่น การให้ตัวละครเคลื่อนที่ต่อเนื่อง การแสดงแอนิเมชัน การนับคะแนน หรือการเล่นเสียงซ้ำ การใช้บล็อกทำซ้ำใน Scratch เช่น repeat, forever หรือ repeat until ช่วยให้ผู้เรียนเห็นแนวคิดเรื่องลูปอย่างเป็นรูปธรรม และเข้าใจว่าการทำงานบางอย่างสามารถเกิดขึ้นซ้ำภายใต้จำนวนรอบหรือเงื่อนไขที่กำหนดได้

7.3.4 การเชื่อมโยงแนวคิดทั้งสามในการสร้างโปรแกรม

ในการสร้างชิ้นงานจริง ผู้เรียนมักต้องใช้ทั้งลำดับคำสั่ง เงื่อนไข และการทำซ้ำร่วมกัน เช่น เกมอย่างง่ายอาจเริ่มจากคำสั่งเมื่อกดธงเขียว ให้ตัวละครเคลื่อนที่ตลอดเวลา หากแตะวัตถุบางชนิดให้เพิ่มคะแนน และทำซ้ำไปเรื่อย ๆ จนกว่าจะจบเกม การเชื่อมโยงแนวคิดทั้งสามนี้ทำให้ผู้เรียนเห็นว่าโปรแกรมมิได้แยกเป็นส่วน ๆ แต่เป็นระบบของคำสั่งที่ทำงานร่วมกันอย่างเป็นเหตุเป็นผล

สำหรับนักศึกษาครู การสอนเรื่องลำดับคำสั่ง เงื่อนไข และการทำซ้ำด้วย Scratch ควรใช้แนวทางค่อยเป็นค่อยไป เริ่มจากกิจกรรมที่เป็นรูปธรรม เช่น การให้ผู้เรียนออกคำสั่งให้เพื่อนเดินตามเส้นทาง การตัดสินใจจากเงื่อนไขใกล้ตัว หรือการทำกิจกรรมซ้ำ ๆ ในชีวิตประจำวัน แล้วจึงเชื่อมโยงไปสู่การใช้บล็อกคำสั่งในโปรแกรม วิธีการเช่นนี้จะช่วยให้ผู้เรียนสร้างความเข้าใจจากประสบการณ์จริง ไม่ใช่เพียงจดจำตำแหน่งของบล็อกคำสั่งเท่านั้น

กล่าวโดยสรุป การออกแบบลำดับคำสั่ง เงื่อนไข และการทำซ้ำด้วยบล็อกคำสั่งเป็นรากฐานสำคัญของการโปรแกรมด้วย Scratch และเป็นแกนกลางของการพัฒนาทักษะการคิดเชิงคำนวณ ซึ่งนักศึกษาครูควรเข้าใจทั้งในมิติของเนื้อหาและในมิติของการถ่ายทอดให้แก่ผู้เรียน

7.4 การสร้างชิ้นงานหรือสื่อการเรียนรู้ด้วยการโปรแกรมแบบบล็อกคำสั่ง

หนึ่งในจุดเด่นสำคัญของ Scratch คือ ความสามารถในการนำไปใช้สร้างชิ้นงานหรือสื่อการเรียนรู้ที่หลากหลาย ไม่ว่าจะเป็นนิทานแบบโต้ตอบ แอนิเมชัน เกมเพื่อการศึกษา แบบฝึกหัดตอบคำถาม สื่ออธิบายเนื้อหา หรือโครงงานสร้างสรรค์อื่น ๆ ความยืดหยุ่นดังกล่าวทำให้ Scratch ไม่ได้เป็นเพียงเครื่องมือฝึกเขียนโปรแกรม แต่ยังเป็นเครื่องมือสำหรับการสร้างสรรค์สื่อที่สนับสนุนการเรียนรู้ได้อย่างมีประสิทธิภาพ

การสร้างชิ้นงานด้วย Scratch ควรเริ่มจากการกำหนดเป้าหมายหรือวัตถุประสงค์ของชิ้นงานให้ชัดเจนก่อน เช่น ต้องการสร้างเกมฝึกคำนวณ ต้องการสร้างนิทานส่งเสริมคุณธรรม หรือต้องการสร้างสื่อ

อธิบายวงจรชีวิตของพืช เมื่อกำหนดจุดมุ่งหมายแล้ว ผู้เรียนจึงวางแผนองค์ประกอบต่าง ๆ ได้แก่ ตัวละคร ฉากหลัง ลำดับเหตุการณ์ ปฏิสัมพันธ์กับผู้ใช้ และคำสั่งที่ต้องใช้ในการทำงาน
ตัวอย่างของชิ้นงานที่เหมาะสมกับ Scratch ได้แก่

1) นิทานหรือเรื่องเล่าแบบโต้ตอบ

ผู้เรียนสามารถสร้างตัวละครหลายตัว กำหนดบทสนทนา เปลี่ยนฉาก และให้ผู้ใช้มีส่วนร่วมในการเลือกเส้นทางของเรื่องได้ ชิ้นงานลักษณะนี้เหมาะกับการบูรณาการกับวิชาภาษาไทย ภาษาอังกฤษ หรือการปลูกฝังค่านิยม

2) เกมเพื่อการเรียนรู้

Scratch สามารถใช้สร้างเกมอย่างง่าย เช่น เกมตอบคำถาม เกมจับคู่ เกมเก็บคะแนน หรือเกมหลบสิ่งกีดขวาง ซึ่งช่วยให้ผู้เรียนเรียนรู้ผ่านการเล่นและการมีปฏิสัมพันธ์ เกมลักษณะนี้เหมาะกับการทบทวนบทเรียนและส่งเสริมแรงจูงใจในการเรียนรู้

3) สื่ออธิบายเนื้อหาวิชา

ผู้เรียนหรือครูสามารถสร้างแอนิเมชันหรือสื่ออินเทอร์แอกทีฟเพื่ออธิบายเนื้อหา เช่น ระบบสุริยะ วัฏจักรน้ำ หลักคณิตศาสตร์ หรือมารยาทในการใช้อินเทอร์เน็ต สื่อประเภทนี้ช่วยทำให้เนื้อหาที่เป็นนามธรรมเข้าใจได้ง่ายขึ้น

4) แบบฝึกหัดหรือกิจกรรมโต้ตอบ

Scratch สามารถใช้สร้างแบบฝึกหัดที่มีการตอบสนองทันที เช่น ถามคำถามแล้วแจ้งผลว่าถูกหรือผิด หรือให้ผู้เรียนทดลองทำกิจกรรมแล้วรับข้อเสนอแนะกลับทันที สื่อประเภทนี้ช่วยให้การเรียนรู้มีความน่าสนใจและเกิดการเรียนรู้แบบมีปฏิสัมพันธ์

สำหรับนักศึกษาครู การสร้างชิ้นงานด้วย Scratch มีความสำคัญมากกว่าการผลิตผลงานเพียงชิ้นหนึ่ง เพราะเป็นกระบวนการที่ช่วยพัฒนาทั้งความคิดสร้างสรรค์ การออกแบบการเรียนรู้ และการบูรณาการเทคโนโลยีกับเนื้อหาวิชา นักศึกษาครูควรฝึกคิดว่าชิ้นงานที่สร้างขึ้นจะช่วยให้ผู้เรียนเข้าใจอะไร เกิดการมีส่วนร่วมอย่างไร และสามารถใช้ในบริบทการจัดการเรียนรู้จริงได้หรือไม่

กล่าวโดยสรุป การสร้างชิ้นงานหรือสื่อการเรียนรู้ด้วย Scratch เป็นกระบวนการที่ส่งเสริมทั้งการเรียนรู้วิทยาการคำนวณและการออกแบบนวัตกรรมการเรียนรู้ นักศึกษาครูจึงควรได้รับการฝึกฝนให้สามารถใช้ Scratch ทั้งในฐานะเครื่องมือสร้างโปรแกรมและในฐานะเครื่องมือสร้างสื่อการเรียนรู้ที่มีคุณภาพ

7.5 แนวทางการจัดกิจกรรมการเรียนรู้ด้วยการโปรแกรมแบบบล็อกคำสั่งสำหรับนักเรียน

การจัดกิจกรรมการเรียนรู้ด้วยการโปรแกรมแบบบล็อกคำสั่งสำหรับนักเรียนควรคำนึงถึงความพร้อมของผู้เรียน พัฒนาการตามวัย และเป้าหมายของการเรียนรู้เป็นสำคัญ แม้ Scratch จะเป็นเครื่องมือที่ใช้งานง่าย แต่การทำให้ผู้เรียนเกิดความเข้าใจอย่างแท้จริงจำเป็นต้องอาศัยการออกแบบกิจกรรมที่เหมาะสม มิใช่เพียงการสอนตำแหน่งของบล็อกคำสั่งหรือให้ผู้เรียนทำตามตัวอย่างโดยไม่มีความหมาย

หลักการสำคัญประการแรก คือ ควรเริ่มจากกิจกรรมที่เชื่อมโยงกับชีวิตจริงหรือความสนใจของผู้เรียน เช่น เกม เรื่องเล่า การ์ตูน หรือสถานการณ์ใกล้ตัว เพราะเมื่อผู้เรียนเห็นความหมายของสิ่งที่จะสร้าง จะเกิดแรงจูงใจในการเรียนรู้มากขึ้น และมองเห็นว่าการเขียนโปรแกรมเป็นเครื่องมือสำหรับการสร้างสรรค์ มิใช่เพียงภารกิจทางเทคนิค

หลักการประการที่สอง คือ ควรใช้แนวทางการเรียนรู้แบบค่อยเป็นค่อยไป เริ่มจากแนวคิดพื้นฐาน เช่น การใช้เหตุการณ์และลำดับคำสั่ง ก่อนจะค่อย ๆ เพิ่มเงื่อนไข การทำซ้ำ ตัวแปร และการสร้างปฏิสัมพันธ์ที่ซับซ้อนขึ้น การจัดลำดับเนื้อหาเช่นนี้จะช่วยลดความสับสนและทำให้ผู้เรียนสร้างความเข้าใจอย่างมั่นคง

หลักการประการที่สาม คือ ควรเปิดโอกาสให้ผู้เรียนได้ลงมือสร้างชิ้นงานของตนเอง ไม่ใช่เพียงทำตามขั้นตอนของครูทั้งหมด เพราะการเรียนรู้ผ่านการสร้าง (learning by making) เป็นจุดแข็งสำคัญของ Scratch เมื่อผู้เรียนได้ทดลอง ออกแบบ แก้ไข และปรับปรุงผลงานของตน จะเกิดทั้งความเข้าใจเชิงลึกและความภาคภูมิใจในผลงาน

แนวทางการจัดกิจกรรมสำหรับระดับประถมศึกษา อาจเน้นกิจกรรมที่เรียบง่าย เช่น ให้ตัวละครเคลื่อนที่ตามคำสั่ง ให้พูดข้อความหรือเปลี่ยนฉาก สร้างนิทานสั้น ๆ หรือเกมถามตอบอย่างง่าย กิจกรรมควรใช้เวลาไม่ยาวเกินไป และมีภาพหรือเรื่องราวที่ดึงดูดความสนใจของผู้เรียน

สำหรับระดับมัธยมศึกษา อาจเพิ่มความซับซ้อนของงาน เช่น การสร้างเกมที่มีคะแนน การใช้เงื่อนไขหลายรูปแบบ การใช้ตัวแปร การสร้างแบบฝึกหัดโต้ตอบ หรือการทำโครงงานที่บูรณาการกับเนื้อหาในรายวิชาอื่น การเรียนรู้ในระดับนี้ควรเปิดพื้นที่ให้ผู้เรียนคิดวางแผน แบ่งหน้าที่ และทำงานร่วมกันมากขึ้น

ในด้านการประเมินผล ครูควรประเมินทั้งกระบวนการและผลงาน เช่น ความเข้าใจเรื่องลำดับคำสั่ง การใช้เงื่อนไขและการทำซ้ำ ความคิดสร้างสรรค์ ความสามารถในการแก้ปัญหา และการอธิบาย

วิธีการทำงานของชิ้นงาน ไม่ควรประเมินจากความสวยงามของผลงานเพียงอย่างเดียว เพราะเป้าหมายสำคัญอยู่ที่กระบวนการคิดและการเรียนรู้ทางวิทยาการคำนวณ

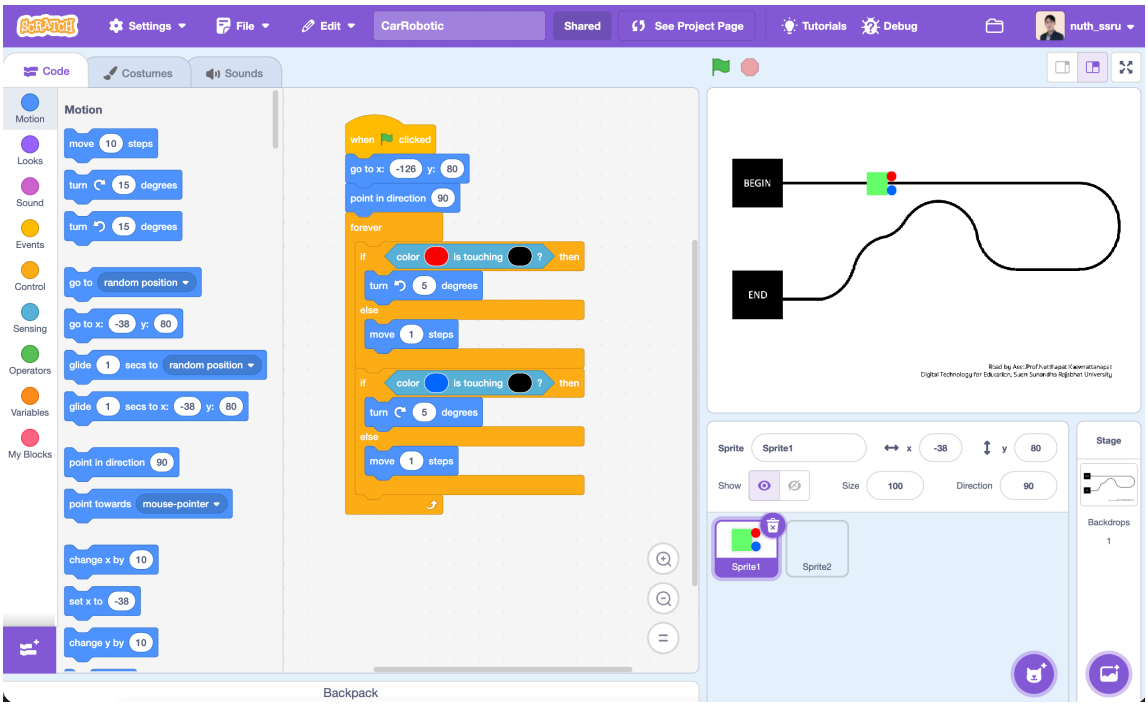
สำหรับนักศึกษาครู การจัดกิจกรรมด้วย Scratch ควรเชื่อมโยงกับการจัดทำแผนการจัดการเรียนรู้ เช่น การกำหนดผลลัพธ์การเรียนรู้ การเลือกกิจกรรมที่เหมาะสมกับวัย การจัดเตรียมสื่อและอุปกรณ์ การออกแบบใบงานหรือภารกิจ และการกำหนดเกณฑ์ประเมินที่สะท้อนทั้งด้านความรู้ ทักษะ และคุณลักษณะของผู้เรียน แนวทางนี้จะช่วยให้การใช้ Scratch ในชั้นเรียนมีความหมายและเกิดผลลัพธ์ทางการศึกษาที่ชัดเจน

กล่าวโดยสรุป การจัดกิจกรรมการเรียนรู้ด้วยการโปรแกรมแบบบล็อกคำสั่งสำหรับนักเรียนควรเน้นความสนุก ความหมาย การลงมือปฏิบัติ และการพัฒนากระบวนการคิดอย่างเป็นระบบ โดยใช้ Scratch เป็นเครื่องมือหลักในการส่งเสริมการเรียนรู้เชิงสร้างสรรค์และการพัฒนาทักษะวิทยาการคำนวณอย่างเหมาะสมตามช่วงวัย

ตัวอย่างกิจกรรมแก้โจทย์ปัญหาด้วย Scratch

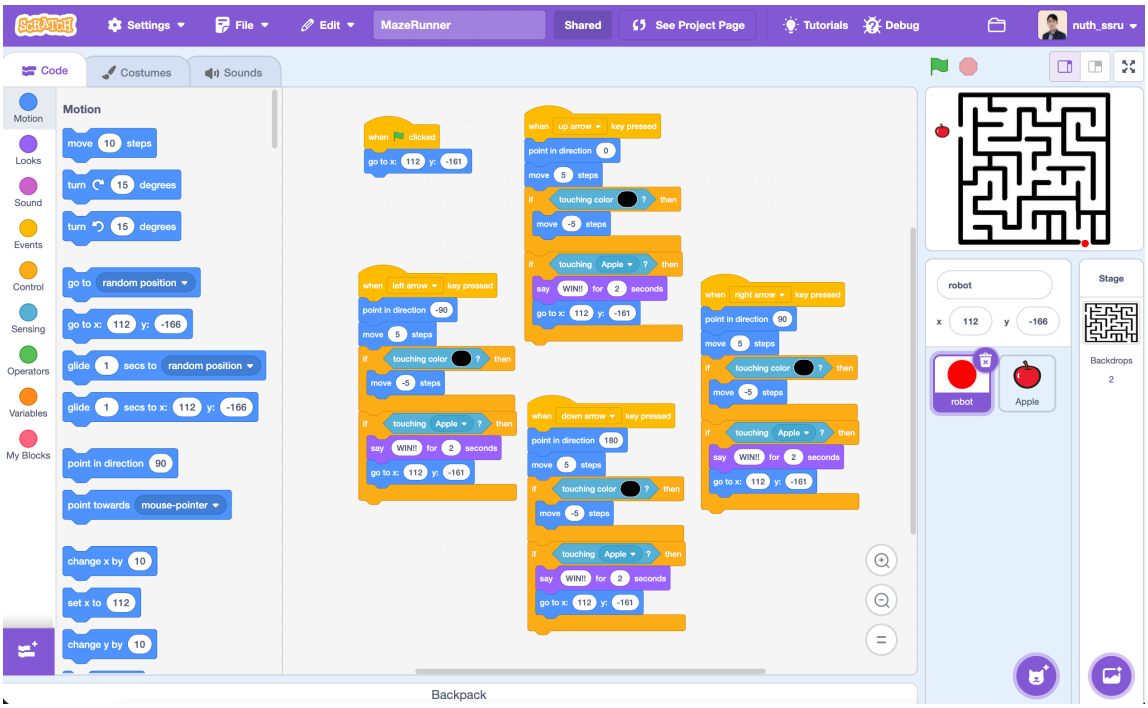
1. รถเดินตามเส้นอัตโนมัติ ดาวน์โหลดตัวอย่างการสาธิตได้ที่

https://scratch.mit.edu/projects/715600028



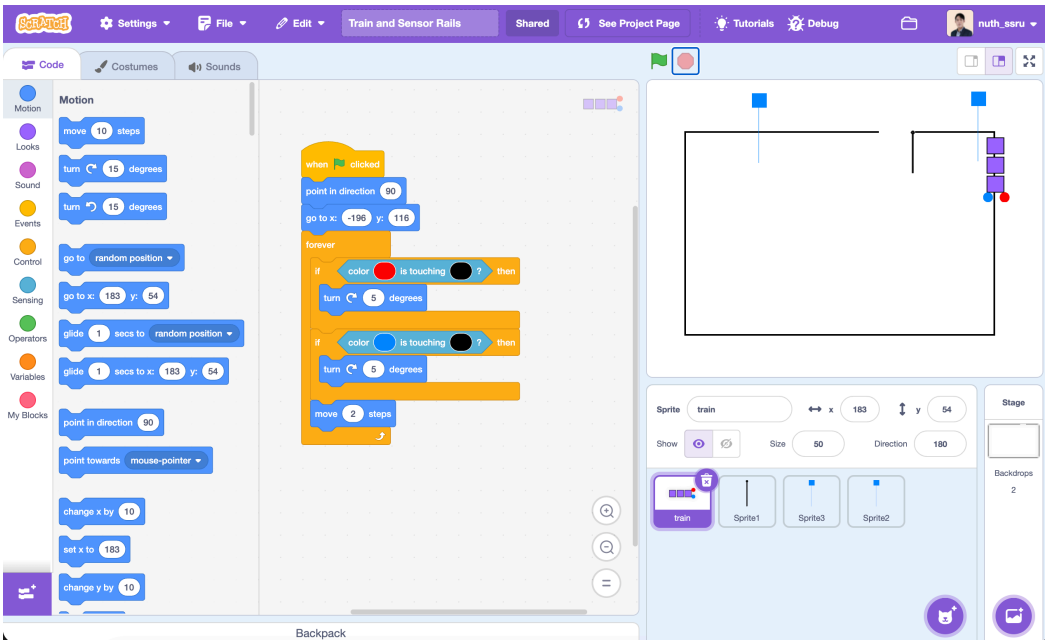
2. เขาวงกต ควบคุมด้วยแป้น cursor ดาวน์โหลดตัวอย่างการสาธิตได้ที่

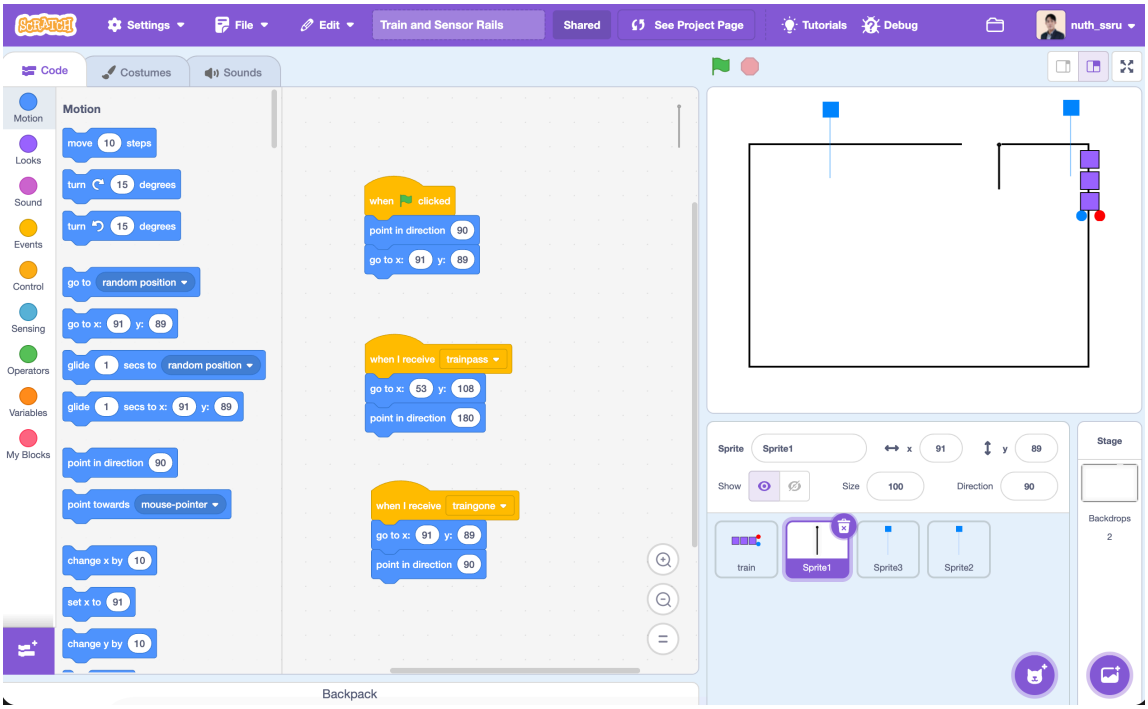
https://scratch.mit.edu/projects/728592636

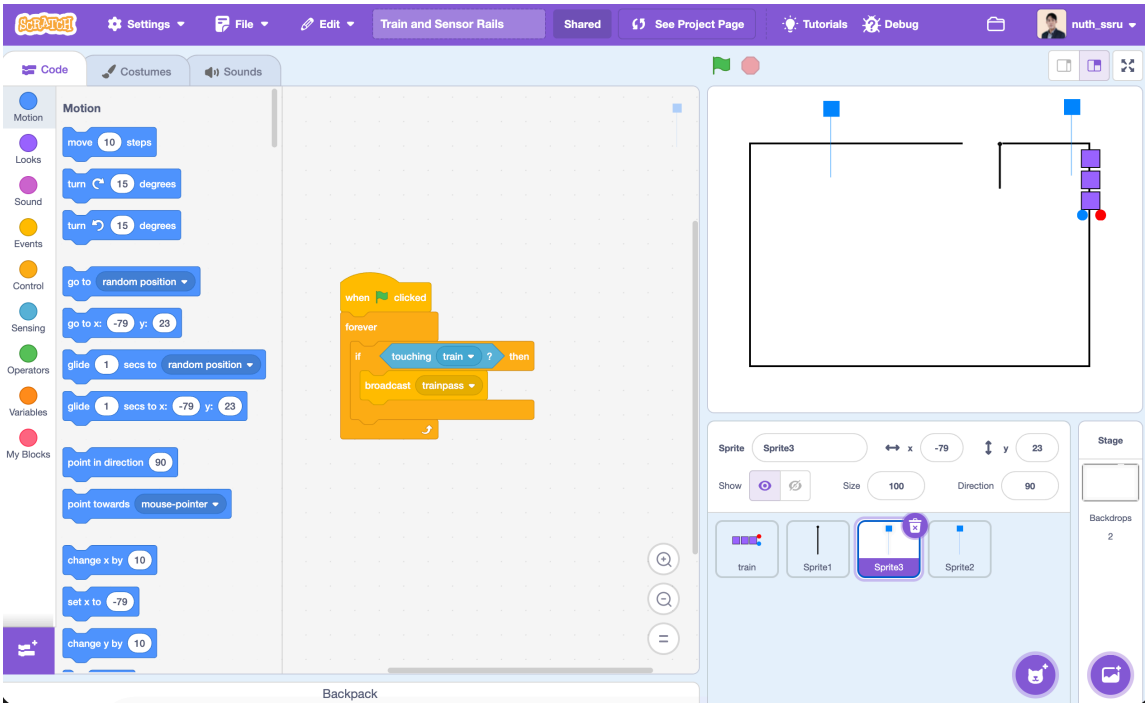


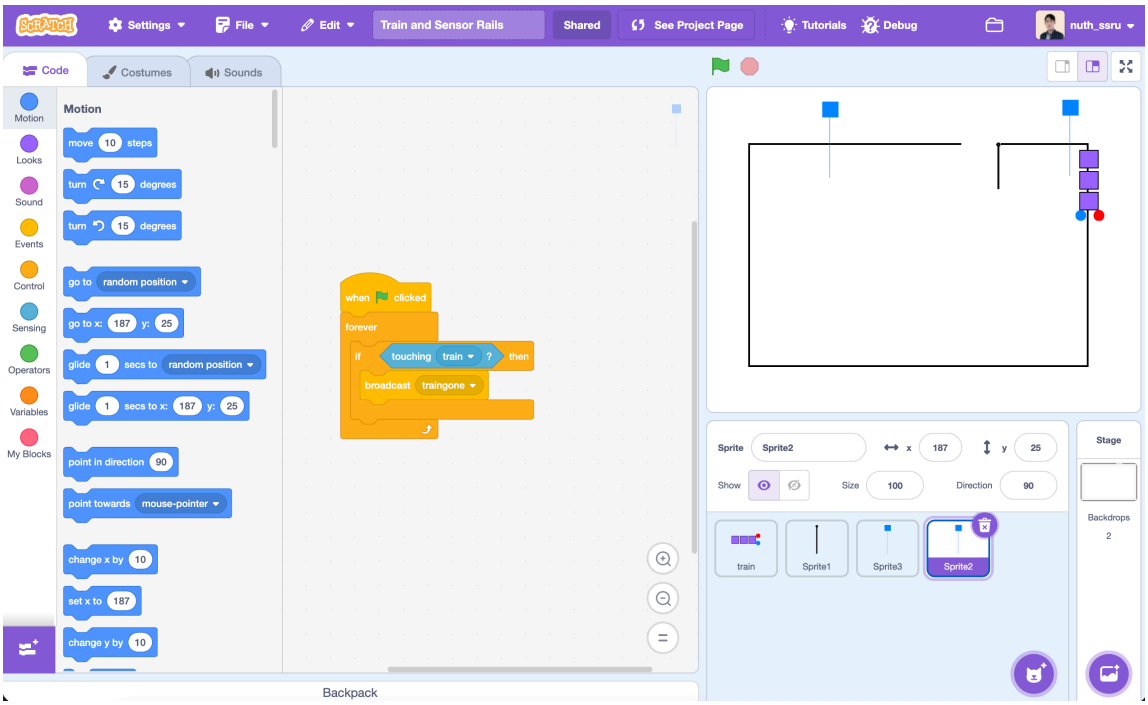
3. รถไฟสับลางอัตโนมัติ ดาวน์โหลดตัวอย่างการสาธิตได้ที่

https://scratch.mit.edu/projects/743127024



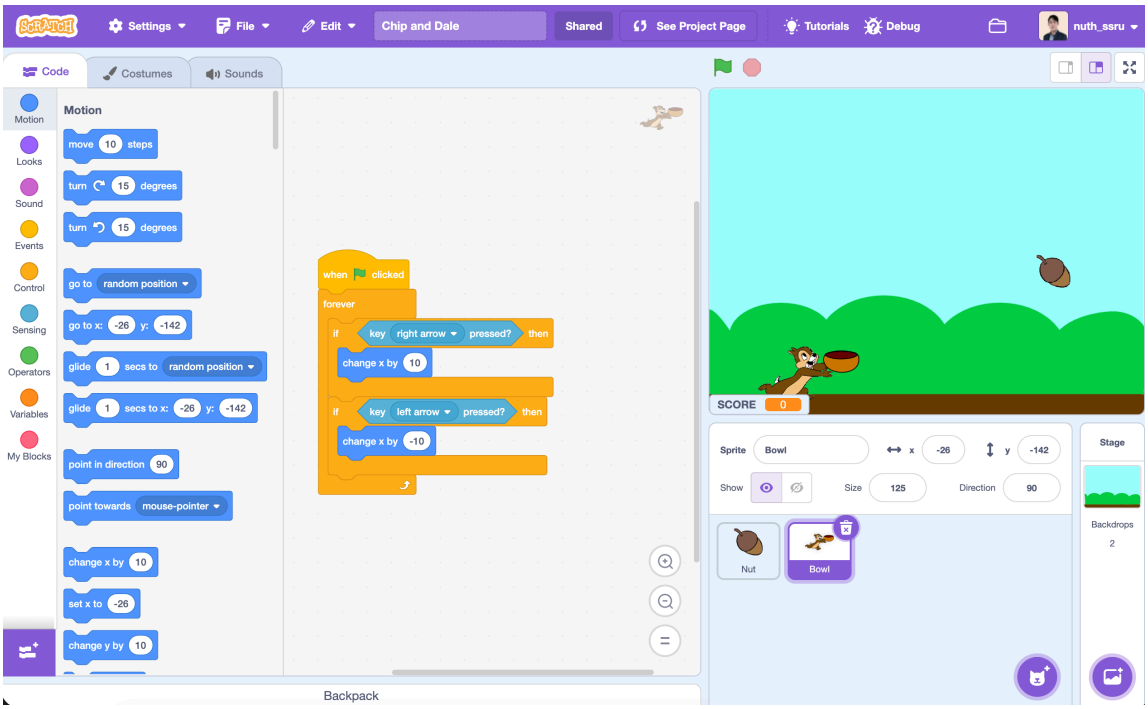

Settings
File
Edit
Train and Sensor Rails
Shared
See Project Page
Tutorials
Debug
nuth_ssru
Code
Costumes
Sounds
Motion
Looks
Sound
Events
Control
Sensing
Operators
Variables
My Blocks
move 10 steps
turn 15 degrees
turn 15 degrees
go to random position
go to x: 91 y: 89
glide 1 secs to random position
glide 1 secs to x: 91 y: 89
point in direction 90
point towards mouse-pointer
change x by 10
set x to 91
change y by 10
when clicked
point in direction 90
go to x: 91 y: 89
when I receive trainpass
go to x: 53 y: 108
point in direction 180
when I receive traingone
go to x: 91 y: 89
point in direction 90
Sprite
Sprite1
x 91
y 89
Show
Size 100
Direction 90
Stage
Backdrops 2
train
Sprite1
Sprite3
Sprite2
Backpack

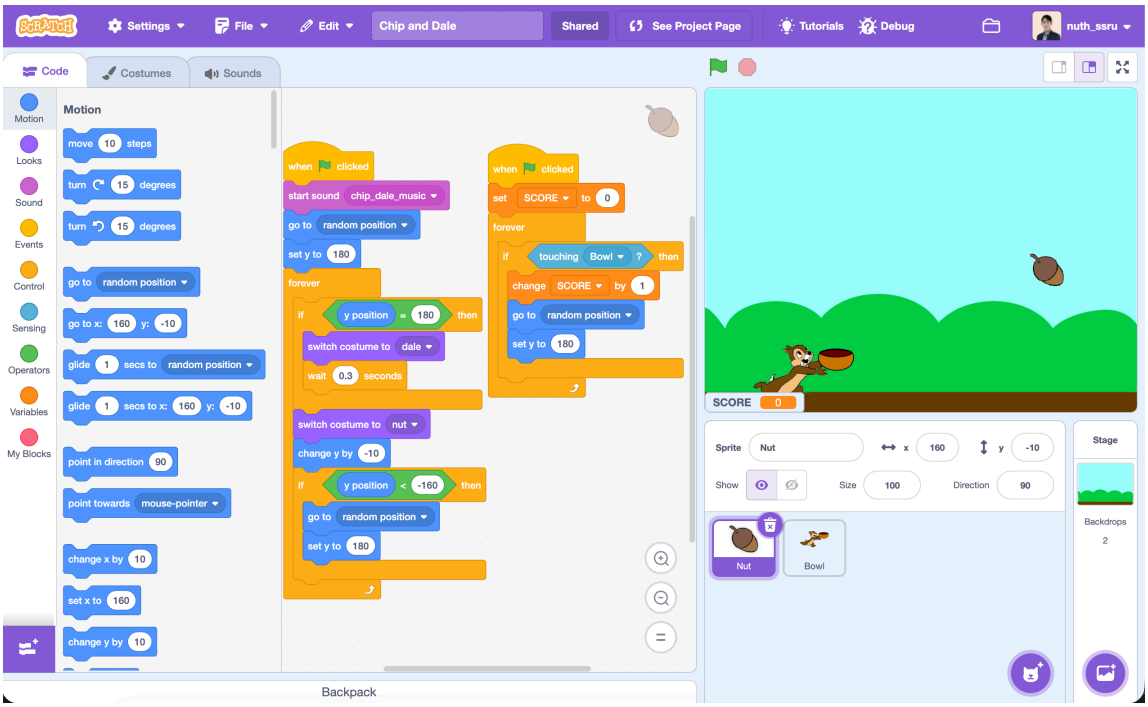

Settings
File
Edit
Train and Sensor Rails
Shared
See Project Page
Tutorials
Debug
nuth_ssru
Code
Costumes
Sounds
Motion
Looks
Sound
Events
Control
Sensing
Operators
Variables
My Blocks
move 10 steps
turn 15 degrees
turn 15 degrees
go to random position
go to x: -79 y: 23
glide 1 secs to random position
glide 1 secs to x: -79 y: 23
point in direction 90
point towards mouse-pointer
change x by 10
set x to -79
change y by 10
when clicked
forever
if touching train ? then
broadcast trainpass
Sprite
Sprite3
x -79
y 23
Show
Size 100
Direction 90
Stage
Backdrops 2
train
Sprite1
Sprite3
Sprite2
Backpack



4. เกมรับผลวอลนัท ดาวน์โหลดตัวอย่างการสาธิตได้ที่

https://scratch.mit.edu/projects/320818783





บทสรุปประจำบทที่ 7

การโปรแกรมแบบบล็อกคำสั่งเป็นแนวทางสำคัญในการเริ่มต้นเรียนรู้การเขียนโปรแกรมสำหรับผู้เรียนระดับประถมศึกษาและมัธยมศึกษา เพราะช่วยลดความซับซ้อนของไวยากรณ์ภาษาโปรแกรม และเปิดโอกาสให้ผู้เรียนมุ่งเน้นที่กระบวนการคิดเชิงตรรกะ การลำดับขั้นตอน และการแก้ปัญหาอย่างเป็นระบบ ในบทนี้ได้อธิบายแนวคิดพื้นฐานของการโปรแกรมแบบบล็อกคำสั่ง ส่วนประกอบและเครื่องมือของ Scratch ของ MIT การออกแบบลำดับคำสั่ง เงื่อนไข และการทำซ้ำ ตลอดจนการสร้างชิ้นงานหรือสื่อการเรียนรู้ด้วย Scratch

นอกจากนี้ ยังได้เสนอแนวทางการจัดกิจกรรมการเรียนรู้ด้วยการโปรแกรมแบบบล็อกคำสั่งสำหรับนักเรียน โดยเน้นการเชื่อมโยงกับชีวิตจริง การค่อย ๆ พัฒนาความซับซ้อนของเนื้อหา การเปิดโอกาสให้ผู้เรียนได้ลงมือสร้างผลงาน และการประเมินทั้งกระบวนการและผลลัพธ์ สำหรับนักศึกษาครูความเข้าใจเรื่องการโปรแกรมแบบบล็อกคำสั่งมิได้มีความสำคัญเฉพาะในฐานะผู้เรียนวิทยาการคำนวณเท่านั้น แต่ยังเป็นพื้นฐานของการออกแบบสื่อและกิจกรรมการเรียนรู้ที่เหมาะสมกับผู้เรียนในโรงเรียน ซึ่งจะช่วยให้การจัดการเรียนรู้วิทยาการคำนวณมีความหมาย สนุก และส่งเสริมทักษะแห่งอนาคตได้อย่างแท้จริง

คำถามท้ายบท

1. จงอธิบายความหมายของการโปรแกรมแบบบล็อกคำสั่ง และเหตุใดจึงเหมาะกับผู้เริ่มต้นเรียนเขียนโปรแกรม
2. Scratch ของ MIT มีจุดเด่นอย่างไรในการใช้เป็นเครื่องมือสำหรับการเรียนรู้การเขียนโปรแกรม
3. ส่วนประกอบสำคัญของ Scratch มีอะไรบ้าง และแต่ละส่วนมีหน้าที่อย่างไร
4. เพราะเหตุใดการออกแบบลำดับคำสั่งจึงมีความสำคัญต่อการสร้างชิ้นงานด้วย Scratch
5. การใช้เงื่อนไขและการทำซ้ำใน Scratch มีบทบาทอย่างไรต่อการสร้างโปรแกรมที่มีปฏิสัมพันธ์
6. จงยกตัวอย่างชิ้นงานที่สามารถสร้างได้ด้วย Scratch อย่างน้อย 3 ประเภท
7. หากท่านต้องการสร้างเกมตอบคำถามด้วย Scratch ท่านจะต้องใช้แนวคิดการเขียนโปรแกรมใดบ้าง
8. หากท่านเป็นครูระดับประถมศึกษา ท่านจะออกแบบกิจกรรม Scratch อย่างไรให้เหมาะกับผู้เรียน
9. หากท่านเป็นครูระดับมัธยมศึกษา ท่านจะใช้ Scratch เพื่อบูรณาการกับรายวิชาอื่นอย่างไร
10. จงอธิบายบทบาทของ Scratch ในการพัฒนาทักษะการคิดเชิงคำนวณและความคิดสร้างสรรค์ของผู้เรียน

เอกสารอ้างอิง

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

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 Annual Meeting of the American Educational Research Association*. Vancouver, Canada.

Maloney, J., Resnick, M., Rusk, N., Silverman, B., & Eastmond, E. (2010). The Scratch programming language and environment. *ACM Transactions on Computing Education, 10*(4), 1–15.

MIT Media Lab. (2024). *Scratch*. Retrieved from the official Scratch website.

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., ... Kafai, Y. (2009). Scratch: Programming for all. *Communications of the ACM, 52*(11), 60–67.

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

บทที่ 8
การโปรแกรมด้วยไมโครบิต
Programming with Micro:bit

บทนำ

ในยุคที่เทคโนโลยีดิจิทัลเข้ามามีบทบาทอย่างกว้างขวางต่อการดำรงชีวิตและการศึกษา การเรียนรู้ด้านวิทยาการคำนวณมิได้จำกัดอยู่เพียงการเขียนโปรแกรมบนหน้าจอคอมพิวเตอร์เท่านั้น แต่ยังครอบคลุมถึงการสั่งงานอุปกรณ์อัจฉริยะและระบบที่สามารถโต้ตอบกับสิ่งแวดล้อมได้จริง การเรียนรู้ในลักษณะนี้ช่วยให้ผู้เรียนเห็นความเชื่อมโยงระหว่างแนวคิดเชิงนามธรรมของโปรแกรมกับผลลัพธ์ที่เกิดขึ้นในโลกจริง ทำให้การเรียนรู้มีความหมาย มีความท้าทาย และเอื้อต่อการพัฒนาทักษะการคิดเชิงคำนวณ การแก้ปัญหา และความคิดสร้างสรรค์อย่างเป็นรูปธรรม

ไมโครบิต (micro:bit) เป็นอุปกรณ์สมองกลฝังตัวขนาดเล็กที่ได้รับการออกแบบเพื่อการศึกษาโดยเฉพาะ และได้รับความนิยมอย่างแพร่หลายในหลายประเทศ รวมถึงประเทศไทย เนื่องจากมีขนาดกะทัดรัด ใช้งานง่าย ราคาเหมาะสม และรองรับการเขียนโปรแกรมได้ทั้งในรูปแบบบล็อกคำสั่งและข้อความ ไมโครบิตช่วยให้ผู้เรียนสามารถสร้างชิ้นงานที่โต้ตอบกับแสง เสียง การเคลื่อนไหว อุณหภูมิ หรือการสัมผัสได้อย่างหลากหลาย จึงเป็นเครื่องมือสำคัญในการเชื่อมโยงการเรียนรู้ด้านการเขียนโปรแกรมเข้ากับโลกจริง

สำหรับนักศึกษาครู การเรียนรู้การโปรแกรมด้วยไมโครบิตมีความสำคัญในสองมิติสำคัญ มิติแรก คือ การเข้าใจหลักการทำงานของระบบสมองกลฝังตัว การเชื่อมโยงระหว่างฮาร์ดแวร์กับซอฟต์แวร์ และการออกแบบคำสั่งเพื่อควบคุมอุปกรณ์ มิติที่สอง คือ การสามารถนำไมโครบิตไปประยุกต์ใช้ในการจัดกิจกรรมการเรียนรู้และสร้างนวัตกรรมทางการศึกษาได้อย่างเหมาะสมกับผู้เรียนในระดับการศึกษาขั้นพื้นฐาน การเข้าใจทั้งตัวอุปกรณ์และแนวทางการสอนจึงเป็นสิ่งจำเป็นสำหรับครูในยุคดิจิทัล

บทนี้มุ่งอธิบายความรู้เบื้องต้นเกี่ยวกับไมโครบิตและระบบสมองกลฝังตัว องค์ประกอบและการทำงานของบอร์ดไมโครบิต การเขียนโปรแกรมควบคุมไมโครบิตเบื้องต้น การประยุกต์ใช้ไมโครบิตในการสร้างชิ้นงานและนวัตกรรมทางการศึกษา ตลอดจนการออกแบบกิจกรรมการเรียนรู้โดยใช้ไมโครบิตสำหรับผู้เรียนในระดับการศึกษาขั้นพื้นฐาน เพื่อให้นักศึกษาครูสามารถนำความรู้ไปประยุกต์ใช้ได้ทั้งในด้านวิชาการและในด้านการออกแบบการเรียนรู้

8.1 ความรู้เบื้องต้นเกี่ยวกับไมโครบิตและระบบสมองกลฝังตัว

ไมโครบิต (micro:bit) คือ บอร์ดไมโครคอนโทรลเลอร์ขนาดเล็กที่พัฒนาขึ้นเพื่อใช้เป็นเครื่องมือทางการศึกษา โดยมีเป้าหมายเพื่อส่งเสริมให้ผู้เรียนเข้าใจแนวคิดพื้นฐานของการเขียนโปรแกรม การทำงานของอุปกรณ์อิเล็กทรอนิกส์ และการประยุกต์ใช้เทคโนโลยีในการแก้ปัญหา ไมโครบิตมีขนาดเล็กพกพาสะดวก มีอุปกรณ์อินพุตและเอาต์พุตพื้นฐานติดตั้งอยู่บนบอร์ด และสามารถเขียนโปรแกรมควบคุมได้ง่าย จึงเหมาะกับการนำไปใช้ในชั้นเรียนตั้งแต่ระดับประถมศึกษาจนถึงมัธยมศึกษา

ไมโครบิตเป็นตัวอย่างของระบบสมองกลฝังตัว (Embedded System) ซึ่งหมายถึง ระบบคอมพิวเตอร์ขนาดเล็กที่ถูกฝังอยู่ในอุปกรณ์เพื่อควบคุมการทำงานเฉพาะอย่าง ระบบสมองกลฝังตัวพบได้ทั่วไปในชีวิตประจำวัน เช่น เครื่องซักผ้า เตาไมโครเวฟ เครื่องปรับอากาศ รถยนต์ สมาร์ตวอทช์ หรืออุปกรณ์อัตโนมัติต่าง ๆ จุดเด่นของระบบสมองกลฝังตัวคือ การออกแบบให้ทำงานเฉพาะด้าน มีประสิทธิภาพ ใช้พลังงานไม่มาก และสามารถตอบสนองต่อสัญญาณจากสิ่งแวดล้อมได้อย่างรวดเร็ว

ในเชิงแนวคิด การทำงานของระบบสมองกลฝังตัวมักอธิบายผ่านกระบวนการพื้นฐาน 3 ส่วน ได้แก่ การรับข้อมูล (Input) การประมวลผล (Process) และการแสดงผลหรือสั่งงาน (Output) ตัวอย่างเช่น หากไมโครบิตรับค่าจากปุ่มกดหรือเซนเซอร์ แล้วนำข้อมูลไปประมวลผลตามคำสั่งที่เขียนไว้ จากนั้นแสดงผลผ่านไฟ LED หรือส่งสัญญาณเสียงออกมา กระบวนการดังกล่าวสะท้อนหลักการทำงานพื้นฐานของระบบสมองกลฝังตัวอย่างชัดเจน

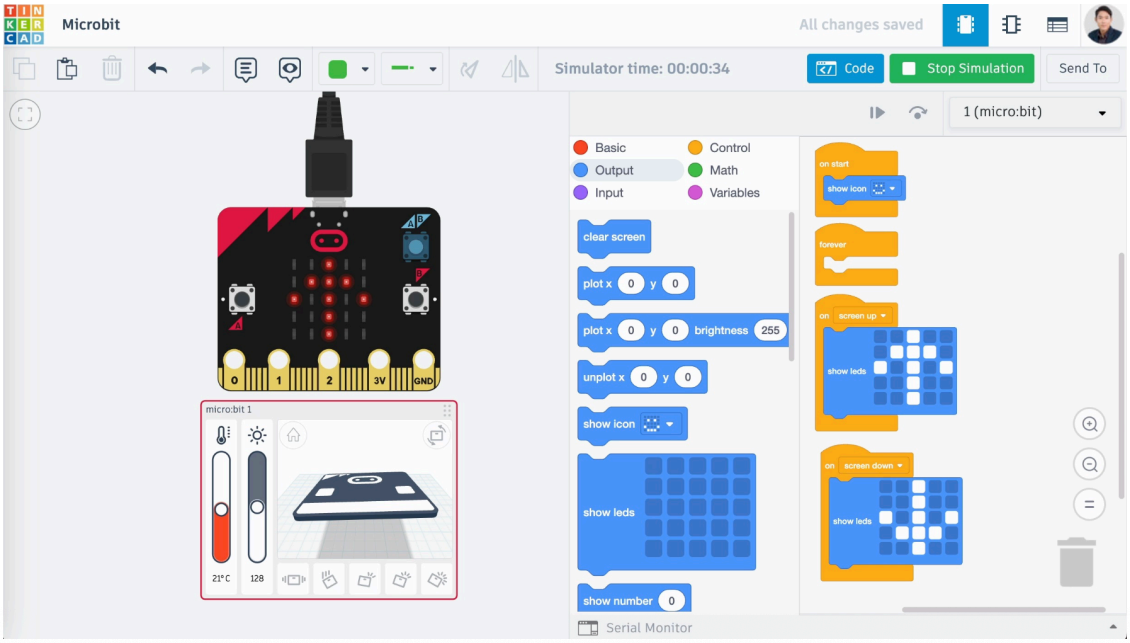
ไมโครบิตมีความเหมาะสมอย่างยิ่งต่อการเรียนรู้วิทยาการคำนวณ เพราะช่วยให้ผู้เรียนเชื่อมโยงการเขียนโปรแกรมกับการทำงานจริงของอุปกรณ์ ผู้เรียนมิได้เพียงเห็นผลลัพธ์บนหน้าจอ แต่สามารถสังเกตการตอบสนองของอุปกรณ์ เช่น การติดของไฟ LED การแสดงตัวอักษร การตรวจจับการเอียง การวัดอุณหภูมิ หรือการสื่อสารกับอุปกรณ์อื่น สิ่งเหล่านี้ช่วยให้แนวคิดเรื่องอัลกอริทึม ตัวแปร เหตุการณ์เงื่อนไข และการทำซ้ำ กลายเป็นสิ่งที่สัมผัสได้จริง

สำหรับนักศึกษาครู การเรียนรู้ไมโครบิตช่วยเปิดมุมมองใหม่ของการจัดการเรียนรู้วิทยาการคำนวณ กล่าวคือ การสอนเขียนโปรแกรมไม่จำเป็นต้องจำกัดอยู่แค่การสร้างภาพเคลื่อนไหวหรือเกมบนหน้าจอเท่านั้น แต่สามารถพัฒนาไปสู่การสร้างชิ้นงานที่มีปฏิสัมพันธ์กับโลกจริง เช่น เครื่องนับก้าว เกมตอบสนองต่อการเอียง ระบบเตือนภัยอย่างง่าย หรือสื่อการเรียนรู้เชิงโต้ตอบ ซึ่งล้วนช่วยให้ผู้เรียนเกิดความสนใจและเข้าใจการทำงานของเทคโนโลยีอย่างลึกซึ้งมากขึ้น

กล่าวโดยสรุป ไมโครบิตเป็นเครื่องมือทางการศึกษาที่ช่วยให้ผู้เรียนเข้าใจทั้งการเขียนโปรแกรมและการทำงานของระบบสมองกลฝังตัวอย่างเป็นรูปธรรม เป็นสะพานเชื่อมสำคัญระหว่างแนวคิดเชิงคำนวณกับการประยุกต์ใช้เทคโนโลยีในชีวิตจริง

8.2 องค์ประกอบและการทำงานของบอร์ดไมโครบิต

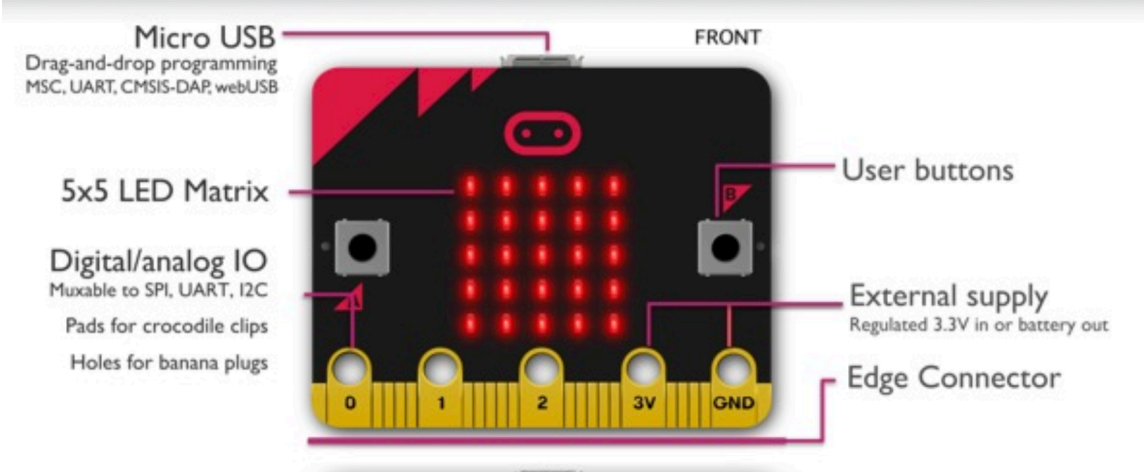
การทำความเข้าใจองค์ประกอบของบอร์ดไมโครบิตเป็นพื้นฐานสำคัญสำหรับการเขียนโปรแกรมและการสร้างชิ้นงาน เนื่องจากผู้เรียนจำเป็นต้องรู้ว่าอุปกรณ์แต่ละส่วนทำหน้าที่อะไร และสามารถนำไปใช้ในการออกแบบกิจกรรมหรือโครงงานได้อย่างไร บอร์ดไมโครบิตถูกออกแบบให้มีส่วนประกอบที่เหมาะสมกับการเรียนรู้ในระดับเริ่มต้น แต่ก็มีศักยภาพเพียงพอสำหรับการพัฒนาชิ้นงานที่สร้างสรรค์และหลากหลาย



องค์ประกอบสำคัญของบอร์ดไมโครบิตสามารถอธิบายได้ดังนี้

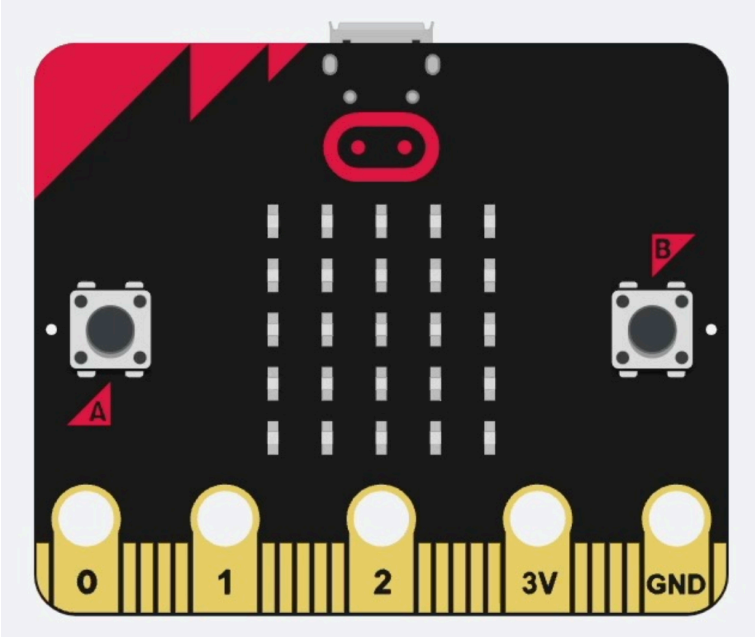
8.2.1 ไฟ LED แบบตาราง 5x5

ไมโครบิตมีไฟ LED จำนวน 25 ดวง จัดเรียงในรูปแบบตาราง 5x5 ซึ่งใช้สำหรับแสดงผลในรูปของตัวอักษร ตัวเลข รูปภาพ ไอคอน หรือแอนิเมชันอย่างง่าย ไฟ LED เป็นส่วนสำคัญที่ช่วยให้ผู้เรียนเห็นผลลัพธ์ของโปรแกรมได้ทันที เช่น การแสดงรูปหัวใจ การนับเลข หรือการแสดงข้อความเลื่อน



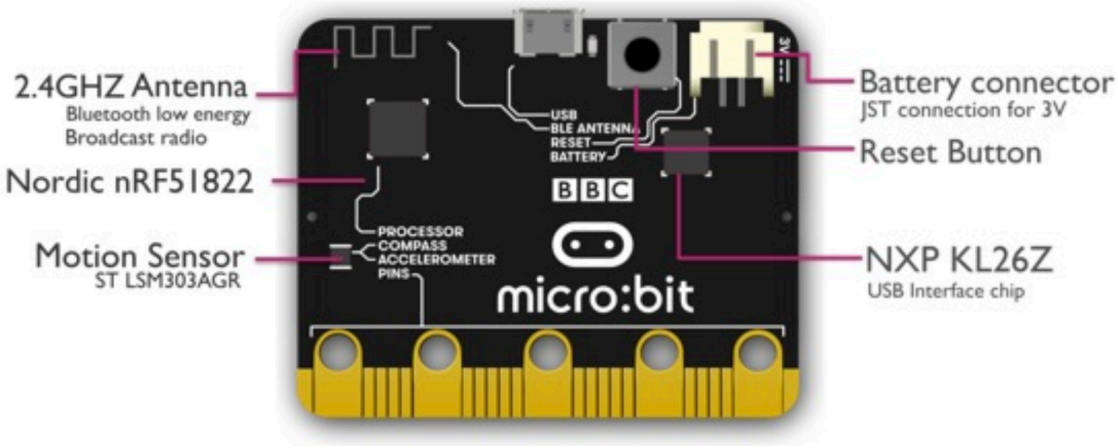
8.2.2 ปุ่มกด A และ B

บอร์ดไมโครบิตมีปุ่มกด 2 ปุ่มหลัก คือ ปุ่ม A และปุ่ม B ซึ่งสามารถใช้เป็นอินพุตให้ผู้เรียนออกแบบกิจกรรมหรือโปรแกรมโต้ตอบได้ เช่น เมื่อกดปุ่ม A ให้แสดงตัวเลข เมื่อกดปุ่ม B ให้เปลี่ยนรูปภาพ หรือเมื่อกดพร้อมกันให้เริ่มเกม ปุ่มกดเหล่านี้ช่วยให้ผู้เรียนเข้าใจแนวคิดเรื่องเหตุการณ์และการตอบสนองของโปรแกรม



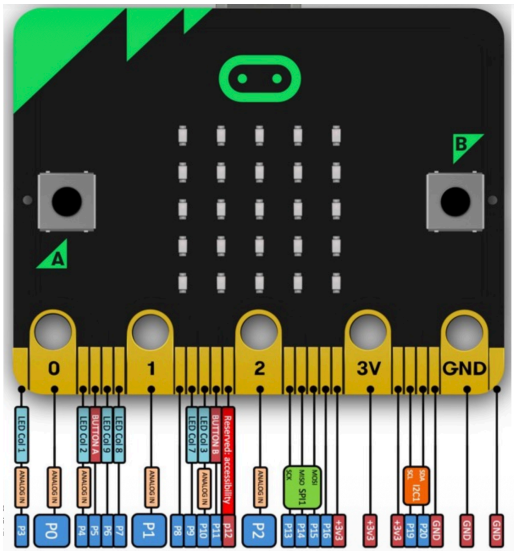
8.2.3 เซนเซอร์และอุปกรณ์ตรวจจับ

ไมโครบิตมีเซนเซอร์หลายประเภทในตัว เช่น เซนเซอร์ตรวจจับความเร่ง (Accelerometer) เซนเซอร์ตรวจจับสนามแม่เหล็ก (Compass) และเซนเซอร์วัดอุณหภูมิหรือแสงในบางฟังก์ชันของระบบ เซนเซอร์เหล่านี้ช่วยให้ไมโครบิตสามารถรับข้อมูลจากสิ่งแวดล้อมและตอบสนองได้ เช่น ตรวจจับการเอียง การเขย่า หรือการหันทิศทาง



8.2.4 ขั้วต่อและพอร์ตเชื่อมต่อ

ไมโครบิตมีขั้วต่อสำหรับเชื่อมต่อกับอุปกรณ์ภายนอก เช่น สายไฟ เซนเซอร์เพิ่มเติม หลอดไฟ มอเตอร์ หรืออุปกรณ์อิเล็กทรอนิกส์อื่น ๆ การมีขั้วต่อเหล่านี้ทำให้ผู้เรียนสามารถขยายศักยภาพของไมโครบิตจากการใช้งานพื้นฐานไปสู่การสร้างโครงงานเชิงประดิษฐ์ที่ซับซ้อนขึ้น



8.2.5 หน่วยประมวลผลและระบบสื่อสาร

ภายในบอร์ดไมโครบิตมีไมโครคอนโทรลเลอร์ทำหน้าที่เป็นหน่วยประมวลผลหลักของอุปกรณ์ คอยรับข้อมูล ประมวลผลตามโปรแกรม และสั่งงานส่วนต่าง ๆ ของบอร์ด นอกจากนี้ ไมโครบิตยังสามารถสื่อสารกับอุปกรณ์อื่นผ่านเทคโนโลยี เช่น Bluetooth หรือการส่งสัญญาณวิทยุอย่างง่าย ซึ่งช่วยเปิดโอกาสให้เกิดกิจกรรมการเรียนรู้ที่มีปฏิสัมพันธ์ระหว่างอุปกรณ์หลายชิ้น

ในด้านการทำงานของบอร์ดไมโครบิต สามารถอธิบายตามหลัก Input–Process–Output ได้อย่างชัดเจน กล่าวคือ อุปกรณ์รับข้อมูลจากปุ่มกดหรือเซนเซอร์ จากนั้นหน่วยประมวลผลจะทำงานตามโปรแกรมที่ผู้เรียนเขียนไว้ และส่งผลลัพธ์ออกทางไฟ LED เสียง หรืออุปกรณ์ภายนอก การเข้าใจวงจรการทำงานเช่นนี้จะช่วยให้ผู้เรียนมองเห็นภาพรวมของระบบและเข้าใจความสัมพันธ์ระหว่างฮาร์ดแวร์กับซอฟต์แวร์

สำหรับนักศึกษาครู การเข้าใจองค์ประกอบของไมโครบิตมีความสำคัญอย่างยิ่ง เพราะจะช่วยให้สามารถออกแบบกิจกรรมที่ใช้ศักยภาพของบอร์ดได้อย่างเหมาะสม เช่น หากต้องการสอนเรื่องเหตุการณ์อาจเน้นใช้ปุ่มกด หากต้องการสอนเรื่องข้อมูลจากสิ่งแวดล้อมอาจใช้เซนเซอร์ หรือหากต้องการสร้างสื่อแสดงผลอย่างง่ายอาจใช้ไฟ LED และเสียง การรู้จักส่วนประกอบของบอร์ดอย่างเป็นระบบจึงเป็นพื้นฐานสำคัญของการจัดการเรียนรู้ด้วยไมโครบิต

กล่าวโดยสรุป บอร์ดไมโครบิตประกอบด้วยส่วนต่าง ๆ ที่ออกแบบมาเพื่อสนับสนุนการเรียนรู้ด้านการเขียนโปรแกรมและระบบสมองกลฝังตัวอย่างเป็นรูปธรรม การเข้าใจองค์ประกอบเหล่านี้จะช่วยให้ผู้เรียนและครูสามารถใช้งานไมโครบิตได้อย่างมีประสิทธิภาพและสร้างสรรค์

8.3 การเขียนโปรแกรมควบคุมไมโครบิตเบื้องต้น

การเขียนโปรแกรมควบคุมไมโครบิตเบื้องต้นเป็นขั้นตอนสำคัญที่ช่วยให้ผู้เรียนเชื่อมโยงความรู้ด้านวิทยาการคำนวณกับการทำงานของอุปกรณ์จริง โดยทั่วไป ผู้เรียนสามารถเขียนโปรแกรมไมโครบิตผ่านแพลตฟอร์มที่ใช้งานง่าย เช่น Microsoft MakeCode ซึ่งรองรับทั้งการเขียนโปรแกรมแบบบล็อกคำสั่งและแบบข้อความ สำหรับผู้เริ่มต้น การเขียนด้วยบล็อกคำสั่งมีความเหมาะสมอย่างยิ่ง เพราะช่วยลดความซับซ้อนด้านไวยากรณ์และเปิดโอกาสให้มุ่งเน้นที่ตรรกะของการควบคุมอุปกรณ์เป็นหลัก

แนวคิดพื้นฐานในการเขียนโปรแกรมไมโครบิตประกอบด้วยการกำหนดเหตุการณ์ การสั่งงานอุปกรณ์ การใช้เงื่อนไข การทำซ้ำ และการรับข้อมูลจากเซนเซอร์หรือปุ่มกด โปรแกรมเบื้องต้นมักเริ่มจากคำสั่งง่าย ๆ เช่น แสดงรูปหัวใจบนไฟ LED เมื่อเริ่มต้นโปรแกรม แสดงข้อความเมื่อกดปุ่ม A หรือ

เปิดไฟแสดงผลเมื่อเขย่าอุปกรณ์ กิจกรรมเหล่านี้ช่วยให้ผู้เรียนเห็นความสัมพันธ์ระหว่างคำสั่งกับผลลัพธ์อย่างชัดเจน

ตัวอย่างพื้นฐานของการเขียนโปรแกรมไมโครบิต ได้แก่

1) การแสดงข้อความหรือภาพบน LED

ผู้เรียนสามารถใช้คำสั่งให้ไมโครบิตแสดงตัวเลข ตัวอักษร หรือรูปภาพ เช่น แสดงคำว่า “HELLO” หรือแสดงรูปหัวใจ ซึ่งช่วยให้เข้าใจการสั่งงานเอาต์พุตอย่างง่าย

2) การใช้ปุ่มกดเพื่อสั่งงาน

ผู้เรียนสามารถกำหนดว่า เมื่อกดปุ่ม A ให้แสดงเลข 1 และเมื่อกดปุ่ม B ให้แสดงเลข 2 กิจกรรมเช่นนี้ช่วยให้เข้าใจแนวคิดเรื่องเหตุการณ์ (Event) และการโต้ตอบระหว่างผู้ใช้กับอุปกรณ์

3) การใช้คำสั่งทำซ้ำ

ผู้เรียนสามารถสร้างคำสั่งให้ไฟ LED กระพริบซ้ำ ๆ หรือให้แสดงภาพเคลื่อนไหวอย่างต่อเนื่อง ซึ่งช่วยให้เข้าใจแนวคิดเรื่องลูปและการควบคุมการทำงานที่เกิดขึ้นต่อเนื่อง

4) การใช้เงื่อนไข

ผู้เรียนอาจสร้างโปรแกรมให้แสดงข้อความต่างกันเมื่ออุณหภูมิสูงหรือต่ำกว่าค่าที่กำหนด หรือให้แสดงรูปภาพต่างกันเมื่ออุปกรณ์ถูกเอียงไปคนละด้าน การเรียนรู้เช่นนี้ช่วยพัฒนาทักษะการตัดสินใจและการตอบสนองต่อข้อมูลจากสิ่งแวดล้อม

5) การใช้ตัวแปรอย่างง่าย

ผู้เรียนสามารถสร้างตัวแปรเพื่อเก็บข้อมูล เช่น คะแนน จำนวนครั้ง หรือค่าที่ใช้ในการนับ การใช้ตัวแปรช่วยให้ผู้เรียนเข้าใจว่าข้อมูลสามารถถูกเก็บและเปลี่ยนแปลงระหว่างการทำงานของโปรแกรมได้

ในกระบวนการเรียนรู้ การทดลองเขียนโปรแกรมควบคุมไมโครบิตควรมาพร้อมกับการทดสอบและการสะท้อนผล ผู้เรียนควรได้สังเกตว่าคำสั่งที่เขียนส่งผลอย่างไรต่อการทำงานของอุปกรณ์ และหากโปรแกรมไม่ทำงานตามที่คาดไว้ ควรตรวจสอบที่จุดใดบ้าง เช่น เหตุการณ์ไม่ตรง คำสั่งซ้อนกันผิดลำดับ หรือเงื่อนไขไม่ครอบคลุม กระบวนการนี้จะช่วยพัฒนาทักษะการแก้ปัญหาและการคิดเชิงระบบ

สำหรับนักศึกษาครู การเขียนโปรแกรมไมโครบิตเบื้องต้นควรถูกมองทั้งในฐานะทักษะทางเทคนิคและในฐานะเครื่องมือสำหรับการออกแบบการเรียนรู้ กล่าวคือ นักศึกษาครูควรฝึกคิดว่า

โปรแกรมอย่างง่ายที่เขียนขึ้นนั้นสามารถนำไปใช้เป็นกิจกรรมในชั้นเรียนได้อย่างไร เช่น ใช้สอนเรื่องเหตุการณ์ ใช้สอนเรื่องข้อมูลจากเซนเซอร์ หรือใช้เป็นสื่อสร้างแรงจูงใจให้ผู้เรียนสนใจการเขียนโปรแกรมมากขึ้น

กล่าวโดยสรุป การเขียนโปรแกรมควบคุมไมโครบิตเบื้องต้นเป็นประสบการณ์การเรียนรู้ที่ช่วยให้ผู้เรียนเห็นผลของการเขียนโปรแกรมในโลกจริงอย่างเป็นรูปธรรม และเป็นพื้นฐานสำคัญในการต่อยอดไปสู่การสร้างชิ้นงานและนวัตกรรมที่ซับซ้อนยิ่งขึ้น

8.4 การประยุกต์ใช้ไมโครบิตในการสร้างชิ้นงานและนวัตกรรมทางการศึกษา

ไมโครบิตเป็นเครื่องมือที่มีศักยภาพสูงในการนำไปสร้างชิ้นงานและนวัตกรรมทางการศึกษา เนื่องจากสามารถเชื่อมโยงการเขียนโปรแกรมกับการทำงานของอุปกรณ์จริงได้อย่างหลากหลาย ผู้เรียนจึงไม่เพียงแต่เรียนรู้วิธีสั่งงานอุปกรณ์ แต่ยังสามารถออกแบบและสร้างผลงานที่ตอบโจทย์การเรียนรู้ การแก้ปัญหา หรือการสร้างประสบการณ์ใหม่ในชั้นเรียนได้อย่างสร้างสรรค์

การประยุกต์ใช้ไมโครบิตในการสร้างชิ้นงานสามารถทำได้หลายรูปแบบ เช่น

1) สื่อการเรียนรู้เชิงโต้ตอบ

ไมโครบิตสามารถใช้สร้างสื่อที่ผู้เรียนมีปฏิสัมพันธ์ได้ เช่น ป้ายคำศัพท์ที่กดแล้วแสดงคำตอบ เกมตอบคำถามอย่างง่าย หรือสื่อแสดงภาพและเสียงประกอบการเรียนรู้ สื่อประเภทนี้ช่วยเพิ่มความน่าสนใจและทำให้ผู้เรียนมีส่วนร่วมมากขึ้น

2) เกมหรือกิจกรรมส่งเสริมการเรียนรู้

ผู้เรียนสามารถใช้ไมโครบิตสร้างเกมตอบสนองต่อการกดปุ่ม การเอียง หรือการเขย่า เช่น เกมนับคะแนน เกมทดสอบปฏิกิริยาตอบสนอง หรือเกมความรู้ในรายวิชาต่าง ๆ กิจกรรมลักษณะนี้ช่วยส่งเสริมการเรียนรู้ผ่านการเล่นและกระตุ้นแรงจูงใจของผู้เรียน

3) โครงงานด้านสิ่งแวดล้อมและวิทยาศาสตร์

ไมโครบิตสามารถใช้ร่วมกับเซนเซอร์เพื่อตรวจวัดข้อมูลจากสิ่งแวดล้อม เช่น อุณหภูมิ แสง หรือการเคลื่อนไหว ทำให้สามารถนำไปสร้างโครงงานด้านวิทยาศาสตร์หรือสิ่งแวดล้อม เช่น เครื่องวัดอุณหภูมิอย่างง่าย ระบบเตือนเมื่อแสงน้อย หรือเครื่องนับก้าวเดิน

4) นวัตกรรมเพื่อแก้ปัญหาในชั้นเรียนหรือชีวิตประจำวัน

ผู้เรียนสามารถออกแบบชิ้นงานที่ตอบโจทย์ปัญหาจริง เช่น เครื่องเตือนเวลาเข้าแถว ป้ายแจ้ง

เตือนระดับเสียงในห้องเรียน หรืออุปกรณ์เตือนการลืมสิ่งของ การสร้างชิ้นงานเช่นนี้ช่วยให้ผู้เรียนเห็นคุณค่าของการเขียนโปรแกรมในฐานะเครื่องมือแก้ปัญหา

ในบริบททางการศึกษา การใช้ไมโครบิตสร้างชิ้นงานมิได้มีเป้าหมายเพียงเพื่อให้ได้ผลงานที่น่าสนใจเท่านั้น แต่ยังมีคุณค่าในเชิงกระบวนการเรียนรู้ เพราะผู้เรียนต้องผ่านขั้นตอนของการวิเคราะห์ปัญหา การออกแบบแนวทางแก้ไข การเลือกใช้ส่วนประกอบของอุปกรณ์ การเขียนโปรแกรม การทดสอบ และการปรับปรุงผลงาน กระบวนการเช่นนี้ช่วยพัฒนาทักษะหลายด้านพร้อมกัน ทั้งการคิดเชิงคำนวณ การออกแบบเชิงวิศวกรรม ความคิดสร้างสรรค์ การทำงานร่วมกัน และความรับผิดชอบต่อผลงาน

สำหรับนักศึกษาครู การเรียนรู้การประยุกต์ใช้ไมโครบิตในการสร้างชิ้นงานมีความสำคัญมาก เพราะช่วยให้สามารถมองเห็นศักยภาพของไมโครบิตในการบูรณาการกับรายวิชาอื่น เช่น คณิตศาสตร์ วิทยาศาสตร์ ภาษาไทย ภาษาอังกฤษ หรือการงานอาชีพ นอกจากนี้ ยังช่วยให้สามารถออกแบบกิจกรรมที่มีลักษณะเป็นโครงงานหรือการเรียนรู้แบบลงมือปฏิบัติ ซึ่งสอดคล้องกับแนวทางการจัดการเรียนรู้ในศตวรรษที่ 21

กล่าวโดยสรุป การประยุกต์ใช้ไมโครบิตในการสร้างชิ้นงานและนวัตกรรมทางการศึกษาเป็นแนวทางที่ช่วยเปลี่ยนการเรียนรู้การเขียนโปรแกรมจากการฝึกคำสั่งไปสู่การสร้างสรรค์ผลงานที่มีความหมาย ช่วยให้ผู้เรียนเห็นคุณค่าของเทคโนโลยีในการแก้ปัญหาและสร้างประโยชน์ในบริบทจริง

8.5 การออกแบบกิจกรรมการเรียนรู้โดยใช้ไมโครบิตสำหรับผู้เรียนในระดับการศึกษาขั้นพื้นฐาน

การออกแบบกิจกรรมการเรียนรู้โดยใช้ไมโครบิตสำหรับผู้เรียนในระดับการศึกษาขั้นพื้นฐานควรคำนึงถึงพัฒนาการของผู้เรียน ความพร้อมด้านอุปกรณ์ เวลาเรียน และจุดประสงค์ของบทเรียนเป็นสำคัญ แม้ไมโครบิตจะเป็นอุปกรณ์ที่ทรงพลัง แต่การทำให้ผู้เรียนเกิดความเข้าใจอย่างแท้จริงจำเป็นต้องอาศัยกิจกรรมที่เชื่อมโยงกับชีวิตจริง เหมาะกับวัย และเปิดโอกาสให้ผู้เรียนได้ลงมือปฏิบัติอย่างมีความหมาย

หลักการสำคัญประการแรก คือ ควรเริ่มจากกิจกรรมง่าย ๆ ที่ผู้เรียนเห็นผลได้รวดเร็ว เช่น การแสดงรูปภาพบนไฟ LED การใช้ปุ่ม A และ B เพื่อเปลี่ยนผลลัพธ์ หรือการให้ไมโครบิตแสดงข้อความเมื่อเขย่าอุปกรณ์ กิจกรรมลักษณะนี้ช่วยสร้างความมั่นใจและทำให้ผู้เรียนรู้สึกสนุกกับการเขียนโปรแกรมตั้งแต่เริ่มต้น

หลักการประการที่สอง คือ ควรค่อย ๆ พัฒนาความซับซ้อนของกิจกรรมจากการใช้คำสั่งพื้นฐานไปสู่การใช้เงื่อนไข การทำซ้ำ ตัวแปร และเซนเซอร์ เช่น เริ่มจากการสั่งแสดงรูปภาพ ต่อด้วยการสร้างเกมกดปุ่มอย่างง่าย จากนั้นจึงพัฒนาไปสู่กิจกรรมที่ใช้ข้อมูลจากการเอียงหรือการวัดอุณหภูมิ การจัดลำดับเช่นนี้จะช่วยลดความสับสนและทำให้ผู้เรียนสร้างความเข้าใจได้เป็นขั้นตอน

หลักการประการที่สาม คือ ควรเชื่อมโยงกิจกรรมกับปัญหาหรือบริบทจริง เช่น การสร้างเครื่องเตือนเสียงดังในห้องเรียน การสร้างป้ายแจ้งเตือนอุณหภูมิ หรือการทำเกมคำศัพท์ กิจกรรมที่มีความหมายต่อชีวิตจริงจะช่วยให้ผู้เรียนเห็นว่าไมโครบิตไม่ใช่เพียงอุปกรณ์สำหรับทดลอง แต่เป็นเครื่องมือสำหรับการสร้างสรรค์และการแก้ปัญหา

ในระดับประถมศึกษา กิจกรรมควรมีลักษณะเรียบง่าย ใช้เวลาสั้น เน้นการมองเห็นผลลัพธ์ชัดเจน และเชื่อมโยงกับสิ่งที่เด็กสนใจ เช่น การทำป้ายชื่อกระพริบ เกมกดปุ่มแสดงอารมณ์ หรือสื่อแสดงรูปสัตว์ การเรียนรู้ควรเน้นการสำรวจ ทดลอง และสนุกกับการค้นพบ

ในระดับมัธยมศึกษา กิจกรรมสามารถซับซ้อนขึ้น เช่น การใช้ตัวแปรนับคะแนน การใช้เซนเซอร์เอียงเพื่อควบคุมเกม การสร้างระบบแจ้งเตือนอย่างง่าย หรือการทำโครงงานขนาดเล็กที่บูรณาการกับรายวิชาอื่น การเรียนรู้ในระดับนี้ควรเปิดพื้นที่ให้ผู้เรียนวางแผน แก้ปัญหา และทำงานร่วมกันมากขึ้น

สำหรับนักศึกษาครู การออกแบบกิจกรรมด้วยไมโครบิตควรเชื่อมโยงกับการจัดทำแผนการจัดการเรียนรู้ เช่น การกำหนดผลลัพธ์การเรียนรู้ การเลือกสื่อและอุปกรณ์ การเขียนขั้นตอนกิจกรรม การเตรียมคำถามกระตุ้นการคิด และการกำหนดวิธีประเมินผลที่ครอบคลุมทั้งความรู้ ทักษะ และผลงานของผู้เรียน ครูควรคำนึงด้วยว่า การใช้อุปกรณ์จริงในห้องเรียนอาจมีข้อจำกัด เช่น จำนวนอุปกรณ์ไม่เพียงพอ เวลาเรียนจำกัด หรือพื้นฐานของผู้เรียนไม่เท่ากัน จึงควรออกแบบกิจกรรมให้ยืดหยุ่นและเหมาะสมกับบริบท

ในด้านการประเมินผล ครูอาจใช้การประเมินจากการสังเกตการมีส่วนร่วม การตรวจโปรแกรมหรือชิ้นงาน การให้ผู้เรียนอธิบายวิธีการทำงานของผลงาน และการใช้แบบสะท้อนคิดเพื่อดูว่าผู้เรียนได้เรียนรู้อะไรและพบปัญหาใดบ้าง การประเมินลักษณะนี้ช่วยให้เห็นพัฒนาการของผู้เรียนอย่างรอบด้านมากกว่าการดูผลลัพธ์สุดท้ายเพียงอย่างเดียว

กล่าวโดยสรุป การออกแบบกิจกรรมการเรียนรู้โดยใช้ไมโครบิตควรมุ่งให้ผู้เรียนได้เรียนรู้ผ่านการลงมือทำ การทดลอง และการเชื่อมโยงกับสถานการณ์จริง โดยค่อย ๆ พัฒนาจากกิจกรรมพื้นฐานไปสู่กิจกรรมสร้างสรรค์ที่ซับซ้อนขึ้น ทั้งนี้ นักศึกษาครูควรได้รับการฝึกให้สามารถออกแบบกิจกรรมที่เหมาะสมกับวัยและบริบท เพื่อให้ไมโครบิตเป็นเครื่องมือที่ส่งเสริมการเรียนรู้ได้อย่างแท้จริง

บทสรุปประจำบทที่ 8

ไมโครบิตเป็นเครื่องมือทางการศึกษาที่มีบทบาทสำคัญในการเชื่อมโยงการเรียนรู้ด้านการเขียนโปรแกรมกับการทำงานของระบบสมองกลฝังตัวในโลกจริง ผู้เรียนสามารถใช้ไมโครบิตเพื่อเรียนรู้หลักการพื้นฐานของการรับข้อมูล การประมวลผล และการแสดงผล ตลอดจนการใช้เหตุการณ์ เงื่อนไข การทำซ้ำ และตัวแปรในการควบคุมการทำงานของอุปกรณ์ บทนี้ได้อธิบายความรู้เบื้องต้นเกี่ยวกับไมโครบิต องค์ประกอบของบอร์ด การเขียนโปรแกรมควบคุมเบื้องต้น การประยุกต์ใช้เพื่อสร้างชิ้นงานและนวัตกรรมทางการศึกษา รวมถึงแนวทางการออกแบบกิจกรรมการเรียนรู้สำหรับผู้เรียนในระดับการศึกษาขั้นพื้นฐาน

สำหรับนักศึกษาครู การเรียนรู้ไมโครบิตมีความสำคัญทั้งในฐานะผู้เรียนวิทยาการคำนวณและในฐานะผู้สอนในอนาคต เพราะไมโครบิตเป็นเครื่องมือที่ช่วยให้ครูสามารถออกแบบกิจกรรมที่ส่งเสริมการคิดเชิงคำนวณ การแก้ปัญหา ความคิดสร้างสรรค์ และการลงมือปฏิบัติได้อย่างเป็นรูปธรรม เมื่อใช้ร่วมกับการออกแบบการเรียนรู้ที่เหมาะสม ไมโครบิตจะไม่เป็นเพียงอุปกรณ์อิเล็กทรอนิกส์ขนาดเล็ก แต่จะเป็นสื่อสำคัญในการเปิดโอกาสให้ผู้เรียนสร้างองค์ความรู้และนวัตกรรมด้วยตนเอง

คำถามท้ายบท

1. จงอธิบายความหมายของไมโครบิตและระบบสมองกลฝังตัว พร้อมยกตัวอย่างอุปกรณ์ในชีวิตประจำวัน
2. เพราะเหตุใดไมโครบิตจึงเหมาะสำหรับใช้เป็นเครื่องมือทางการศึกษาในการเรียนรู้วิทยาการคำนวณ
3. องค์ประกอบสำคัญของบอร์ดไมโครบิตมีอะไรบ้าง และแต่ละส่วนมีหน้าที่อย่างไร
4. จงอธิบายหลักการทำงานของไมโครบิตตามแนวคิด Input–Process–Output
5. การเขียนโปรแกรมควบคุมไมโครบิตเบื้องต้นควรเริ่มจากแนวคิดใดบ้าง
6. จงยกตัวอย่างชิ้นงานหรือสื่อการเรียนรู้ที่สามารถสร้างได้ด้วยไมโครบิตอย่างน้อย 3 ตัวอย่าง
7. การใช้ไมโครบิตสามารถช่วยส่งเสริมทักษะใดของผู้เรียนได้บ้าง
8. หากท่านเป็นครูระดับประถมศึกษา ท่านจะออกแบบกิจกรรมไมโครบิตอย่างไรให้เหมาะกับวัยของผู้เรียน
9. หากท่านเป็นครูระดับมัธยมศึกษา ท่านจะใช้ไมโครบิตบูรณาการกับรายวิชาอื่นอย่างไร

10. จงอธิบายบทบาทของไมโครบิตในการพัฒนานวัตกรรมทางการศึกษาและการเรียนรู้เชิงสร้างสรรค์

เอกสารอ้างอิง

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

BBC micro:bit Educational Foundation. (2024). *Micro:bit*. Retrieved from the official micro:bit website.

Sentance, S., Waite, J., Hodges, S., MacLeod, E., & Yeomans, L. (2017). “Creating cool stuff”: Pupils’ experience of the BBC micro:bit. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education* (pp. 531–536).

Microsoft. (2024). *MakeCode for micro:bit*. Retrieved from the official MakeCode website.

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

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 Annual Meeting of the American Educational Research Association*. Vancouver, Canada.

บทที่ 9
การโปรแกรมภาษาไพทอนเบื้องต้น
Introduction to Python Programming

บทนำ

ภาษาไพทอน (Python) เป็นหนึ่งในภาษาโปรแกรมที่ได้รับความนิยมอย่างแพร่หลายในระดับโลก ทั้งในด้านการศึกษา การพัฒนาโปรแกรม การวิเคราะห์ข้อมูล ปัญญาประดิษฐ์ ระบบอัตโนมัติ และงานประยุกต์อื่น ๆ อีกมากมาย สาเหตุสำคัญที่ทำให้ภาษาไพทอนได้รับความนิยม คือ การมีรูปแบบคำสั่งที่อ่านง่าย เข้าใจง่าย และมีโครงสร้างที่เอื้อต่อการเรียนรู้สำหรับผู้เริ่มต้น ขณะเดียวกันก็มีศักยภาพเพียงพอสำหรับการประยุกต์ใช้ในงานที่มีความซับซ้อนสูง ภาษาไพทอนจึงเหมาะอย่างยิ่งสำหรับใช้เป็นเครื่องมือในการเรียนรู้พื้นฐานการเขียนโปรแกรมในรายวิชาวิทยาการคำนวณ

สำหรับผู้เรียนที่เริ่มต้นศึกษาการเขียนโปรแกรม ภาษาไพทอนช่วยให้สามารถทำความเข้าใจแนวคิดสำคัญต่าง ๆ ได้อย่างเป็นระบบ เช่น ตัวแปร ชนิดข้อมูล การรับและแสดงผลข้อมูล การใช้คำสั่งควบคุมแบบเรียงลำดับ ทางเลือก และทำซ้ำ ตลอดจนการประยุกต์ใช้โปรแกรมในการแก้ปัญหาอย่างง่าย การเรียนภาษาไพทอนจึงมิได้มุ่งเน้นเพียงการจำคำสั่งหรือไวยากรณ์ของภาษา แต่เป็นการพัฒนาความสามารถในการคิดเชิงตรรกะ การออกแบบลำดับขั้นตอนการทำงาน และการแปลงแนวคิดไปสู่คำสั่งที่คอมพิวเตอร์สามารถทำงานได้

ในบริบทของรายวิชาวิทยาการคำนวณสำหรับนักศึกษาครู การเรียนรู้ภาษาไพทอนมีความสำคัญในสองมิติ มิติแรก คือ การพัฒนาความเข้าใจพื้นฐานด้านการเขียนโปรแกรมในรูปแบบข้อความ ซึ่งเป็นขั้นต่อยอดจากการเรียนรู้ผังงาน การโปรแกรมแบบบล็อกคำสั่ง และการใช้เครื่องมืออย่าง Scratch หรือ Flowgorithm มิติที่สอง คือ การเตรียมความพร้อมให้นักศึกษาครูสามารถนำภาษาไพทอนไปประยุกต์ใช้ในการจัดการเรียนรู้สำหรับนักเรียนในอนาคต โดยเฉพาะการเลือกใช้เครื่องมือที่เหมาะสมและการออกแบบกิจกรรมที่ช่วยลดความซับซ้อนในการเรียนรู้

ในเอกสารประกอบการสอนนี้ ใช้ **Google Colab** เป็นเครื่องมือหลักในการรันโปรแกรมภาษาไพทอน เนื่องจากใช้งานสะดวก ไม่ต้องติดตั้งโปรแกรมเพิ่มเติม รองรับการทำงานผ่านเว็บเบราว์เซอร์ และเอื้อต่อการเรียนรู้ทั้งในห้องเรียนและนอกห้องเรียน นอกจากนี้ ยังใช้ **Python Tutor** เป็นเครื่องมือช่วยอธิบายลำดับขั้นตอนการทำงานของโปรแกรมภาษาไพทอน โดยเฉพาะการติดตามค่าของตัวแปร

และการทำงานของคำสั่งทีละขั้นตอน ซึ่งช่วยให้ผู้เรียนเข้าใจการทำงานภายในของโปรแกรมได้ง่ายขึ้น และลดความสับสนที่มักเกิดขึ้นในผู้เริ่มต้นเรียนเขียนโปรแกรม

บทนี้มุ่งอธิบายความรู้เบื้องต้นเกี่ยวกับภาษาไพทอน ตัวแปร ชนิดข้อมูล และการรับ–แสดงผลข้อมูล การใช้คำสั่งควบคุมแบบเรียงลำดับ ทางเลือก และทำซ้ำในภาษาไพทอน การเขียนโปรแกรมแก้ปัญหาอย่างง่าย ตลอดจนการออกแบบกิจกรรมการเรียนรู้เบื้องต้นสำหรับการสอนภาษาไพทอน เพื่อให้นักศึกษาครูสามารถใช้ภาษาไพทอนได้อย่างมีพื้นฐานที่มั่นคง และสามารถนำไปประยุกต์ใช้ในการจัดการเรียนรู้ได้อย่างเหมาะสม

9.1 ความรู้เบื้องต้นเกี่ยวกับภาษาไพทอน

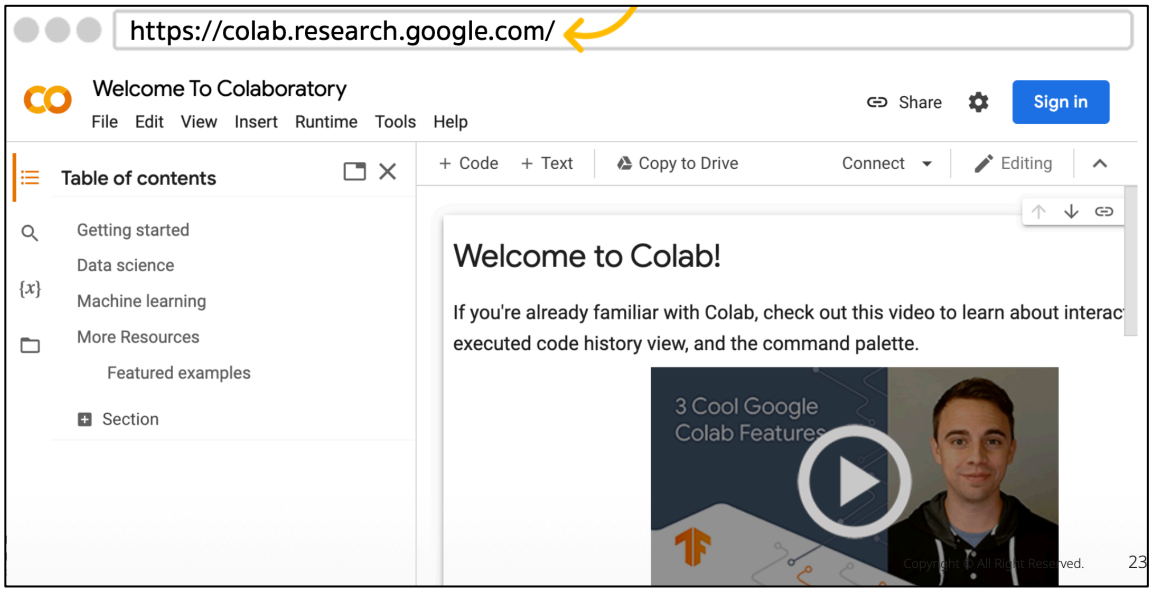
ภาษาไพทอนเป็นภาษาโปรแกรมระดับสูง (High-level Programming Language) ที่ถูกออกแบบมาให้มีรูปแบบคำสั่งอ่านง่าย คล้ายภาษามนุษย์ และเน้นความชัดเจนของโครงสร้างโปรแกรม ไพทอนได้รับการพัฒนาโดย Guido van Rossum และถูกนำมาใช้อย่างแพร่หลายในหลากหลายสาขา เนื่องจากเป็นภาษาที่มีความยืดหยุ่น รองรับการทำงานหลายรูปแบบ และมีชุมชนผู้ใช้งานขนาดใหญ่ที่ช่วยพัฒนาเครื่องมือและแหล่งเรียนรู้จำนวนมาก

จุดเด่นสำคัญของภาษาไพทอน คือ ความเรียบง่ายของไวยากรณ์ เมื่อเปรียบเทียบกับภาษาโปรแกรมอื่น ผู้เรียนสามารถอ่านและทำความเข้าใจโค้ดได้ง่ายกว่า เพราะไพทอนไม่ใช้สัญลักษณ์ซับซ้อนมากเกินจำเป็น เช่น เครื่องหมายปีกกาในการกำหนดขอบเขตของคำสั่ง แต่ใช้การเยื้องบรรทัด (Indentation) เป็นส่วนสำคัญของโครงสร้างภาษา ลักษณะนี้ช่วยส่งเสริมให้ผู้เรียนเขียนโปรแกรมอย่างเป็นระเบียบและมองเห็นลำดับการทำงานได้ชัดเจน

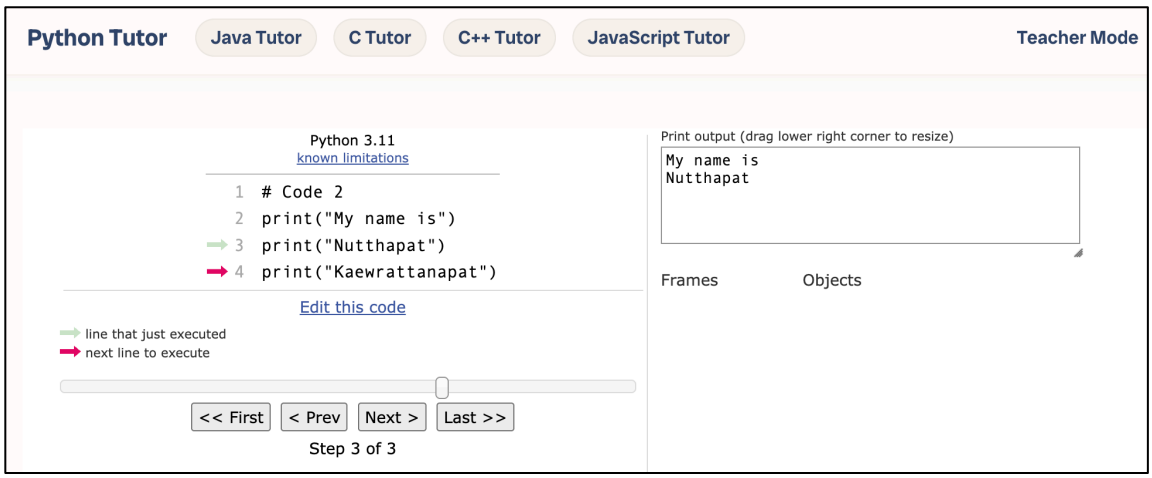
ภาษาไพทอนมีลักษณะเป็นภาษาตีความ (Interpreted Language) หมายถึง โปรแกรมจะถูกแปลและทำงานทีละบรรทัดหรือทีละส่วนในขณะรัน ทำให้ผู้เรียนสามารถทดลองคำสั่ง ตรวจสอบผลลัพธ์ และแก้ไขข้อผิดพลาดได้อย่างรวดเร็ว คุณลักษณะนี้เหมาะอย่างยิ่งต่อการเรียนการสอนในระดับเริ่มต้น เพราะช่วยให้ผู้เรียนเห็นผลของคำสั่งที่ตนเขียนทันที

ในบริบทการศึกษา ภาษาไพทอนได้รับความนิยมอย่างมากเพราะเหมาะสำหรับการเรียนรู้แนวคิดพื้นฐานทางการเขียนโปรแกรม เช่น ตัวแปร การคำนวณ การควบคุมลำดับคำสั่ง การตัดสินใจ และการทำซ้ำ อีกทั้งยังสามารถต่อยอดไปสู่การเรียนรู้ขั้นสูง เช่น การวิเคราะห์ข้อมูล ปัญญาประดิษฐ์ หรือการพัฒนาโปรแกรมประยุกต์ได้ในอนาคต

สำหรับเอกสารประกอบการสอนฉบับนี้ ผู้เขียนเลือกใช้ Google Colab เป็นเครื่องมือในการรันโปรแกรมภาษาไพทอน เนื่องจากผู้เรียนสามารถใช้งานได้สะดวกผ่านเว็บเบราว์เซอร์ โดยไม่จำเป็นต้องติดตั้งโปรแกรมลงในเครื่องคอมพิวเตอร์ ช่วยลดข้อจำกัดด้านอุปกรณ์และเวลาในการเตรียมความพร้อม อีกทั้งยังเหมาะกับการจัดการเรียนรู้ที่เน้นการทดลองและการปรับแก้โค้ดอย่างต่อเนื่อง



นอกจากนี้ การใช้ Python Tutor ยังมีบทบาทสำคัญในการช่วยให้ผู้เรียนเข้าใจการทำงานของภาษาไพทอนในระดับลึกขึ้น โดยเฉพาะในประเด็นที่ผู้เรียนมักสับสน เช่น การเปลี่ยนแปลงค่าของตัวแปร การทำงานของลูป หรือการตัดสินใจตามเงื่อนไข Python Tutor แสดงลำดับการทำงานของโปรแกรมทีละขั้นตอน ทำให้ผู้เรียนสามารถเห็นภาพการทำงานภายในได้อย่างเป็นระบบ



สำหรับนักศึกษาครู ความรู้เบื้องต้นเกี่ยวกับภาษาไพทอนไม่ควรจำกัดอยู่เพียงการรู้จักคำสั่ง แต่ควรครอบคลุมถึงความเข้าใจว่าทำไมไพทอนจึงเป็นภาษาที่เหมาะต่อการเรียนรู้เบื้องต้น และเครื่องมือใดบ้างที่ช่วยให้ผู้เรียนเข้าถึงการเขียนโปรแกรมได้ง่ายขึ้น ความเข้าใจเช่นนี้จะช่วยให้สามารถออกแบบการเรียนรู้ที่ลดความซับซ้อนและส่งเสริมความเข้าใจอย่างแท้จริง

กล่าวโดยสรุป ภาษาไพทอนเป็นภาษาโปรแกรมที่เหมาะสำหรับการเรียนรู้พื้นฐานการเขียนโปรแกรม เนื่องจากมีโครงสร้างที่เรียบง่าย อ่านง่าย และเอื้อต่อการทดลองปฏิบัติ เมื่อใช้ร่วมกับ Google Colab และ Python Tutor จะช่วยให้ผู้เรียนสามารถเรียนรู้ได้อย่างสะดวก เข้าใจง่าย และเห็นกระบวนการทำงานของโปรแกรมได้อย่างชัดเจน

9.2 ตัวแปร ชนิดข้อมูล และการรับ–แสดงผลข้อมูล

ตัวแปร (Variable) เป็นแนวคิดพื้นฐานสำคัญของการเขียนโปรแกรม หมายถึง ชื่อที่ใช้แทนพื้นที่จัดเก็บข้อมูลในหน่วยความจำของคอมพิวเตอร์ เพื่อให้โปรแกรมสามารถนำข้อมูลนั้นไปใช้งานหรือปรับเปลี่ยนค่าได้ระหว่างการทำงาน ผู้เรียนสามารถเข้าใจตัวแปรได้ในลักษณะของ “กล่องเก็บข้อมูล” ที่มีชื่อกำกับไว้ เมื่อโปรแกรมต้องการเก็บค่าบางอย่าง เช่น คะแนน ชื่อ อายุ หรือผลลัพธ์จากการคำนวณ ก็จะนำข้อมูลเหล่านั้นไปเก็บไว้ในตัวแปร

การตั้งชื่อตัวแปรในภาษาไพทอนควรสื่อความหมาย ชัดเจน และสอดคล้องกับข้อมูลที่เก็บ เช่น score, name, total, average หรือ age การใช้ชื่อตัวแปรที่เหมาะสมจะช่วยให้โปรแกรมอ่านง่ายและเข้าใจได้สะดวกยิ่งขึ้น โดยเฉพาะในบริบทการเรียนการสอนที่ผู้เรียนยังต้องพัฒนาความเข้าใจเกี่ยวกับโครงสร้างของโปรแกรมอย่างเป็นระบบ

ชนิดข้อมูล (Data Type) หมายถึง ประเภทของข้อมูลที่เก็บอยู่ในตัวแปร ซึ่งภาษาไพทอนรองรับข้อมูลหลายชนิด แต่ในการเรียนรู้เบื้องต้นมักเน้นชนิดข้อมูลสำคัญ เช่น จำนวนเต็ม (Integer) จำนวนทศนิยม (Float) ข้อความ (String) และค่าความจริง (Boolean) การเข้าใจชนิดข้อมูลมีความสำคัญเพราะส่งผลต่อวิธีการประมวลผลและผลลัพธ์ของโปรแกรม เช่น การบวกตัวเลขย่อมต่างจากการเชื่อมข้อความ หรือการตรวจสอบเงื่อนไขย่อมเกี่ยวข้องกับค่าจริงและเท็จ

การรับข้อมูล (Input) เป็นกระบวนการที่โปรแกรมรับค่าจากผู้ใช้เพื่อนำไปประมวลผลต่อ ในภาษาไพทอนคำสั่งพื้นฐานที่ใช้รับข้อมูล คือ input() ตัวอย่างเช่น การรับชื่อหรือคะแนนจากผู้ใช้ อย่างไรก็ตาม ผู้เรียนควรเข้าใจด้วยว่า ข้อมูลที่รับผ่าน input() มักอยู่ในรูปข้อความ (String) ก่อน หากต้องการ

นำไปคำนวณเป็นตัวเลข จำเป็นต้องแปลงชนิดข้อมูลก่อน เช่น ใช้ int() สำหรับจำนวนเต็ม หรือ float() สำหรับจำนวนทศนิยม

การแสดงผลข้อมูล (Output) คือ กระบวนการนำข้อมูลหรือผลลัพธ์ออกมาแสดงให้ผู้ใช้เห็น ในภาษาไพทอนมักใช้คำสั่ง print() ในการแสดงข้อความ ตัวเลข หรือค่าของตัวแปร การแสดงผลอย่างเหมาะสมช่วยให้ผู้ใช้เข้าใจสิ่งที่โปรแกรมทำได้ชัดเจน เช่น การแสดงผลลัพธ์พร้อมข้อความอธิบาย หรือการแสดงค่าที่คำนวณได้ในรูปแบบที่อ่านง่าย

ตัวอย่างโปรแกรมเบื้องต้นเกี่ยวกับการรับและแสดงผลข้อมูล เช่น โปรแกรมรับชื่อนักเรียนแล้วแสดงข้อความต้อนรับ หรือโปรแกรมรับคะแนนสองวิชาแล้วคำนวณผลรวมและค่าเฉลี่ย กิจกรรมลักษณะนี้ช่วยให้ผู้เรียนเข้าใจบทบาทของตัวแปร การจัดเก็บข้อมูล การรับข้อมูลจากผู้ใช้ และการแสดงผลลัพธ์อย่างเป็นระบบ

สำหรับนักศึกษาครู การสอนเรื่องตัวแปรและชนิดข้อมูลควรใช้การอธิบายที่เชื่อมโยงกับสิ่งที่ผู้เรียนเข้าใจง่าย เช่น เปรียบตัวแปรเป็นกล่องหรือป้ายชื่อของข้อมูล และควรใช้ Python Tutor ช่วยแสดงให้เห็นว่าเมื่อโปรแกรมทำงาน ค่าของตัวแปรถูกสร้าง เปลี่ยนแปลง หรือถูกนำไปใช้อย่างไร เครื่องมือเช่นนี้ช่วยลดความเป็นนามธรรมของแนวคิดและทำให้ผู้เรียนเข้าใจการทำงานภายในของโปรแกรมได้ชัดเจนขึ้น

กล่าวโดยสรุป ตัวแปร ชนิดข้อมูล และการรับ–แสดงผลข้อมูล เป็นพื้นฐานสำคัญของการเขียนโปรแกรมภาษาไพทอน เพราะเป็นกลไกหลักในการจัดการข้อมูลที่โปรแกรมต้องใช้ การเข้าใจประเด็นเหล่านี้อย่างถูกต้องจะช่วยให้ผู้เรียนสามารถเขียนโปรแกรมแก้ปัญหาอย่างง่ายได้อย่างมีประสิทธิภาพ

9.3 การใช้คำสั่งควบคุมแบบเรียงลำดับ ทางเลือก และทำซ้ำในภาษาไพทอน

การเขียนโปรแกรมภาษาไพทอนมิได้อาศัยเพียงการกำหนดตัวแปรหรือการรับ–แสดงผลข้อมูลเท่านั้น แต่ยังต้องอาศัยคำสั่งควบคุม (Control Structures) เพื่อกำหนดลำดับและรูปแบบการทำงานของโปรแกรม คำสั่งควบคุมพื้นฐานที่สำคัญประกอบด้วย การทำงานแบบเรียงลำดับ การทำงานแบบทางเลือก และการทำงานแบบทำซ้ำ ซึ่งเป็นแนวคิดเดียวกับที่ผู้เรียนเคยศึกษาในรูปของผังงานและ Flowgorithm มาก่อน

9.3.1 การทำงานแบบเรียงลำดับ

การทำงานแบบเรียงลำดับ คือ การที่โปรแกรมดำเนินคำสั่งทีละคำสั่งจากบนลงล่างตามลำดับที่เขียนไว้ โดยไม่มีการข้ามขั้นตอนหรือการตัดสินใจเพิ่มเติม ตัวอย่างเช่น โปรแกรมรับคะแนนสองจำนวน

คำนวณผลรวม และแสดงผลลัพธ์ โปรแกรมลักษณะนี้ช่วยให้ผู้เรียนเข้าใจว่า ลำดับของคำสั่งมีผลโดยตรงต่อผลลัพธ์ หากสลับขั้นตอนอาจทำให้โปรแกรมทำงานผิดพลาดได้

9.3.2 การทำงานแบบทางเลือก

การทำงานแบบทางเลือกใช้เมื่อโปรแกรมต้องตัดสินใจภายใต้เงื่อนไขบางประการ ในภาษาไพทอนมักใช้คำสั่ง if, if-else หรือ if-elif-else เพื่อกำหนดแนวทางการทำงานที่แตกต่างกันตามผลของเงื่อนไข ตัวอย่างเช่น หากคะแนนมากกว่าหรือเท่ากับ 50 ให้แสดงผลว่า “ผ่าน” แต่หากน้อยกว่า 50 ให้แสดงผลว่า “ไม่ผ่าน” การเรียนรู้เรื่องนี้ช่วยให้ผู้เรียนเข้าใจความสัมพันธ์ระหว่างเงื่อนไขกับผลลัพธ์ของโปรแกรม

9.3.3 การทำงานแบบทำซ้ำ

การทำงานแบบทำซ้ำใช้เมื่อโปรแกรมต้องดำเนินคำสั่งเดิมหลายครั้ง ในภาษาไพทอนสามารถใช้คำสั่ง for และ while เพื่อควบคุมการทำงานซ้ำ เช่น การแสดงตัวเลข 1 ถึง 10 การหาผลรวมของข้อมูลหลายค่า หรือการรับค่าจนกว่าจะถูกต้อง การทำซ้ำช่วยลดความซ้ำซ้อนของโปรแกรมและทำให้การจัดการงานที่เกิดซ้ำเป็นไปอย่างมีประสิทธิภาพ

การสอนคำสั่งควบคุมในภาษาไพทอนควรเชื่อมโยงกับสิ่งที่ผู้เรียนเรียนมาก่อน เช่น ผังงานแบบเรียงลำดับ ทางเลือก และทำซ้ำ เพื่อให้ผู้เรียนเห็นว่า ภาษาไพทอนเป็นเพียงอีกวิธีหนึ่งในการถ่ายทอดแนวคิดเดียวกันจากรูปแบบภาพไปสู่รูปแบบข้อความ การเชื่อมโยงเช่นนี้จะช่วยลดความรู้สึกว่าการเขียนโค้ดเป็นเรื่องใหม่ที่แยกขาดจากความรู้เดิม

ในบริบทของการเรียนรู้ ผู้สอนสามารถใช้ Google Colab เพื่อให้ผู้เรียนทดลองเขียนและรันคำสั่งควบคุมแต่ละประเภทได้ทันที และใช้ Python Tutor เพื่อช่วยอธิบายลำดับการทำงานของโปรแกรม เช่น การตรวจสอบว่าเงื่อนไขเป็นจริงหรือเท็จอย่างไร การวนซ้ำเปลี่ยนค่าตัวแปรอย่างไร หรือเหตุใดโปรแกรมจึงให้ผลลัพธ์เช่นนั้น เครื่องมือทั้งสองนี้ช่วยให้การเรียนรู้คำสั่งควบคุมชัดเจนและตรวจสอบได้มากขึ้น

สำหรับนักศึกษาครู ความเข้าใจเรื่องคำสั่งควบคุมควรครอบคลุมทั้งการเขียนโปรแกรมและการอธิบายแก่ผู้เรียนในระดับการศึกษาขั้นพื้นฐาน ครูควรสามารถเลือกตัวอย่างง่าย ๆ ที่ใกล้ตัว เช่น การตรวจสอบผลสอบ การนับเลข หรือการทำกิจกรรมซ้ำ ๆ ในชีวิตประจำวัน แล้วค่อยเชื่อมไปสู่โค้ดในภาษาไพทอน เพื่อให้ผู้เรียนเห็นว่าคำสั่งควบคุมเป็นส่วนหนึ่งของการคิดอย่างมีระบบ ไม่ใช่เพียงรูปแบบคำสั่งที่ต้องท่องจำ

กล่าวโดยสรุป การใช้คำสั่งควบคุมแบบเรียงลำดับ ทางเลือก และทำซ้ำในภาษาไพทอน เป็นแกนหลักของการพัฒนาโปรแกรมที่มีความหมายและสามารถแก้ปัญหาได้อย่างเป็นระบบ ผู้เรียนที่เข้าใจแนวคิดนี้จะสามารถต่อยอดไปสู่การเขียนโปรแกรมที่ซับซ้อนขึ้นได้อย่างมั่นคง

9.4 การเขียนโปรแกรมแก้ปัญหาอย่างง่ายด้วยภาษาไพทอน

การเขียนโปรแกรมแก้ปัญหาอย่างง่ายด้วยภาษาไพทอนเป็นขั้นตอนที่ช่วยให้ผู้เรียนนำความรู้เรื่องตัวแปร ชนิดข้อมูล การรับ–แสดงผล และคำสั่งควบคุม มาประยุกต์ใช้ร่วมกันในการสร้างโปรแกรมที่ตอบโจทย์ปัญหาจริง การเรียนรู้ในส่วนนี้มีความสำคัญอย่างยิ่ง เพราะเป็นจุดที่ผู้เรียนเปลี่ยนจากการเรียนรู้คำสั่งแยกส่วน ไปสู่การใช้แนวคิดทางวิทยาการคำนวณเพื่อออกแบบวิธีแก้ปัญหาอย่างเป็นระบบ

กระบวนการเขียนโปรแกรมแก้ปัญหาอย่างง่ายควรเริ่มจากการวิเคราะห์โจทย์หรือสถานการณ์ก่อนว่า ต้องการข้อมูลนำเข้าอะไร ต้องประมวลผลอย่างไร และควรแสดงผลลัพธ์ในรูปแบบใด หลักการนี้สอดคล้องกับแนวคิด Input–Process–Output ที่ผู้เรียนได้ศึกษาในบทก่อนหน้า เมื่อเข้าใจโจทย์อย่างชัดเจนแล้ว จึงค่อยออกแบบอัลกอริทึมและแปลงเป็นโปรแกรมภาษาไพทอน

ตัวอย่างของปัญหาอย่างง่ายที่เหมาะกับการฝึกเขียนโปรแกรมในระดับเริ่มต้น ได้แก่ โปรแกรมคำนวณพื้นที่รูปเรขาคณิต โปรแกรมหาค่าเฉลี่ยคะแนน โปรแกรมตรวจสอบผลสอบ โปรแกรมแปลงหน่วยอุณหภูมิ โปรแกรมคำนวณส่วนลดสินค้า หรือโปรแกรมแสดงตารางสูตรคูณ ปัญหาเหล่านี้มีลักษณะไม่ซับซ้อนมากนัก แต่ช่วยให้ผู้เรียนฝึกใช้แนวคิดพื้นฐานครบถ้วน

ตัวอย่างเช่น หากโจทย์กำหนดให้เขียนโปรแกรมรับคะแนน 3 วิชา แล้วหาค่าเฉลี่ยและแสดงผลว่าผ่านหรือไม่ ผู้เรียนต้องวิเคราะห์ว่า ข้อมูลนำเข้าคือคะแนน 3 ค่า กระบวนการคือการหาผลรวมและหารด้วย 3 และผลลัพธ์คือค่าเฉลี่ยพร้อมข้อความว่า “ผ่าน” หรือ “ไม่ผ่าน” จากนั้นจึงเลือกใช้ตัวแปร คำสั่งรับข้อมูล คำสั่งคำนวณ และคำสั่งเงื่อนไขให้เหมาะสม กระบวนการเช่นนี้ช่วยให้ผู้เรียนเห็นความเชื่อมโยงระหว่างโจทย์กับโค้ดอย่างเป็นขั้นตอน

อีกตัวอย่างหนึ่ง เช่น โปรแกรมแสดงตัวเลข 1 ถึง 10 และหาผลรวมของตัวเลขทั้งหมด ผู้เรียนต้องใช้แนวคิดเรื่องการทำซ้ำและตัวแปรสะสมผลร่วมกัน การฝึกโจทย์ลักษณะนี้ช่วยให้เข้าใจว่าการทำซ้ำมิใช่เพียงการแสดงผลหลายครั้ง แต่ยังใช้เพื่อจัดการปัญหาที่เกี่ยวข้องกับข้อมูลหลายค่าได้ด้วย

ในการเรียนการสอน ผู้เขียนใช้ Google Colab เป็นพื้นที่ให้ผู้เรียนทดลองเขียนโปรแกรม แก้ไข และรันโปรแกรมได้สะดวก ขณะเดียวกัน ใช้ Python Tutor เพื่อช่วยให้ผู้เรียนติดตามลำดับการทำงานของโปรแกรมทีละขั้น โดยเฉพาะในโจทย์ที่เกี่ยวข้องกับการเปลี่ยนแปลงค่าของตัวแปรหรือการทำงาน

ของลูป เครื่องมือทั้งสองนี้ช่วยให้ผู้เรียนไม่เพียงเห็นผลลัพธ์สุดท้าย แต่เข้าใจเส้นทางที่นำไปสู่ผลลัพธ์นั้นด้วย

สำหรับนักศึกษาครู การเขียนโปรแกรมแก้ปัญหาอย่างง่ายควรถูกมองเป็นทั้งการฝึกทักษะทางวิทยาการคำนวณและการเตรียมความพร้อมสำหรับการสอน กล่าวคือ นักศึกษาครูควรสามารถคัดเลือกโจทย์ที่เหมาะสมกับวัยของผู้เรียน อธิบายวิธีวิเคราะห์โจทย์ได้อย่างชัดเจน และใช้เครื่องมืออย่าง Google Colab หรือ Python Tutor เพื่อช่วยลดความซับซ้อนในการเรียนรู้ของนักเรียน

กล่าวโดยสรุป การเขียนโปรแกรมแก้ปัญหาอย่างง่ายด้วยภาษาไพทอนเป็นกระบวนการสำคัญที่ช่วยให้ผู้เรียนบูรณาการความรู้พื้นฐานต่าง ๆ เข้าด้วยกัน และพัฒนาความสามารถในการวิเคราะห์ปัญหา ออกแบบวิธีการ และสร้างโปรแกรมที่มีความหมายในระดับเบื้องต้น

9.5 การออกแบบกิจกรรมการเรียนรู้เบื้องต้นสำหรับการสอนภาษาไพทอน

การออกแบบกิจกรรมการเรียนรู้เบื้องต้นสำหรับการสอนภาษาไพทอนควรคำนึงถึงธรรมชาติของผู้เรียนระดับเริ่มต้น ซึ่งมักยังไม่คุ้นเคยกับการเขียนโปรแกรมแบบข้อความ ผู้สอนจึงต้องเลือกแนวทางที่ช่วยลดความซับซ้อนของภาษา เชื่อมโยงเนื้อหากับประสบการณ์เดิม และเปิดโอกาสให้ผู้เรียนได้ทดลองปฏิบัติจริงอย่างต่อเนื่อง การเรียนรู้ภาษาไพทอนจะเกิดประสิทธิภาพสูงสุดเมื่อผู้เรียนไม่เพียงมองเห็นคำสั่ง แต่เข้าใจเหตุผลของคำสั่งและสามารถนำไปใช้แก้ปัญหาได้

หลักการสำคัญประการแรก คือ ควรเริ่มจากกิจกรรมที่ง่าย ชัดเจน และให้ผลลัพธ์รวดเร็ว เช่น การใช้ print() แสดงข้อความ การสร้างตัวแปรเก็บชื่อหรืออายุ หรือการรับข้อมูลอย่างง่ายผ่าน input() กิจกรรมเหล่านี้ช่วยสร้างความคุ้นเคยกับรูปแบบของภาษาไพทอน และทำให้ผู้เรียนเกิดความมั่นใจตั้งแต่เริ่มต้น

หลักการประการที่สอง คือ ควรเชื่อมโยงภาษาไพทอนกับแนวคิดที่ผู้เรียนเคยเรียนมาก่อน เช่น ผังงาน โครงสร้างแบบเรียงลำดับ แบบทางเลือก และแบบทำซ้ำ การให้ผู้เรียนเปรียบเทียบผังงานกับโค้ดไพทอน หรือเปรียบเทียบบล็อกคำสั่งใน Scratch กับคำสั่งในภาษาไพทอน จะช่วยให้ผู้เรียนมองเห็นความต่อเนื่องขององค์ความรู้และลดความรู้สึกว่าไพทอนเป็นเรื่องใหม่ที่ยากเกินไป

หลักการประการที่สาม คือ ควรใช้เครื่องมือที่สนับสนุนการเรียนรู้อย่างเหมาะสม ในบริบทนี้ Google Colab เป็นเครื่องมือที่มีประโยชน์มาก เพราะผู้เรียนสามารถเปิดใช้งานผ่านเว็บเบราว์เซอร์ได้ทันที เขียนโค้ด รันโปรแกรม และบันทึกงานได้สะดวก ส่วน Python Tutor เป็นเครื่องมือช่วยวิเคราะห์

การทำงานของโปรแกรมทีละขั้นตอน ซึ่งเหมาะอย่างยิ่งสำหรับการอธิบายตัวแปร เงื่อนไข และการทำซ้ำในระดับเริ่มต้น

รูปแบบกิจกรรมการเรียนรู้ที่เหมาะสมสำหรับการสอนภาษาไพทอนเบื้องต้น อาจประกอบด้วย

1) กิจกรรมทดลองเขียนและรันคำสั่งสั้น ๆ

เช่น การแสดงข้อความ การคำนวณอย่างง่าย หรือการรับข้อมูลแล้วแสดงผล ช่วยให้ผู้เรียนคุ้นเคยกับสภาพแวดล้อมของภาษา

2) กิจกรรมวิเคราะห์การทำงานของโปรแกรม

โดยใช้ Python Tutor ให้ผู้เรียนติดตามค่าของตัวแปรและลำดับการทำงานทีละขั้นตอน วิธีนี้ช่วยให้เข้าใจการทำงานภายในของโปรแกรมอย่างชัดเจน

3) กิจกรรมแก้ไขโปรแกรมที่ผิดพลาด

ให้ผู้เรียนวิเคราะห์ข้อผิดพลาดของโค้ดที่กำหนด เช่น ลืมแปลงชนิดข้อมูล เขียนเงื่อนไขผิด หรือเยื้องบรรทัดไม่ถูกต้อง กิจกรรมลักษณะนี้ช่วยพัฒนาทักษะการตรวจสอบและแก้ปัญหา

4) กิจกรรมออกแบบโปรแกรมจากสถานการณ์จริง

เช่น โปรแกรมคำนวณค่าใช้จ่าย โปรแกรมตัดเกรด หรือโปรแกรมคำนวณคะแนนรวม กิจกรรมเหล่านี้ช่วยให้ผู้เรียนเห็นความหมายของภาษาไพทอนในฐานะเครื่องมือแก้ปัญหา

สำหรับนักศึกษาครู การออกแบบกิจกรรมการสอนภาษาไพทอนควรเชื่อมโยงกับการจัดทำแผนการจัดการเรียนรู้ เช่น การกำหนดผลลัพธ์การเรียนรู้ การเลือกกิจกรรมที่เหมาะสมกับระดับของผู้เรียน การจัดลำดับเนื้อหา การเลือกใช้ Google Colab และ Python Tutor ให้สอดคล้องกับจุดประสงค์ และการกำหนดเกณฑ์ประเมินที่สะท้อนทั้งความรู้ ทักษะ และความสามารถในการคิดวิเคราะห์ของผู้เรียน

ในด้านการประเมินผล ครูควรใช้ทั้งการประเมินจากชิ้นงาน การสังเกตกระบวนการทำงาน การให้ผู้เรียนอธิบายโค้ดของตนเอง และการวิเคราะห์ลำดับการทำงานของโปรแกรม การประเมินลักษณะนี้จะช่วยให้เห็นว่าผู้เรียนเข้าใจจริงหรือเพียงคัดลอกคำสั่งตามตัวอย่างเท่านั้น

กล่าวโดยสรุป การออกแบบกิจกรรมการเรียนรู้เบื้องต้นสำหรับการสอนภาษาไพทอนควรมุ่งเน้นความเข้าใจเป็นลำดับขั้น ลดความซับซ้อนด้วยเครื่องมือที่เหมาะสม และเปิดโอกาสให้ผู้เรียนได้ลงมือปฏิบัติ วิเคราะห์ และสะท้อนผลการเรียนรู้ของตนเองอย่างต่อเนื่อง

บทสรุปประจำบทที่ 9

ภาษาไพทอนเป็นภาษาโปรแกรมที่เหมาะสำหรับการเรียนรู้พื้นฐานการเขียนโปรแกรม เนื่องจากมีรูปแบบคำสั่งที่เรียบง่าย อ่านง่าย และเอื้อต่อการพัฒนาทักษะการคิดเชิงตรรกะและการแก้ปัญหาอย่างเป็นระบบ ในบทนี้ได้อธิบายความรู้เบื้องต้นเกี่ยวกับภาษาไพทอน แนวคิดเรื่องตัวแปร ชนิดข้อมูล การรับ–แสดงผลข้อมูล การใช้คำสั่งควบคุมแบบเรียงลำดับ ทางเลือก และทำซ้ำ ตลอดจนการเขียนโปรแกรมแก้ปัญหาอย่างง่ายด้วยภาษาไพทอน

นอกจากนี้ ยังได้เน้นการใช้ Google Colab เป็นเครื่องมือในการรันโปรแกรม เพื่อให้ผู้เรียนเข้าถึงการใช้งานได้สะดวกโดยไม่ต้องติดตั้งซอฟต์แวร์เพิ่มเติม และใช้ Python Tutor เป็นเครื่องมือช่วยอธิบายลำดับขั้นตอนการทำงานของโปรแกรม ซึ่งช่วยให้ผู้เรียนเข้าใจการเปลี่ยนแปลงค่าของตัวแปรและการทำงานของโปรแกรมได้ชัดเจนมากขึ้น สำหรับนักศึกษาครู การเรียนรู้ภาษาไพทอนควรเชื่อมโยงทั้งกับการพัฒนาทักษะการเขียนโปรแกรมของตนเอง และกับการออกแบบกิจกรรมการเรียนรู้ที่เหมาะสมสำหรับผู้เรียนในระดับการศึกษาขั้นพื้นฐาน เพื่อให้การสอนภาษาไพทอนเป็นไปอย่างมีความหมาย เข้าใจง่าย และส่งเสริมทักษะวิทยาการคำนวณได้อย่างแท้จริง

คำถามท้ายบท

1. จงอธิบายลักษณะเด่นของภาษาไพทอน และเหตุใดภาษาไพทอนจึงเหมาะกับผู้เริ่มต้นเรียนเขียนโปรแกรม
2. Google Colab มีบทบาทอย่างไรในการสนับสนุนการเรียนรู้ภาษาไพทอน
3. Python Tutor มีประโยชน์อย่างไรต่อการทำความเข้าใจลำดับการทำงานของโปรแกรมภาษาไพทอน
4. จงอธิบายความหมายของตัวแปรและชนิดข้อมูลในภาษาไพทอน พร้อมยกตัวอย่าง
5. เพราะเหตุใดจึงต้องแปลงชนิดข้อมูลเมื่อรับค่าจาก input() ในบางกรณี
6. จงอธิบายความแตกต่างระหว่างโครงสร้างควบคุมแบบเรียงลำดับ ทางเลือก และทำซ้ำในภาษาไพทอน
7. หากต้องการเขียนโปรแกรมตรวจสอบผลสอบจากคะแนน ผู้เรียนควรใช้คำสั่งควบคุมประเภทใด เพราะเหตุใด
8. จงยกตัวอย่างโปรแกรมแก้ปัญหาอย่างง่ายด้วยภาษาไพทอนอย่างน้อย 2 ตัวอย่าง พร้อมอธิบายแนวคิดในการออกแบบ

9. หากท่านเป็นครูระดับมัธยมศึกษา ท่านจะออกแบบกิจกรรมการเรียนรู้ภาษาไพทอนโดยใช้ Google Colab อย่างไร
10. หากท่านเป็นครูผู้สอน ท่านจะใช้ Python Tutor เพื่อช่วยลดความสับสนของผู้เรียนเกี่ยวกับตัวแปรและลูปได้อย่างไร

เอกสารอ้างอิง

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

Downey, A. (2015). *Think Python: How to think like a computer scientist* (2nd ed.). Needham, MA: Green Tea Press.

Google. (2024). *Google Colaboratory*. Retrieved from the official Google Colab website.

Guo, P. J. (2013). Online Python Tutor: Embeddable web-based program visualization for CS education. In *Proceedings of the 44th ACM Technical Symposium on Computer Science Education* (pp. 579–584).

Python Software Foundation. (2024). *Python documentation*. Retrieved from the official Python documentation website.

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

เอกสารแนบ

ภาษาไพทอน (Python) เป็นภาษาการเขียนโปรแกรมที่ได้รับความนิยมอย่างมากในปัจจุบัน เนื่องจากมีรูปแบบคำสั่งที่อ่านง่าย เขียนง่าย และเหมาะสำหรับผู้เริ่มต้นเรียนรู้การเขียนโปรแกรม อีกทั้งยังสามารถประยุกต์ใช้ได้หลากหลาย ทั้งด้านการคำนวณ การพัฒนาเว็บไซต์ การวิเคราะห์ข้อมูล ปัญญาประดิษฐ์ และงานอัตโนมัติต่าง ๆ

ในการเรียนรู้ภาษาไพทอนสำหรับผู้เริ่มต้น สิ่งสำคัญคือการทำความเข้าใจ **Syntax** หรือรูปแบบการเขียนคำสั่งให้ถูกต้อง เพราะคอมพิวเตอร์จะประมวลผลตามโครงสร้างของคำสั่งอย่างเคร่งครัด หากเขียนผิด แม้เพียงเล็กน้อย โปรแกรมอาจไม่สามารถทำงานได้ ดังนั้น เอกสารแนบฉบับนี้จึงจัดทำขึ้นเพื่อให้นักศึกษาได้ศึกษาไวยากรณ์พื้นฐานของภาษาไพทอน พร้อมตัวอย่างโจทย์และโค้ดประกอบ เพื่อใช้เป็นแนวทางในการฝึกปฏิบัติและพัฒนาทักษะการเขียนโปรแกรมอย่างเป็นขั้นตอน

1. ความรู้พื้นฐานเกี่ยวกับ Syntax ของภาษาไพทอน

1.1 การแสดงผลข้อมูลด้วย print()

คำสั่ง print() ใช้สำหรับแสดงข้อความหรือค่าตัวแปรออกทางหน้าจอ

รูปแบบคำสั่ง

print("ข้อความที่ต้องการแสดง")

ตัวอย่าง

```
print("Hello, Python")
print("ยินดีต้อนรับสู่การเขียนโปรแกรม")
```

ผลลัพธ์

```
Hello, Python
ยินดีต้อนรับสู่การเขียนโปรแกรม
```

1.2 การรับข้อมูลด้วย input()

คำสั่ง input() ใช้สำหรับรับข้อมูลจากผู้ใช้ทางแป้นพิมพ์

รูปแบบคำสั่ง

ตัวแปร = input("ข้อความแจ้งผู้ใช้: ")

ตัวอย่าง

```
name = input("กรุณาป้อนชื่อของคุณ: ")
print("สวัสดี", name)
```

หมายเหตุ

ค่าที่รับด้วย input() จะเป็นชนิดข้อความ (string) โดยอัตโนมัติ

1.3 ตัวแปร (Variable)

ตัวแปรใช้สำหรับเก็บข้อมูล เช่น ชื่อ อายุ คะแนน หรือผลลัพธ์จากการคำนวณ

ตัวอย่าง

```
name = "Nutthapat"
age = 40
score = 85.5
```

การตั้งชื่อตัวแปร

- ควรสื่อความหมาย
- ห้ามขึ้นต้นด้วยตัวเลข
- ห้ามเว้นวรรค
- สามารถใช้ตัวอักษร ตัวเลข และเครื่องหมาย _

ตัวอย่างชื่อตัวแปรที่ถูกต้อง

```
student_name = "Somchai"
score1 = 90
```

1.4 ชนิดข้อมูลพื้นฐาน

ภาษาไพทอนมีชนิดข้อมูลพื้นฐานที่พบบ่อย ดังนี้

ชนิดข้อมูล	ความหมาย	ตัวอย่าง
int	จำนวนเต็ม	10, 25, -5
float	จำนวนทศนิยม	3.14, 99.5
str	ข้อความ	"Hello", "Python"
bool	ค่าความจริง	True, False

ตัวอย่าง

```
x = 10
y = 5.5
name = "Python"
is_pass = True
```

1.5 การแปลงชนิดข้อมูล

บางครั้งจำเป็นต้องแปลงข้อมูลจากข้อความให้เป็นตัวเลขก่อนนำไปคำนวณ

ตัวอย่าง

```
age = int(input("กรุณาป้อนอายุ: "))
weight = float(input("กรุณาป้อนน้ำหนัก: "))
```

คำสั่งที่ใช้บ่อย

- int() แปลงเป็นจำนวนเต็ม
- float() แปลงเป็นจำนวนทศนิยม
- str() แปลงเป็นข้อความ

1.6 การคำนวณทางคณิตศาสตร์

เครื่องหมาย	ความหมาย	ตัวอย่าง
+	บวก	a + b
-	ลบ	a - b
*	คูณ	a * b
/	หาร	a / b
//	หารเอาส่วน	a // b
%	หารเอาเศษ	a % b
**	ยกกำลัง	a ** b

ตัวอย่าง

```
num1 = 10
num2 = 3
print(num1 + num2)
print(num1 - num2)
print(num1 * num2)
print(num1 / num2)
print(num1 // num2)
print(num1 % num2)
print(num1 ** num2)
```

1.7 การเปรียบเทียบ

ใช้เปรียบเทียบค่าระหว่างข้อมูล 2 ค่า โดยผลลัพธ์จะเป็น True หรือ False

เครื่องหมาย	ความหมาย
==	เท่ากับ
!=	ไม่เท่ากับ
>	มากกว่า
<	น้อยกว่า
>=	มากกว่าหรือเท่ากับ
<=	น้อยกว่าหรือเท่ากับ

ตัวอย่าง

```
x = 10
y = 20
print(x < y)
print(x == y)
```

1.8 การตัดสินใจด้วย if, elif, else

ใช้สำหรับตรวจสอบเงื่อนไขและเลือกทำงานตามเงื่อนไขที่กำหนด

รูปแบบคำสั่ง

```
if เงื่อนไข:
    คำสั่ง
elif เงื่อนไข:
    คำสั่ง
else:
    คำสั่ง
```

ตัวอย่าง

```
score = int(input("กรุณาป้อนคะแนน: "))
if score >= 80:
    print("เกรด A")
elif score >= 70:
    print("เกรด B")
elif score >= 60:
    print("เกรด C")
else:
    print("ควรพัฒนา")
```

1.9 การทำซ้ำด้วย for

ใช้เมื่อต้องการทำงานซ้ำตามจำนวนรอบที่กำหนด

ตัวอย่าง

```
for i in range(5):
    print("รอบที่", i)
```

ผลลัพธ์

```
รอบที่ 0
รอบที่ 1
```

รอบที่ 2

รอบที่ 3

รอบที่ 4

1.10 การทำซ้ำด้วย while

ใช้เมื่อต้องการทำงานซ้ำตราบใดที่เงื่อนไขยังเป็นจริง

ตัวอย่าง

```
count = 1
while count <= 5:
    print("จำนวน", count)
    count += 1
```

1.11 ความสำคัญของการย่อหน้า (Indentation)

ภาษาไพทอนใช้การย่อหน้าเพื่อกำหนดขอบเขตของคำสั่ง เช่น ภายใน if หรือภายในลูป หากย่อหน้าไม่ถูกต้อง โปรแกรมจะเกิดข้อผิดพลาด

ตัวอย่างที่ถูกต้อง

```
score = 80
if score >= 50:
    print("ผ่าน")
```

ตัวอย่างที่ผิด

```
score = 80
if score >= 50:
print("ผ่าน")
```

2. ตัวอย่างโจทย์พร้อมโค้ด

โจทย์ที่ 1 การแสดงข้อความทักทาย

โจทย์ จงเขียนโปรแกรมเพื่อแสดงข้อความว่า “ยินดีต้อนรับสู่ภาษาไพทอน”

โค้ดตัวอย่าง

```
print("ยินดีต้อนรับสู่ภาษาไพทอน")
```

โจทย์ที่ 2 รับชื่อและแสดงผล

โจทย์ จงเขียนโปรแกรมรับชื่อจากผู้ใช้ แล้วแสดงข้อความทักทาย

โค้ดตัวอย่าง

```
name = input("กรุณาป้อนชื่อ: ")
print("สวัสดี", name)
```

โจทย์ที่ 3 การหาผลบวกของจำนวน 2 จำนวน

โจทย์ จงเขียนโปรแกรมรับค่าจำนวนเต็ม 2 จำนวน แล้วแสดงผลรวม

โค้ดตัวอย่าง

```
num1 = int(input("ป้อนจำนวนที่ 1: "))
num2 = int(input("ป้อนจำนวนที่ 2: "))
result = num1 + num2
print("ผลรวม =", result)
```

โจทย์ที่ 4 การหาพื้นที่สี่เหลี่ยมผืนผ้า

โจทย์ จงเขียนโปรแกรมรับค่าความกว้างและความยาว แล้วคำนวณหาพื้นที่สี่เหลี่ยมผืนผ้า

โค้ดตัวอย่าง

```
width = float(input("ป้อนความกว้าง: "))
length = float(input("ป้อนความยาว: "))
area = width * length
print("พื้นที่สี่เหลี่ยมผืนผ้า =", area)
```

โจทย์ที่ 5 ตรวจสอบเลขคู่หรือเลขคี่

โจทย์ จงเขียนโปรแกรมรับจำนวนเต็ม 1 จำนวน แล้วตรวจสอบว่าเป็นเลขคู่หรือเลขคี่

โค้ดตัวอย่าง

```
number = int(input("ป้อนจำนวนเต็ม: "))
if number % 2 == 0:
    print("เป็นเลขคู่")
else:
    print("เป็นเลขคี่")
```

โจทย์ที่ 6 ตรวจสอบคะแนนสอบ

โจทย์ จงเขียนโปรแกรมรับคะแนนสอบ แล้วแสดงผลว่าผ่านหรือไม่ผ่าน โดยกำหนดให้ 50 คะแนนขึ้นไปถือว่าผ่าน

โค้ดตัวอย่าง

```
score = int(input("ป้อนคะแนน: "))
if score >= 50:
    print("ผ่าน")
else:
    print("ไม่ผ่าน")
```

โจทย์ที่ 7 แสดงตัวเลข 1 ถึง 10 ด้วย for

โจทย์ จงเขียนโปรแกรมแสดงตัวเลขตั้งแต่ 1 ถึง 10

โค้ดตัวอย่าง

```
for i in range(1, 11):
    print(i)
```

โจทย์ที่ 8 หาผลรวมของตัวเลข 1 ถึง 5

โจทย์ จงเขียนโปรแกรมหาผลรวมของตัวเลขตั้งแต่ 1 ถึง 5

โค้ดตัวอย่าง

```
sum_num = 0
for i in range(1, 6):
    sum_num += i

print("ผลรวม =", sum_num)
```

โจทย์ที่ 9 แม่สูตรคูณ

โจทย์ จงเขียนโปรแกรมรับตัวเลข 1 จำนวน แล้วแสดงสูตรคูณแม่ของจำนวนนั้นตั้งแต่ 1 ถึง 12

โค้ดตัวอย่าง

```
number = int(input("ป้อนแม่สูตรคูณ: "))

for i in range(1, 13):
    print(number, "x", i, "=", number * i)
```

โจทย์ที่ 10 นับถอยหลังด้วย while

โจทย์ จงเขียนโปรแกรมนับถอยหลังจาก 5 ถึง 1

โค้ดตัวอย่าง

```
count = 5

while count >= 1:
    print(count)
    count -= 1
```

3. แบบฝึกหัดเพิ่มเติมสำหรับฝึกปฏิบัติ

1. เขียนโปรแกรมรับอายุของผู้ใช้ แล้วแสดงผลว่าอีก 5 ปี จะมีอายุเท่าไร
2. เขียนโปรแกรมรับรัศมีของวงกลม แล้วคำนวณหาพื้นที่วงกลม
3. เขียนโปรแกรมรับจำนวนเต็ม 1 จำนวน แล้วตรวจสอบว่าเป็นจำนวนบวก จำนวนลบ หรือศูนย์
4. เขียนโปรแกรมแสดงเลขคู่ตั้งแต่ 2 ถึง 20
5. เขียนโปรแกรมหาผลคูณของตัวเลขตั้งแต่ 1 ถึง 5

4. สรุป

การเขียนโปรแกรมภาษาไพทอนสำหรับผู้เริ่มต้นควรเริ่มจากการเรียนรู้ Syntax พื้นฐาน ได้แก่ การแสดงผล การรับข้อมูล ตัวแปร ชนิดข้อมูล การคำนวณ การเปรียบเทียบ การตัดสินใจ และการทำซ้ำ เมื่อเข้าใจหลักการเหล่านี้แล้ว ผู้เรียนจะสามารถพัฒนาโปรแกรมที่มีความซับซ้อนมากขึ้นได้ต่อไป

การฝึกเขียนโปรแกรมจากโจทย์ง่าย ๆ พร้อมดูตัวอย่างโค้ด จะช่วยให้เกิดความเข้าใจทั้งในเชิงโครงสร้างคำสั่งและการประยุกต์ใช้จริง ดังนั้น ผู้เรียนควรฝึกทดลองแก้ไขโค้ด เปลี่ยนค่า และสังเกตผลลัพธ์ เพื่อสร้างความเข้าใจอย่างลึกซึ้งและพัฒนาทักษะการคิดเชิงตรรกะควบคู่กันไป

5. ข้อเสนอแนะในการเรียนรู้เพิ่มเติม

- ฝึกพิมพ์โค้ดด้วยตนเองแทนการคัดลอก
- ทดลองเปลี่ยนค่าตัวแปรและเงื่อนไขเพื่อดูผลลัพธ์
- ตรวจสอบการย่อหน้าให้ถูกต้องทุกครั้ง
- ฝึกอ่านข้อผิดพลาด (Error Message) เพื่อเรียนรู้การแก้ปัญหา
- เริ่มจากโจทย์ง่าย แล้วค่อยเพิ่มความซับซ้อน

เอกสารแนบฉบับนี้สามารถใช้เป็นพื้นฐานสำหรับการเรียนรู้ภาษาไพทอนในบทเริ่มต้น และใช้ต่อยอดสู่การเขียนโปรแกรมในระดับที่สูงขึ้นได้ต่อไป

บทที่ 10

โครงงานทางด้านวิทยาการคำนวณเพื่อการศึกษา

Computational Science projects for education

บทนำ

การเรียนรู้วิทยาการคำนวณในศตวรรษที่ 21 มิได้มุ่งเพียงให้ผู้เรียนเข้าใจแนวคิดพื้นฐานของการคิดเชิงคำนวณ การเขียนโปรแกรม หรือการใช้เทคโนโลยีดิจิทัลเท่านั้น แต่ยังมุ่งส่งเสริมให้ผู้เรียนสามารถนำความรู้เหล่านั้นไปประยุกต์ใช้ในการแก้ปัญหา สร้างสรรค์นวัตกรรม และพัฒนาชิ้นงานที่มีความหมายต่อชีวิตจริงได้อย่างเป็นรูปธรรม หนึ่งในแนวทางสำคัญที่ช่วยให้ผู้เรียนเกิดการบูรณาการความรู้ ทักษะ และประสบการณ์เข้าด้วยกัน คือ การพัฒนา “โครงงานทางด้านวิทยาการคำนวณ” ซึ่งเป็นรูปแบบการเรียนรู้ที่เปิดโอกาสให้ผู้เรียนได้ลงมือคิด วางแผน ออกแบบ สร้าง ทดลอง ปรับปรุง และนำเสนอผลงานด้วยตนเอง

โครงงานทางด้านวิทยาการคำนวณมีความสำคัญอย่างยิ่งต่อการเรียนรู้ในยุคดิจิทัล เพราะช่วยให้ผู้เรียนไม่ได้หยุดอยู่เพียงการฝึกคำสั่งหรือการทำแบบฝึกหัดเฉพาะเรื่อง แต่ได้เรียนรู้จากการแก้ปัญหาในสถานการณ์จริงหรือสถานการณ์จำลองที่มีความหมายต่อชีวิตและบริบทของตน การทำโครงงานช่วยพัฒนาทักษะหลายด้านพร้อมกัน ได้แก่ การคิดวิเคราะห์ การคิดสร้างสรรค์ การทำงานเป็นทีม การวางแผน การจัดการเวลา การใช้เทคโนโลยี และการสื่อสารผลงาน ซึ่งล้วนเป็นสมรรถนะสำคัญของผู้เรียนในศตวรรษที่ 21

สำหรับนักศึกษาครู โครงงานทางด้านวิทยาการคำนวณมีความสำคัญในสองมิติสำคัญ มิติแรก คือ การเป็นผู้พัฒนาโครงงานด้วยตนเอง ซึ่งช่วยให้เข้าใจกระบวนการคิดเชิงโครงงาน การวิเคราะห์ปัญหา การเลือกเครื่องมือ และการสร้างชิ้นงานที่สอดคล้องกับวัตถุประสงค์ มิติที่สอง คือ การสามารถออกแบบโครงงานทางด้านวิทยาการคำนวณเพื่อใช้ในการจัดการเรียนรู้ในสถานศึกษา โดยคำนึงถึงบริบทของผู้เรียน ระดับชั้น และสาระการเรียนรู้ที่เกี่ยวข้อง ครูในยุคใหม่จึงควรมีสมรรถนะในการใช้โครงงานเป็นฐานการเรียนรู้ เพื่อส่งเสริมให้ผู้เรียนเกิดการเรียนรู้อย่างลึกซึ้งและสร้างสรรค์

บทนี้มุ่งอธิบายความหมายและความสำคัญของโครงงานทางด้านวิทยาการคำนวณ ประเภทและกระบวนการพัฒนาโครงงานทางด้านวิทยาการคำนวณเพื่อการศึกษา การวิเคราะห์ปัญหา การกำหนดวัตถุประสงค์ และการออกแบบโครงงาน การดำเนินงาน การทดสอบ และการนำเสนอผลงาน ตลอดจนการออกแบบโครงงานวิทยาการคำนวณเพื่อใช้ในการจัดการเรียนรู้ในสถานศึกษา เพื่อให้นักศึกษาครู

สามารถนำความรู้ไปประยุกต์ใช้ได้ทั้งในฐานะผู้สร้างโครงงานและในฐานะผู้ออกแบบการเรียนรู้ในอนาคต

10.1 ความหมายและความสำคัญของโครงงานทางด้านวิทยาการคำนวณ

โครงงานทางด้านวิทยาการคำนวณ หมายถึง กิจกรรมการเรียนรู้หรือการพัฒนาชิ้นงานที่ผู้เรียนใช้ความรู้ ทักษะ และกระบวนการทางวิทยาการคำนวณในการวิเคราะห์ปัญหา ออกแบบแนวทางแก้ไข สร้างชิ้นงาน ทดสอบ และปรับปรุงผลงาน เพื่อให้ได้ผลลัพธ์ที่ตอบโจทย์วัตถุประสงค์ที่กำหนดไว้ โครงงานลักษณะนี้อาจอยู่ในรูปของโปรแกรม เกม สื่อการเรียนรู้ ระบบอัตโนมัติ อุปกรณ์อัจฉริยะ งานวิเคราะห์ข้อมูล หรือชิ้นงานเชิงนวัตกรรมอื่น ๆ ที่ใช้หลักการของวิทยาการคำนวณเป็นฐาน

ลักษณะสำคัญของโครงงานทางด้านวิทยาการคำนวณ คือ การเปิดโอกาสให้ผู้เรียนได้เรียนรู้จากการลงมือทำจริง โดยเริ่มจากปัญหาหรือความต้องการที่มีความหมาย จากนั้นจึงพัฒนาวิธีการแก้ปัญหาอย่างเป็นระบบ กระบวนการนี้แตกต่างจากการเรียนรู้แบบรับสารเพียงอย่างเดียว เพราะผู้เรียนต้องมีบทบาทในการตั้งคำถาม ค้นคว้าหาข้อมูล ออกแบบแนวคิด ตัดสินใจเลือกวิธีการ และรับผิดชอบต่อผลงานของตนเอง

ความสำคัญของโครงงานทางด้านวิทยาการคำนวณสามารถพิจารณาได้หลายประการ ประการแรก คือ ช่วยให้ผู้เรียนบูรณาการความรู้หลายด้านเข้าด้วยกัน เช่น การคิดเชิงคำนวณ การเขียนโปรแกรม การใช้ข้อมูล การออกแบบเชิงวิศวกรรม และการใช้เทคโนโลยีดิจิทัล ประการที่สอง คือ ช่วยส่งเสริมการเรียนรู้แบบมีความหมาย เพราะผู้เรียนได้ทำงานกับปัญหาที่เกี่ยวข้องกับชีวิตจริงหรือบริบทที่ตนสนใจ ประการที่สาม คือ ช่วยพัฒนาทักษะสำคัญในศตวรรษที่ 21 เช่น การคิดวิเคราะห์ การทำงานร่วมกัน การสื่อสาร และความคิดสร้างสรรค์ และประการที่สี่ คือ ช่วยสร้างทัศนคติที่ดีต่อการเรียนรู้วิทยาการคำนวณ เพราะผู้เรียนเห็นคุณค่าของความรู้ผ่านการสร้างผลงานที่จับต้องได้

ในบริบททางการศึกษา โครงงานทางด้านวิทยาการคำนวณยังเป็นเครื่องมือสำคัญในการเปลี่ยนบทบาทของผู้เรียนจากผู้รับความรู้ไปสู่ผู้สร้างองค์ความรู้ด้วยตนเอง ผู้เรียนได้มีโอกาสเผชิญปัญหาอย่างแท้จริง ฝึกการตัดสินใจและการแก้ปัญหาอย่างเป็นระบบ ขณะเดียวกัน ครูก็เปลี่ยนบทบาทจากผู้ถ่ายทอดสาระไปสู่ผู้อำนวยความสะดวกและผู้ให้คำปรึกษาในกระบวนการเรียนรู้

สำหรับนักศึกษาครู ความเข้าใจในความหมายและความสำคัญของโครงงานทางด้านวิทยาการคำนวณจะช่วยให้สามารถมองเห็นศักยภาพของโครงงานในฐานะเครื่องมือจัดการเรียนรู้ มิใช่เพียง

กิจกรรมปลายภาคหรือกิจกรรมเสริมเท่านั้น แต่เป็นแนวทางที่ช่วยให้ผู้เรียนได้ใช้ความรู้วิทยาการคำนวณอย่างมีเป้าหมายและเกิดการเรียนรู้อย่างลึกซึ้ง

กล่าวโดยสรุป โครงงานทางด้านวิทยาการคำนวณเป็นรูปแบบการเรียนรู้ที่มีความสำคัญอย่างยิ่งต่อการพัฒนาผู้เรียนในยุคดิจิทัล เพราะช่วยส่งเสริมการบูรณาการความรู้ การแก้ปัญหาอย่างสร้างสรรค์ และการพัฒนาสมรรถนะที่จำเป็นต่อการเรียนรู้และการดำรงชีวิตในโลกปัจจุบัน

10.2 ประเภทและกระบวนการพัฒนาโครงงานทางด้านวิทยาการคำนวณเพื่อการศึกษา

โครงงานทางด้านวิทยาการคำนวณเพื่อการศึกษาสามารถจำแนกได้หลายประเภทตามลักษณะของปัญหา วัตถุประสงค์ และรูปแบบของชิ้นงานที่พัฒนา การเข้าใจประเภทของโครงงานจะช่วยให้นักศึกษาครูสามารถเลือกแนวทางที่เหมาะสมกับบริบทของผู้เรียนและเป้าหมายของการจัดการเรียนรู้ได้ดียิ่งขึ้น

ประเภทของโครงงานทางด้านวิทยาการคำนวณเพื่อการศึกษาอาจจำแนกได้ดังนี้

10.2.1 โครงงานประเภทโปรแกรมหรือซอฟต์แวร์อย่างง่าย

เป็นโครงงานที่มุ่งพัฒนาโปรแกรมเพื่อแก้ปัญหาหรืออำนวยความสะดวกในการเรียนรู้ เช่น โปรแกรมคำนวณคะแนน โปรแกรมตอบคำถาม โปรแกรมช่วยทบทวนคำศัพท์ หรือระบบจัดเก็บข้อมูลอย่างง่าย โครงงานลักษณะนี้มักเน้นการออกแบบอัลกอริทึมและการเขียนโปรแกรมเป็นสำคัญ

10.2.2 โครงงานประเภทเกมและสื่อการเรียนรู้

เป็นโครงงานที่ใช้การเขียนโปรแกรมเพื่อสร้างเกมเพื่อการศึกษา แอนิเมชัน นิทานเชิงโต้ตอบหรือสื่อการเรียนรู้ดิจิทัล เช่น เกมฝึกคำนวณ เกมตอบคำถาม หรือสื่ออธิบายเนื้อหาวิชาต่าง ๆ โครงงานประเภทนี้เหมาะอย่างยิ่งสำหรับการบูรณาการกับการจัดการเรียนรู้ในโรงเรียน

10.2.3 โครงงานประเภทอุปกรณ์หรือระบบสมองกลฝังตัว

เป็นโครงงานที่ใช้ไมโครบิตหรืออุปกรณ์อิเล็กทรอนิกส์ร่วมกับการเขียนโปรแกรม เพื่อสร้างอุปกรณ์หรือระบบอัจฉริยะอย่างง่าย เช่น เครื่องวัดอุณหภูมิ ป้ายเตือนระดับเสียง เครื่องนับก้าว หรือระบบแจ้งเตือนอัตโนมัติ โครงงานลักษณะนี้เชื่อมโยงการเขียนโปรแกรมกับโลกจริงได้อย่างชัดเจน

10.2.4 โครงงานประเภทวิเคราะห์ข้อมูล

เป็นโครงงานที่ใช้การรวบรวม จัดการ และวิเคราะห์ข้อมูลเพื่อหาคำตอบของปัญหาหรือสรุปสารสนเทศ เช่น การสำรวจพฤติกรรมการใช้สื่อของนักเรียน การวิเคราะห์ผลการเรียน หรือการนำเสนอ

ข้อมูลในรูปแบบตารางและกราฟ โครงงานลักษณะนี้ช่วยพัฒนาทักษะการใช้ข้อมูลและการตีความผลลัพธ์อย่างมีเหตุผล

10.2.5 โครงงานประเภทนวัตกรรมเพื่อแก้ปัญหาในสถานศึกษา

เป็นโครงงานที่มุ่งตอบโจทย์ปัญหาจริงในโรงเรียนหรือชุมชน เช่น ระบบเตือนเข้าแถว แอปพลิเคชันช่วยทบทวนบทเรียน สื่อช่วยสอนเฉพาะเรื่อง หรืออุปกรณ์ช่วยจัดการห้องเรียน โครงงานลักษณะนี้ทำให้ผู้เรียนเห็นความเชื่อมโยงระหว่างวิทยาการคำนวณกับการพัฒนาสภาพแวดล้อมการเรียนรู้

นอกจากการเข้าใจประเภทของโครงงานแล้ว ผู้เรียนควรเข้าใจกระบวนการพัฒนาโครงงานอย่างเป็นระบบ โดยทั่วไปกระบวนการพัฒนาโครงงานทางด้านวิทยาการคำนวณสามารถอธิบายได้ดังนี้

1. การเลือกหัวข้อหรือปัญหาที่สนใจ
2. การศึกษาข้อมูลและวิเคราะห์ปัญหา
3. การกำหนดวัตถุประสงค์และขอบเขตของโครงงาน
4. การออกแบบวิธีการหรือชิ้นงาน
5. การพัฒนาและดำเนินงานตามแผน
6. การทดสอบและปรับปรุงผลงาน
7. การสรุปผลและนำเสนอผลงาน

กระบวนการดังกล่าวช่วยให้ผู้เรียนทำงานอย่างเป็นลำดับ มีเป้าหมายชัดเจน และสามารถตรวจสอบความก้าวหน้าของโครงงานได้อย่างมีระบบ

สำหรับนักศึกษาครู การเข้าใจประเภทและกระบวนการพัฒนาโครงงานมีความสำคัญอย่างมาก เพราะช่วยให้สามารถออกแบบโครงงานที่เหมาะกับผู้เรียนแต่ละระดับ เลือกประเภทของชิ้นงานที่สอดคล้องกับทรัพยากรที่มี และวางแผนการจัดกิจกรรมการเรียนรู้ที่มีความเป็นไปได้ในบริบทของสถานศึกษา

กล่าวโดยสรุป ประเภทและกระบวนการพัฒนาโครงงานทางด้านวิทยาการคำนวณเพื่อการศึกษามีความหลากหลายและยืดหยุ่น การเข้าใจทั้งสองส่วนนี้จะช่วยให้ผู้เรียนและครูสามารถพัฒนาโครงงานได้อย่างมีทิศทาง มีคุณภาพ และสอดคล้องกับเป้าหมายของการเรียนรู้

10.3 การวิเคราะห์ปัญหา การกำหนดวัตถุประสงค์ และการออกแบบโครงงาน

การวิเคราะห์ปัญหา การกำหนดวัตถุประสงค์ และการออกแบบโครงงาน เป็นขั้นตอนสำคัญที่กำหนดทิศทางและคุณภาพของโครงงานทางด้านวิทยาการคำนวณ หากผู้เรียนเริ่มต้นโครงงานโดยไม่เข้าใจปัญหาอย่างชัดเจน หรือกำหนดวัตถุประสงค์ไม่เฉพาะเจาะจง อาจทำให้การดำเนินงานขาดเป้าหมาย ชิ้นงานไม่ตอบโจทย์ หรือเกิดความสับสนระหว่างการพัฒนา ดังนั้น ขั้นตอนเริ่มต้นของโครงงานจึงควรได้รับการวางแผนอย่างรอบคอบ

การวิเคราะห์ปัญหา (Problem Analysis) หมายถึง การทำความเข้าใจสถานการณ์หรือความต้องการที่เป็นจุดเริ่มต้นของโครงงาน ผู้เรียนควรพิจารณาว่า ปัญหานั้นคืออะไร เกิดขึ้นกับใคร มีสาเหตุจากอะไร และเหตุใดจึงควรได้รับการแก้ไข ในบริบททางการศึกษา ปัญหาอาจเกี่ยวข้องกับการเรียนรู้ของผู้เรียน การจัดกิจกรรมในชั้นเรียน การเข้าถึงสื่อการเรียนรู้ หรือความยากในการทำความเข้าใจเนื้อหาบางเรื่อง การตั้งต้นจากปัญหาที่มีความหมายจะช่วยให้โครงงานมีคุณค่าและสามารถตอบสนองต่อบริบทจริงได้

เมื่อวิเคราะห์ปัญหาแล้ว ขั้นต่อไปคือการกำหนดวัตถุประสงค์ของโครงงาน (Objectives) วัตถุประสงค์ควรระบุให้ชัดเจนว่า โครงงานนี้ต้องการพัฒนาอะไร เพื่อใคร และเพื่อให้เกิดผลอย่างไร เช่น เพื่อสร้างสื่อการเรียนรู้วิชาคณิตศาสตร์สำหรับนักเรียนชั้นประถมศึกษา เพื่อพัฒนาเกมทบทวนคำศัพท์ภาษาอังกฤษ หรือเพื่อสร้างอุปกรณ์แจ้งเตือนระดับเสียงในห้องเรียน วัตถุประสงค์ที่ชัดเจนจะช่วยให้การออกแบบและการประเมินผลของโครงงานเป็นไปอย่างมีทิศทาง

การออกแบบโครงงาน (Project Design) เป็นขั้นตอนที่ผู้เรียนนำผลจากการวิเคราะห์ปัญหาและวัตถุประสงค์มาพิจารณาเป็นแนวทางการพัฒนา โดยผู้เรียนต้องตัดสินใจว่าจะใช้เครื่องมือใด ใช้เทคโนโลยีรูปแบบใด กำหนดคุณลักษณะของชิ้นงานอย่างไร และวางแผนการดำเนินงานอย่างไร การออกแบบที่ดีควรคำนึงถึงความเป็นไปได้ ความเหมาะสมกับเวลาและทรัพยากร และความสอดคล้องกับวัตถุประสงค์ของโครงงาน

องค์ประกอบสำคัญที่ควรพิจารณาในการออกแบบโครงงาน ได้แก่

1. กลุ่มเป้าหมายของชิ้นงาน
2. ขอบเขตของโครงงาน
3. เครื่องมือหรือแพลตฟอร์มที่ใช้
4. ขั้นตอนการพัฒนา
5. เกณฑ์ในการทดสอบและประเมินผล

6. รูปแบบการนำเสนอผลงาน

ตัวอย่างเช่น หากนักศึกษาครูต้องการพัฒนาเกมการศึกษาด้วย Scratch สำหรับนักเรียนประถมศึกษา ต้องเริ่มจากการวิเคราะห์ว่า นักเรียนมีปัญหาในการเรียนรู้เรื่องใด กำหนดวัตถุประสงค์ว่าต้องการให้เกมช่วยเสริมทักษะอะไร จากนั้นจึงออกแบบตัวละคร ฉาก รูปแบบการเล่น การให้คะแนน และวิธีประเมินว่าเกมนั้นช่วยส่งเสริมการเรียนรู้ได้จริงหรือไม่

สำหรับนักศึกษาครู การฝึกวิเคราะห์ปัญหาและออกแบบโครงงานมีความสำคัญอย่างยิ่ง เพราะเป็นทักษะที่เชื่อมโยงโดยตรงกับการออกแบบการเรียนรู้ในอนาคต ครูที่มีทักษะในด้านนี้จะสามารถวิเคราะห์ความต้องการของผู้เรียน กำหนดเป้าหมายการเรียนรู้ และออกแบบกิจกรรมหรือสื่อที่ตอบโจทย์ได้อย่างเป็นระบบ

กล่าวโดยสรุป การวิเคราะห์ปัญหา การกำหนดวัตถุประสงค์ และการออกแบบโครงงานเป็นฐานสำคัญของการพัฒนาโครงงานทางด้านวิทยาการคำนวณ เพราะช่วยให้โครงงานมีทิศทางชัดเจน ตอบโจทย์ปัญหาจริง และสามารถดำเนินการได้อย่างเป็นระบบและมีประสิทธิภาพ

10.4 การดำเนินงาน การทดสอบ และการนำเสนอผลงานโครงงาน

หลังจากวิเคราะห์ปัญหา กำหนดวัตถุประสงค์ และออกแบบโครงงานแล้ว ขั้นตอนต่อไปคือการดำเนินงาน การทดสอบ และการนำเสนอผลงาน ซึ่งเป็นช่วงที่ผู้เรียนจะได้แปลงแนวคิดไปสู่การปฏิบัติจริง กระบวนการในระยะนี้มีความสำคัญอย่างมาก เพราะเป็นจุดที่ทำให้โครงงานปรากฏเป็นรูปธรรม และเปิดโอกาสให้ผู้เรียนได้ฝึกทักษะการลงมือทำ การแก้ปัญหาเฉพาะหน้า และการสื่อสารผลงานอย่างเป็นระบบ

การดำเนินงาน (Implementation) หมายถึง การลงมือพัฒนาโครงงานตามแผนที่วางไว้ โดยอาจประกอบด้วยการสร้างโปรแกรม การออกแบบสื่อ การเขียนคำสั่ง การเชื่อมต่ออุปกรณ์ การเก็บรวบรวมข้อมูล หรือการสร้างชิ้นงานต้นแบบ ขั้นตอนนี้ผู้เรียนควรทำงานอย่างมีลำดับและบันทึกความก้าวหน้าอย่างต่อเนื่อง เพื่อให้สามารถตรวจสอบและปรับปรุงงานได้ตลอดกระบวนการ

ในการดำเนินงานจริง ผู้เรียนอาจพบปัญหาหรือข้อจำกัดที่ไม่ได้คาดการณ์ไว้ เช่น เครื่องมือไม่ทำงานตามที่ตั้งใจ โปรแกรมมีข้อผิดพลาด เวลาไม่เพียงพอ หรือแนวคิดเดิมไม่เหมาะสมกับสถานการณ์จริง การเผชิญปัญหาเหล่านี้ไม่ควรถูกมองว่าเป็นความล้มเหลว แต่เป็นส่วนหนึ่งของกระบวนการเรียนรู้ที่สำคัญ เพราะช่วยให้ผู้เรียนได้ฝึกคิด ปรับแผน และตัดสินใจแก้ไขปัญหาอย่างยืดหยุ่น

การทดสอบ (Testing) เป็นขั้นตอนที่มีความสำคัญอย่างมากต่อคุณภาพของโครงงาน ผู้เรียนควรตรวจสอบว่า ชิ้นงานทำงานได้ตรงตามวัตถุประสงค์หรือไม่ ใช้งานได้ถูกต้องหรือไม่ มีข้อผิดพลาดใดเกิดขึ้น และกลุ่มเป้าหมายสามารถใช้งานได้สะดวกหรือไม่ การทดสอบอาจทำได้หลายรูปแบบ เช่น การทดลองใช้งานด้วยตนเอง การให้เพื่อนหรือผู้ใช้เป้าหมายทดลองใช้ การตรวจสอบผลลัพธ์ในหลายสถานการณ์ หรือการเปรียบเทียบกับเกณฑ์ที่กำหนดไว้ตั้งแต่ต้น

เมื่อพบข้อบกพร่องจากการทดสอบ ผู้เรียนควรนำข้อมูลที่ได้มาปรับปรุงผลงานให้ดีขึ้น เช่น แก้ไขคำสั่งที่ผิด ปรับหน้าจอให้ใช้งานง่ายขึ้น เพิ่มคำอธิบาย หรือปรับลำดับการทำงานของระบบ กระบวนการปรับปรุงนี้สะท้อนแนวคิดสำคัญของการพัฒนางานทางวิทยาการคำนวณ คือ การทำงานอย่างต่อเนื่องและยอมรับการแก้ไขเพื่อให้ชิ้นงานมีประสิทธิภาพมากขึ้น

การนำเสนอผลงาน (Project Presentation) เป็นขั้นตอนที่ทำให้ผู้เรียนได้สื่อสารแนวคิด กระบวนการ และผลลัพธ์ของโครงงานให้ผู้อื่นเข้าใจ ซึ่งมีความสำคัญไม่แพ้การพัฒนาชิ้นงานเอง ผู้เรียนควรนำเสนอให้ครอบคลุมประเด็นสำคัญ เช่น ปัญหาที่เลือกศึกษา วัตถุประสงค์ วิธีดำเนินงาน เครื่องมือที่ใช้ ผลการพัฒนา ข้อดี ข้อจำกัด และข้อเสนอแนะในการพัฒนาต่อ การนำเสนออาจอยู่ในรูปแบบรายงาน สไลด์ วิดีโอ การสาธิตการทำงานของชิ้นงาน หรือการจัดนิทรรศการผลงาน

สำหรับนักศึกษาครู การฝึกนำเสนอผลงานโครงงานมีความสำคัญอย่างยิ่ง เพราะช่วยพัฒนาทักษะการสื่อสารเชิงวิชาการ การอธิบายเหตุผล และการสะท้อนการเรียนรู้ของตนเอง นอกจากนี้ ยังช่วยเตรียมความพร้อมสำหรับบทบาทครูที่ต้องสามารถอธิบายแนวคิดทางวิชาการและจัดการเรียนรู้แบบใช้โครงงานเป็นฐานในอนาคต

กล่าวโดยสรุป การดำเนินงาน การทดสอบ และการนำเสนอผลงานโครงงานเป็นกระบวนการสำคัญที่ทำให้แนวคิดของผู้เรียนกลายเป็นผลงานที่มีคุณภาพและสามารถสื่อสารคุณค่าของโครงงานออกไปได้อย่างชัดเจน ทั้งยังเป็นช่วงที่ผู้เรียนได้พัฒนาทักษะเชิงปฏิบัติและทักษะการสะท้อนคิดอย่างรอบด้าน

10.5 การออกแบบโครงงานวิทยาการคำนวณเพื่อใช้ในการจัดการเรียนรู้ในสถานศึกษา

การออกแบบโครงงานวิทยาการคำนวณเพื่อใช้ในการจัดการเรียนรู้ในสถานศึกษา เป็นทักษะสำคัญสำหรับนักศึกษาครู เพราะช่วยให้สามารถเปลี่ยนโครงงานจากกิจกรรมเชิงเทคนิคไปสู่กิจกรรมการเรียนรู้ที่มีเป้าหมายทางการศึกษาชัดเจน การออกแบบในลักษณะนี้มิได้มุ่งเพียงให้ผู้เรียนสร้างชิ้นงานได้

เท่านั้น แต่ต้องคำนึงถึงผลลัพธ์การเรียนรู้ พัฒนาการของผู้เรียน บริบทของโรงเรียน และความเป็นไปได้ในการนำไปใช้จริงในชั้นเรียน

หลักการสำคัญประการแรก คือ การออกแบบโครงงานให้สอดคล้องกับระดับชั้นและความพร้อมของผู้เรียน โครงงานสำหรับนักเรียนประถมศึกษาควรมีความเรียบง่าย เน้นความสนุก เห็นผลได้เร็ว และใช้เครื่องมือที่ไม่ซับซ้อนมาก เช่น Scratch หรือไมโครบิตในกิจกรรมพื้นฐาน ส่วนโครงงานสำหรับนักเรียนมัธยมศึกษาสามารถเพิ่มความซับซ้อนในด้านการวิเคราะห์ปัญหา การใช้เงื่อนไข การทำซ้ำ การใช้ข้อมูล หรือการสร้างชิ้นงานที่มีปฏิสัมพันธ์มากขึ้น

หลักการประการที่สอง คือ การเชื่อมโยงโครงงานกับบริบทจริงในสถานศึกษา เช่น ปัญหาในห้องเรียน ความต้องการของผู้เรียน หรือกิจกรรมในโรงเรียน ตัวอย่างเช่น การสร้างเกมทบทวนบทเรียน การสร้างสื่ออธิบายเนื้อหายาก การออกแบบระบบเตือนเวลาเข้าแถว หรือการพัฒนาอุปกรณ์ช่วยดูแลสิ่งแวดล้อมในโรงเรียน โครงงานที่เชื่อมโยงกับบริบทจริงจะช่วยให้ผู้เรียนเห็นคุณค่าของสิ่งที่ทำ และเกิดแรงจูงใจในการเรียนรู้มากขึ้น

หลักการประการที่สาม คือ การวางแผนโครงงานให้สอดคล้องกับกระบวนการเรียนรู้ ครูควรกำหนดว่าผู้เรียนจะได้เรียนรู้อะไรจากการทำโครงงาน เช่น การวิเคราะห์ปัญหา การออกแบบอัลกอริทึม การเขียนโปรแกรม การทำงานร่วมกัน หรือการนำเสนอผลงาน จากนั้นจึงจัดลำดับกิจกรรมให้ผู้เรียนได้พัฒนาทักษะเหล่านี้อย่างเป็นขั้นตอน ตั้งแต่การเลือกหัวข้อ การศึกษาข้อมูล การออกแบบ การพัฒนา การทดสอบ และการสะท้อนผล

หลักการประการที่สี่ คือ การออกแบบการประเมินผลให้ครอบคลุม ไม่ควรประเมินเพียงความสำเร็จของชิ้นงานสุดท้าย แต่ควรประเมินกระบวนการทำงานด้วย เช่น ความสามารถในการตั้งปัญหา การวางแผน การเลือกใช้เครื่องมือ การแก้ปัญหาระหว่างทำงาน การทำงานเป็นทีม และการอธิบายผลงาน การประเมินเช่นนี้จะช่วยให้ผู้เรียนให้ความสำคัญกับกระบวนการเรียนรู้มากกว่าการมุ่งเพียงผลลัพธ์สุดท้าย

ตัวอย่างของโครงงานวิทยาการคำนวณที่สามารถใช้ในการจัดการเรียนรู้ในสถานศึกษา ได้แก่

1. เกมการศึกษาเพื่อทบทวนบทเรียนในรายวิชาต่าง ๆ
2. สื่อแอนิเมชันหรือสื่อโต้ตอบอธิบายเนื้อหาที่ยาก
3. ระบบแจ้งเตือนด้วยไมโครบิตในกิจกรรมของโรงเรียน
4. โปรแกรมคำนวณหรือช่วยจัดการข้อมูลในห้องเรียน
5. โครงงานวิเคราะห์ข้อมูลอย่างง่ายจากบริบทของโรงเรียน

สำหรับนักศึกษาครู การออกแบบโครงงานเพื่อใช้ในการจัดการเรียนรู้ควรเชื่อมโยงกับการจัดทำแผนการจัดการเรียนรู้ โดยกำหนดผลลัพธ์การเรียนรู้ จุดประสงค์ กิจกรรม สื่อ เครื่องมือ และเกณฑ์ประเมินอย่างชัดเจน การทำเช่นนี้จะช่วยให้โครงงานไม่ใช่เพียงกิจกรรมสร้างชิ้นงาน แต่เป็นกระบวนการเรียนรู้ที่มีเป้าหมาย มีความหมาย และสามารถนำไปใช้ได้จริงในสถานศึกษา

กล่าวโดยสรุป การออกแบบโครงงานวิทยาการคำนวณเพื่อใช้ในการจัดการเรียนรู้ในสถานศึกษาเป็นกระบวนการที่ต้องอาศัยทั้งความเข้าใจในเนื้อหาวิทยาการคำนวณและศาสตร์การสอน ครูที่มีทักษะในด้านนี้จะสามารถสร้างการเรียนรู้ที่บูรณาการความรู้ ทักษะ และการลงมือปฏิบัติได้อย่างมีคุณภาพ และช่วยให้ผู้เรียนเห็นคุณค่าของวิทยาการคำนวณในฐานะเครื่องมือสร้างสรรค์และแก้ปัญหาในชีวิตจริง

บทสรุปประจำบทที่ 10

โครงงานทางด้านวิทยาการคำนวณเพื่อการศึกษาเป็นรูปแบบการเรียนรู้ที่มีคุณค่าอย่างยิ่งต่อการพัฒนาผู้เรียนในศตวรรษที่ 21 เพราะเปิดโอกาสให้ผู้เรียนได้บูรณาการความรู้ด้านการคิดเชิงคำนวณ การเขียนโปรแกรม การใช้เทคโนโลยี และการแก้ปัญหาเข้าด้วยกันผ่านการลงมือสร้างชิ้นงานจริง ในบทนี้ได้อธิบายความหมายและความสำคัญของโครงงาน ประเภทและกระบวนการพัฒนาโครงงาน การวิเคราะห์ปัญหา การกำหนดวัตถุประสงค์ การออกแบบโครงงาน การดำเนินงาน การทดสอบ และการนำเสนอผลงาน ตลอดจนแนวทางการออกแบบโครงงานเพื่อใช้ในการจัดการเรียนรู้ในสถานศึกษา

สำหรับนักศึกษาครู การเรียนรู้เรื่องโครงงานทางด้านวิทยาการคำนวณมีความสำคัญทั้งในฐานะผู้พัฒนาโครงงานและในฐานะผู้ออกแบบการเรียนรู้ เพราะช่วยให้สามารถใช้โครงงานเป็นฐานในการจัดการเรียนรู้ที่ส่งเสริมการคิด วิเคราะห์ สร้างสรรค์ และการทำงานร่วมกันได้อย่างมีความหมาย หากครูสามารถออกแบบโครงงานที่สอดคล้องกับบริบทของผู้เรียนและสถานศึกษาได้อย่างเหมาะสม โครงงานจะกลายเป็นกลไกสำคัญในการทำให้วิทยาการคำนวณเป็นการเรียนรู้ที่มีชีวิต มีคุณค่า และเชื่อมโยงกับโลกจริงอย่างแท้จริง

คำถามท้ายบท

1. จงอธิบายความหมายของโครงงานทางด้านวิทยาการคำนวณ และชี้ให้เห็นว่ามีความสำคัญต่อการเรียนรู้อย่างไร
2. โครงงานทางด้านวิทยาการคำนวณเพื่อการศึกษาสามารถแบ่งออกเป็นประเภทใดบ้าง พร้อมยกตัวอย่าง
3. กระบวนการพัฒนาโครงงานทางด้านวิทยาการคำนวณมีขั้นตอนอะไรบ้าง
4. เพราะเหตุใดการวิเคราะห์ปัญหาและการกำหนดวัตถุประสงค์จึงมีความสำคัญต่อการพัฒนาโครงงาน
5. จงอธิบายองค์ประกอบที่ควรพิจารณาในการออกแบบโครงงานวิทยาการคำนวณ
6. การทดสอบโครงงานมีบทบาทอย่างไรต่อคุณภาพของชิ้นงาน
7. การนำเสนอผลงานโครงงานควรครอบคลุมประเด็นใดบ้าง
8. หากท่านเป็นครูระดับประถมศึกษา ท่านจะออกแบบโครงงานวิทยาการคำนวณแบบใดให้เหมาะกับผู้เรียน เพราะเหตุใด
9. หากท่านเป็นครูระดับมัธยมศึกษา ท่านจะใช้โครงงานวิทยาการคำนวณเพื่อบูรณาการกับรายวิชาอื่นอย่างไร
10. จงอธิบายบทบาทของครูในการใช้โครงงานวิทยาการคำนวณเป็นฐานการเรียนรู้ในสถานศึกษา

เอกสารอ้างอิง

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

Bell, T., Andreae, P., & Robins, A. (2014). A case study of the introduction of computer science in NZ schools. *ACM Transactions on Computing Education, 14*(2), 1–31.

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 Annual Meeting of the American Educational Research Association*. Vancouver, Canada.

Krajcik, J. S., & Blumenfeld, P. C. (2006). Project-based learning. In R. K. Sawyer (Ed.), *The Cambridge handbook of the learning sciences* (pp. 317–334). New York: Cambridge University Press.

Thomas, J. W. (2000). *A review of research on project-based learning*. San Rafael, CA: Autodesk Foundation.

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