Chulalongkorn University
Chula Digital Collections

Chulalongkorn University Theses and Dissertations (Chula ETD)

2020

การพัฒนาบทเรียนออนไลน์เพื่อส่งเสริมความสามารถด้านการเขียนโค้ดโดยใช้การเรียนรู้แบบจุลภาคสำหรับนักศึกษาครู : การวิจัยเชิงทดลองแบบปรับเหมาะ

กฤษณ์ วิริยะสิทธารถ
คณะครุศาสตร์

Follow this and additional works at: https://digital.car.chula.ac.th/chulaetd

Part of the Educational Assessment, Evaluation, and Research Commons

Recommended Citation

วิริยะสิทธารถ, กฤษณ์, "การพัฒนาบทเรียนออนไลน์เพื่อส่งเสริมความสามารถด้านการเขียนโค้ดโดยใช้การเรียนรู้แบบจุลภาคสำหรับนักศึกษาครู : การวิจัยเชิงทดลองแบบปรับเหมาะ" (2020). *Chulalongkorn University Theses and Dissertations (Chula ETD)*. 3713.
https://digital.car.chula.ac.th/chulaetd/3713

This Thesis is brought to you for free and open access by Chula Digital Collections. It has been accepted for inclusion in Chulalongkorn University Theses and Dissertations (Chula ETD) by an authorized administrator of Chula Digital Collections. For more information, please contact ChulaDC@car.chula.ac.th.

การพัฒนาบทเรียนออนไลน์เพื่อส่งเสริมความสามารถด้านการเขียนโค้ดโดยใช้การเรียนรู้แบบจุลภาค สำหรับนักศึกษาครู : การวิจัยเชิงทดลองแบบปรับเหมาะ

นายกฤษณ์ วิริยะสิทธารถ

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาครุศาสตรมหาบัณฑิต
สาขาวิชาวิธีวิทยาการวิจัยการศึกษา ภาควิชาวิจัยและจิตวิทยาการศึกษา
คณะครุศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2563

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

DEVELOPMENT OF ONLINE COURSES FOR ENHANCING CODING ABILITIES WITH MICROLEARNING FOR STUDENT TEACHERS: ADAPTIVE EXPERIMENTAL RESEARCH

Mr. Krit Viriyasittharod

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Education in Educational Research Methodology
Department of Educational Research and Psychology
FACULTY OF EDUCATION
Chulalongkorn University
Academic Year 2020
Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การพัฒนาบทเรียนออนไลน์เพื่อส่งเสริมความสามารถด้านการเขียนโค้ดโดยใช้การเรียนรู้แบบจุลภาคสำหรับนักศึกษาครู : การวิจัยเชิงทดลองแบบปรับเหมาะ
โดย	นายกฤษณ์ วิริยะสิทธารถ
สาขาวิชา	วิธีวิทยาการวิจัยการศึกษา
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	อาจารย์ ดร.ชยุตม์ ภิรมย์สมบัติ

คณะครุศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาครุศาสตรมหาบัณฑิต

..... คณบดีคณะครุศาสตร์

(รองศาสตราจารย์ ดร.ศิริเดช สุชีวะ)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ

(รองศาสตราจารย์ ดร.ศจีมาจ ณ วิเชียร)

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

(อาจารย์ ดร.ชยุตม์ ภิรมย์สมบัติ)

..... กรรมการ

(อาจารย์ ดร.สิวะโชติ ศรีสุทธิยากร)

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

กฤษณ์ วิริยะสิทธารถ : การพัฒนาบทเรียนออนไลน์เพื่อส่งเสริมความสามารถด้านการเขียนโค้ดโดยใช้การเรียนรู้แบบจุลภาคสำหรับนักศึกษาครู : การวิจัยเชิงทดลองแบบปรับเหมาะ. (DEVELOPMENT OF ONLINE COURSES FOR ENHANCING CODING ABILITIES WITH MICROLEARNING FOR STUDENT TEACHERS: ADAPTIVE EXPERIMENTAL RESEARCH) อ.ที่ปรึกษาหลัก : อ. ดร.ชยุตม์ ภิรมย์สมบัติ

ความสามารถหนึ่งที่สำคัญในยุคดิจิทัลสำหรับนักศึกษาครู คือ ความสามารถด้านการเขียนโค้ด ซึ่งนักศึกษาครูสามารถได้รับการพัฒนาด้วยบทเรียนออนไลน์ที่สร้างขึ้นตามแนวคิดการเรียนรู้จุลภาคที่ตอบสนองความต้องและปรับเหมาะกับความสามารถพื้นฐานของแต่ละคน การวิจัยครั้งนี้มีวัตถุประสงค์เพื่อ 1) เปรียบเทียบความสามารถด้านการเขียนโค้ดของนักศึกษาครูที่มีภูมิหลังแตกต่างกัน 2) ออกแบบและพัฒนาบทเรียนออนไลน์ด้านการเขียนโค้ดโดยใช้การเรียนรู้แบบจุลภาค และ 3) เปรียบเทียบโปรไฟล์และความสามารถด้านการเขียนโค้ดระหว่างได้รับบทเรียนออนไลน์ด้านการเขียนโค้ดที่ออกแบบตามการเรียนรู้แบบจุลภาคที่มีรูปแบบแตกต่างกันของนักศึกษาครูที่มีภูมิหลังแตกต่างกัน การวิจัยครั้งนี้เป็นการวิจัยเชิงทดลองแบบปรับเหมาะเพื่อส่งเสริมความสามารถด้านการเขียนโค้ดของนักศึกษาครู โดยกลุ่มทดลอง คือ นักศึกษาครูชั้นปีที่ 1-4 กำหนดขนาดตัวอย่าง 30 คน ซึ่งได้มาจากการเลือกตามสะดวกเครื่องมือวิจัย ประกอบด้วย แบบสำรวจความพึงพอใจต่อลักษณะบทเรียนการเรียนรู้แบบจุลภาค และแบบทดสอบออนไลน์เกี่ยวกับความสามารถด้านการเขียนโค้ด ผลการวิจัยสรุปได้ดังนี้

1. คะแนนรวมและคะแนนองค์ประกอบด้านโปรแกรมระหว่างกลุ่มแตกต่างกันอย่างมีนัยสำคัญทางสถิติ เมื่อเปรียบเทียบระหว่างกลุ่มที่เคยเรียนการเขียนโค้ดกับกลุ่มที่ไม่เคยเรียนการเขียนโค้ด กลุ่มวิชาเอกคอมพิวเตอร์กับลุ่มวิชาเอกอื่น ๆ กลุ่มวิชาเอกคณิตศาสตร์-วิทยาศาสตร์กับกลุ่มวิชาเอกอื่น ๆ

2. บทเรียนออนไลน์ด้านการเขียนโค้ดที่ออกแบบโดยแนวคิดการเรียนรู้แบบจุลภาคประกอบด้วย 3 โมดูลได้แก่ โมดูลการเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ โมดูล ภาษา Scratch และ โมดูลภาษา Python ซึ่งแต่ละโมดูลก่อนเรียนจะได้รับการประเมินความสามารถด้านการเขียนโค้ดว่าอยู่ในระดับใดเพื่อที่จะได้รับบทเรียนที่ตอบสนองความต้องการของผู้เรียน บทเรียนออนไลน์ด้านการเขียนโค้ดออกแบบขึ้นบนแพลตฟอร์มการเรียนรู้ออนไลน์ Canvas learning management system (Canvas lms) โดยมีสื่อการเรียนรู้ที่สร้างขึ้นตามแนวคิดการเรียนรู้แบบจุลภาค ประกอบด้วย ข้อความบรรยาย เกม วิดีโอ วิดีโอเชิงโต้ตอบ (interactive video) เอกสารเชิงโต้ตอบ (interactive slide) โปรแกรมเขียนผังงานบนเว็บไซต์ draw.io เว็บเขียนภาษา Scratch Microsoft MakeCode arcade และหน้าต่างการเขียนโค้ดเชิงโต้ตอบบนเว็บ (interactive Python Console)

3. ผู้เข้าร่วมวิจัยได้รับการจำแนกตามความสามารถด้านการเขียนโค้ดเป็นกลุ่มต่ำ (L) กลุ่มกลาง (M) และกลุ่มสูง (H) เช่น LHM เป็นผู้เรียนที่เริ่มจากความสามารถในโมดูลที่ 1 ต่ำ และมีความสามารถปานกลางในโมดูลที่ 2 และมีความสามารถสูงในโมดูลที่ 3 พบว่า กลุ่ม MLL ได้คะแนนสัมพัทธ์สูงที่สุดเท่ากับ .57 น้อยที่สุดคือ กลุ่ม HML ได้คะแนนเท่ากับ .00 กลุ่ม HMM ได้คะแนนสูงที่สุด ร้อยละ 100 ในโมดูลที่ 1 การเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ แต่ HML ได้คะแนนต่ำที่สุดร้อยละ 25 กลุ่ม MHH ได้คะแนนสูงที่สุด ร้อยละ 71.25 ในโมดูลที่ 2 ภาษา Scratch และ กลุ่ม HHH ได้คะแนนสูงที่สุด ร้อยละ 68.75 ในโมดูลที่ 3 ภาษา Python ขณะที่กลุ่ม HML ได้คะแนนต่ำที่สุด ทั้งในโมดูลที่ 2 และ 3

สาขาวิชา	วิธีวิทยาการวิจัยการศึกษา	ลายมือชื่อนิสิต .....
ปีการศึกษา	2563	ลายมือชื่อ อ.ที่ปรึกษาหลัก .....

CHULALONGKORN UNIVERSITY

6280005327 : MAJOR EDUCATIONAL RESEARCH METHODOLOGY
KEYWORD: ADAPTIVE EXPERTIMENTAL RESEARCH, CODING ABILITY, MICROLEARNING

Krit Viriyasittharod : DEVELOPMENT OF ONLINE COURSES FOR ENHANCING CODING ABILITIES WITH MICROLEARNING FOR STUDENT TEACHERS: ADAPTIVE EXPERIMENTAL RESEARCH. Advisor: CHAYUT PIROMSOMBAT, Ph.D.

One of the most important abilities in digital age for student teacher is coding abilities. Student teacher can be developed with online courses built on microlearning concept and adapt from their basic abilities. This research aims to 1) compare coding abilities of student teacher in different backgrounds. 2) design and develop online courses using microlearning 3) compare profile and coding abilities of student teacher who learning on coding course in different microlearning style and backgrounds. This research is adaptive experimental research that develop coding abilities of student teacher. The experimental group consisted of 30 first year student teacher to senior has been collected using the convenience sampling. The research instruments were a microlearning satisfaction survey and coding abilities online test. The key finding can be summarizing as follows:

1. Total scores and program component scores between groups were statistically different when compared to those who had learned to code with those who had not learned to code, computer major and other, major mathematics-science major and other major.

2. This online coding courses designed around the microlearning concept consists of three modules: the unplugged coding module, the Scratch module, and the Python module. for each of modules pretest was assessed at what level of coding abilities in order to receive lessons that suitable for learners' needs. Online coding coures designed on an online learning platform (Canvas LMS). There are learning materials created according to the concept of microlearning, consisting of text, games, video, interactive video, interactive slides, draw.io, Microsoft MakeCode Arcade and interactive Python console.

3. Comparison of pre -post coding abilities using relative score found that the MLL group got the highest relative score of .57, the lowest was the HML group got the score of .00. The scores on coding abilities during the course were found that module 1: unplugged Coding, the HMM group scored the highest 8 points, representing 100%, and the HML scored the lowest 2 points, representing 25%. Module 2: Scratch, the MHH group received the highest score of 5.70, or 71.25%, and HML, the lowest score of 0, with 0%, and module 3: Python, the HHH group, had the highest score of 5.50, or 68.75%, and the HML. The lowest score is 1 point, or 12.50 %.3. The participants in this study were classified based on their performance (L = low, M = moderate, and H = high) on coding across learning routes. For example, LMH refers to students who start with low performance in Module 1, moderate level in Module 2, and high level in Module 3. It was found that the students in MML group had the highest relative score (M = .57), while those in HML group had the lowest score (M = .00). The HMM group had the highest level of performance (100%) in Module 1 (Unplugged Coding), but the HML had the lowest (25%). The MHH group performed best (71.25%) in Module 2 (Scratch Coding) while the HHH group outperformed others (68.75%) in Module 3 (Python Coding). The HML group had the lowest performance in both Modules 2 & 3.

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

Field of Study:	Educational Research Methodology	Student's Signature .....
Academic Year:	2020	Advisor's Signature .....

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยความเมตตาและความกรุณาอย่างสูงยิ่งจาก อาจารย์ ดร.ชยุตม์ ภิรมย์สมบัติ อาจารย์ที่ปรึกษาวิทยานิพนธ์ที่ได้เสียสละเวลาให้แนวคิด วิธีการ และกระบวนการทำงานแบบนักวิจัย ให้คำปรึกษาในการปรับปรุงแก้ไขข้อบกพร่องต่าง ๆ ของงานวิจัยด้วยความเอาใจใส่ ผู้วิจัยซาบซึ้งและขอกราบขอบพระคุณมา ณ โอกาสนี้

ขอกราบขอบพระคุณรองศาสตราจารย์ ดร.ศจีมาจ ณ วิเชียร ประธานกรรมการสอบวิทยานิพนธ์ และอาจารย์ ดร.สิวะโชติ ศรีสุทธิยากร กรรมการสอบวิทยานิพนธ์ที่ได้ให้ข้อเสนอแนะที่เป็นประโยชน์ในการแก้ไขวิทยานิพนธ์ฉบับนี้ให้มีความสมบูรณ์ยิ่งขึ้น และขอขอบคุณอาจารย์ทุก ๆ ท่านในภาควิชาวิจัยและจิตวิทยาการศึกษา ที่ได้ให้ความรู้ คำแนะนำ และให้ความช่วยเหลือตลอดระยะเวลาที่กำลังศึกษาและทำวิทยานิพนธ์ฉบับนี้

ขอขอบคุณอาจารย์ ธีรพล พากเพียรกิจ นางสาวพรภัทร จตุพร และนางสาวจุฑามาศ คำมั่น ที่คอยให้ความช่วยเหลือในการทำวิทยานิพนธ์ ประสานงานในการเก็บข้อมูล ให้คำแนะนำ และคอยให้กำลังตลอดในช่วงระยะเวลาที่ทำวิทยานิพนธ์ฉบับนี้ และขอขอบคุณผู้ที่เกี่ยวข้องทุก ๆ คน รวมถึงผู้ที่ให้ข้อมูล และสละเวลามาเป็นกลุ่มทดลอง จนทำให้วิทยานิพนธ์ฉบับนี้เสร็จสมบูรณ์ไปได้ด้วยดี

ขอขอบคุณนางสาวกัทลี จันดี นางสาวจุฑามาศ เตชะภัททวรกุล เพื่อน พี่ และน้องในสาขาวิธีวิทยาการวิจัยการศึกษา ทุก ๆ คน ที่ให้ความช่วยเหลือและคำแนะนำตลอดมา

สุดท้ายนี้ขอกราบขอบพระคุณคุณพ่อสุรพล วิริยะสิทธารถ และคุณแม่ทิพวัลย์ วิริยะสิทธารถ ที่เป็นกำลังใจสำคัญในการขับเคลื่อนการทำงานวิจัย คอยสนับสนุนและให้กำลังใจในการทำวิทยานิพนธ์ครั้งนี้ให้สำเร็จลุล่วงด้วยดี

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

กฤษณ์ วิริยะสิทธารถ

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ค

บทคัดย่อภาษาอังกฤษ ..... ง

กิตติกรรมประกาศ ..... จ

สารบัญ ..... ฉ

สารบัญตาราง ..... ฌ

สารบัญภาพ ..... ญ

บทที่ 1 บทนำ 1

ความเป็นมาและความสำคัญ 1

คำถามวิจัย 3

วัตถุประสงค์การวิจัย 3

ขอบเขตของการวิจัย 4

คำจำกัดความ 4

ประโยชน์ที่ได้รับจากการวิจัย 4

บทที่ 2 เอกสารและงานวิจัยที่เกี่ยวข้อง 6

ตอนที่ 1 การเรียนรู้แบบจุลภาค 6

1.1 ความหมายของการเรียนรู้แบบจุลภาค 6

1.2 องค์ประกอบของการเรียนรู้แบบจุลภาค 6

1.3 การออกแบบการเรียนรู้แบบจุลภาค 8

ตอนที่ 2 การเขียนโค้ด 9

2.1 ความหมายของการเขียนโค้ด 9

2.2 องค์ประกอบของความสามารถด้านการเขียนโค้ด 10

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

2.3 การสอนเขียนโค้ด 12
ตอนที่ 3 แนวคิดการพัฒนาบทเรียนการเรียนรู้แบบจุลภาค 12
กรอบแนวคิดการวิจัย 13
บทที่ 3 วิธีดำเนินการวิจัย 15
ระยะที่ 1 การพัฒนาบทเรียนการเรียนรู้แบบจุลภาคด้านการเขียนโค้ด 15
ตอนที่ 1.1 การออกแบบโครงสร้างของบทเรียนการเรียนรู้แบบจุลภาคด้านการเขียนโค้ด ... 15
ตอนที่ 1.2 การพัฒนาบทเรียนการเรียนรู้แบบจุลภาคด้านการเขียนโค้ด 16
ระยะที่ 2 การศึกษาผลของการใช้บทเรียนออนไลน์ด้านการเขียนโค้ดที่ออกแบบโดยแนวคิดการเรียนรู้แบบจุลภาค 17
ตอนที่ 2.1 กลุ่มเป้าหมาย และกลุ่มทดลอง 17
ตอนที่ 2.2 รูปแบบการทดลอง 18
ตอนที่ 2.3 เครื่องมือวิจัย 20
ตอนที่ 2.4 การตรวจสอบคุณภาพเครื่องมือ 25
ตอนที่ 2.5 การเก็บรวบรวมข้อมูลและการวิเคราะห์ข้อมูล 26
บทที่ 4 ผลการวิจัย 27
ตอนที่ 1 ความสามารถด้านการเขียนโค้ดของนักศึกษาครูที่มีภูมิหลังแตกต่างกัน 27
ตอนที่ 2 ผลของการออกแบบและพัฒนาบทเรียนออนไลน์ด้านการเขียนโค้ดโดยใช้การเรียนรู้แบบจุลภาค 33
ตอนที่ 3 การเปรียบเทียบโปรไฟล์และความสามารถด้านการเขียนโค้ดระหว่างได้รับบทเรียนออนไลน์ด้านการเขียนโค้ดที่ออกแบบตามการเรียนรู้แบบจุลภาคที่มีรูปแบบแตกต่างกันของนักศึกษาครูที่มีภูมิหลังแตกต่างกัน 59
บทที่ 5 สรุปผล อภิปรายผล และข้อเสนอแนะการวิจัย 70
สรุปผลการวิจัย 70
อภิปรายผลการวิจัย 72
ข้อจำกัดของการวิจัย 74

ข้อเสนอแนะจากการวิจัย 74
บรรณานุกรม 77
ภาคผนวก 82
ภาคผนวก ก รายชื่อผู้ทรงคุณวุฒิในการตรวจสอบคุณภาพเครื่องมือวิจัย 83
ภาคผนวก ข เครื่องมือที่ใช้ในการวิจัย 85
ภาคผนวก ค ผลงานของกลุ่มทดลองในการเรียนบทเรียนออนไลน์ด้านการเขียนโค้ด 93
ประวัติผู้เขียน 98

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

สารบัญตาราง

หน้า

ตาราง 3.1 ตารางวิเคราะห์ข้อสอบของแบบทดสอบก่อนเรียน.....21
ตาราง 3.2 ตารางวิเคราะห์ข้อสอบของแบบทดสอบหลังเรียน.....23
ตาราง 3.3 ผลการตรวจสอบความตรงเชิงเนื้อหาของแบบทดสอบความสามารถด้านการเขียนโค้ดก่อนเรียน.....25
ตาราง 3.4 ผลการตรวจสอบความตรงเชิงเนื้อหาของแบบทดสอบความสามารถด้านการเขียนโค้ดหลังเรียน.....25
ตาราง 4.1 ความถี่ ร้อยละของตัวอย่างวิจัยจำแนกตามเพศ ชั้นปี สาขาวิชา และวิชาเอก.....27
ตาราง 4.2 ความถี่ ร้อยละของตัวอย่างวิจัยจำแนกตามชั้นปี สาขาวิชา และวิชาเอก.....28
ตาราง 4.3 ค่าเฉลี่ย ส่วนเบี่ยงเบนมาตรฐานของคะแนนความสามารถด้านการเขียนโค้ด จำแนกตามสาขาวิชา และประสบการณ์การเขียนโค้ด.....30
ตาราง 4.4 ผลของการเปรียบเทียบคะแนนรวม และคะแนนองค์ประกอบด้านโปรแกรมจำแนกตามประสบการณ์ด้านการเขียนโค้ดและวิชาเอก โดยใช้สถิติทดสอบประเภทต่าง ๆ.....32
ตาราง 4.5 เนื้อหาจุลภาคของโมดูลที่ 1 การเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ (การแก้ปัญหา).....44
ตาราง 4.6 เนื้อหาจุลภาคของโมดูลที่ 2 ภาษา Scratch บน Microsoft MakeCode Arcade.....51
ตาราง 4.7 ตัวอย่างฟังก์ชัน (built-in function).....54
ตาราง 4.8 ตัวอย่างฟังก์ชันอื่น ๆ.....54
ตาราง 4.9 เนื้อหาจุลภาคของโมดูลที่ 3 ภาษา Python 55
ตาราง 4.10 ความถี่ ร้อยละ ของตัวอย่างวิจัยจำแนกตามเพศ ชั้นปี สาขาวิชา และวิชาเอก.....60
ตาราง 4.11 สาขาวิชา วิชาเอก และประสบการณ์การเขียนโค้ดจำแนกตามเส้นทางการเรียนรู้.....62
ตาราง 4.12 คะแนนก่อนเรียน คะแนนหลังเรียน คะแนนสัมพัทธ์ และร้อยละคะแนนสัมพัทธ์จำแนกตามกลุ่มที่ได้รับบทเรียนแตกต่างกัน.....63
ตาราง 4.13 ค่าเฉลี่ยคะแนนก่อนเรียน หลังเรียนในโมดูลที่ 1 การเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ จำแนกตามกลุ่มที่ได้รับบทเรียนแตกต่างกัน.....64
ตาราง 4.14 ค่าเฉลี่ยคะแนนก่อนเรียน หลังเรียนในโมดูลที่ 2 ภาษา Scratch จำแนกตามกลุ่มที่ได้รับบทเรียนแตกต่างกัน 67
ตาราง 4.15 ค่าเฉลี่ยคะแนนก่อนเรียน หลังเรียนในโมดูลที่ 3 ภาษา Python จำแนกตามกลุ่มที่ได้รับบทเรียนแตกต่างกัน 67

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สารบัญภาพ

หน้า

ภาพ 2.1 กรอบแนวคิดการวิจัย.....14
ภาพ 3.1 โครงสร้างของบทเรียนการเรียนรู้แบบจุลภาคด้านการเขียนโค้ด.....15
ภาพ 3.2 แบบแผนการทดลอง.....19
ภาพ 4.1 ลำดับการเรียนของโมดูลเริ่มต้น PreClass.....34
ภาพ 4.2 ลำดับการเรียนของโมดูลเริ่มต้น PreClass บน Canvas LMS.....34
ภาพ 4.3 ลำดับการเรียนของ Introduction of coding.....35
ภาพ 4.4 ลำดับการเรียนของ Introduction of coding บน Canvas LMS.....36
ภาพ 4.5 ลำดับการเรียนของ Scratch.....36
ภาพ 4.6 ลำดับการเรียนของ Scratch บน Canvas LMS.....37
ภาพ 4.7 ลำดับการเรียนของ Python.....37
ภาพ 4.8 ลำดับการเรียนของ Python บน Canvas LMS.....38
ภาพ 4.9 แพลตฟอร์มการเรียนรู้ออนไลน์ Canvas LMS บคอมพิวเตอร์.....38
ภาพ 4.10 แพลตฟอร์มการเรียนรู้ออนไลน์ Canvas LMS บนเท็บเล็ต.....39
ภาพ 4.11 บทเรียนย่อยในโมดูลการเรียนรู้จุลภาค เรื่อง การเขียนโค้ดเบื้องต้น.....39
ภาพ 4.12 ฟังก์ชันการปรับเหมาะโดยใช้ Mystery Path บน Canvas LMS.....40
ภาพ 4.13 กิจกรรมเกม lightbot hour.....56
ภาพ 4.14 วิดีโอเชิงโต้ตอบ เรื่อง ผังงาน.....56
ภาพ 4.15 ผังงาน.....57
ภาพ 4.16 เอกสารเชิงโต้ตอบ เรื่อง ภาษา Scratch.....58
ภาพ 4.17 เว็บไซต์ ww.arcade.makecode.com.....58
ภาพ 4.18 หน้าต่างการเขียนโค้ดภาษา Python เชิงโต้ตอบ.....59
ภาพ 4.19 เส้นทางการเรียนบทเรียนออนไลน์ด้านการเขียนโค้ด.....61
ภาพ 4.20 แผนภาพแสดงคะแนนหลังเรียนจากโมดูลที่ 1 การเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์.....65
ภาพ 4.21 แผนภาพแสดงคะแนนหลังเรียนจากโมดูลที่ 2 ภาษา Scratch.....66
ภาพ 4.22 แผนภูมิแท่งแสดงคะแนนหลังเรียนจากโมดูลที่ 3 ภาษา Python.....68
ภาพ 4.23 แผนภาพระดับความสามารถด้านการเขียนโค้ดระหว่างเรียน.....69

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

บทที่ 1
บทนำ

ความเป็นมาและความสำคัญ

ในยุคดิจิทัล สิ่งที่สำคัญประการหนึ่งคือ ความสามารถในการใช้เทคโนโลยี ซึ่งทำให้การเรียนรู้เกี่ยวกับการเขียนโปรแกรม (programming) หรือการเขียนโค้ด (coding) มีความจำเป็น เนื่องจากการเขียนโค้ดจะเปลี่ยนสิ่งรอบตัวเราให้อยู่ในรูปแบบของดิจิทัลได้เกือบทั้งหมด (Crow, 2014) ปัจจุบันไม่ว่าจะเป็นสังคมแบบใดหรืออาชีพใด เทคโนโลยีได้มีบทบาทสำคัญต่อการทำงานและการใช้ชีวิตเป็นอย่างมาก ในบริบทด้านการศึกษานั้น ความสามารถด้านการเขียนโค้ดเป็นสิ่งจำเป็นต่อครูทุกคน ทั้งครูที่สอนรายวิชาวิทยาการคำนวณ และครูที่สอนรายวิชาอื่น ๆ

สำหรับครูที่สอนในรายวิชาวิทยาการคำนวณที่เป็นผู้จัดกิจกรรมการเรียนรู้ในเรื่องการเขียนโค้ดให้กับนักเรียนมีความจำเป็นที่จะต้องมีความรู้ความเข้าใจในเรื่องการเขียนโค้ดเป็นอย่างดี เพื่อให้นักเรียนสามารถบรรลุเป้าหมายของการจัดการเรียนการสอนได้อย่างมีประสิทธิภาพตามหลักสูตรการศึกษาขั้นพื้นฐาน พ.ศ. 2551 ฉบับปรับปรุง พ.ศ. 2560 ที่ตั้งเป้าหมายไว้ว่า นักเรียนต้องสามารถเขียนโค้ดอย่างง่ายและพัฒนานวัตกรรมที่เชื่อมโยงกับศาสตร์อื่นได้ (รวิวรรณ เทนอิสสระ และคณะ, 2560) ในปี พ.ศ. 2561 เป็นช่วงแรกของการเตรียมความพร้อมของครู จึงมีการจัดอบรมครูโดยสถาบันส่งเสริมการสอนวิทยาศาสตร์และเทคโนโลยี (*สสวท.*) เกี่ยวกับการสอนเขียนโค้ดในระดับประถมศึกษา ซึ่งเป็นการเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ (unplugged coding) โดยมีลักษณะกิจกรรมการเรียนรู้ที่สามารถจัดให้นักเรียนได้โดยไม่ต้องใช้คอมพิวเตอร์ แต่ต่อไปนักเรียนจะต้องเรียนการเขียนโค้ดด้วยคอมพิวเตอร์ที่เป็นภาษาแบบภาพด้วยโปรแกรม Scratch และภาษาแบบข้อความ เช่น ภาษา C, Java และ Python ดังนั้น ครูจึงต้องเรียนรู้การเขียนโค้ดทั้งแบบที่ใช้คอมพิวเตอร์และไม่ใช้คอมพิวเตอร์

ในขณะที่ครูทั่วไปที่ไม่ได้สอนการเขียนโค้ดโดยตรง หากมีความสามารถด้านการเขียนโค้ดจะเป็นพื้นฐานที่สามารถต่อยอดเพื่อให้ครูสามารถสร้างและพัฒนาโปรแกรมหรือโปรแกรมประยุกต์ที่ช่วยสอนในรายวิชาของตนเอง ครูสามารถสร้างแชทบอทได้ เช่น Nguyen et al. (2019) ได้พัฒนาแชทบอทเพื่อแก้ปัญหาคณิตศาสตร์ในระดับมัธยมศึกษา นอกจากนี้ การเขียนโค้ดภาษา Python สามารถสร้างแชทบอทให้คำแนะนำหรือข้อมูล เช่น คะแนนสอบ คะแนนเก็บ ให้กับนักเรียนได้อย่างอัตโนมัติ และยังช่วยอำนวยความสะดวกในการทำงานด้านอื่น เช่น การพัฒนาโปรแกรมเพื่อช่วยในการรวบรวมข้อมูลและวิเคราะห์ข้อมูลของนักเรียน หรือข้อมูลขนาดใหญ่ของทั้งโรงเรียน

ดังนั้น การเขียนโค้ดจึงมีประโยชน์และความสำคัญสำหรับครูทุก ๆ คนในการทำงานด้านการศึกษาและมีความจำเป็นอย่างยิ่งที่จะเตรียมความพร้อมและพัฒนาความสามารถด้านการเขียนโค้ดให้กับนักศึกษาครูตั้งแต่ยังศึกษาอยู่ในระดับอุดมศึกษาเพื่อให้มีพื้นฐานความรู้และเป็นเครื่องมือสำคัญในการพัฒนานักเรียนต่อไป

การสอนเขียนโค้ดหรือการสอนภาษาคอมพิวเตอร์ในอดีตจัดขึ้นโดยผู้เชี่ยวชาญด้านการเขียนโปรแกรมจึงไม่ได้มีการออกแบบการเรียนการสอนตามทฤษฎีการเรียนรู้ จึงเป็นการเรียนการสอนแบบปกติและการมอบหมายแบบฝึกหัด ต่อมามีการใช้ทฤษฎีการเรียนรู้ เช่น การเรียนรู้ผ่านเกม (gamification) โดยเน้นการแข่งขันและสังคมในกลุ่มผู้เรียน และ Skalka และ Drlik (2018) กล่าวว่าการเรียนการสอนเกี่ยวกับการเขียนโค้ดที่ประสบความสำเร็จเป็นการเรียนรู้โดยมีเว็บเป็นฐาน (web-based learning) แต่มีความยุ่งยากในการจัดการเรียนรู้ให้กับผู้เรียนในด้านอุปกรณ์ การเรียนรู้โดยมีเว็บเป็นฐานจึงใช้เป็นเพียงฐานในการพัฒนาต่อไป และได้เป็นการเรียนรู้แบบจุลภาค (microlearning) ที่สามารถเรียนได้ง่ายผ่านอุปกรณ์เคลื่อนที่

การเรียนรู้แบบจุลภาคเป็นการเรียนรู้ที่ใช้แรงและเวลาในการเรียนน้อย เพราะมีเนื้อหาที่เรียนปริมาณน้อย ไม่กว้าง เรียนรู้ไปทีละขั้น (step by step) ซึ่งมีความเหมาะสมในการสอนการเขียนโค้ดที่มีลักษณะเนื้อหาที่เฉพาะเจาะจง แคบ สามารถออกแบบหรือแบ่งเนื้อหาออกให้มีปริมาณน้อย ๆ และเรียนได้ในเวลาที่สั้น โดยทั่วไปการเรียนภาษาคอมพิวเตอร์มีเนื้อหาในการเรียนคือ การวิเคราะห์ปัญหา การเขียนโค้ดและการหาข้อผิดพลาดของคำสั่ง ซึ่งมีความยากง่ายแตกต่างกัน ในเรื่องการเขียนโค้ดที่เป็นเนื้อหาที่ไม่ได้ซับซ้อนมากจนเกินไป สามารถนำมาออกแบบเป็นหน่วยการเรียนขนาดเล็กได้ (Robin, Rountree, & Rountree, 2003)

Skalka และ Drlik (2018) ได้ศึกษาเกี่ยวกับการสอนเขียนโค้ดโดยใช้การเรียนรู้แบบจุลภาค ได้มีผู้ศึกษาและใช้กรอบแนวคิดสำหรับการพัฒนาทักษะการเขียนโปรแกรมในระดับมหาวิทยาลัย ซึ่งเป็นกรอบแนวคิดอิงการเรียนรู้แบบจุลภาค พบว่า การเรียนรู้แบบจุลภาคเป็นส่วนสำคัญที่ใช้ในการออกแบบการสอนเขียนโปรแกรม สามารถทำให้ผลการเรียนที่ดีกว่าการเรียนแบบดั้งเดิมหากผสมผสานกันระหว่างการออกแบบเนื้อหา และการออกแบบกิจกรรมการเรียนรู้ได้อย่างเหมาะสมและสามารถสร้างปฏิสัมพันธ์กับผู้เรียนได้มากเพียงพอ นอกจากนี้ Gabrielli et al. (2006) ได้เสนอไว้ว่าการเรียนรู้แบบจุลภาคจะต้องตอบสนองความต้องการของผู้เรียนเป็นรายบุคคลตามความชอบและความสามารถ

งานวิจัยเกี่ยวกับการพัฒนาบทเรียนออนไลน์มักจะเป็นการทดลองที่เปรียบเทียบผลของการใช้บทเรียนออนไลน์ระหว่างกลุ่มทดลองและกลุ่มควบคุม ซึ่งทรีตเมนต์ที่นำมาทดลองจะมีลักษณะแบบเดียว ต่อมา Green (2020) กล่าวว่า การทดลองแบบปรับเหมาะมีประสิทธิภาพในการค้นพบทรีตเมนต์ที่ดีที่สุดได้มากกว่าการทดลองแบบอื่น การทดลองแบบปรับเหมาะคือการจัดสรรทรีตเมนต์

ให้กับกลุ่มทดลองโดยขึ้นอยู่กับการวิเคราะห์ข้อมูลระหว่างการทดลอง การปรับเปลี่ยนกลุ่มทดลองให้ได้รับทรีตเมนต์ที่แตกต่างกัน หรือการปรับเปลี่ยนทรัพยากรที่จะให้กับกลุ่มทดลองตามผลการประเมินที่แตกต่างกัน

ในการทำวิจัยครั้งนี้จึงมีการปรับปรุงและเปลี่ยนแปลงทรีตเมนต์ที่ตัวอย่างวิจัยได้รับระหว่างการทดลองโดยอิงจากผลการประเมินระหว่างทำการทำวิจัยในด้านความชอบ ความสามารถ เป็นต้น เพื่อให้ตอบสนองความต้องการเป็นขั้น ๆ ต่อไป ดังนั้น งานวิจัยครั้งนี้จึงใช้แบบแผนการวิจัยเป็นการวิจัยเชิงทดลองแบบปรับเหมาะ เพื่อให้สามารถค้นพบรูปแบบของทรีตเมนต์ที่มีประสิทธิภาพมากที่สุดและเหมาะกับความสามารถด้านการเขียนโค้ดของนักศึกษาครูที่แตกต่างกันออกไป

ดังนั้น งานวิจัยนี้จึงต้องการพัฒนานักศึกษาครูให้มีความสามารถด้านการเขียนโค้ดทั้งแบบใช้คอมพิวเตอร์และแบบที่ไม่ใช้คอมพิวเตอร์เพื่อตอบสนองความต้องการของระบบการศึกษาผ่านการจัดการเรียนการสอนบทเรียนออนไลน์ที่ออกแบบขึ้นตามแนวคิดการเรียนรู้แบบจุภาคภายใต้การทดลองแบบปรับเหมาะ

คำถามวิจัย

1. นักศึกษาครูมีความสามารถด้านการเขียนโค้ดแตกต่างกันตามภูมิหลังอย่างไร
2. บทเรียนออนไลน์ด้านการเขียนโค้ดที่ออกแบบตามการเรียนรู้แบบจุลภาคมีลักษณะเป็นอย่างไร
3. นักศึกษาครูที่มีภูมิหลังต่างกัน มีโปรไฟล์และความสามารถด้านการเขียนโค้ดระหว่างที่ได้รับบทเรียนออนไลน์ด้านการเขียนโค้ดที่ออกแบบตามการเรียนรู้แบบจุลภาคแตกต่างกันอย่างไร

วัตถุประสงค์การวิจัย

1. เพื่อเปรียบเทียบความสามารถด้านการเขียนโค้ดของนักศึกษาครูที่มีภูมิหลังแตกต่างกัน
2. เพื่อออกแบบและพัฒนาบทเรียนออนไลน์ด้านการเขียนโค้ดโดยใช้การเรียนรู้แบบจุลภาค
3. เพื่อเปรียบเทียบโปรไฟล์และความสามารถด้านการเขียนโค้ดระหว่างได้รับบทเรียนออนไลน์ด้านการเขียนโค้ดที่ออกแบบตามการเรียนรู้แบบจุลภาคที่มีรูปแบบแตกต่างกันของนักศึกษาครูที่มีภูมิหลังแตกต่างกัน

ขอบเขตของการวิจัย

ภาษาคอมพิวเตอร์ที่ศึกษาในงานวิจัยนี้ใช้ภาษาบล็อก คือ ภาษา Scratch และภาษาข้อความ คือภาษา Python เพราะ เป็นภาษาที่ระบุไว้ให้นักเรียนในระดับมัธยมศึกษาได้เรียน ในสาระการเรียนรู้ของวิชาวิทยาการคำนวณ นอกจากนี้ ยังเป็นภาษาที่เหมาะกับผู้เริ่มเรียนการเขียนโค้ดใหม่ (Zyl et al., 2020; Meerbaum-Salant, Armoni, & Ben-Ari, 2010; รวิวรรณ เทนอิสสระ และคณะ, 2560)

คำจำกัดความ

การเรียนรู้แบบจุลภาค (microlearning) หมายถึง การเรียนรู้ที่ผู้เรียนได้เรียนเนื้อหาเป็นหน่วยย่อย แต่ละหน่วยย่อยประกอบด้วย 2 ส่วนคือ 1) สื่อการเรียนรู้เป็นวิดีโอที่มีการบรรยายเนื้อหาภาพกราฟิกหรือแอนิเมชัน และ 2) แบบฝึกหัดและแบบทดสอบ โดยผู้เรียนแต่ละคนจะได้เรียนหน่วยย่อยที่แตกต่างกันตามลักษณะของผู้เรียน

ความสามารถด้านการเขียนโค้ด หมายถึง ความสามารถในการทำความเข้าใจปัญหาออกแบบหรือกำหนดวิธีการแก้ไขปัญหาในรูปแบบที่เข้ากับคอมพิวเตอร์ แปลวิธีคิดและเขียนออกมาเป็นคำสั่งภาษาโปรแกรมที่ถูกต้องตามวากยสัมพันธ์ ซึ่งประกอบด้วย 2 องค์ประกอบคือ องค์ประกอบด้านการแก้ไขปัญหา และองค์ประกอบด้านโปรแกรม

องค์ประกอบที่ 1 ด้านแก้ไขปัญหา หมายถึง ความสามารถในการทำความเข้าใจปัญหาออกแบบหรือกำหนดวิธีการแก้ไขปัญหาในรูปแบบของผังงาน ขั้นตอนวิธี และอื่น ๆ

องค์ประกอบที่ 2 ด้านโปรแกรม หมายถึง การแปลวิธีคิด วิธีการแก้ปัญหาที่ออกแบบไว้ และเขียนคำสั่งเป็นภาษาโปรแกรมที่มีวากยสัมพันธ์ที่ถูกต้อง

ความพึงพอใจต่อการเรียนรู้ หมายถึง ความรู้สึกชอบ หรือไม่ชอบต่อลักษณะบทเรียนการเรียนรู้แบบจุลภาคในด้านระยะเวลาในการเรียน ประเภทของสื่อการเรียนรู้ ลำดับเนื้อหา และสังคมการเรียนรู้

ประโยชน์ที่ได้รับจากการวิจัย

1. **ประโยชน์เชิงนโยบาย**

1) องค์กรที่เกี่ยวข้องกับการผลิตหรือกำหนดรูปแบบการให้ความรู้ในการเขียนโค้ดกับครูประจำการหรือนักศึกษาครู สามารถนำผลการวิจัยไปใช้เป็นแนวทางในการปรับปรุงและพัฒนารูปแบบการให้ความรู้เพื่อให้ตอบสนองต่อการเรียนรู้ที่หลากหลายและเกิดประสิทธิภาพสูงสุด

2) สถานศึกษาระดับอุดมศึกษาและมัธยมศึกษาสามารถนำผลการวิจัยไปใช้เป็นแนวทางในการพัฒนาบทเรียนหรือรายวิชาเพื่อส่งเสริมความสามารถด้านการเขียนโค้ดให้กับนักเรียน

2. ประโยชน์เชิงวิชาการ

1) ได้องค์ความรู้เกี่ยวกับการออกแบบบทเรียนออนไลน์ด้านการเขียนโค้ดที่ใช้การเรียนรู้แบบจุลภาคสำหรับนักศึกษาครูทั้งทฤษฎี ขั้นตอนการออกแบบ การนำไปใช้ เพื่อให้ผู้ที่สนใจนำองค์ความรู้ไปวิจัยและพัฒนาต่อไป

2) ได้หลักการออกแบบบทเรียนออนไลน์ด้านการเขียนโค้ดที่ใช้แนวคิดการเรียนรู้แบบจุลภาคที่ช่วยชี้นำคุณลักษณะของบทเรียนและกระบวนการที่ควรจะเป็นในการสร้างและออกแบบ

3. ประโยชน์เชิงปฏิบัติ

1) คณะหรือมหาวิทยาลัยฝึกหัดครูสามารถนำบทเรียนออนไลน์ด้านการเขียนโค้ดที่ใช้การเรียนรู้แบบจุลภาคไปใช้ส่งเสริมความรู้และทักษะในการแก้ปัญหา การเขียนโค้ดภาษา Scratch และภาษา Python

2) นักศึกษาครูและครูประจำการ หรือบุคคลทั่วไปที่สนใจสามารถเรียนรู้เกี่ยวกับการแก้ปัญหา การเขียนโค้ดภาษา Scratch และภาษา Python ได้จากบทเรียนออนไลน์ด้านการเขียนโค้ดที่ใช้การเรียนรู้แบบจุลภาค

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 2
เอกสารและงานวิจัยที่เกี่ยวข้อง

ผู้วิจัยได้ศึกษาเอกสารและงานวิจัยที่เกี่ยวข้องกับการพัฒนารูปแบบการเรียนรู้แบบจุลภาคบนบทเรียนออนไลน์ด้านการเขียนโค้ดสำหรับนักศึกษาครู โดยแบ่งออกเป็น 3 ตอน คือ 1) การเรียนรู้แบบจุลภาค 2) การเขียนโค้ด และ 3) แนวคิดการพัฒนาบทเรียนการเรียนรู้แบบจุลภาค ซึ่งมีรายละเอียดดังนี้

ตอนที่ 1 การเรียนรู้แบบจุลภาค

1.1 ความหมายของการเรียนรู้แบบจุลภาค

การเรียนรู้สามารถแบ่งออกเป็น 2 ระดับคือ การเรียนรู้ในระดับมหภาค (macro level) และการเรียนรู้ในระดับจุลภาค (micro level) ซึ่งการเรียนรู้ทั้งสองระดับมีความแตกต่างกันในหลายมิติ เช่น 1) เนื้อหาการเรียนรู้ในระดับมหภาคคือ วิชา หัวข้อเรื่อง แต่ในระดับจุลภาคคือ วัตถุประสงค์การเรียนรู้ เนื้อหาย่อย 2) สังคมวิทยาในระดับมหภาคคือ การเรียนรู้ในระดับสังคม แต่ในระดับจุภาคคือ การเรียนรู้รายบุคคล (Hug, 2007) โดยเบื้องต้นการเรียนรู้แบบจุลภาคจะรับรู้ได้ว่าเป็นการเรียนรู้ที่มีบริบทของเนื้อหาการเรียนและเป้าหมายการเรียนในระดับย่อย

นักการศึกษาได้ให้ความหมายของ การเรียนรู้แบบจุลภาค (microlearning) ไว้หลากหลายความหมาย ซึ่งส่วนที่มีความหมายสอดคล้องกันเป็นดังนี้ การเรียนรู้แบบจุลภาค (microlearning) เป็นการเรียนรู้ที่มีเนื้อหาปริมาณน้อย ละเอียด และใช้เวลาในการเรียนสั้น ๆ เพื่อให้บรรลุเป้าหมายของการเรียนที่มีความเฉพาะ (Buchem & Hemelmann, 2010; Kapp & Defelice, 2019; Langreiter & Bolka, 2005; Lindner, 2006; Schmidt, 2007) นอกจากนั้นยังมีการอธิบายถึงความสัมพันธ์ที่สอดคล้องกับการเรียนรู้รูปแบบอื่น คือ การเรียนรู้เป็นรายบุคคล (individual learning) ที่ผู้เรียนสามารถเลือกเรียนได้ตามความสนใจของตนเอง (Chisholm, 2005)

1.2 องค์ประกอบของการเรียนรู้แบบจุลภาค

Alqurashi (2017) ได้ศึกษาจากนักวิชาการหลากหลายคนและสร้างโมเดลการเรียนรู้แบบจุลภาค ซึ่งแบ่งออกเป็น 3 องค์ประกอบต่อไปนี้

1) เนื้อหา (content) เป็นเนื้อหาที่สามารถย่อยเป็นหน่วยเล็ก ๆ มีความเหมาะสมกับเวลา มักเป็นจุดผิดที่พบบ่อยในการเรียน (Alqurashi, 2017)

2) ศาสตร์การสอน (pedagogy) แบ่งออกเป็น 4 ระยะ ได้แก่ ระยะที่ 1 ผู้เรียนซึมซับความรู้ที่เป็นพื้นฐานและง่าย ระยะที่ 2 ผู้เรียนรับเอาความรู้ผ่านการมีปฏิสัมพันธ์กับบรรยากาศการเรียนรู้ที่

ผู้สอนออกแบบและเกิดการเรียนรู้ด้วยตนเอง ระยะที่ 3 การสร้างความรู้ที่เกิดจากการร่วมมือกันระหว่างผู้สอนกับผู้เรียน และระยะที่ 4 นักเรียนเกิดมโนทัศน์ในระดับที่สูงขึ้น (Baumgartner, 2013)

3) เทคโนโลยี (technology) เป็นเครื่องมือกระตุ้นในการเรียนรู้ที่เกิดขึ้นได้นอกห้องเรียน และได้รับการออกแบบตามลักษณะเฉพาะของการเรียนรู้แบบจุลภาค (Alqurashi, Gokbel, และ Carbonara, 2016)

นอกจากนี้ Hug (2007) ได้เสนอมิติของการเรียนรู้แบบจุลภาคที่ได้จากการเรียนรู้หลากหลายรูปแบบที่มีอยู่ในขณะนั้น และได้ออกมาเป็นมิติของการเรียนรู้แบบจุลภาคทั้งหมด 7 มิติ และได้ยกตัวอย่างสิ่งที่เกี่ยวข้องไว้ดังนี้ 1) เวลา: การใช้เวลาค่อนข้างสั้น 2) เนื้อหา: เนื้อหาปริมาณน้อย ไม่ยากและไม่ลึกจนเกินไป 3) หลักสูตร: เนื้อหาเป็นส่วนหนึ่งของหลักสูตร โมดูล และการเรียนรู้แบบไม่เป็นทางการ 4) รูปแบบ: การเรียนรู้แบ่งเป็นตอนย่อย ๆ หรือแบ่งทักษะเป็นองค์ประกอบย่อย 5) กระบวนการ: กิจกรรมแบบบูรณาการ กระบวนการที่เกิดขึ้นซ้ำ ๆ การจัดการกับความสนใจของผู้เรียน 6) สื่อกลาง: การเรียนรู้แบบเผชิญหน้า การเรียนรู้ผ่านมัลติมีเดีย 7) ประเภทการเรียนรู้: เป็นสิ่งที่เกิดขึ้นระหว่างการเรียนรู้แบบจุลภาค เช่น การเรียนรู้ที่เน้นการทบทวน การเรียนรู้ผ่านกิจกรรม การสร้างความรู้ด้วยตนเอง การเชื่อมโยงความรู้ การเรียนรู้จากตัวอย่าง งานหรือแบบฝึกหัด การเรียนรู้แบบร่วมมือ

Buchem และ Hemelmann (2010) ได้กล่าวถึงการเรียนรู้แบบจุลภาค ซึ่งมีประเด็นที่สอดคล้อง และได้ขยายความจากมิติการเรียนรู้แบบจุลภาคที่เสนอโดย Hug (2007) ดังนี้ มิติที่ 1 *เวลา*: การเรียนรู้สามารถใช้เวลาเป็นวินาทีได้จนถึงประมาณ 15 นาที มิติที่ 2 *เนื้อหา*: เนื้อหาจุลภาคเป็นเนื้อหาที่มีข้อมูลขนาดเล็ก เน้นที่ความคิดหรือหัวข้อเดียว และยังต้องสามารถเรียนรู้ให้เข้าใจได้โดยที่ไม่ต้องใช้ข้อมูลภายนอกอื่น ๆ ในขณะเดียวกันก็ต้องระวังไม่ให้เนื้อหาถูกแบ่งออกเป็นเสียความหมายที่สมบูรณ์ไป มิติที่ 3 *หลักสูตร*: การเรียนรู้แบบจุลภาคเป็น การเรียนรู้แบบไม่เป็นทางการซึ่งง่ายและยืดหยุ่นสามารถนำเข้าไปเป็นส่วนหนึ่งในกิจวัตรประจำวันได้ มิติที่ 4 *หลักสูตร*: การสร้างมโนทัศน์และทักษะการแก้ปัญหาเป็นเป้าหมายของการเรียนรู้แบบจุลภาค มิติที่ 5 *กระบวนการ*: กระบวนการเรียนรู้เป็นพลวัตรและหยืดหยุ่นเกิดจากการสร้างสรรค์ของผู้เรียน นอกเหนือจากประเด็นที่สอดคล้องกับมิติของ Hug (2007) ยังกล่าวถึง การมีปฏิสัมพันธ์กับสังคมว่า การเรียนรู้แบบจุลภาคเน้นให้บทบาทและการมีส่วนร่วมของผู้เรียนเกิดขึ้นระหว่างผู้เรียนด้วยกัน

องค์ประกอบและมิติของการเรียนรู้แบบจุลภาคที่เสนอโดย Alqurashi (2017) Hug (2007) และ Buchem และ Hemelmann (2010) แบ่งออกเป็น 3 องค์ประกอบได้แก่ องค์ประกอบด้านเนื้อหา ด้านศาสตร์การสอน และด้านเทคโนโลยี โดยสามารถแบ่งเป็นองค์ประกอบย่อยที่มีความสอดคล้องกันได้ดังนี้

องค์ประกอบด้านเนื้อหาประกอบด้วยมิติด้านเวลา การกำหนดเนื้อหา หลักสูตร และรูปแบบ ซึ่งเกี่ยวข้องกับ การกำหนดเนื้อหาที่มีปริมาณน้อย มีความเฉพาะ ละเอียดเข้าใจง่าย เนื้อหาจบในตัว มักเป็นการเรียนมโนทัศน์หรือทักษะย่อย ๆ ในการแก้ปัญหา ทั้งนี้ต้องใช้เวลาในการเรียนแต่ละครั้งน้อย ๆ

องค์ประกอบด้านศาสตร์การสอนประกอบด้วยมิติด้าน กระบวนการและประเภทการเรียนรู้ ซึ่งเกี่ยวข้องกับการจัดการเรียนรู้ที่มีความยืดหยุ่น สามารปรับเปลี่ยนได้ มีลักษณะเป็นทั้งการเรียนรู้แบบไม่เป็นทางการ การเรียนรู้เป็นรายบุคคล การเรียนรู้ทางสังคมร่วมกันกับเพื่อน เป็นต้น

องค์ประกอบด้านเทคโนโลยี ประกอบด้วยมิติด้านสื่อกลาง ซึ่งเกี่ยวข้องกับสื่อการเรียนรู้ที่หลากหลาย เครื่องมือเป็นแบบอิเล็กทรอนิกส์ และสามารถเข้าถึงได้ด้วยอุปกรณ์ที่หลากหลาย

งานวิจัยที่ศึกษาเกี่ยวกับผลของการจัดการเรียนรู้แบบจุลภาคนั้นมีความหลากหลายและครอบคลุมถึงเนื้อหาของหลายศาสตร์ โดยตัวอย่างของการศึกษาเป็นรายกรณีของ Buchem และ Hemelmann (2010) มีรายละเอียดของการเรียนรู้ตามองค์ประกอบดังนี้ การออกแบบการเรียนรู้แบบจุลภาคเพื่อฝึกฝนทักษะการออกเสียงภาษาอังกฤษ ซึ่งมีระยะเวลาในการเรียน 2 เดือน สามารถวางแผนการเข้าเรียนได้อย่างอิสระไม่ว่าเรียนที่ใด เมื่อใด ก็ได้ โดยแต่ละบทเรียนแต่ละตอนมีความยาวตอนละ 10-15 นาที แบ่งออกเป็น 40 ตอน ประบกอบด้วย บทนำ 5 ตอน เนื้อหา 30 ตอน (เนื้อหาย่อยละ 10 ตอน) สรุป 5 ตอน ซึ่งมีสื่อการเรียนรู้เป็น wiki บทเว็บ 2.0 การศึกษาครั้งนี้จะเน้นการส่งเสริมการเรียนรู้แบบร่วมมือ การเรียนรู้อิงการแก้ปัญหา

1.3 การออกแบบการเรียนรู้แบบจุลภาค

การออกแบบการเรียนรู้แบบจุลภาคสามารถออกแบบได้โดยยึดองค์ประกอบของการเรียนรู้แบบจุลภาคเป็นหลักเพื่อให้กิจกรรมการจัดการเรียนรู้ที่ออกแบบนั้นครบถ้วน และสอดคล้องกับทุกองค์ประกอบย่อย

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

Gabrielli, Kimani, และ Catarci (2006) ได้เสนอแนวทางการออกแบบเนื้อหาและกิจกรรมการเรียนรู้กับเทคโนโลยีในการจัดสภาพแวดล้อมของการเรียนรู้แบบจุลภาคไว้ดังนี้ 1) สามารถเรียนรู้ได้ทุกที่ทุกเวลา ใช้งานได้ง่าย และเหมาะกับคนที่มีความสามารถด้านเทคโนโลยีที่แตกต่างกัน 2) ใช้กิจกรรมการเรียนรู้ที่หลากหลายในแต่ละเป้าหมายของการเรียนรู้ 3) ผู้เรียนได้รับความช่วยเหลือเป็นรายบุคคล และยังสามารถได้รับการสนับสนุนหรือความช่วยเหลือจากเพื่อน ผู้สอน ผ่านเทคโนโลยีการสื่อสาร และ4) สามารถปรับเปลี่ยนให้เข้ากับความต้องการของผู้เรียนได้เป็นรายบุคคลตามความชอบและความสามารถ หรือระบบสามารถปรับให้เหมาะสมได้อย่างอัตโนมัติ ยังต้องระมัดระวังไม่ให้มีความซับซ้อนมากจนเกินไป และยังต้องทำให้ผู้เรียนเกิดการเรียนรู้ได้ทุก ๆ วัน

การออกแบบการเรียนรู้แบบจุลภาค พิจารณาจาก 2 ส่วนที่สำคัญคือ การออกแบบเนื้อหาแบบจุลภาค (microcontent design) และการออกแบบกิจกรรมการเรียนรู้แบบจุลภาค (microlearning activities design) (Buchem & Hemelmann, 2010)

1. หลักการออกแบบหน่วยของเนื้อหาแบบจุลภาค ประกอบด้วย 5 ส่วน ได้แก่ 1) รูปแบบ หมายถึง รูปแบบที่เข้าถึงได้ง่ายและรวดเร็ว เช่น การแสดงภาพบนหน้าจอ2) จุดเด่น หมายถึง จุดเด่นที่ชัดเจน เฉพาะเจาะจง รู้ว่าสื่อถึงอะไรอย่างชัดเจน3) อิสระ หมายถึง เนื้อหาที่ผู้เรียนรู้ไม่ต้องสืบค้นเพิ่มเติม 4) โครงสร้าง หมายถึง การประกอบกันขององค์ประกอบย่อย เช่น ชื่อ หัวข้อ ผู้เขียน วันที่ ลิงค์ 5) ความสามารถในการระบุ หมายถึง การใช้ลิงค์เดียว ที่เข้าใจได้ว่าหมายถึงอะไร เช่น permalink

2. หลักการออกแบบกิจกรรมการเรียนรู้แบบจุลภาค สามารถอาศัยหลักการต่อไปนี้เข้ามาช่วยสนับสนุน ได้แก่ 1) กลวิธีการสอนเด่น ๆ ที่นำมาใช้ เช่น การเรียนรู้ด้วยตนเอง การเรียนรู้ที่เหมาะสม การจัดการเรียนรู้โดยใช้ชุมชนเป็นฐาน การเรียนรู้แบบร่วมมือ และโมเดลการออกแบบสื่อ เช่น แนวทางเชิงกระบวนการในการสร้างสื่อและแลกเปลี่ยน 2) กระบวนการเรียนรู้ถูกออกแบบให้เผชิญกับสถานการณ์ โดยที่ไม่ได้จัดแจงไว้ล่วงหน้า สามารถแบ่งขั้นตอนออกเป็นช่วง ๆ โดยแต่ละช่วงเป็นอิสระต่อกัน ผู้เรียนสามารถเลือกได้ 3) กิจกรรมการเรียนรู้แบบจุลภาค ควรเป็นกิจกรรมที่ขับเคลื่อนด้วยนักเรียน และเป็นกิจกรรมที่บูรณาการ ช่วยกระตุ้นให้ผู้เรียนค้นหา ใช้ หรือสร้างความรู้ผ่านเครื่องมือที่ให้ผู้เรียนมีส่วนร่วม 4) วัสดุการเรียนรู้แบบจุลภาค ต้องมีความสมดุล สั้น แต่ต้องเพียงพอต่อความเข้าใจ และ 5) การเรียนรู้ในชุมชนการเรียนรู้ การเรียนสามารถเน้นเพื่อให้ผู้เรียนบรรลุจุดมุ่งหมายที่แตกต่างกันได้ สามารถแลกเปลี่ยนเรียนรู้กันในสังคมการเรียนรู้

นอกจากนี้ Park และ Kim (2018) ศึกษาลักษณะสำหรับการพัฒนาการเรียนรู้แบบจุลภาคบนการเรียนรู้อิเล็กทรอนิกส์ (e-learning) โดยแบ่งออกเป็น 5 ด้าน ได้แก่ 1) ความพอเพียง (adequacy) หมายถึง ระดับความยากในการสร้างสำหรับผู้ที่ไม่เชี่ยวชาญด้านเทคโนโลยี 2) ความสะดวก (convenience) หมายถึงเนื้อหาและเครื่องมือสามารถเชื่อมต่อกันและปรับเปลี่ยนได้ 3) ประสิทธิภาพ (efficiency) หมายถึง สามารถสร้างได้ด้วยเวลาที่น้อย 4) การใช้งาน (usability) เป็นรูปแบบเปิดกว้างสำหรับทุกคนและไม่ซับซ้อน 5) ความทันสมัย (recency) ใช้เทคโนโลยีที่ทันสมัย

ตอนที่ 2 การเขียนโค้ด

2.1 ความหมายของการเขียนโค้ด

จิระศักดิ์ สุวรรณโณ และคณะ (2561) กล่าวว่า การเขียนโค้ด (coding) เป็นขั้นตอนการเขียนทดสอบ และดูแลรหัสต้นฉบับหรือซอร์ซโค้ด (source code) ของโปรแกรมคอมพิวเตอร์ ซึ่งซอร์ซโค้ดนั้นจะเขียนด้วยภาษาโปรแกรม การเขียนโปรแกรมคอมพิวเตอร์โดยทั่วไปผู้เขียนจำเป็นต้อง

จดจำคำสั่งและไวยากรณ์ของภาษาคอมพิวเตอร์ให้แม่นยำก่อน จึงจะสามารถสร้างชุดคำสั่งที่ต้องการได้ ซึ่งคำสั่งเหล่านั้นเป็นภาษาอังกฤษและมีรูปแบบที่แน่นอนตายตัว ถ้าเขียนคำสั่งผิดพลาดเพียงเล็กน้อยโปรแกรมจะไม่สามารถทำงานได้ ทำให้เป็นอุปสรรคสำคัญต่อการเรียนรู้ของผู้เริ่มต้นใหม่ มักเกิดความเบื่อหน่ายต่อคำสั่งเหล่านั้น จึงเกิดวิธีการสร้างชุดคำสั่งแบบบล็อกขึ้น ทำให้การสร้างชุดคำสั่งทำได้ง่าย เพียงใช้การลากบล็อกคำสั่งที่ต้องการมาเรียงต่อกัน ช่วยให้ผู้เรียนโฟกัสไปที่กระบวนการคิดมากกว่าการแก้ปัญหาเรื่องการพิมพ์คำสั่งผิด ที่ให้ผู้ใช้ลากบล็อกมาต่อกัน โดยบล็อกแต่ละตัวจะเป็นตัวแทนคำสั่งที่แตกต่างกันออกไป เมื่อนำบล็อกมาเรียงต่อกันเป็นลำดับตามขั้นตอนการทำงาน ก็สามารถทำงานได้เหมือนกับโปรแกรม เช่นเดียวกับโปรแกรมภาษาอื่น ๆ

การเขียนโค้ดที่เป็นที่แพร่หลายมากกว่าคือ ภาษาแบบข้อความ (text-based language) ซึ่งเป็นที่ทราบกันโดยทั่วไปว่า ภาษา Python เป็นภาษาที่ได้รับความนิยมสูงเป็นอันดับต้น ๆ ของโลกในปัจจุบันนี้เนื่องจากเป็นภาษาที่สามารถทำงานได้หลากลักษณะและใช้งานกับสถานการณ์จริงได้หลากหลาย ซึ่งสำหรับผู้ที่เรียนต้นเรียนการเขียนโค้ด ภาษา Python ก็เป็นภาษาที่สามารถอ่านและมีโครงสร้างการเขียนที่ง่ายกว่าภาษาอื่น ๆ เนื่องจากมีไวยกรณ์คล้ายภาษาอังกฤษ และยังบังคับให้เขียนโปรแกรมได้ตามโครงสร้างที่ถูกต้อง (Zyl et al., 2020)

งานวิจัยนี้ศึกษาเกี่ยวกับภาษาโปรแกรม 2 ภาษาคือ ภาษา Scratch ที่เป็นภาษาบล็อก และภาษา Python ที่เป็นภาษาข้อความ เพื่อให้เหมาะกับบริบทของผู้เรียนใหม่ที่เริ่มเรียนการเขียนโค้ด

2.2 องค์ประกอบของความสามารถด้านการเขียนโค้ด

การเขียนโค้ดเป็นกระบวนการแก้ไขปัญหาอย่างหนึ่งซึ่งต้องอาศัยความรู้เฉพาะทางด้านคอมพิวเตอร์ประกอบกันถึงจะสามารถเขียนโค้ดได้สำเร็จ ซึ่ง Raadt (2008) ได้ให้ควาหมายของการแก้ปัญหา ไว้ว่า “การแก้ปัญหาเป็นกลไกที่เกิดขึ้นเพื่อให้ได้มาซึ่งคำตอบของปัญหาของการเขียนโปรแกรม” ดังนั้น การเขียนโค้ดจึงแบ่งออกเป็น 2 ส่วน คือ การแก้ไขปัญหาและความรู้เกี่ยวกับด้านคอมพิวเตอร์

Winslow (1996) ได้เสนอวิธีการแก้ปัญหาออกเป็น 4 ขั้นตอน ดังนี้ ขั้นที่ 1 ทำความเข้าใจปัญหา ขั้นที่ 2 กำหนดการแก้ปัญหาว่าทำอย่างไรโดยให้คิดออกมาในรูปแบบใดก็ได้ จากนั้นเปลี่ยนมาเป็นรูปแบบที่เข้ากับคอมพิวเตอร์ ขั้นที่ 3 แปลวิธีคิดออกมาเป็นภาษาโปรแกรม ขั้นที่ 4 ขั้นทดสอบและแก้จุดบกพร่องของโปรแกรม และได้ระบุว่า ขั้นที่ 2 เป็นขั้นตอนที่ยากที่สุดสำหรับผู้เรียนใหม่ ซึ่งขั้นที่ 2 สอดคล้องกับแนวคิดการถ่ายโยงที่มีบริบทระหว่างรูปแบบที่ไม่ใช่คอมพิวเตอร์กับคอมพิวเตอร์ ซึ่ง Mayer (1992) ได้เสนอแนวทางการสอนเพื่อให้เกิดการพัฒนาการถ่ายโยงไว้ 3 วิธี ดังนี้ 1) การสอนมโนทัศน์โดย เริ่มจากการระบุมโนทัศน์เบื้องต้นที่อาจจะแตกต่างไปจากบริบททั่วไปในชีวิตจริง ระบุมโนทัศน์ที่คลาดเคลื่อน และนำเสนอมโนทัศน์ 2) การสอนทักษะเบื้องต้นที่จำเป็น

เช่น ทักษะการทำความเข้าใจกระบวนการ และ 3) การกระตุ้นให้ผู้เรียนพัฒนากลวิธีการแก้ปัญหา เช่น การแยกแยะปัญหาออกเป็นส่วนย่อย การทำให้เกิดการวนซ้ำ

ดังนั้น ผู้เรียนควรจะต้องได้เรียนรู้การแก้ไขปัญหาเพื่อให้มีความสามารถในการแก้ปัญหาก่อน จากงานวิจัยของ Koulouriet (2015) พบว่า การสอนแก้ปัญหาก่อนการสอนเขียนโปรแกรมสามารถทำให้ผู้เรียนมีประสิทธิภาพมากกว่า ในการสอนเขียนโค้ดที่เน้นการแก้ปัญหาสามารถเรียนรู้ได้จากการเขียนโค้ดที่เป็นภาษาบล็อก Soloway และ Spohrer (2013) ได้เสนอว่า การใช้ภาษาบล็อกทำให้เห็นถึงกระบวนการทำงานว่า มีความง่าย ไม่ต้องให้ความช่วยเหลือในช่วงแรกมาก ลดข้อจำกัดในการเขียนโค้ดลง และ Meerbaum-Salant, Armoni, และ Ben-Ari (2010) กล่าวว่า การเริ่มต้นศึกษาด้วยภาษาบล็อกทำให้ได้ผลลัพธ์ของการเรียนรู้ที่ยอดเยี่ยม

เมื่อผู้เรียนสามารถเข้าใจภาษาบล็อกได้แล้ว การเรียนรู้ภาษาโปรแกรมแบบข้อความก็เป็นสิ่งจำเป็นเนื่องจากเป็นภาษาที่ใช้ในบริบทของการทำงานจริงมากกว่า โดยจะต้องศึกษาเกี่ยวกับวากยสัมพันธ์ หรือ syntax ของภาษาคอมพิวเตอร์ และความรู้ด้านการเขียนโปรแกรมอื่น ๆ เช่น ตัวแปร ฟังก์ชัน โครงสร้างของข้อมูล Su et al. (2005) ได้เสนอ องค์ความรู้ด้านการเขียนโปรแกรมประกอบด้วย 3 องค์ประกอบคือ ขั้นตอนวิธี (algorithm) ความซับซ้อน (complexity) และโครงสร้างข้อมูล (data structure) โดย ขั้นตอนวิธี หมายถึง ขั้นตอนในการแก้ไขปัญหาหรือการทำงานโดยมีลำดับขั้นตอนของคำสั่งหรือวิธีการที่ชัดเจนที่คอมพิวเตอร์สามารถปฏิบัติตามได้ ความซับซ้อน หมายถึง ปริมาณของเนื้อที่และเวลาที่ใช้ในการแก้ปัญหาหนึ่งของโปรแกรมและโครงสร้างข้อมูลหมายถึง รูปแบบของการจัดระเบียบของข้อมูล ซึ่งมีอยู่หลายรูปแบบ เช่น เขตข้อมูล (field) แถวลำดับ (array) ระเบียน (record) ต้นไม้ (tree) และรายการโยง (linked list)

นอกจากนี้ McGill และ Volet (1997) ได้แบ่งองค์ประกอบของความรู้เกี่ยวกับการเขียนโปรแกรมไว้ 3 องค์ประกอบ ซึ่งมีลักษณะเป็นทั้งความรู้ชัดแจ้งและความรู้เชิงกระบวนการ ไว้ดังนี้ 1) ความรู้เกี่ยวกับมโนทัศน์ เป็นความรู้เกี่ยวกับความหมายของการปฏิบัติงานของโปรแกรม การออกแบบขั้นตอนการแก้ปัญหา 2) ความรู้เกี่ยวกับวากยสัมพันธ์ เป็นความรู้เกี่ยวกับวากยสัมพันธ์ของภาษาโปรแกรม การนำไปประยุกต์ใช้ในการเขียนโปรแกรม และ 3) ความรู้เกี่ยวกับกลวิธี เป็นการออกแบบ เขียน และทดสอบโปรแกรม

งานวิจัยนี้แบ่งองค์ประกอบของการเขียนโค้ดออกเป็น 2 องค์ประกอบได้แก่ 1) องค์ประกอบด้านการแก้ปัญหา และ 2) องค์ประกอบด้านโปรแกรม ซึ่งแบ่งออกเป็น ขั้นตอนวิธี ความซับซ้อน โครงสร้างข้อมูล และวากยสัมพันธ์ (McGill & Volet, 1997; Su et al., 2005; Winslow, 1996)

2.3 การสอนเขียนโค้ด

สำหรับการสอนเขียนโค้ดในประเทศไทยอาจแบ่งออกเป็น 2 ระดับคือ การสอนเขียนโค้ดให้กับนักเรียนระดับประถมศึกษาและมัธยมศึกษาตามแนวทางของสสวท. และการสอนเขียนโค้ดให้กับครูผ่านการอบรมจากหน่วยงานทั้งรัฐและเอกชน โดยการสอนเขียนโค้ดให้กับนักเรียนระดับประถมศึกษาและมัธยมศึกษาตามแนวทางของสสวท. นั้นได้มีการกำหนดแนวทางสำหรับครูไว้ในคู่มือการใช้หลักสูตรรายวิชาพื้นฐานวิทยาศาสตร์ ฯ กล่าวโดยสรุปคือ เริ่มจากการสอนเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ผ่านกิจกรรมการเรียนรู้ในห้องเรียน การเขียนโค้ดด้วยภาษาโปรแกรมเริ่มจาก Scratch ต่อมาเป็นภาษา Python และมีเป้าหมายเพื่อพัฒนาแอปพลิเคชันที่บูรณาการกับศาสตร์อย่างสร้างสรรค์ซึ่งเป็นตัวชี้วัดหนึ่งของนักเรียนระดับชั้นมัธยมศึกษาปีที่ 3

ในส่วนของการสอนเขียนโค้ดให้กับครูผ่านการอบรมจากหน่วยงานทั้งรัฐและเอกชนสำหรับครูในประเทศไทย เริ่มในปี พ.ศ. 2561 ซึ่งเป็นช่วงแรกของการเตรียมความพร้อมของครู สถาบันส่งเสริมการสอนวิทยาศาสตร์และเทคโนโลยี (*สสวท.*) ได้จัดการอบรมเกี่ยวกับการสอนเขียนโค้ดในระดับประถมศึกษา ซึ่งเป็นการเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ (unplugged coding) มีเนื้อหาเป็นกิจกรรมการเรียนรู้ที่จัดขึ้นได้ง่าย ๆ ในห้องเรียน โดยสามารถใช้สื่อที่ไม่ใช่สื่ออิเล็กทรอนิกส์ได้ อีกหน่วยงานหนึ่งคือ สวทช. ส่งเสริมให้ครูใช้บอร์ดสมองกลฝังตัวในการสอนนักเรียนผ่านการเรียนรู้ควบคู่กับการเล่น นอกจากนี้ ยังมีการอบรบเกี่ยวกับการเขียนโค้ดที่จัดขึ้นโดยอาจารย์มหาวิทยาลัยหรือองค์กรเอกชน ซึ่งมีเนื้อหาที่หลากหลายกว่าเช่น การสอนภาษา Scratch ภาษา Python ภาษา HTML5 และภาษา CSS

การสอนการเขียนโค้ดในต่างประเทศได้มีการศึกษากันอย่างแพร่หลายจึงได้พบข้อเสนอแนะและแนวทางในการสอนมากมาย Li และ Prasad (2005) พบว่า การสอนเขียนโค้ดต้องมีการยกตัวอย่าง การทำแบบฝึกหัด การประเมินร่วมด้วย และการให้ผลป้อนกลับทันที เน้นปฏิบัติมากกว่าการบรรยาย และหลีกเลี่ยงการสอนในห้องปฏิบัติการทางคอมพิวเตอร์ที่ใช้เวลามากต่อวันหรือต่อสัปดาห์ในการฝึกฝนทักษะพื้นฐาน แต่ควรสอนผู้เรียนมีความกระตือรือร้น และสอนในภาษาและคำสั่งที่นักเรียนต้องการจะเรียน (Jacobson, 2016) นอกจากนั้นยังมีการศึกษาเกี่ยวการสอนการเขียนโค้ด 2 แบบ คือการเขียนโค้ดด้วยภาษาโปรแกรมแบบดั้งเดิมกับการเขียนโค้ดด้วยภาษาบล็อก พบว่า การเขียนโค้ดด้วยภาษาบล็อก จะสะดวกมากกว่าเนื่องจากลดการเขียนคำสั่ง (syntax) และลดภาระทางความคิดลง (cognitive load) (Kelleher & Pausch, 2005)

ตอนที่ 3 แนวคิดการพัฒนาบทเรียนการเรียนรู้แบบจุลภาค

บทเรียนการเรียนรู้แบบจุลภาคมีลักษณะเป็นโมดูลที่สามารถปรับเปลี่ยนได้ตามความเหมาะสมกับธรรมชาติของผู้เรียนที่แตกต่างกัน ซึ่งมีแนวคิดเกี่ยวกับการเรียนรู้แบบปรับเหมาะ

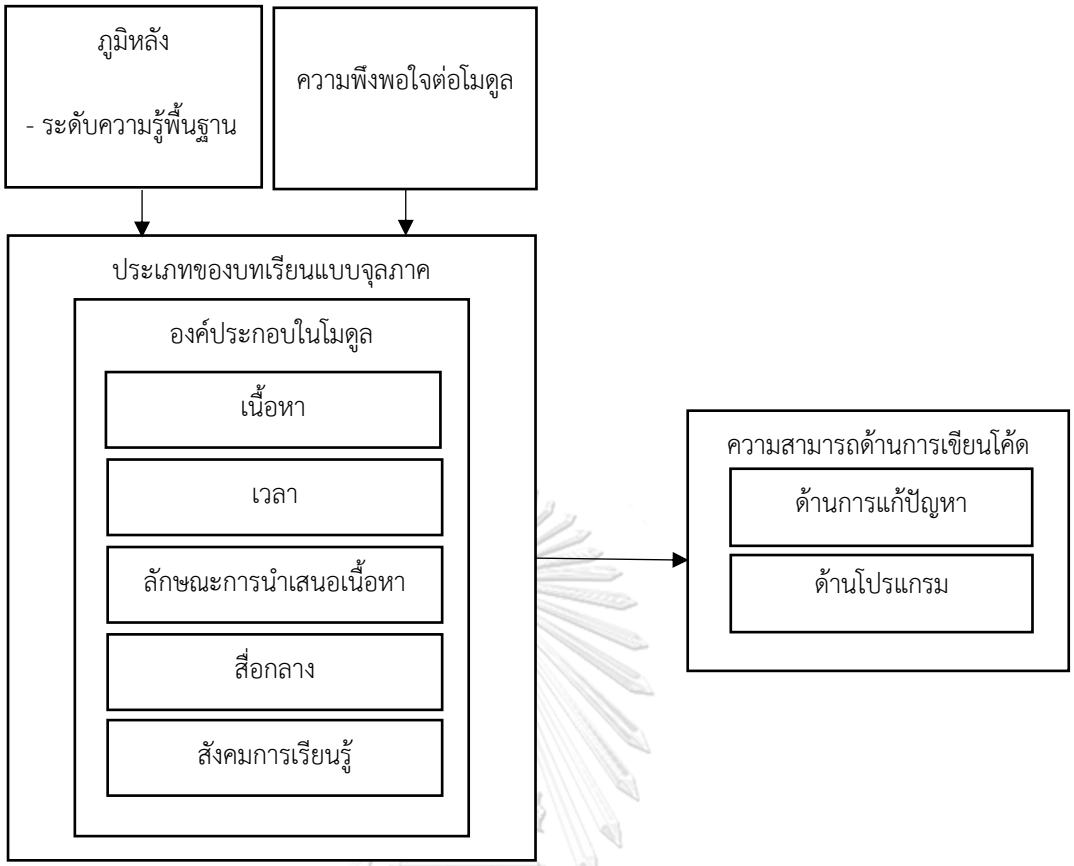
(adaptive learning) เป็นแนวคิดในการพัฒนาดังนี้ การเรียนรู้แบบปรับเหมาะ เป็นกระบวนการเรียนรู้ที่สื่อการเรียนรู้หรือเนื้อหา สามารถปรับเปลี่ยนเพื่อให้เหมาะสมกับความต้องการของผู้เรียนได้ ซึ่งจะได้การเรียนการสอนที่เป็นรายบุคคลโดยขึ้นอยู่กับระดับของความรู้ของผู้เรียน (Thalmann, 2014; Chaplot et al., 2016) นอกจากความรู้ของผู้เรียนที่นำมากำหนดการปรับเปลี่ยนบทเรียนให้กับผู้เรียนยังสามารถพิจารณาจากสิ่งอื่นอีกด้วย Somyurek (2009) ได้เสนอว่าสามารถพิจารณาจาก ตัวแปรคุณลักษณะ ตัวแปรภูมิหลัง และความชอบ เช่น เป้าหมายของผู้เรียน ขีดความสามารถ ประสบการณ์ด้านมัลติมีเดีย ความสนใจ และบุคลิกภาพ นอกจากนี้ Yang et al. (2013) ได้ศึกษาเกี่ยวกับการพัฒนาระบบการเรียนรู้แบบปรับเหมาะ (adaptive learning system) พบว่า ลีลาการเรียนรู้ที่แตกต่างกัน และภาระงานทางจิตใจที่น้อยลงเป็นตัวแปรสำคัญที่ส่งผลต่อผลสัมฤทธิ์ทางการเรียน ดังนั้น การเรียนรู้แบบจุลภาคที่มีลักษณะเด่นเรื่องการลดภาระทางจิตใจและภาระทางปัญญาจึงเป็นลักษณะสำคัญหนึ่งที่ทำให้เกิดผลสัมฤทธิ์ทางการเรียนภายใต้รูปแบบการเรียนรู้แบบปรับเหมาะ

การออกแบบลักษณะภายในโมดูลของบทเรียนการเรียนรู้แบบจุลภาคนั้นมีหลายองค์ประกอบซึ่งสามารถปรับเปลี่ยนได้หลากหลายรูแบบ โดยบทเรียนออนไลน์โดยทั่วไปมักจะเป็นการเรียนรู้ผ่านวิดีโอและออกแบบโดยผู้สอนทั้งหมดว่าแต่ละครั้งจะเรียนเนื้อหาใดบ้าง ลำดับการเรียนเป็นอย่างไร เพื่อให้บรรลุวัตถุประสงค์ของการเรียนบทเรียนนั้น ซึ่งการเรียนจะมีระยะเวลามากมาย ผู้เรียนสามารถเข้ามาศึกษาได้ตลอดเวลาตามความต้องการของผู้เรียน ซึ่งงานวิจัยนี้จะปรับเปลี่ยนบทเรียนออนไลน์ให้มีลักษณะแตกต่างออกไปตามแนวคิดของการเรียนรู้แบบจุลภาค จากงานวิจัยของ Bothe et al. (2019) เสนอว่า ควรให้วิดีโอมีความยาว 4-6 นาที เพื่อง่ายต่อการแบ่งเนื้อหาและสามารถครอบคลุมเนื้อหาได้ 1-3 สไลด์ ต้องตรวจสอบภาพและเสียงให้พร้อมกัน นอกจากนี้ ผู้เรียนควรสามารถข้ามหรือเรียงลำดับเนื้อหาใหม่ได้เองตามความชอบของผู้เรียน

CHULALONGKORN UNIVERSITY

กรอบแนวคิดการวิจัย

การวิจัยครั้งนี้เป็นการวิจัยเชิงทดลองแบบปรับเหมาะที่พัฒนาบทเรียนการเรียนรู้แบบจุลภาค โดยมีกระบวนการในการจัดการเรียนรู้ให้กับผู้เรียนตามแนวคิดการเรียนรู้แบบปรับเหมาะ โดยการให้โมดูลกับผู้เรียนที่ภูมิหลังต่างกันนั้นจะประเมินจากระดับความรู้พื้นฐานและความพึงพอใจต่อโมดูลที่เคยได้รับ เพื่อนำไปสู่โมดูลใหม่ในการเรียนเนื้อหาต่อไป เกิดการทำซ้ำจนครบทั้งหลักสูตรเพื่อให้ผู้เรียนได้เกิดการเรียนรู้แบบจุลภาคที่ปรับเหมาะกับผู้เรียนแต่ละคนจนเกิดผลลัพธ์ที่คาดหวังคือความสามารถด้านการเขียนโค้ด



ภาพ 2.1 กรอบแนวคิดการวิจัย

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

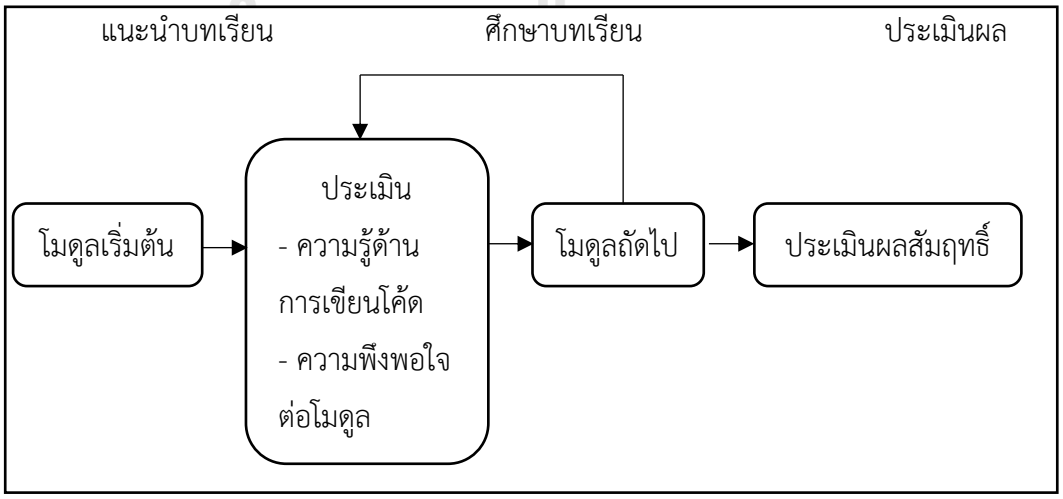
บทที่ 3
วิธีดำเนินการวิจัย

การวิจัยครั้งนี้เป็นการวิจัยเชิงทดลองแบบปรับเหมาะเพื่อส่งเสริมความสามารถด้านการเขียนโค้ดของนักศึกษาครู ผู้วิจัยได้แบ่งการดำเนินการวิจัยเป็น 2 ระยะ ดังนี้

ระยะที่ 1 การพัฒนาบทเรียนการเรียนรู้แบบจุลภาคด้านการเขียนโค้ด

การพัฒนาบทเรียนการเรียนรู้แบบจุลภาคด้านการเขียนโค้ดแบ่งออกเป็น 2 ส่วนคือ การออกแบบโครงสร้างของบทเรียนการเรียนรู้แบบจุลภาคด้านการเขียนโค้ดเป็นการออกแบบตามแนวคิดการเรียนรู้แบบปรับเหมาะ และการพัฒนาบทเรียนการเรียนรู้แบบจุลภาคด้านการเขียนโค้ดเป็นส่วนของการออกแบบองค์ประกอบต่าง ๆ ภายในโมดูล

ตอนที่ 1.1 การออกแบบโครงสร้างของบทเรียนการเรียนรู้แบบจุลภาคด้านการเขียนโค้ด แบ่งออกเป็น 3 ขั้นตอนย่อยได้แก่ แนะนำบทเรียน ศึกษาบทเรียน และประเมินผล โดยโมดูลเริ่มต้นจะเป็นการแนะนำบทเรียนซึ่งโมดูลนี้ออกแบบตามแนวคิดของการเรียนรู้แบบจุลภาค ตัวอย่างวิจัยทุกคนจะได้รับโมดูลนี้เหมือนกัน หลังจากนั้นตัวอย่างวิจัยจะได้รับการประเมินความรู้พื้นฐานด้านการเขียนโค้ดแบ่งเป็นกลุ่มที่มีความรู้พื้นฐานในระดับดี ระดับพอใช้ และไม่มีพื้นฐาน และความพึงพอใจต่อรูปแบบโมดูลที่ได้รับแบ่งเป็นกลุ่มที่ชอบเรียนด้วยตนเองและกลุ่มที่ชอบเรียนกับเพื่อน เพื่อจะได้รับโมดูลถัดไปที่ตอบสนองความต้องการบนพื้นฐานของความรู้และความชอบของตัวอย่างวิจัยแต่ละคน การศึกษาบทเรียนจะทำซ้ำจนกระทั่งตัวอย่างวิจัยได้เรียนครบทั้งหลักสูตร จากนั้นตัวอย่างวิจัยทุกคนจะได้รับการประเมินผลด้วยแบบทดสอบชุดเดียวกัน ดังภาพต่อไปนี้



ภาพ 3.1 โครงสร้างของบทเรียนการเรียนรู้แบบจุลภาคด้านการเขียนโค้ด

ความรู้ด้านการเขียนโค้ดประเมินตามเนื้อหาที่ได้ออกแบบไว้ในแต่ละโมดูลเป็นการทำแบบทดสอบย่อยหลังจากที่ได้ศึกษาเนื้อหานั้น ๆ จากกิจกรรมการเรียนรู้แบบจุลภาคด้วยสื่อการเรียนรู้ที่กำหนดไว้ซึ่งตัวอย่างแต่ละกลุ่มจะได้รับโมดูลแบ่งออกเป็นกลุ่ม ๆ เช่น กลุ่มที่ได้รับการเรียนรู้ด้วยสื่อเอกสารบรรยาย กลุ่มที่ได้รับการเรียนรู้ด้วยวิดีโอแบบโต้ตอบ

ความพึงพอใจต่อโมดูลประเมินตามองค์ประกอบของโมดูลที่ได้รับ โดยองค์ประกอบที่ทำการประเมินมีดังนี้ เวลา สื่อการเรียนรู้ ลำดับเนื้อหา และสังคมการเรียนรู้ โดยการประเมินจะเป็นการให้คะแนนความพึงพอใจและการแสดงความคิดเห็นเพื่อเป็นข้อมูลในการปรับเปลี่ยนโมดูลถัดไป โดยขั้นตอนในการปรับเปลี่ยนโมดูลมีดังนี้ 1) ประเมินระดับความพึงพอใจและข้อคิดเห็นต่อโมดูลแต่ละแบบ 2) พิจารณาโมดูลที่มีลักษณะตามองค์ประกอบที่ตรงกับข้อเสนอแนะ เป็นองค์ประกอบที่ได้คะแนนสูง และไม่เป็นองค์ประกอบที่ได้คะแนนต่ำ 3) จัดกลุ่มตัวอย่างวิจัยและให้โมดูลเดียวกันตามการประเมินในขั้นที่ 2

ตอนที่ 1.2 การพัฒนาบทเรียนการเรียนรู้แบบจุลภาคด้านการเขียนโค้ด

การศึกษาแนวคิดทฤษฎีเพื่อพัฒนาบทเรียนการเรียนรู้แบบจุลภาคด้านการเขียนโค้ด พบว่าองค์ประกอบของการเรียนรู้แบบจุลภาคประกอบด้วย 3 องค์ประกอบหลัก ได้แก่ เนื้อหา ศาสตร์การสอน และเทคโนโลยี ดังนั้นจึงแบ่งเป็น 1) การออกแบบเนื้อหาจุลภาค ซึ่งมีความเกี่ยวข้องกับองค์ประกอบด้านเนื้อหาและเทคโนโลยี (แพลตฟอร์มการเรียนรู้) และ 2) การออกแบบกิจกรรมจุลภาค ซึ่งมีความเกี่ยวข้องกับองค์ประกอบด้านศาสตร์การสอนและเทคโนโลยี (สื่อการเรียนรู้)

1.1 การออกแบบเนื้อหาจุลภาค

Buchem และ Hemelmann (2010) ได้เสนอหลักการออกแบบเนื้อหาจุลภาค โดยพิจารณาจาก 5 ด้านได้แก่ รูปแบบ จุดเด่น อิสระ โครงสร้าง และความสามารถในการระบุ ซึ่งมีรายละเอียดดังต่อไปนี้ งานวิจัยนี้เลือกสร้างบทเรียนการเรียนรู้แบบจุลภาคด้านการเขียนโค้ดบนแพลตฟอร์มการเรียนรู้แบบจุลภาคออนไลน์ที่มี *รูปแบบ* ที่เข้าถึงได้ง่ายและรวดจากทั้งโทรศัพท์มือถือ แทบเลต และคอมพิวเตอร์ และมี *จุดเด่น* ด้านเนื้อหาที่เฉพาะเจาะจงในแต่ละหัวข้อย่อยที่เรียนว่ากำลังเรียนเรื่องอะไรเช่น ความหมายของการเขียนโปรแกรม การใช้ฟังชัน for...loop (Python) และ *อิสระ* ผู้เรียนสามารถเรียนจบในหัวข้อนั้นแบบสมบูรณ์ในแต่ละครั้งจากเนื้อหาที่เตรียมไว้อย่างเหมาะสม แพลตฟอร์มการเรียนรู้แบบจุลภาคนี้ได้มีการจัด *โครงสร้าง* ที่แบ่งออกเป็นหัวข้อย่อยที่ชัดเจน ครบถ้วน ทั้งหัวข้อหลัก หัวข้อย่อย คำอธิบาย นอกจากนี้ ยังสามารถเข้าถึงบทเรียนได้จากลิงค์ที่มี *ความสามารถในการระบุ* ซึ่งเป็นลิงค์ที่เข้าถึงได้โดยตรง

เนื้อหาของบทเรียนการเรียนรู้จุลภาคประกอบด้วย การแก้ไขปัญหา ภาษา Scratch และภาษา Python โดยเนื้อหาเกี่ยวกับการแก้ไขปัญหาเป็นกิจกรรมที่อิงกับปัญหาหรือสถานการณ์ที่ให้

ผู้เรียนได้ตอบคำถามโดยเป็นการฝึกกระบวนการคิด เช่น เกม เกมปริศนา หรือชุดกิจกรรมการเรียนรู้ ส่วนเนื้อหาเกี่ยวกับโปรแกรมพื้นฐาน ประกอบด้วยความหมายของการเขียนโปรแกรม ภาษาโปรแกรม และรายละเอียดของเนื้อหาในภาษา Scratch และ ภาษา Python มีดังต่อไปนี้ ภาษา Scratch: ส่วนต่อประสานโปรแกรม ประเภทของบล็อก ขั้นตอนการทำงานของบล็อก การคำนวณค่าของข้อมูล และ การใช้งานฟังก์ชัน ภาษา Python: ตัวแปร ประเภทของข้อมูล การดำเนินการทางตรรกะและแขนง และฟังก์ชัน

1.2 การออกแบบกิจกรรมจุลภาค

Buchem และ Hemelmann (2010) ได้เสนอหลักการออกแบบกิจกรรมจุลภาค โดยพิจารณาจาก 5 ด้านได้แก่ 1) กลวิธีการสอน 2) กระบวนการเรียนรู้ 3) กิจกรรมการเรียนรู้แบบจุลภาค 4) วัสดุการเรียนรู้แบบจุลภาค 5) การเรียนรู้ในชุมชนการเรียนรู้ ซึ่งมีรายละเอียดดังต่อไปนี้ *กลวิธีการสอน* ใช้การส่งเสริมการเรียนรู้ด้วยตนเองกระตุ้นให้ผู้เรียนเข้ามาศึกษาบทเรียนตามระยะเวลาที่กำหนดจากในแพลตฟอร์มการเรียนรู้แบบจุลภาค *กระบวนการเรียนรู้* เป็นขั้นตอนทั่วไปประกอบด้วยขั้นนำ ขั้นกิจกรรม และขั้นสรุป แต่บทเรียนการเรียนรู้แบบจุลภาคนี้ ช่วยให้ผู้เรียนออกแบบได้เองในการลำดับเนื้อหา โดยการเลือกหัวข้อ กดข้าม กดดูซ้ำในสื่อการเรียนรู้ที่จัดไว้ให้ *กิจกรรมการเรียนรู้แบบจุลภาค* เป็นกิจกรรมที่ให้ผู้เรียนเกิดปฏิสัมพันธ์ระหว่างบทเรียนเช่น การตอบโต้กับ interactive video หรือ interactive video PDFs *วัสดุการเรียนรู้แบบจุลภาค* จัดเตรียมไว้ให้ผู้เรียนในโมดูลเพื่อให้ผู้เรียนได้เข้าถึงได้อย่างสะดวกและมีเนื้อหา ปริมาณที่เหมาะสม *การเรียนรู้ในชุมชนการเรียนรู้* บทเรียนการเรียนรู้แบบจุลภาคเปิดโอกาสให้ผู้เรียนแลกเปลี่ยนเรียนรู้กับเพื่อนและผู้สอนผ่อนกระดานถามตอบหรือการสนทนาส่วนตัวในแพลตฟอร์มการเรียนรู้แบบจุลภาค

ระยะที่ 2 การศึกษาผลของการใช้บทเรียนออนไลน์ด้านการเขียนโค้ดที่ออกแบบโดยแนวคิดการเรียนรู้แบบจุลภาค

ตอนที่ 2.1 กลุ่มเป้าหมาย และกลุ่มทดลอง

กลุ่มเป้าหมาย คือ นิสิตครูชั้นปีที่ 1-4 ของคณะครุศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย เนื่องจากเป็นนักศึกษาครูที่กำลังศึกษาเกี่ยวกับวิธีการสอนและประเด็นทางการศึกษาเพื่อพัฒนาตนเองให้เป็นครูที่มีประสิทธิภาพ งานวิจัยนี้จึงได้ศึกษากับกลุ่มเป้าหมายนี้เพื่อส่งเสริมความสามารถด้านเขียนโค้ดที่เป็นความสามารถที่สำคัญหนึ่งในการประกอบอาชีพครูในอนาคต

กลุ่มทดลอง คือ นักศึกษาครูชั้นปีที่ 1-4 กำหนดขนาดตัวอย่าง 30 คน ซึ่งได้มาจากการเลือกตามสะดวก

ตอนที่ 2.2 รูปแบบการทดลอง

ผู้วิจัยใช้หลักการ MOST และ SMART (Collins et al., 2007, 2014) เป็นแนวทางในการกำหนดรูปแบบการทดลอง โดยแบ่งออกเป็น 3 ระยะ ตามหลักการของ MOST ดังนี้ ระยะค้นหารูปแบบของทรีตเมนต์ ระยะตรวจสอบทรีตเมนต์กับลักษณะของผู้เรียน และระยะยืนยันผลการทดลอง โดยในระยะตรวจสอบทรีตเมนต์กับลักษณะของผู้เรียนจะใช้กลยุทธ์ SMART ร่วมด้วย เพื่อกำหนดวิธีการสุ่มให้ทรีตเมนต์กับผู้เรียน

1. ระยะค้นหารูปแบบของทรีตเมนต์

ทรีตเมนต์เป็นบทเรียนการเรียนรู้จุลภาคที่สร้างขึ้นตามแนวคิดการเรียนรู้แบบจุลภาคที่ได้ศึกษาจากเอกสารและงานวิจัยที่เกี่ยวข้อง โดยผู้วิจัยได้คัดเลือกส่วนสำคัญที่นำมาเป็นหลักในการสร้างทรีตเมนต์คือ องค์ประกอบหลักที่สำคัญทั้งหมด 4 องค์ประกอบ ตามระดับของตัวแปรทดลองทั้ง 4 ตัวแปร ซึ่งทรีตเมนต์ที่ได้ทั้งหมดนี้เป็นทรีตเมนต์ต้นแบบที่จะนำไปศึกษาต่อในระยะที่ 2 ต่อไป โดยแต่ละระดับของตัวแปรเป็นดังต่อไปนี้

ลำดับเนื้อหาของการเรียน แบ่งเป็นสองแบบคือผู้เรียนลำดับเนื้อหาได้เอง และผู้สอนลำดับเนื้อหาเอาไว้แล้ว

สื่อการเรียนรู้ สื่อการเรียนรู้ควรที่จะกระตุ้นให้สำรวจ ค้นหา และส่งเสริมการมีส่วนร่วมของผู้เรียน (Kerres, 2007) ดังนั้น สื่อการเรียนรู้จึงแบ่งเป็นสองแบบคือแบบ active กับแบบ passive ได้แก่ เอกสารเชิงโต้ตอบ (interactive document) วิดีโอเชิงโต้ตอบ (interactive video) รูปภาพกราฟิก และเอกสารการบรรยาย โดยเอกสารเชิงโต้ตอบเป็นเอกสารออนไลน์ที่ผู้เรียนสามารถมีปฏิสัมพันธ์กับเนื้อหาได้ เช่น การพิมพ์ตอบคำถาม การกดเพื่อให้แสดงภาพเคลื่อนไหว วิดีโอเชิงโต้ตอบเป็นวิดีโอที่ผู้เรียนสามารถพิมพ์ข้อความ เลือกคำตอบ หรือให้ข้อมูลแล้วแสดงผลได้ทันทีในวิดีโอ

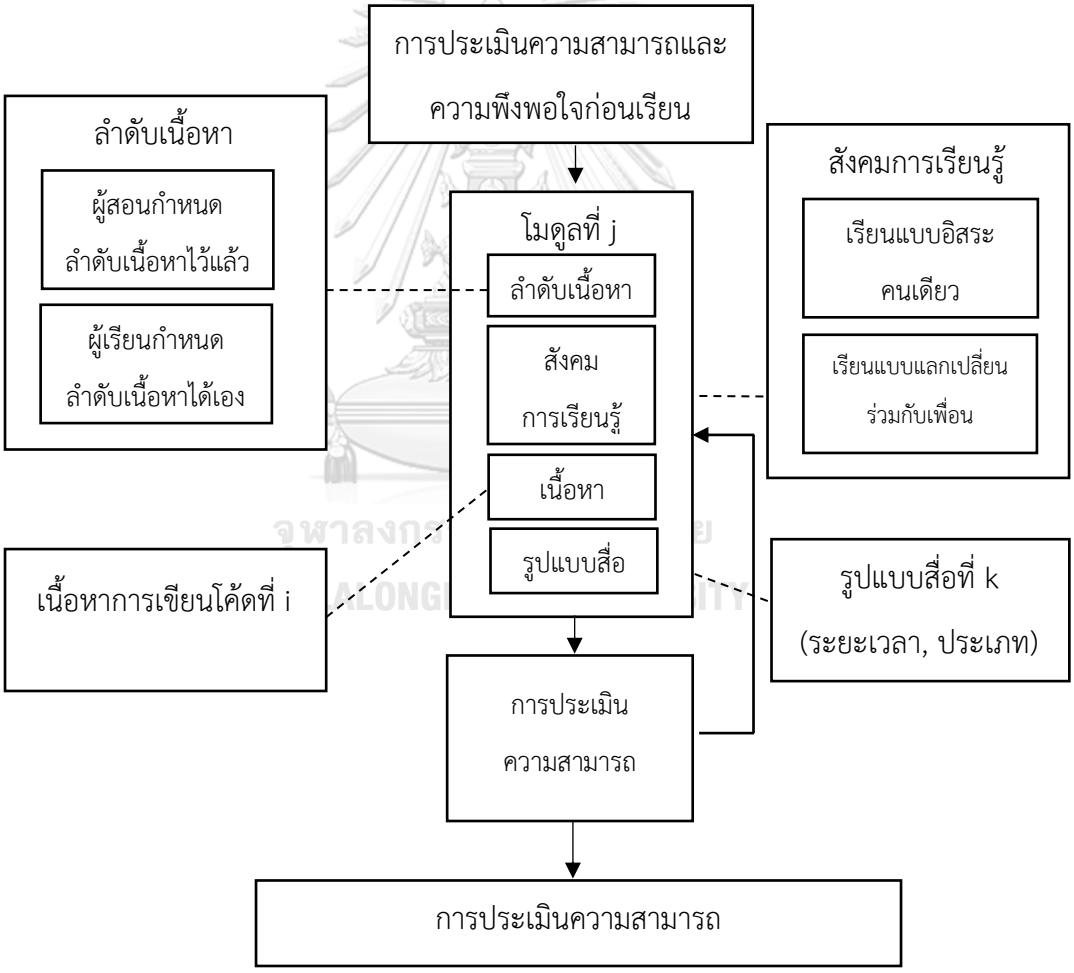
ระยะเวลา การกำหนดเวลาในการเรียนรู้แบบจุลภาคนั้น ไม่ได้มีข้อกำหนดที่เจาะจงขึ้นอยู่กับความเหมาะสมกับการถ่ายทอดความรู้และการฝึกหัดของผู้เรียน (Dave 2018; Dillon 2018) แต่ก็ยังมีการนำเสนอระยะเวลาของการจัดการเรียนรู้แบบจุลภาคโดยนักวิชาการไว้โดยทั่วไปเช่น Kapp และ Delice (2018) ได้เสนอไว้ว่าควรใช้เวลาประมาณ 5 นาที Bothe et al. (2019) ได้เสนอว่าควรใช้เวลาประมาณ 4–6 นาที และ Alqurashi (2017) ได้เสนอไว้ว่าควรใช้เวลาประมาณ 15–20 นาทีดังนั้น จึงกำหนดให้มีระดับด้านเวลาทั้งหมด 3 ระดับคือ 5-10 นาที (น้อย) 10-15 นาที (ปานกลาง) 15-20 นาที (มาก)

สังคมการเรียนรู้ แบ่งเป็นสองแบบคือ เรียนแบบอิสระคนเดียวและเรียนแบบแลกเปลี่ยนร่วมกับเพื่อน

2. ระยะตรวจสอบทรีตเมนต์กับลักษณะของผู้เรียน

ผู้วิจัยนำทรีตเมนต์ต้นแบบไปศึกษาต่อ โดยให้ทรีตเมนต์เป็นลำดับขั้นตามระยะเวลา ซึ่งพิจารณาจากความสามารถด้านการเขียนโค้ดและความพึงพอใจต่อลักษณะบทเรียนการเรียนรู้แบบจุลภาคตามกลยุทธ์ SMART ร่วมกับการเรียนรู้แบบปรับเหมาะในขั้นของการให้ทรีตเมนต์จะใช้ข้อมูลจากผู้เรียนแทนการสุ่มให้ทรีตเมนต์ โดยมีรายละเอียดดังนี้

ทรีตเมนต์ที่ให้กับตัวอย่างวิจัยในโมดูลที่ 1 จะปรับให้เหมาะกับข้อมูลที่ได้มาจากการประเมินความสามารถด้านการเขียนโค้ดก่อนเรียน (มีพื้นฐานระดับดี พอใช้ และไม่มีพื้นฐาน) หลังจากจบการเรียนครั้งที่ 1 แล้วผู้เรียนจะได้รับการประเมินความสามารถด้านการเขียนโค้ดเพื่อเป็นข้อมูลในการปรับเนื้อหาให้เหมาะสม ดำเนินการเช่นนี้ไปจนถึงการทดลองครั้งสุดท้าย จากนั้นจึงประเมินความสามารถด้านการเขียนโค้ดหลังเรียน



ภาพ 3.2 แบบแผนการทดลอง

หมายเหตุ เนื้อหาของการเขียนโค้ดกำหนดขึ้นจากการสังเคราะห์เอกสารและความคิดเห็นของผู้เชี่ยวชาญ จำนวนโมดูล รูปแบบสื่อ และเนื้อหาที่ตัวอย่างวิจัยได้รับขึ้นอยู่กับผลของการประเมินก่อนหน้า

หากพิจารณาตัวอย่างวิจัยเป็นรายบุคคลจะมีลำดับของการได้รับทรีตเมนต์ต่างกัน ตัวอย่างเช่นตัวอย่างวิจัยเริ่มเรียนบทเรียนที่ผู้สอนได้จัดลำดับเนื้อหาไว้ให้แล้ว เรียนร่วมกับเพื่อนสามารถแลกเปลี่ยนเรียนรู้กันได้ ในบทเรียนแรก ผู้เรียนได้รับเนื้อหา 1 ซึ่งเหมาะกับความสามารถด้านการเขียนโค้ดที่มี และรูปแบบสื่อ 1 ซึ่งเหมาะกับความชอบของผู้เรียน หลังจากจบการเรียนในบทเรียนแรกตัวอย่างวิจัยประเมินความสามารถด้านการเขียนโค้ดและความพึงพอใจ ในบทเรียนที่ 2 จะเป็นเนื้อหา 2 แต่รูปแบบสื่อ 1 เหมือนเดิมเนื่องจากตัวอย่างวิจัยนี้มีความพึงพอใจในรูปแบบสื่อนี้เพียงพออยู่แล้ว จากนั้นดำเนินการซ้ำจนถึงบทเรียนสุดท้าย ซึ่งอาจเป็นเนื้อหา 16 และรูปแบบสื่อเปลี่ยนไปเป็นรูปแบบสื่อ 8 แทน

3. ระยะยืนยันผลการทดลอง

ผู้วิจัยวิเคราะห์ผลการทดลองโดยใช้การวิเคราะห์โปรไฟล์ (profile analysis) เพื่อเปรียบเทียบโปรไฟล์และความสามารถด้านการเขียนโค้ดระหว่างได้รับบทเรียนออนไลน์ด้านการเขียนโค้ดที่ออกแบบตามการเรียนรู้แบบจุลภาคที่มีรูปแบบแตกต่างกันของนักศึกษาครูที่มีภูมิหลังแตกต่างกัน ซึ่งภูมิหลังประกอบด้วยระดับความรู้พื้นฐานและสาขาวิชา

ตอนที่ 2.3 เครื่องมือวิจัย

เครื่องมือวิจัย ประกอบด้วย 1) แบบสำรวจความพึงพอใจต่อลักษณะบทเรียนการเรียนรู้แบบจุลภาค 2) แบบทดสอบออนไลน์เกี่ยวกับความสามารถด้านการเขียนโค้ด

1. แบบสำรวจความพึงพอใจต่อลักษณะบทเรียนการเรียนรู้แบบจุลภาค

แบบสำรวจเป็นข้อคำถามแบบเลือกตอบ (checklist) ประกอบด้วยข้อคำถามตามองค์ประกอบของการเรียนรู้แบบจุลภาคได้แก่

1) ท่านชอบศึกษาหาความรู้แบบใด

☐ ศึกษาเรื่องต่าง ๆ ด้วยตนเอง ☐ ศึกษาหาความรู้ร่วมกับเพื่อน ๆ

2) ท่านชอบการเรียนโดยมีระยะเวลาในการเรียนแบบใด

☐ 5-10 นาที ☐ 10-15 นาที ☐ 15-20 นาที

3) ท่านชอบสื่อการเรียนรู้แบบใด

☐ เอกสารที่มีคำอธิบาย ข้อความ ☐ ภาพเคลื่อนไหว วีดิโอ

4) ท่านชอบลำดับเนื้อหาในการเรียนแบบใด

☐ เลือกเรียนเนื้อหาด้วยตนเอง ☐ เรียนเนื้อหาตามที่มีการจัดลำดับไว้ให้แล้ว

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

2. แบบทดสอบออนไลน์เกี่ยวกับความสามารถด้านการเขียนโค้ด

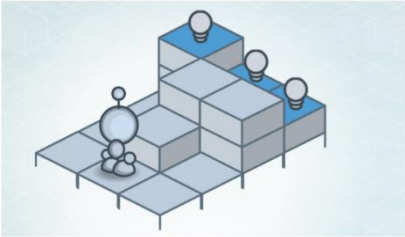
2.1) แบบทดสอบประกอบด้วย แบบทดสอบก่อนเรียนที่ออกแบบขึ้นเพื่อวัดความสามารถด้านการเขียนโค้ดตามองค์ประกอบของความสามารถด้านการเขียนโค้ด โดยแต่ละข้อสร้างขึ้นเพื่อวัดพฤติกรรมตามวัตถุประสงค์เชิงพฤติกรรม โดยในแต่ละข้อจะวัดในระดับพฤติกรรมตั้งแต่ ระดับเข้าใจ นำไปใช้ และสร้างสรรค์ ดังรายละเอียดต่อไปนี้

ตาราง 3.1 ตารางวิเคราะห์ข้อสอบของแบบทดสอบก่อนเรียน

วัตถุประสงค์เชิงพฤติกรรม	**ระดับพฤติกรรม**		
	เข้าใจ	**นำไปใช้**	**สร้างสรรค์**
องค์ประกอบที่ 1 ด้านแก้ไขปัญหา ความสามารถในการทำความเข้าใจปัญหา ออกแบบหรือกำหนดวิธีการแก้ไขปัญหาในรูปแบบของผังงาน ขั้นตอนวิธี และอื่น ๆ			
1.1 เขียนอธิบายขั้นตอนในการทำงานที่เป็นลำดับทางเดียว		1	
1.2 ตรวจสอบการทำงานของคำสั่งแบบมีเงื่อนไข		1	
1.3 ออกแบบขั้นตอนในการทำงานที่มีเงื่อนไขและการวนซ้ำ			1
			รวม 3 ข้อ
องค์ประกอบที่ 2 ด้านโปรแกรม การแปลวิธีคิด วิธีการแก้ปัญหาที่ออกแบบไว้ และเขียนคำสั่งเป็นภาษาโปรแกรมที่มีวากยสัมพันธ์ที่ถูกต้อง			
2.1 แปลความหมายวิธีคิด วิธีการแก้ปัญหาที่ออกแบบไว้	1		
2.2 เขียนคำสั่งเป็นภาษาโปรแกรมที่ถูกต้องตามวากยสัมพันธ์		1	
รวมทั้งหมด 5 ข้อ	1	3	1

ตัวอย่างข้อสอบในองค์ประกอบที่ 1

1. (นำไปใช้)



จากภาพ จงใช้คำสั่งจากข้อที่ 1-5 เพื่อแสดงขั้นตอนให้หุ่นยนต์ไปเปิดไฟทั้ง 3 ดวง

1. หุ่นยนต์เดินไปด้านหน้า 1 ช่อง (เดินได้ก็ต่อเมื่อพื้นอยู่ระดับเดียวกัน)
2. หุ่นยนต์หันซ้าย
3. หุ่นยนต์หันขวา
4. กระโดดขึ้นหรือลงไปข้างหน้า 1 ช่อง
5. เปิดไฟ

การตอบคำถาม : ให้ตอบด้วยตัวเลขเรียงตามลำดับขั้น เช่น 432531231 เป็นต้น

เกณฑ์การให้คะแนน (คะแนนเต็ม 3 คะแนน)

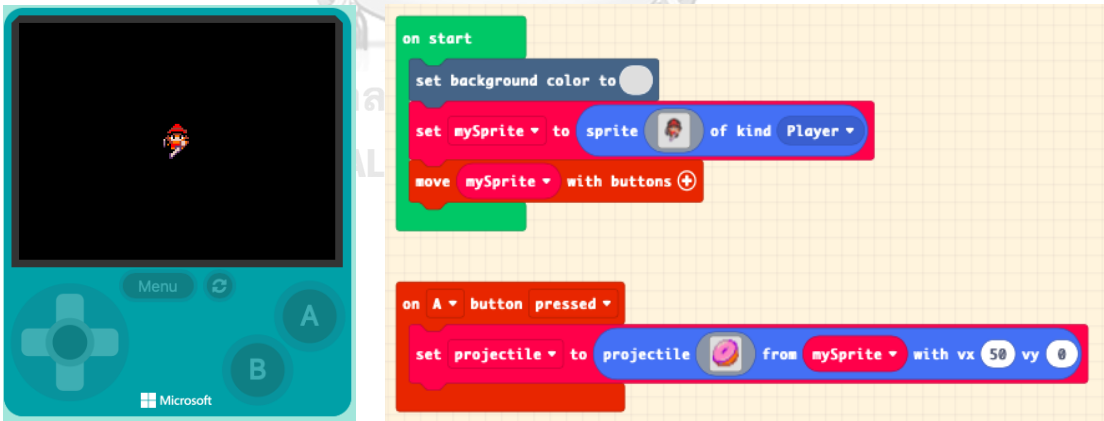
ได้ 3 คะแนน เมื่อ เขียนอธิบายให้หุ่นยนต์เปิดไฟจนครบ และใช้คำสั่งได้สอดคล้องกับข้อที่ 1

ได้ 2 คะแนน เมื่อ เขียนอธิบายให้หุ่นยนต์เปิดไฟจนครบ แต่ใช้คำสั่งไม่สอดคล้องกับข้อที่ 1 <u>หรือ</u> เขียนอธิบายให้หุ่นยนต์เดินได้ ไม่เกิดการเดินชน แต่เปิดไฟไม่ครบ และใช้คำสั่งได้สอดคล้องกับข้อที่ 1

ได้ 1 คะแนน เมื่อ เขียนอธิบายให้หุ่นยนต์เดินได้ ไม่เกิดการเดินชน แต่เปิดไฟไม่ครบ และใช้คำสั่งไม่สอดคล้องกับข้อที่ 1 <u>หรือ</u> เขียนอธิบายให้หุ่นยนต์เดินไม่ได้ แต่ใช้คำสั่งได้สอดคล้องกับข้อที่ 1

ได้ 0 คะแนน เมื่อ หุ่นยนต์เดินไม่ได้ และใช้คำสั่งไม่สอดคล้องกับข้อที่ 1 หรือไม่ตอบคำถาม

ตัวอย่างข้อสอบในองค์ประกอบที่ 2



2. จงอธิบายการทำงานของบล็อกคำสั่งต่อไปนี้ โดยให้อธิบายทุกบล็อกคำสั่งย่อยให้ครบทั้ง 4 บล็อก ว่าแต่ละบล็อกมีการทำงานอย่างไร (เข้าใจ)

บล็อกที่ 1

บล็อกที่ 2

<u>คำตอบ</u>

บล็อกที่ 1

- ตั้งสีพื้นหลัง
- สร้างตัวละครเป็นรูปนักดาบ
- บังคับตัวละครให้เคลื่อนที่จากปุ่มบนเครื่องเล่น

บล็อกที่ 2

- ตั้งให้ปุ่ม A ทำงานโดยการปล่อยรูปโดนัทจากตัวละครไปทางขวา

2.2) แบบทดสอบหลังเรียนที่ออกแบบขึ้นเพื่อวัดความสามารถด้านการเขียนโค้ดในองค์ประกอบด้านการแก้ไขปัญหาและองค์ประกอบด้านโปรแกรมจำแนกตามเนื้อหาบทเรียนที่ได้เรียนได้แก่เรื่อง การเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ ภาษา Scratch และภาษา Python ข้อสอบแต่ละข้อจะวัดในระดับพฤติกรรมตั้งแต่ระดับเข้าใจ สร้างสรรค์ และประเมินค่า

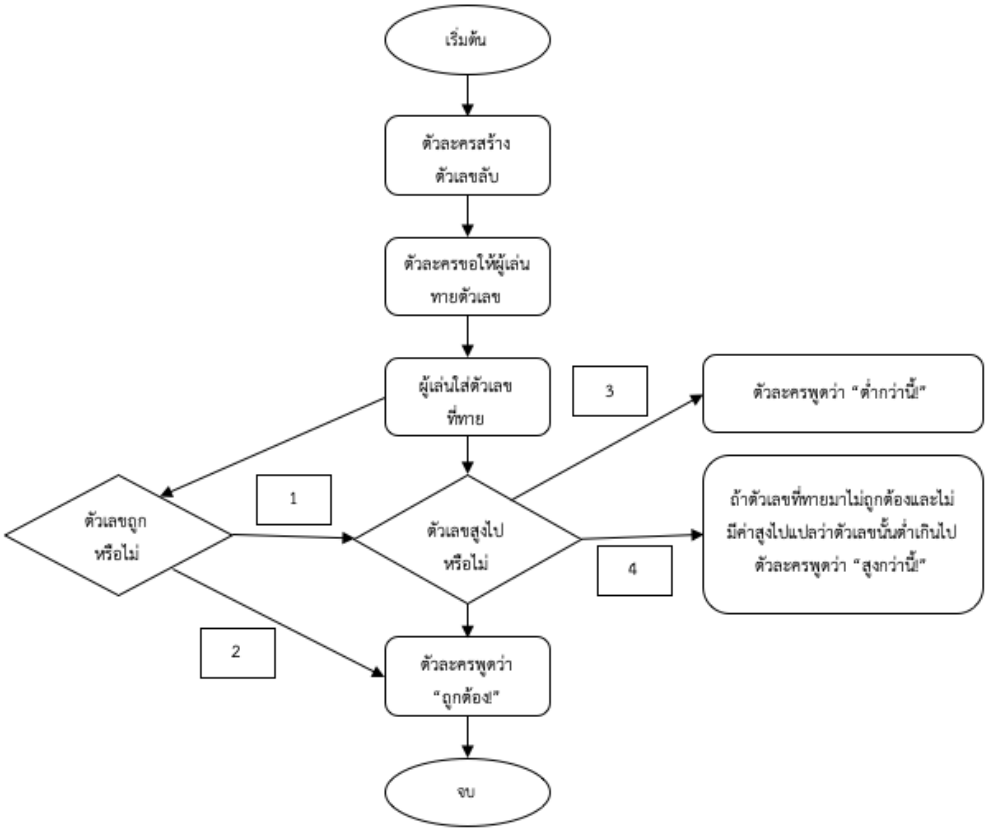
ตาราง 3.2 ตารางวิเคราะห์ข้อสอบของแบบทดสอบก่อนเรียน

เนื้อหา / องค์ประกอบ	**การเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์**	**ภาษา Scratch**	**ภาษา Python**
ด้านการแก้ไขปัญหา	ข้อ 1 (ประเมินค่า), ข้อ 2 (เข้าใจ)	ข้อ 4 (เข้าใจ)	ข้อ 5.1 (สร้างสรรค์)
ด้านภาษาโปรแกรม	-	ข้อ 3 (เข้าใจ)	ข้อ 5.2 (สร้างสรรค์) , ข้อ 6 (ประเมินค่า)
รวม	2	2	2

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ตัวอย่างข้อสอบหลังเรียน

จากผังงานต่อไปนี้จงตอบคำถามข้อที่ 1 – 2



1. การทำงานมีข้อผิดพลาดตรงส่วนใด และสามารถแก้ไขได้อย่างไร จงอธิบาย (ประเมินค่า)

<u>ตอบ</u>

การทำงานขาดตอนในกล่องด้านขวามือ ทำให้การทำงานหยุดเมื่อทำงานไปถึงเงื่อนไขนั้น เมื่อพิจารณาจากการทำงานของคำสั่ง จึงควรเพิ่มเส้นทางคำสั่งต่อจากกล่อง 2 กล่องขวาไปสู่กล่องที่มีคำสั่งว่า "ตัวละครขอให้ผู้เล่นทายตัวเลข"

2. พิจารณาเส้นทางการทำงาน 1 – 4 เงื่อนไขที่ทำให้คำสั่งในผังงานทำงานได้อย่างถูกต้องคือข้อใดตามลำดับ (เข้าใจ)

ก. ใช่ ไม่ ใช่ ไม่

ข. ไม่ ใช่ ไม่ ใช่

ค. ใช่ ไม่ ไม่ ใช่

ง. ไม่ ใช่ ใช่ ไม่*

ตอนที่ 2.4 การตรวจสอบคุณภาพเครื่องมือ

การตรวจสอบคุณภาพเครื่องมือมีการตรวจสอบความตรงเชิงเนื้อหา (content validity) และตรวจสอบความเที่ยงแบบสอดคล้องภายใน (internal consistency reliability) ดังนี้

1) การตรวจสอบคุณภาพเครื่องมือมีการตรวจสอบความตรงเชิงเนื้อหา (content validity) โดยผู้ทรงคุณวุฒิจำนวน 3 ท่าน พบว่าข้อสอบมีดัชนี IOC อยู่ระหว่าง .67 – 1.00 และได้แก้ไขปรับปรุงแบบทดสอบตามคำแนะนำของผู้ทรงคุณวุฒิ ในเรื่องการใช้ภาษา ความซับซ้อนของคำถาม เพื่อให้สามารถสื่อความหมายได้ดียิ่งขึ้น พร้อมทั้งระบุเกณฑ์การให้คะแนนให้ชัดเจนมากยิ่งขึ้น ดังนี้

ตาราง 3.3 ผลการตรวจสอบความตรงเชิงเนื้อหาของแบบทดสอบความสามารถด้านการเขียนโค้ดก่อนเรียน

องค์ประกอบ	วัตถุประสงค์เชิงพฤติกรรม	ค่า IOC
1. ด้านแก้ไขปัญหา	1.1 เขียนอธิบายขั้นตอนในการทำงานที่เป็นลำดับทางเดียว	1.00
	1.2 ตรวจสอบการทำงานของคำสั่งแบบมีเงื่อนไข	1.00
	1.3 ออกแบบขั้นตอนในการทำงานที่มีเงื่อนไขและการวนซ้ำ	1.00
2. ด้านโปรแกรม	2.1 แปลความหมายวิธีคิด วิธีการแก้ปัญหาที่ออกแบบไว้	.67
	2.2 เขียนคำสั่งเป็นภาษาโปรแกรมที่ถูกต้องตามวากยสมพันธ์	.67

ตาราง 3.4 ผลการตรวจสอบความตรงเชิงเนื้อหาของแบบทดสอบความสามารถด้านการเขียนโค้ดหลังเรียน

องค์ประกอบ	วัตถุประสงค์เชิงพฤติกรรม	จำนวนข้อ	ค่า IOC
1. ด้านแก้ไขปัญหา	ความสามารถในการทำความเข้าใจปัญหา ออกแบบหรือกำหนดวิธีการแก้ไขปัญหาในรูปแบบของ ผังงาน ขั้นตอนวิธี และอื่น ๆ	4	.67 - 1.00
2. ด้านโปรแกรม	การแปลวิธีคิด วิธีการแก้ปัญหาที่ออกแบบไว้ และเขียนคำสั่งเป็นภาษาโปรแกรมที่มีวากยสัมพันธ์ที่ถูกต้อง	3	1.00

2) การตรวจสอบความเที่ยงแบบสอดคล้องภายใน (internal consistency reliability) โดยการนำแบบทดสอบก่อนเรียนซึ่งเป็นข้อสอบแบบอัตนัยไปเก็บข้อมูลนำร่องจากนิสิตคณะครุศาสตร์จำนวน 46 คน ที่ไม่ใช่ตัวอย่างวิจัย พบว่า มีค่าสัมประสิทธิ์แอลฟาของครอนบาค (Cronbach's alpha coefficient) เท่ากับ .726

ตอนที่ 2.5 การเก็บรวบรวมข้อมูลและการวิเคราะห์ข้อมูล

การเก็บรวบรวมข้อมูลแบ่งออกเป็น 2 ระยะ ได้แก่ การเก็บข้อมูลก่อนการทดลอง และการเก็บข้อมูลระหว่างการทดลอง

1. การเก็บข้อมูลก่อนการทดลองใช้แบบฟอร์มออนไลน์เพื่อเก็บข้อมูลภูมิหลังและความสามารถด้านการเขียนโค้ด จากตัวอย่างจำนวน 46 คนที่ไม่ใช่ตัวอย่างวิจัยที่เป็นกลุ่มทดลองในระยะต่อไป เพื่อนำมาวิเคราะห์ข้อมูลโดยใช้สถิติพื้นฐาน ได้แก่ ความถี่ ร้อยละ ค่าเฉลี่ยเลขคณิต และส่วนเบี่ยงเบนมาตรฐาน สถิติทดสอบ independent sample t-test, Mann-Whitney U test, วิเคราะห์ความแปรปรวนทางเดียว (One-Way ANOVA), Kruskal-Wallis test และเทคนิค bootstrap

2. การเก็บข้อมูลระหว่างการทดลองใช้บทเรียนออนไลน์เรื่องการเขียนโค้ดเพื่อเก็บข้อมูลความสามารถด้านการเขียนโค้ดระหว่างเรียนและหลังเรียน จากตัวอย่างจำนวน 31 คนที่เป็นนิสิตคณะครุศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ชั้นปีที่ 1 – 4 ในสาขาวิชาและวิชาเอกที่หลากหลาย ใช้การเลือกตัวอย่างแบบตามสะดวก เพื่อนำข้อมูลมาวิเคราะห์โดยใช้สถิติต่อไปนี้ ความถี่ ร้อยละ ค่าเฉลี่ยเลขคณิต ส่วนเบี่ยงเบนมาตรฐาน และการวิเคราะห์โปรไฟล์ และการเก็บรวบรวมข้อมูลในขั้นนี้เมื่อได้ตัวอย่างวิจัยมาเรียบร้อยแล้วได้ให้คำแนะนำในการใช้บทเรียนออนไลน์บนแพลตฟอร์มการเรียนรู้ Canvas LMS บนเครื่องคอมพิวเตอร์ และแท็บเล็ต เพื่อเตรียมความพร้อมให้กับตัวอย่างวิจัยในการทดลองได้อย่างสะดวกยิ่งขึ้น

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 4
ผลการวิจัย

การวิจัยครั้งนี้มีวัตถุประสงค์ 3 ข้อได้แก่ 1) เพื่อเปรียบเทียบความสามารถด้านการเขียนโค้ดของนักศึกษาครูที่มีภูมิหลังแตกต่างกัน 2) เพื่อออกแบบและพัฒนาบทเรียนออนไลน์ด้านการเขียนโค้ดโดยใช้การเรียนรู้แบบจุลภาค 3) เพื่อเปรียบเทียบโปรไฟล์และความสามารถด้านการเขียนโค้ดระหว่างได้รับบทเรียนออนไลน์ด้านการเขียนโค้ดที่ออกแบบตามการเรียนรู้แบบจุลภาคที่มีรูปแบบแตกต่างกันของนักศึกษาครูที่มีภูมิหลังแตกต่างกัน จึงแบ่งการนำเสนอออกเป็น 3 ตอนดังต่อไปนี้

ตอนที่ 1 ความสามารถด้านการเขียนโค้ดของนักศึกษาครูที่มีภูมิหลังแตกต่างกัน

ข้อมูลภูมิหลัง

ตัวอย่างวิจัยที่ได้เก็บรวบรวมข้อมูลเป็นนิสิตคณะครุศาสตร์จุฬาลงกรณ์มหาวิทยาลัย จำนวน 46 คน หลังจากการตรวจสอบพบว่ามีตัวอย่างที่ให้ข้อมูลครบถ้วนจำนวน 31 คน เป็นผู้หญิงจำนวน 20 คน (ร้อยละ 64.52) ชั้นปีที่ 2 มากที่สุด จำนวน 9 คน คิดเป็นร้อยละ 29 สาขาวิชามัธยมศึกษาวิทยาศาสตร์มากที่สุด จำนวน 7 คน (ร้อยละ 22.6) และวิชาเอก คณิตศาสตร์มากที่สุด จำนวน 5 คน (ร้อยละ 25) รายละเอียดดังตารางต่อไปนี้

ตาราง 4.1 ความถี่ ร้อยละของตัวอย่างวิจัยจำแนกตามเพศ ชั้นปี สาขาวิชา และวิชาเอก

	เพศ					
	ชาย		หญิง		รวม	
	n	%	*n*	%	*n*	%
ชั้นปี						
1	4	36.36	3	15.00	7	22.58
2	2	18.18	7	35.00	9	29.03
3	1	9.10	6	30.00	7	22.58
4	4	36.36	4	20.00	8	25.81
สาขาวิชา						
มัธยมศึกษา (วิทย์)	3	27.27	4	20.00	7	22.58
มัธยมศึกษา (ศิลป์)	3	27.27	1	5.00	4	12.90
เทคโนโลยีการศึกษา	1	9.10	3	15.00	4	12.90
การศึกษานอกระบบโรงเรียน	1	9.10	4	20.00	5	16.13

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

	เพศ					
	ชาย		หญิง		รวม	
	n	%	*n*	%	*n*	%
อื่น ๆ	3	27.27	8	40.00	11	35.48
วิชาเอก						
คณิตศาสตร์	2	18.18	3	15.00	5	16.13
คอมพิวเตอร์	1	9.10	3	15.00	4	12.90
วิทยาศาสตร์	1	9.10	1	5.00	2	6.45
ภาษา	3	27.27	2	10.00	5	16.13
อื่น ๆ	4	36.36	11	55.00	15	48.39
รวม	11	100.00	20	100.00	31	100.00

เมื่อพิจารณาตัวอย่างวิจัยจากสาขาวิชา วิชาเอก โดยจำแนกตามชั้นปี พบว่า ชั้นปีที่ 1 อยู่สาขาเทคโนโลยีการศึกษา และวิชาเอกคอมพิวเตอร์มากที่สุด จำนวน 3 คน (ร้อยละ 42.86) ชั้นปีที่ 2 อยู่สาขาการศึกษานอกระบบโรงเรียนมากที่สุดจำนวน 3 คน (ร้อยละ 42.86) วิชาเอกอื่น ๆ จำนวน 7 คน (ร้อยละ 77.77) ชั้นปีที่ 3 อยู่สาขาอื่น ๆ และวิชาเอกอื่น ๆ มากที่สุดจำนวน 5 คน (ร้อยละ 71.42) และชั้นปีที่ 4 อยู่สาขามัธยมศึกษา (ศิลป์) มากที่สุดจำนวน 3 คน (ร้อยละ 37.50) และวิชาเอกภาษาจำนวน 5 คน (ร้อยละ 62.50)

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ตาราง 4.2 ความถี่ ร้อยละของตัวอย่างวิจัยจำแนกตามชั้นปี สาขาวิชา และวิชาเอก

	ชั้นปี									
	1		2		3		4		รวม	
	n	%	*n*	%	*n*	%	*n*	%	*n*	%
สาขาวิชา										
มัธยมศึกษา (วิทย์)	2	28.57	2	22.22	1	14.29	2	25.00	7	22.58
มัธยมศึกษา (ศิลป์)	0	0.00	1	11.11	0	0.00	3	37.50	4	12.90
เทคโนโลยีการศึกษา	3	42.86	0	0.00	1	14.29	0	0.00	4	12.90
การศึกษานอกระบบโรงเรียน	0	0.00	3	33.33	0	0.00	2	25.00	5	16.13
อื่น ๆ	2	28.57	3	33.33	5	71.42	1	12.50	11	35.48

	ชั้นปี									
	1		2		3		4		รวม	
	n	%	*n*	%	*n*	%	*n*	%	*n*	%
วิชาเอก										
คณิตศาสตร์	2	28.57	1	11.11	1	14.29	1	12.50	5	16.13
คอมพิวเตอร์	3	42.86	0	0.00	1	14.29	0	0.00	4	12.90
วิทยาศาสตร์	0	0.00	1	11.11	0	0.00	1	12.50	2	6.45
ภาษา	0	0.00	0	0.00	0	0.00	5	62.50	5	16.13
อื่น ๆ	2	28.57	7	77.77	5	71.42	1	12.50	15	48.39
รวม	7	100.00	9	100.00	7	100.00	8	100.00	31	100.00

นอกจากข้อมูลพื้นฐานเหล่านี้ ยังพบว่า มีตัวอย่างจำนวน 13 คน (ร้อยละ 41.94) ที่ไม่เคยเรียนการเขียนโค้ด และจำนวนที่เหลือหากจำแนกตามภาษาที่เคยเรียน (เรียนมากกว่า 1 ภาษา) ได้ดังนี้ C จำนวน 9 คน C++ จำนววน 9 คน Python จำนวน 8 คน และอื่น ๆ (C#, Java, JavaScript, PHP, R) จำนวน 6 คน

ความสามารถด้านการเขียนโค้ด

ผลการวิเคราะห์ความสามารถด้านการเขียนโค้ดของตัวอย่าง พบว่า จากคะแนนเต็ม 16 คะแนน ได้คะแนนเฉลี่ยเท่ากับ 6.95 ส่วนเบี่ยงเบนมาตรฐานเท่ากับ 3.72 คะแนนสูงสุดและคะแนนต่ำสุด เท่ากับ 15 และ 0 คะแนน คะแนนความสามารถด้านการเขียนโค้ดแบ่งตามองค์ประกอบได้ดังนี้ องค์ประกอบที่ 1 ด้านการแก้ไขปัญหา ได้คะแนนเฉลี่ยเท่ากับ 4.36 ส่วนเบี่ยงเบนมาตรฐานเท่ากับ 2.10 องค์ประกอบที่ 2 ด้านโปรแกรม ได้คะแนนเฉลี่ยเท่ากับ 2.60 คิดส่วนเบี่ยงเบนมาตรฐานเท่ากับ 2.33

การเปรียบเทียบความสามารถด้านการเขียนโค้ดจำแนกตามประสบการณ์ด้านการเขียนโค้ด และสาขาวิชา

เมื่อพิจารณาคะแนนแยกตามกลุ่มโดยใช้ประสบการณ์ในการเรียนเขียนโค้ด กับสาขาวิชาในการจำแนก ซึ่งจะแยกเป็น 3 กลุ่มได้แก่ กลุ่มที่เกี่ยวกับคอมพิวเตอร์ กลุ่มคณิตศาสตร์-วิทยาศาสตร์ และกลุ่มอื่น ๆ พบว่า กลุ่มที่เคยเรียนเขียนโค้ดจะมีความสามารถด้านการเขียนโค้ดสูงกว่ากลุ่มที่ไม่เคยเรียนเขียนโค้ดทั้ง 2 องค์ประกอบและคะแนนรวม กลุ่มที่มีวิชาเอกคอมพิวเตอร์มีคะแนน

ความสามารถด้านการเขียนโค้ดสูงที่สุด รองลงมาเป็นวิชาเอกคณิตศาสตร์ – วิทยาศาสตร์ และ อื่น ๆ ในทั้ง 2 องค์ประกอบ และคะแนนรวม ดังตารางต่อไปนี้

ตาราง 4.3 ค่าเฉลี่ย ส่วนเบี่ยงเบนมาตรฐานของคะแนนความสามารถด้านการเขียนโค้ด จำแนกตามสาขาวิชา และประสบการณ์การเขียนโค้ด

กลุ่ม	**จำนวนคน**	**คะแนนองค์ประกอบด้านการแก้ปัญหา**		**คะแนนองค์ประกอบด้านโปรแกรม**		**คะแนนรวม**	
		M	*SD*	*M*	*SD*	*M*	*SD*
ประสบการณ์ด้านการเขียนโค้ด							
ไม่เคยเรียนเขียนโค้ด	13	4.31	1.80	0.73	1.05	5.04	2.31
เคยเรียนเขียนโค้ด	18	4.39	2.28	3.94	2.06	8.33	3.99
วิชาเอก							
คอมพิวเตอร์	4	5.75	1.90	5.25	2.10	11.00	3.81
คณิตศาสตร์-วิทยาศาสตร์	7	5.43	2.44	4.07	2.05	9.50	2.81
อื่น ๆ	20	3.70	1.72	1.55	1.78	5.25	2.86

เมื่อวิเคราะห์ความแตกต่างของค่าเฉลี่ยของคะแนนรวม และคะแนนในแต่ละองค์ประกอบระหว่างกลุ่มที่เคยเรียนการเขียนโค้ดกับกลุ่มที่ไม่เคยเรียนการเขียนโค้ด พบว่า ข้อตกลงเบื้องต้นเกี่ยวกับตัวแปรสุ่มแบบปกติ (normality assumption) โดยการทดสอบ Shapiro-Wilk test ของคะแนนรวมและคะแนนองค์ประกอบด้านการแก้ไขปัญหาไม่พบนัยสำคัญ แต่คะแนนองค์ประกอบด้านโปรแกรม พบว่า W=0.908, p=.012 ซึ่งทำให้ไม่ผ่านข้อตกลงเบื้องต้นเกี่ยวกับตัวแปรสุ่มแบบปกติ ดังนั้น ในการนำเสนอจึงใช้เทคนิคการสุ่มซ้ำ bootstrap และการทดสอบแบบไม่ใช่พารามิเตอร์เพิ่มเติมได้ผลดังนี้

1) independent sample t-test พบว่า คะแนนรวมระหว่างกลุ่มที่เคยเรียนการเขียนโค้ดกับกลุ่มที่ไม่เคยเรียนการเขียนโค้ดแตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ 0.5, $t(29) = 2.68$, $p = .01$, $d = 0.97$ และคะแนนองค์ประกอบด้านโปรแกรมระหว่างกลุ่มที่เคยเรียนการเขียนโค้ดกับกลุ่มที่ไม่เคยเรียนการเขียนโค้ดแตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ 0.5, $t(29) = 5.14$, $p < .001$, $d = 1.87$

นอกจากนี้ Bootstrap for independent sample t-test พบว่า คะแนนรวมระหว่างกลุ่มที่เคยเรียนการเขียนโค้ดกับกลุ่มที่ไม่เคยเรียนการเขียนโค้ดแตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ 0.5 (t_b = .043, p = .007) และและคะแนนองค์ประกอบด้านโปรแกรมระหว่างกลุ่มที่เคยเรียนการเขียนโค้ดกับกลุ่มที่ไม่เคยเรียนการเขียนโค้ดแตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ 0.5 (t_b .0084, p = .003)

2) Mann-Whitney U test พบว่า คะแนนรวมระหว่างกลุ่มที่เคยเรียนการเขียนโค้ดกับกลุ่มที่ไม่เคยเรียนแตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ 0.5, U = 49.00, p = .007, d = 0.97 และคะแนนองค์ประกอบด้านโปรแกรมระหว่างกลุ่มที่เคยเรียนการเขียนโค้ดกับกลุ่มที่ไม่เคยเรียนการเขียนโค้ดแตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ 0.5, U = 117.00, p < .001, d = 1.87

เมื่อวิเคราะห์ความแตกต่างของค่าเฉลี่ยของคะแนนรวม และคะแนนในแต่ละองค์ประกอบระหว่างวิชาเอกพบว่า ข้อตกลงเบื้องต้นเกี่ยวกับตัวแปรสุ่มแบบปกติ (normality assumption) โดยการทดสอบ Shapiro-Wilk test ของคะแนนรวมและคะแนนองค์ประกอบด้านการแก้ไขปัญหาไม่พบนัยสำคัญ แต่คะแนนองค์ประกอบด้านโปรแกรม พบว่า W=0.920, p=.024 ซึ่งทำให้ไม่ผ่านข้อตกลงเบื้องต้นเกี่ยวกับตัวแปรสุ่มแบบปกติ ดังนั้น ในการนำเสนอจึงใช้เทคนิคการสุ่มซ้ำ bootstrap และการทดสอบแบบไม่ใช่พารามิเตอร์ เพิ่มเติมได้ผลดังนี้

1) วิเคราะห์ความแปรปรวนทางเดียว (One-Way ANOVA) พบว่า คะแนนรวมระหว่างวิชาเอกอย่างน้อย 1 คู่ มีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ .05(F(2, 28) = 9.59, p < .001, η^2=.41) และคะแนนองค์ประกอบด้านโปรแกรมระหว่างวิชาเอกอย่างน้อย 1 คู่ มีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ .05 (F(2, 28) = 9.29, p < .001, η^2=.41) เมื่อพิจารณาเปรียบเทียบเป็นรายคู่พบว่า คะแนนรวมระหว่างกลุ่มวิชาเอกคอมพิวเตอร์กับลุ่มวิชาเอกอื่น ๆ แตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ .05 (t(28)=3.54, p = .040, d = .15) คะแนนรวมระหว่างกลุ่มวิชาเอกคณิตศาสตร์-วิทยาศาสตร์กับกลุ่มวิชาเอกอื่น ๆ แตกต่างกับอย่างมีนัยสำคัญทางสถิติที่ระดับ .05 (t(28)=3.26, p = .008, d = .12) คะแนนองค์ประกอบด้านโปรแกรมระหว่างวิชาเอกคอมพิวเตอร์กับกลุ่มวิชาเอกอื่น ๆ แตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ .05 (t(28)=3.60, p = .003, d = .12) และคะแนนองค์ประกอบด้านโปรแกรมระหว่างวิชาเอกวิชาเอกคณิตศาสตร์-วิทยาศาสตร์กับกลุ่มวิชาเอกอื่น ๆ แตกต่างกับอย่างมีนัยสำคัญทางสถิติที่ระดับ .05 (t(28)=3.06, p = .013, d = .09)

นอกจากนี้ Bootstrap for independent sample t-test พบว่า คะแนนรวมระหว่างกลุ่มวิชาเอกคอมพิวเตอร์กับลุ่มวิชาเอกอื่น ๆ แตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ .05 (ช่วงความเชื่อมั่น 95% มีค่าอยู่ระหว่าง 1.627 ถึง 9.120) คะแนนรวมระหว่างกลุ่มวิชาเอกคณิตศาสตร์-วิทยาศาสตร์กับกลุ่มวิชาเอกอื่น ๆ แตกต่างกับอย่างมีนัยสำคัญทางสถิติที่ระดับ .05 (ช่วงความเชื่อมั่น

95% มีค่าอยู่ระหว่าง 1.713 ถึง 6.858) คะแนนองค์ประกอบด้านโปรแกรมระหว่างวิชาเอกคอมพิวเตอร์กับกลุ่มวิชาเอกอื่น ๆ แตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ .05 (ช่วงความเชื่อมั่น 95% มีค่าอยู่ระหว่าง 1.670 ถึง 5.259) และคะแนนองค์ประกอบด้านโปรแกรมระหว่างวิชาเอกวิชาเอกคณิตศาสตร์-วิทยาศาสตร์กับกลุ่มวิชาเอกอื่น ๆ แตกต่างกับอย่างมีนัยสำคัญทางสถิติที่ระดับ .05 (ช่วงความเชื่อมั่น 95% มีค่าอยู่ระหว่าง 0.732 ถึง 4.462)

2) Kruskal-Wallis test พบว่า คะแนนรวมระหว่างวิชาเอกอย่างน้อย 1 คู่ มีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ .05(χ^2 (2,31)=12.28 p < .002, η^2=.41) และคะแนนองค์ประกอบด้านโปรแกรมระหว่างวิชาเอกอย่างน้อย 1 คู่ มีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ .05 (χ^2 (2, 31) = 10.68, p < .005, η^2=.36) เมื่อพิจารณาเปรียบเทียบเป็นรายคู่ พบว่า คะแนนรวมระหว่างกลุ่มวิชาเอกคอมพิวเตอร์กับลุ่มวิชาเอกอื่น ๆ แตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ .05 (W=3.41, p = .042) คะแนนรวมระหว่างกลุ่มวิชาเอกคณิตศาสตร์-วิทยาศาสตร์กับกลุ่มวิชาเอกอื่น ๆ แตกต่างกับอย่างมีนัยสำคัญทางสถิติที่ระดับ .05 (W=4.17, p = .009) คะแนนองค์ประกอบด้านโปรแกรมระหว่างวิชาเอกคอมพิวเตอร์กับกลุ่มวิชาเอกอื่น ๆ แตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ .05 (W=3.67, p = .026) และคะแนนองค์ประกอบด้านโปรแกรมระหว่างวิชาเอกวิชาเอกคณิตศาสตร์-วิทยาศาสตร์กับกลุ่มวิชาเอกอื่น ๆ แตกต่างกับอย่างมีนัยสำคัญทางสถิติที่ระดับ .05 (W=3.52, p = .034)

ตาราง 4.4 ผลของการเปรียบเทียบคะแนนรวม และคะแนนองค์ประกอบด้านโปรแกรมจำแนกตามประสบการณ์ด้านการเขียนโค้ดและวิชาเอก โดยใช้สถิติทดสอบประเภทต่าง ๆ

	สถิติทดสอบ			
	t-test	**bootstrap**	**Mann-Whitney U test**	**ผลลัพธ์**
ประสบการณ์ด้านการเขียนโค้ด				
คะแนนรวม	p = .01	p = .007	p = .007	พบนัยสำคัญ
คะแนนองค์ประกอบด้านโปรแกรม	p < .001	p = .003	p < .001	พบนัยสำคัญ
วิชาเอกคอม - วิชาเอกอื่น ๆ				
คะแนนรวม	p = .040	CI=(1.627 - 9.120)	p = .042	พบนัยสำคัญ
คะแนนองค์ประกอบด้านโปรแกรม	p = .003	CI=(1.670 - 5.259)	p = .026	พบนัยสำคัญ

	สถิติทดสอบ			
	t-test	bootstrap	Mann-Whitney U test	ผลลัพธ์
วิชาเอกวิทย์คณิต - วิชาเอกอื่น ๆ				
คะแนนรวม	p = .008	CI=(1.713 - 6.858)	p = .009	พบนัยสำคัญ
คะแนนองค์ประกอบด้านโปรแกรม	p = .013	CI=(0.732 - 4.462)	p = .034	พบนัยสำคัญ

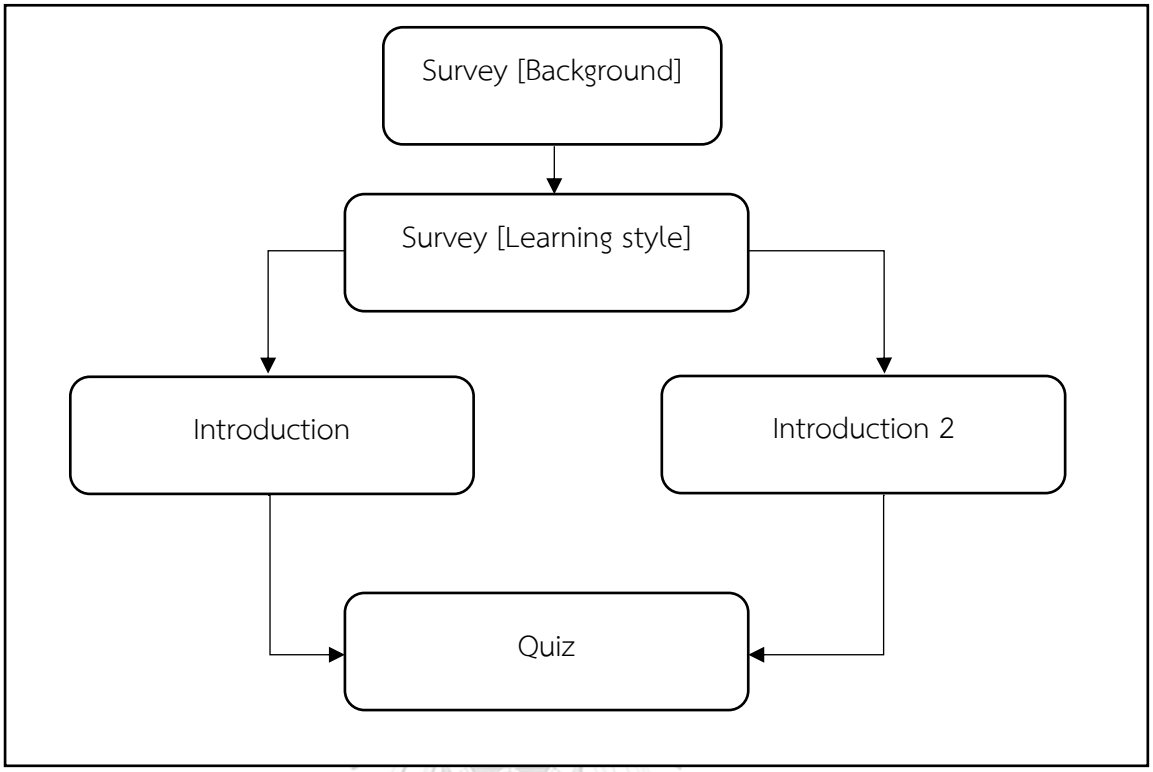
ตอนที่ 2 ผลของการออกแบบและพัฒนาบทเรียนออนไลน์ด้านการเขียนโค้ดโดยใช้การเรียนรู้แบบจุลภาค

การนำเสนอผลจะแบ่งออกเป็น 2 ตอน ได้แก่ โครงสร้างของบทเรียนออนไลน์ด้านการเขียนโค้ดโดยใช้การเรียนรู้แบบจุลภาค และบทเรียนออนไลน์ด้านการเขียนโค้ดโดยใช้การเรียนรู้แบบจุลภาค

1. โครงสร้างของบทเรียนออนไลน์ด้านการเขียนโค้ดโดยใช้การเรียนรู้แบบจุลภาค

การออกแบบและพัฒนาบทเรียนออนไลน์ด้านการเขียนโค้ดโดยใช้การเรียนรู้แบบจุลภาค แบ่งบทเรียนออกเป็นโมดูล 5 โมดูล ได้แก่ PreClass, Introduction of coding, Scratch, Python และ Posttest ซึ่งสอดคล้องกับโครงสร้างของบทเรียนการเรียนรู้แบบจุลภาคด้านการเขียนโค้ดที่ได้ออกแบบไว้ ซึ่งมีรายละเอียดดังตารางต่อไปนี้

1.1) PreClass เป็นโมดูลเริ่มต้นที่ทุกคนจะได้รับ ประกอบด้วยแบบสำรวจข้อมูลภูมิหลังและความพึงพอใจต่อการเรียนรู้แบบจุลภาค เมื่อได้รับการประเมินจะได้รับบทเรียนตามความเหมาะสมแบ่งออกเป็น 2 ลักษณะคือ บทเรียนมีสื่อการเรียนรู้เป็นแบบ interactive กับแบบข้อความ เมื่อสิ้นสุดบทเรียนจะได้รับการประเมินความสามารถด้านการเขียนโค้ด เป็นลำดับดังแผนภาพต่อไปนี้



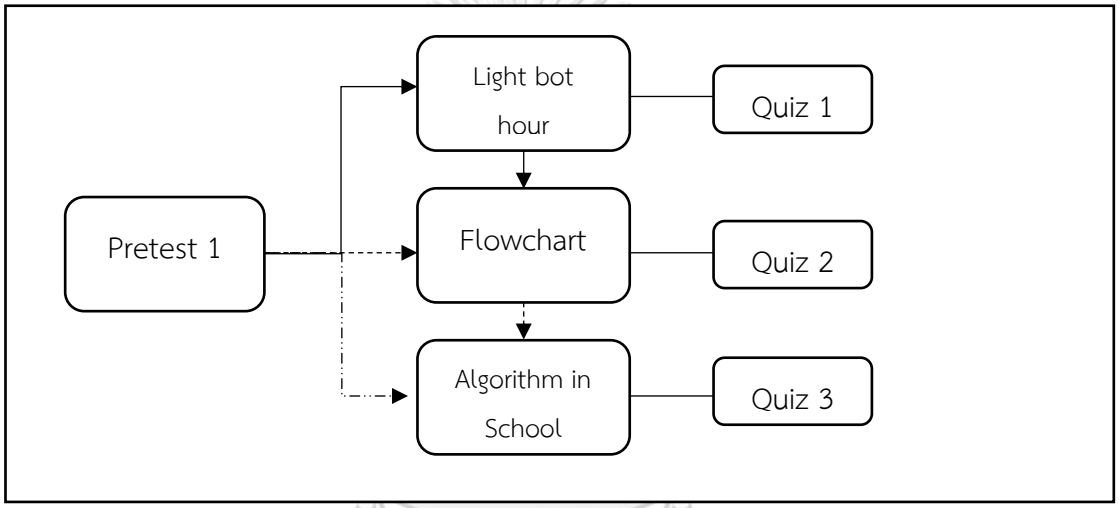
ภาพ 4.1 ลำดับการเรียนของโมดูลเริ่มต้น PreClass

กลุ่มที่ 1 จะได้รับบทเรียน Introduction ซึ่งประกอบด้วย 3 บทเรียนย่อยได้แก่ พื้นฐานในการทำงาน ประเภทข้อมูล และการเขียนคำสั่ง ที่มีสื่อการเรียนรู้เป็นแบบ interactive ส่วนกลุ่มที่ 2 จะได้รับบทเรียนที่เนื้อหาเดียวกันแต่มีสื่อการเรียนรู้เป็นแบบข้อความ และทั้งสองกลุ่มจะได้รับการประเมินเดียวกัน

จุฬาลงกรณ์มหาวิทยาลัย

ภาพ 4.2 ลำดับการเรียนของโมดูลเริ่มต้น PreClass บน Canvas LMS

1.2) Introduction of coding เป็นโมดูลที่ประกอบด้วย แบบทดสอบก่อนเรียนเพื่อประเมินความสามารถของผู้เรียน โดยจะแบ่งผู้เรียนออกเป็น 3 กลุ่ม คือกลุ่มเก่ง กลุ่มกลาง กลุ่มอ่อน เพื่อให้ได้รับบทเรียนตามเงื่อนไขต่อไปนี้ กลุ่มอ่อนจะได้รับบทเรียนตั้งแต่พื้นฐานคือ บทเรียน lightbot hour พร้อมทั้งแบบทดสอบหลังเรียน ต่อไปเป็นบทเรียนที่ยากขึ้นคือ บทเรียน Flowchart พร้อมแบบทดสอบหลังเรียน และบทเรียนที่ท้าทายคือ บทเรียน Algorithm in School พร้อมแบบทดสอบหลังเรียน กลุ่มกลางจะได้รับบทเรียน Flowchart และ Algorithm in School เนื่องจากมีความสามารถพื้นฐานเพียงพอแล้ว และกลุ่มเก่งจะได้รับบทเรียนท้าท้ายคือ บทเรียน Algorithm in School เนื่องจากเป็นกลุ่มที่มีพื้นฐานที่ดีอยู่แล้ว โดยลำดับการได้รับบทเรียนเป็นดังแผนภาพต่อไปนี้



ภาพ 4.3 ลำดับการเรียนของ Introduction of coding

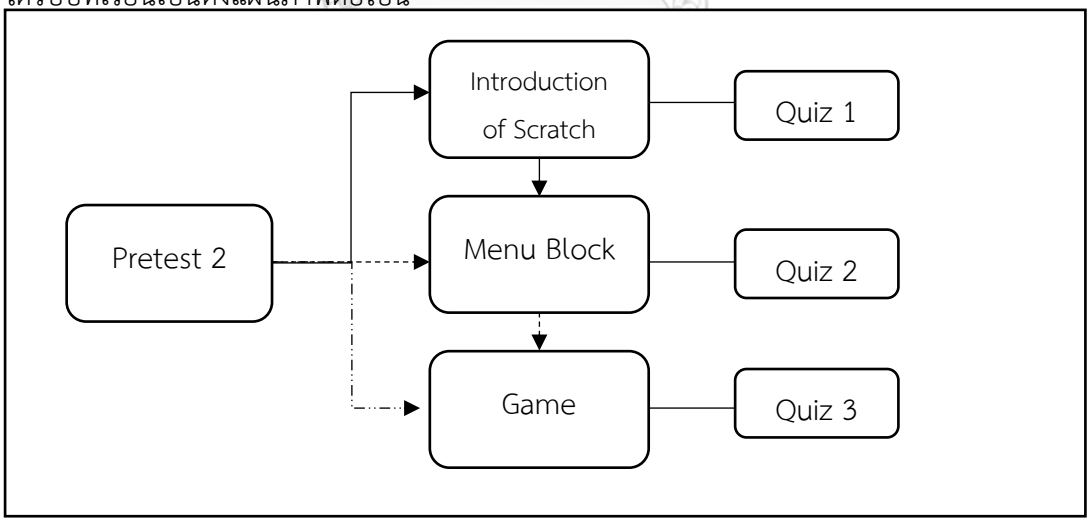
กลุ่มที่ 1 กลุ่มเก่งได้รับบทเรียน Algorithm in School และแบบทดสอบหลังเรียน

กลุ่มที่ 2 กลุ่มกลางได้รับบทเรียน Flowchart, Algorithm in School และแบบทดสอบหลังเรียน

กลุ่มที่ 3 กลุ่มอ่อนได้รับบทเรียน lightbot hour พร้อมทั้งแบบทดสอบหลังเรียน ต่อไปเป็น บทเรียน Flowchart พร้อมแบบทดสอบหลังเรียน และ บทเรียน Algorithm in School พร้อมแบบทดสอบหลังเรียน

ภาพ 4.4 ลำดับการเรียนของ Introduction of coding บน Canvas LMS

1.3) Scratch เป็นโมดูลที่ประกอบด้วย แบบทดสอบก่อนเรียนเพื่อประเมินความสามารถของผู้เรียน โดยจะแบ่งผู้เรียนออกเป็น 3 กลุ่ม คือกลุ่มเก่ง กลุ่มกลาง กลุ่มอ่อน เพื่อให้ได้รับบทเรียนตามเงื่อนไขต่อไปนี้ กลุ่มอ่อนจะได้รับบทเรียนตั้งแต่พื้นฐานคือ Introduction of Scratch พร้อมทั้งแบบทดสอบหลังเรียน ต่อไปเป็นบทเรียนที่ยากขึ้นคือ บทเรียน Menu Block พร้อมแบบทดสอบหลังเรียน และบทเรียนที่ท้าทายคือ บทเรียน Game พร้อมแบบทดสอบหลังเรียน กลุ่มกลางจะได้รับบทเรียน Menu Block และ Game เนื่องจากมีความสามารถพื้นฐานเพียงพอแล้ว และกลุ่มเก่งจะได้รับบทเรียนท้าท้ายคือ บทเรียน Game เนื่องจากเป็นกลุ่มที่มีพื้นฐานที่ดีอยู่แล้ว โดยลำดับการได้รับบทเรียนเป็นดังแผนภาพต่อไปนี้

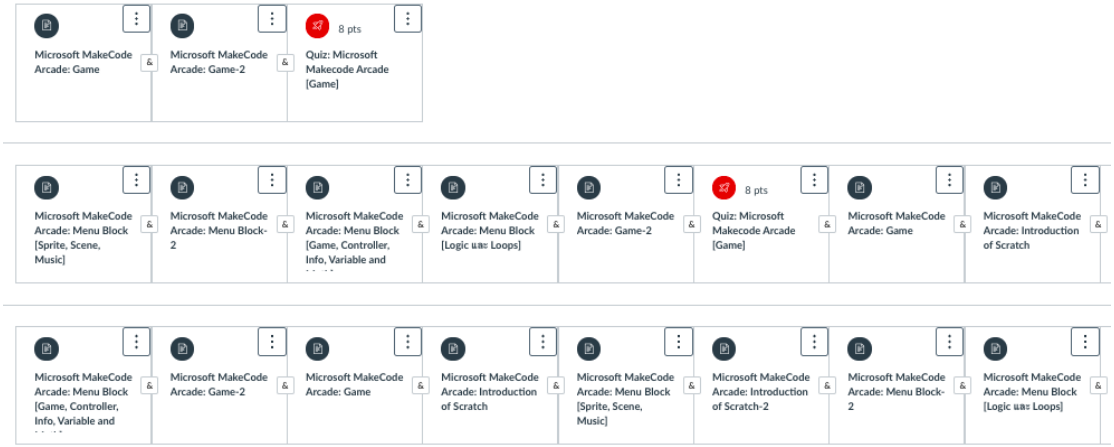


ภาพ 4.5 ลำดับการเรียนของ Scratch

กลุ่มที่ 1 กลุ่มเก่งได้รับบทเรียน Microsoft MakeCode Arcade: Game และแบบทดสอบหลังเรียน

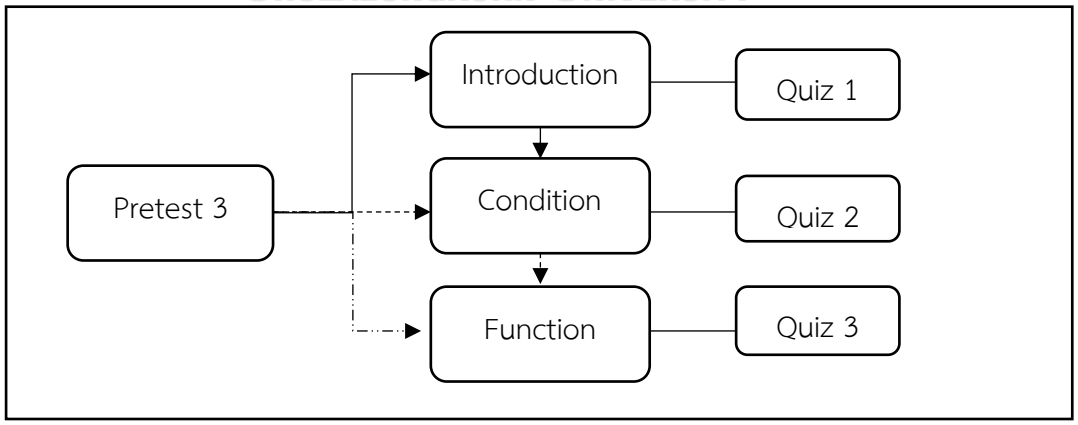
กลุ่มที่ 2 กลุ่มกลางได้รับบทเรียน Microsoft MakeCode Arcade: Menu Block, Microsoft MakeCode Arcade: Game และแบบทดสอบหลังเรียน

กลุ่มที่ 3 กลุ่มอ่อนได้รับบทเรียน Microsoft MakeCode Arcade: Introduction of Scratch ต่อไปเป็น บทเรียน Microsoft MakeCode Arcade: Menu Block และ บทเรียน MicrosoftMakeCode Arcade: Game พร้อมแบบทดสอบหลังเรียน



ภาพ 4.6 ลำดับการเรียนของ Scratch บน Canvas LMS

1.4) Python เป็นโมดูลที่ประกอบด้วย แบบทดสอบก่อนเรียนเพื่อประเมินความสามารถของผู้เรียน โดยจะแบ่งผู้เรียนออกเป็น 3 กลุ่ม คือกลุ่มเก่ง กลุ่มกลาง กลุ่มอ่อน เพื่อให้ได้รับบทเรียนตามเงื่อนไขต่อไปนี้ กลุ่มอ่อนจะได้รับบทเรียนตั้งแต่พื้นฐานคือ Introduction of Python พร้อมทั้งแบบทดสอบหลังเรียน ต่อไปเป็นบทเรียนที่ยากขึ้นคือ บทเรียน Condition พร้อมแบบทดสอบหลังเรียน และบทเรียนที่ท้าทายคือ บทเรียน Function and Libraries พร้อมแบบทดสอบหลังเรียน กลุ่มกลางจะได้รับบทเรียน Condition และ Function and Libraries เนื่องจากมีความสามารถพื้นฐานเพียงพอแล้ว และกลุ่มเก่งจะได้รับบทเรียนท้าท้ายคือ บทเรียน Function and Libraries เนื่องจากเป็นกลุ่มที่มีพื้นฐานที่ดีอยู่แล้ว โดยลำดับการได้รับบทเรียนเป็นดังแผนภาพต่อไปนี้



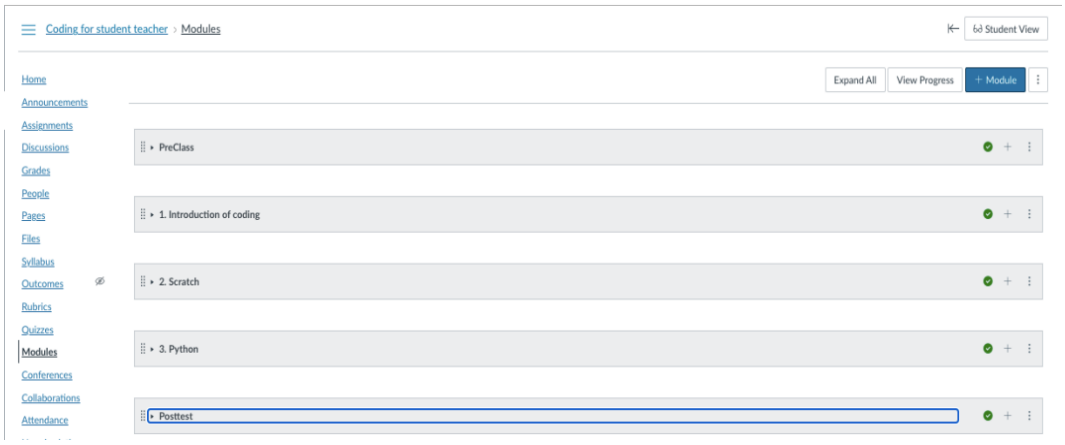
ภาพ 4.7 ลำดับการเรียนของ Python

กลุ่มที่ 1 กลุ่มเก่งได้รับบทเรียน Function and Libraries และแบบทดสอบหลังเรียน
กลุ่มที่ 2 กลุ่มกลางได้รับบทเรียน Condition, Function and Libraries และแบบทดสอบหลังเรียน
กลุ่มที่ 3 กลุ่มอ่อนได้รับบทเรียน Introduction of Python ต่อไปเป็น บทเรียน Condition และบทเรียน Function and Libraries พร้อมแบบทดสอบหลังเรียน

ภาพ 4.8 ลำดับการเรียนของ Python บน Canvas LMS

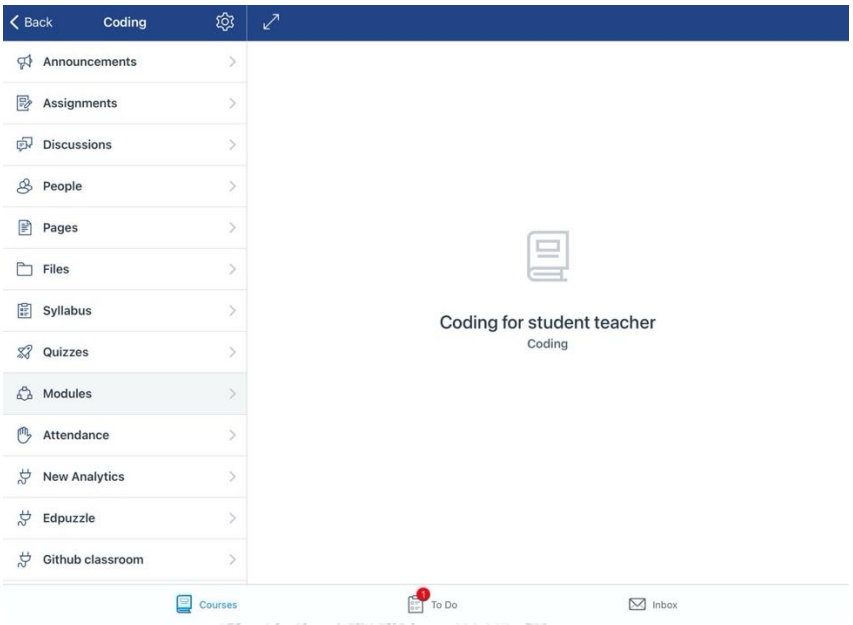
1.5) Posttest เป็นแบบทดสอบความสามารถด้านการเขียนโค้ดหลังเรียนที่กลุ่มทดลองทุกคนเมื่อเรียนรู้ครบทั้ง 3 โมดูลแล้วจะได้รับการทดลองหลังเรียนเพื่อประเมินความสามารถด้านการเขียนโค้ด โดยแบบทดสอบจะเป็นการประเมินความสามารถด้านการเขียนโค้ดในแต่ละองค์ประกอบตามเนื้อหาของทั้ง 3 โมดูลที่ได้เรียนครบแล้ว

การออกแบบและพัฒนาบทเรียนออนไลน์ด้านการเขียนโค้ดโดยใช้การเรียนรู้แบบจุลภาค สร้างขึ้นบทแพลตฟอร์มการเรียนรู้ออนไลน์ของ Canvas LMS (Canvas Learning management system) เนื่องจากสอดคล้องกับหลักการ 5 ด้านได้แก่ รูปแบบ จุดเด่น อิสระ โครงสร้าง และความสามารถในการระบุ โดยมีรายละเอียดดังต่อไปนี้



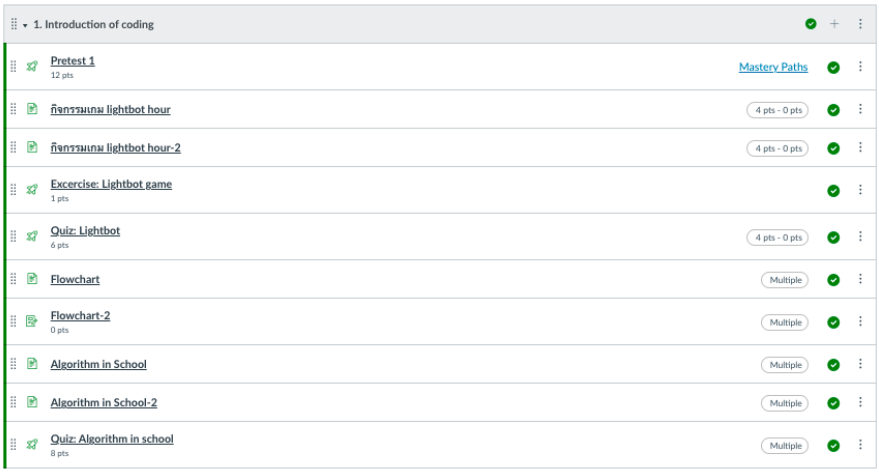
ภาพ 4.9 แพลตฟอร์มการเรียนรู้ออนไลน์ Canvas Learning Management System บนคอมพิวเตอร์

1) รูปแบบ Canvas LMS เป็นแพลตฟอร์มการเรียนรู้ออนไลน์ที่สามารถเข้าถึงได้จากทั้งคอมพิวเตอร์ แทบเล็ต และมือถือ เนื่องจากมีทั้งเว็บไซต์ และแอปพลิเคชัน จึงมีความสะดวกรวดเร็วในการเข้าถึง



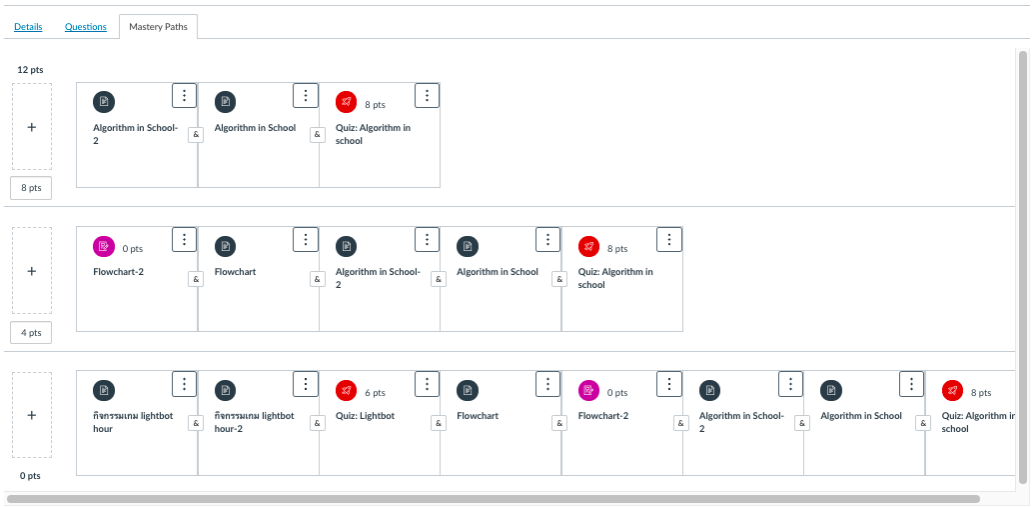
ภาพ 4.10 แพลตฟอร์มการเรียนรู้ออนไลน์ Canvas Learning Management System บนเท็บเล็ต

2) จุดเด่น สามารถสร้างบทเรียนแยกเป็นหัวข้อที่เฉพาะเจาะจงได้ตามต้องการ **3) อิสระ** ผู้เรียนสามารถเลือกจากบทเรียนใดก็ได้ตามที่ต้องการ **4) โครงสร้าง** แพลตฟอร์มสามารถออกแบบเป็นหัวข้อหลัก หัวข้อย่อยได้ตามต้องการพร้อมทั้งเพิ่มเติมรายละเอียดต่าง ๆ ได้ **5) ความสามารถในการระบุ** บทเรียนสามารถเข้าถึงได้จากลิงค์โดยตรง



ภาพ 4.11 บทเรียนย่อยในโมดูลการเรียนรู้จุลภาค เรื่อง การเขียนโค้ดเบื้องต้น

นอกจากนี้ยังสามารถกำหนดการให้บทเรียนกับผู้เรียนตามเงื่อนไขที่วางไว้ ในที่นี้ใช้เกณฑ์เป็นระดับของคะแนนก่อนเรียนจากสูงไปต่ำ ซึ่งทำงานโดยฟังก์ชัน mystery path ของแพลตฟอร์ม เพื่อให้สามารถสร้างการทดลองแบบปรับเหมาะได้



ภาพ 4.12 ฟังก์ชันการปรับเหมาะโดยใช้ Mystery Path บน Canvas LMS

2. บทเรียนออนไลน์ด้านการเขียนโค้ดโดยใช้การเรียนรู้แบบจุลภาค

การนำเสนอในส่วนนี้จะแบ่งออกเป็น 2 ส่วนคือ ผลของการออกแบบเนื้อหาจุลภาค และผลของการออกแบบกิจกรรมจุลภาค

2.1) ผลของการออกแบบเนื้อหาจุลภาค บทเรียนถูกออกแบบเนื้อหาจุลภาคขึ้นเพื่อให้สอดคล้องกับองค์ประกอบของความสามารถด้านการเขียนโค้ดได้แก่ ด้านการแก้ไขปัญหา และด้านโปรแกรม ซึ่งเนื้อหาที่ออกแบบขึ้นจะอิงบริบทที่เกี่ยวข้องกับตัวอย่างวิจัยโดยใช้บริบทของโรงเรียน นักเรียน และวัยรุ่น โดยเนื้อหาแบ่งออกเป็น 4 โมดูล ดังต่อไปนี้

โมดูลเริ่มต้น พื้นฐานเกี่ยวกับการเขียนโค้ด

พื้นฐานในการทำงานของโปรแกรม ประกอบด้วย 1) การรับข้อมูล (input) เช่น การกรอกชื่อเมนูอาหารในโปรแกรมสั่งอาหาร การกรอกชื่อสถานที่ในโปรแกรม google maps 2) การประมวลผล (process) เป็นขั้นตอนที่นำข้อมูลมาประมวลผลตามคำสั่งที่ระบุไว้ ซึ่งเรียกว่า การเขียนโค้ด 3) การแสดงผล (output) เป็นการแสดงผลลัพธ์ที่ได้จากประมวลผลของโปรแกรม ผ่านหน้าจอ

การประมวลผล (Process)

ข้อมูลที่ได้รับมาเพื่อที่จะนำไปประมวลผล ประเภทของข้อมูล ข้อมูลจำนวนเต็ม เช่น -3, 0, 12 ข้อมูลทศนิยม เช่น 3.32, -6.5 ข้อมูลบูลีน เป็นข้อมูลที่มีค่าเป็นจริงหรือเป็นเท็จ เช่น ผู้ใช้งาน

โปรแกรมเป็นนิสิตหรือไม่ ถ้าผู้ใช้งานเป็นนิสิตก็เป็นจริง ถ้าผู้ใช้งานทำอาชีพอื่นก็เป็นเท็จ ข้อมูลตัวอักษร เช่น A, B, C และข้อมูลสตริง เป็นการเรียงกันของตัวษรเช่น Education, Teacher

การเขียนคำสั่งให้คอมพิวเตอร์ทำงานตามที่ต้องการจะมีการกำหนดตัวแปรเพื่อที่จะเรียกใช้ข้อมูลต่าง ๆ ได้ง่ายขึ้น ตัวแปรต่าง ๆ สามารถนำมาดำเนินการกันทางคณิตศาสตร์เช่น การบวก ลบ คูณ หาร การดำเนินการทางตรรกศาสตร์ เช่น และ หรือ ถ้า...แล้ว... ก็ต่อเมื่อ การทำงานของคอมพิวเตอร์ไม่ได้ทำงานเป็นลำดับจากบนลงล่างเท่านั้น แต่ต้องมีการตัดสินใจด้วย เช่น โปรแกรมการตัดเกรด กำหนดไว้ว่า ถ้าคะแนนที่ได้มากกว่าหรือเท่ากับ 80 คะแนน จะได้เกรด 4 นอกจากนี้ยังมีการทำงานแบบวนซ้ำ เช่น กำหนดให้โปรแกรมทำงานวนซ้ำกี่รอบ หรือ กำหนดให้วนซ้ำจนกว่าเงื่อนไขที่กำหนดไว้จะเป็นจริง เช่น สั่งให้โปรแกรมตัดเกรดทำงานจนกว่าจะครบ 100 คน หรือทำจนกว่าจะครบทุกคน นอกจากการเขียนคำสั่งให้สั้นลง สามารถทำได้โดยการใช้ฟังก์ชัน ซึ่งเป็นการเรียกใช้คำสั่งเดิมที่เคยเขียนไว้แล้วเพื่อให้สะดวกมากขึ้น

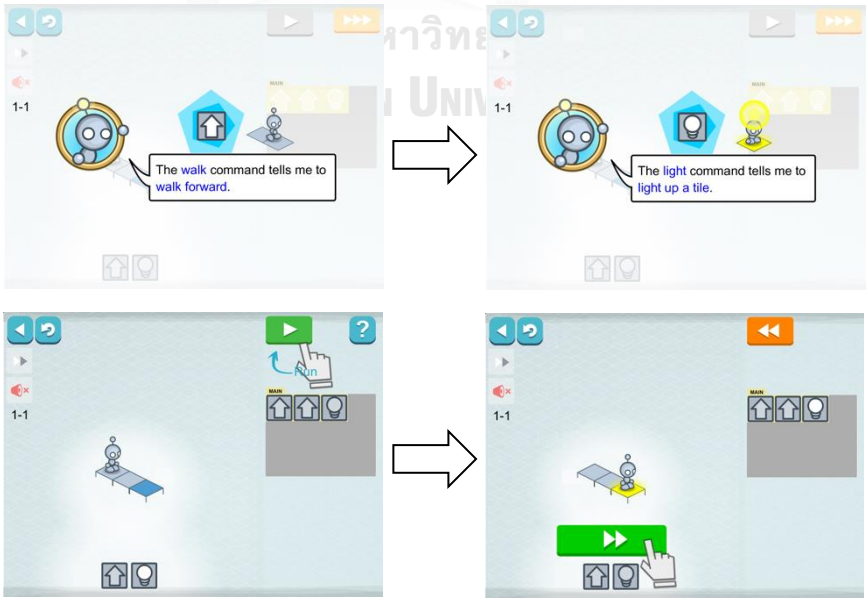
โมดูลที่ 1 การเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ (การแก้ปัญหา)

1) กิจกรรมเกม lightbot hour

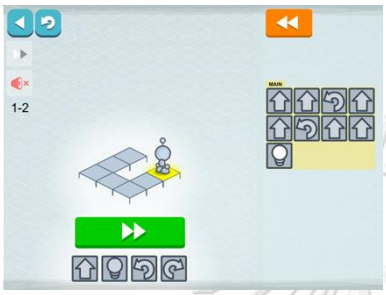
ขั้นตอนการทำงานเป็นลำดับทางเดียว ควบคุมหุ่นยนต์ให้เดินและเปิดไฟตามคำสั่ง โดยใช้เกม lightbot hour โดยมีเงื่อนไขคือ จำนวนครั้งที่จำกัด ในด่านที่ยากขึ้นจะมีการใช้คำสั่งซ้ำและการวนซ้ำเปรียบเสมือนการใช้ฟังก์ชันและการวนซ้ำในการเขียนโค้ด

1.1) เรียนรู้คำสั่งและตัวอย่างการใช้งาน

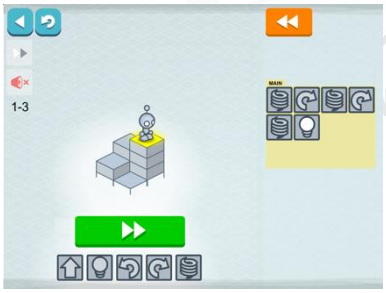
การเดินไปด้านหน้าและเปิดไฟ



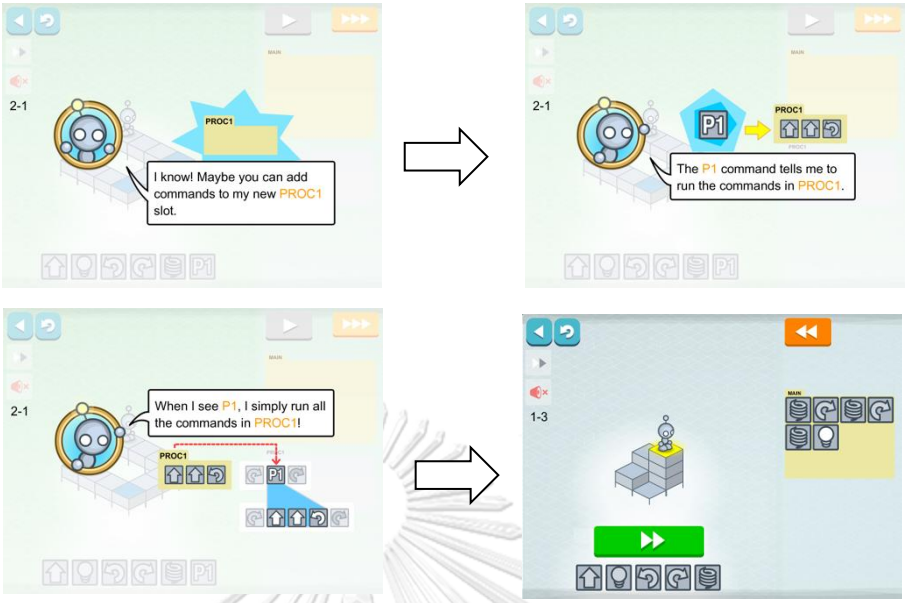
การเดินหันซ้ายและขวาของหุ่นยนต์



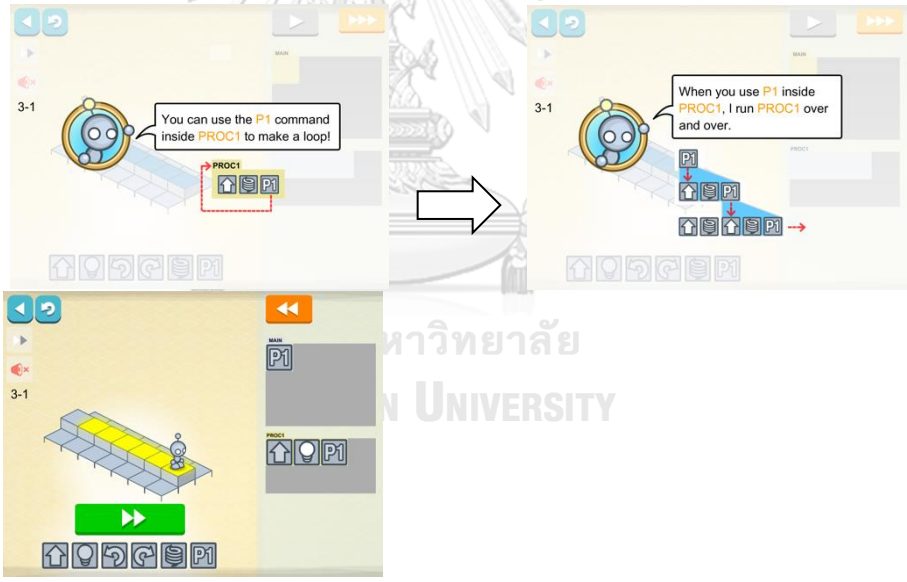
การกระโดด



1.2) เรียนรู้การใช้งาน P1 ที่สามารถเรียกใช้คำสั่งเดิมซ้ำได้



1.3) เรียนรู้การใช้คำสั่งแบบ



2) Algorithm in School

นิสิตต้องการจัดกลุ่มนักเรียนจำนวน 100 คน ให้สามารถเข้ากลุ่มกิจกรรมแบบเวียนฐาน โดยนักเรียนแต่ละคนเลือกลำดับของกิจกรรมฐานที่ต้องการจะเข้าร่วม 3 อันดับ จากทั้งหมด 8 ฐาน ในการเวียนฐานแต่ละครั้งรับจำนวนนักเรียนได้ครั้งละ 30 คน ในช่วงเวลาทำกิจกรรมนี้มีเวลาเพียงพอให้เวียนฐานได้เพียง 3 ฐานเท่านั้น จงอธิบายเป็นขั้นตอนอย่างละเอียดในการจัดกลุ่มนักเรียนให้เข้าร่วมกิจกรรมเวียนฐานอย่างเหมาะสมที่สุดตามความคิดของนิสิต โดยมีเงื่อนไขว่า มีนักเรียนที่ต้องการเข้าร่วมบางฐานมากกว่าจำนวนที่รับได้ต่อรอบ

1. จัดกลุ่มให้นักเรียนเข้าฐานโดยอิงจากลำดับที่เลือกมาเป็นลำดับ 1

2. ถ้าฐานที่มีนักเรียนเลือกมากกว่าจำนวนที่รับได้ต่อรอบแต่ไม่เกิน 3 เท่า ให้นักเรียนเหล่านั้นได้เข้าฐานนั้น ถ้าฐานที่มีนักเรียนเลือกมากกว่าจำนวนที่รับได้ต่อรอบมากกว่า 3 เท่า ให้สุ่มนักเรียนตามจำนวนที่เกินให้เข้าฐานที่เลือกเป็นลำดับที่ 2 แทน (นักเรียนเหล่านี้จะไม่ได้เข้าฐานที่เลือกลำดับที่ 1)

3. แสดงรายชื่อนักเรียนทุกคนตามฐานที่จัดกลุ่ม และไม่นำกิจกรรมฐานที่เต็ม 3 รอบแล้วไปจัดต่อ (ให้ถือว่าเป็นฐาน 1 ใน 3 ที่ได้เข้าร่วม)

4. ทำซ้ำขั้นที่ 1 โดยเปลี่ยนจากการอิงลำดับที่เลือกเป็นลำดับที่ 2
   ทำซ้ำขั้นที่ 2 โดยเปลี่ยนเงื่อนไขที่ 2 เป็นลำดับที่ 3
   ทำซ้ำขั้นที่ 3 ถือว่าได้ฐานที่ 2 ใน 3 ฐาน

5. ทำซ้ำขั้นที่ 1 โดยเปลี่ยนจากการอิงลำดับที่เลือกเป็นลำดับที่ 3
   ทำซ้ำขั้นที่ 2 โดยเปลี่ยนเงื่อนไขที่ 2 เป็นสุ่มเลือกเข้าฐานใดก็ได้ที่ยังไม่เต็ม
   ทำซ้ำขั้นที่ 3 ถือว่าได้ครบทั้ง 3 ฐาน

เมื่อนำเนื้อหาที่ได้ออกแบบไว้มากำหนดเป็นเนื้อหาจุลภาคเพื่อกำหนดเป็นบทเรียนย่อยในโมดูล จะได้เนื้อหาจุลภาคดังตารางต่อไปนี้

ตาราง 4.5 เนื้อหาจุลภาคของโมดูลที่ 1 การเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ (การแก้ปัญหา)

หัวข้อ	เนื้อหาจุลภาค
การเขียนคำสั่งที่เป็นลำดับ	ขั้นตอนการทำงานเป็นลำดับทางเดียว การควบคุมหุ่นยนต์ให้เดินและเปิดไฟตามคำสั่ง โดยใช้เกม lightbot hour
การเขียนคำสั่งที่ต้องแยกย่อยขั้นตอน	ขั้นตอนการทำงานอย่างเป็นขั้นตอน มีการกำหนดคำสั่งเป็นชุดย่อย การควบคุมหุ่นยนต์ให้เดินและเปิดไฟตามคำสั่ง โดยใช้เกม lightbot hour
การเขียนคำสั่งที่ต้องหาแบบรูปและการวนซ้ำ	ขั้นตอนการทำงานอย่างเป็นขั้นตอน มีการกำหนดชุดคำสั่งและวนซ้ำ การควบคุมหุ่นยนต์ให้เดินและเปิดไฟตามคำสั่ง โดยใช้เกม lightbot hour
การออกแบบขั้นตอนวิธี และการเขียนผังงาน	ขั้นตอนการทำงานโดยมีเงื่อนไขต้องตัดสินใจ มี1,2 และ 3 ทางเลือก
	การเขียนผังงาน
ประเภทของข้อมูล โครงสร้างข้อมูล ฟังก์ชัน	ข้อมูลประเภทตัวอักษร จำนวนเต็ม จำนวนจริง บูลีน โครงสร้างข้อมูลแบบเซต ลิสต์ และการจัดกลุ่มการทำงานซ้ำ ๆ เป็นฟังก์ชัน

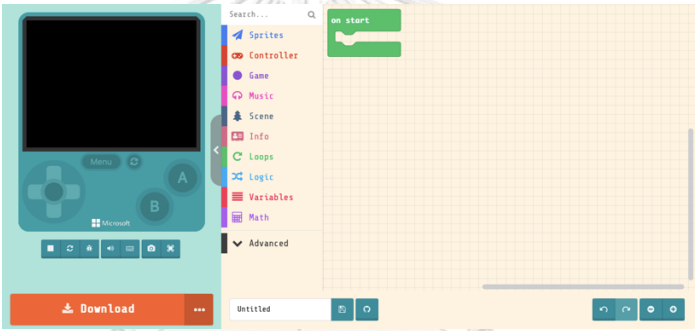
โมดูลที่ 2 ภาษา Scratch

2.1) Introduction of Scratch: Microsoft MakeCode Arcade

ภาษา Scratch เป็นภาษาคอมพิวเตอร์ประเภท visual programming language เป็นภาษาที่สามารถสร้างแอนิเมชันหรือเกมอย่างง่าย การสร้างคำสั่งเป็นการนำเอาบล็อกคำสั่งมาวางต่อ ๆ กัน ซึ่งช่วยให้ลดความผิดพลาดเรื่องการพิมพ์ การใช้สัญลักษณ์ต่าง ๆ สำหรับมือใหม่

Microsoft MakeCode Arcade เป็นโปรแกรมที่ใช้ในการสร้างเกมอย่างง่าย โดยใช้ภาษา Scratch ที่สร้างชุดคำสั่งโดยการลากบล็อกคำสั่งมาเรียงต่อกัน ในโปรแกรมจะมีเมนูบล็อก บล็อกคำสั่งที่เป็นการสร้างเกมโดยเฉพาะ จึงมีความแตกต่างกับภาษา Scratch ตัวเต็ม

หน้าจอของโปรแกรมประกอบด้วย 2 ส่วนคือ 1. หน้าจอเครื่องเล่นเกมที่มีปุ่มควบคุมทิศทางและปุ่มคำสั่ง 2 ปุ่ม 2. หน้าจอการเขียนคำสั่ง และเมนูคำสั่งที่จัดกลุ่มบล็อกคำสั่งที่มีหน้าที่คล้ายกันไว้ด้วยกัน

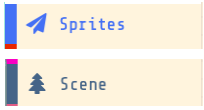


2.2) เมนูบล็อก

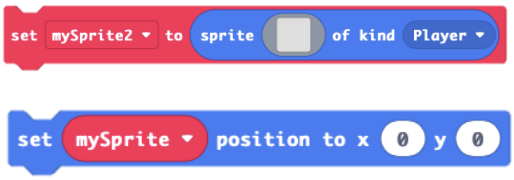
เมนูบล็อกประกอบด้วยบล็อก 3 ชนิด ได้แก่ 1) Sprite, Scene, Music (ออกแบบตัวละครหลัก พื้นหลัง และเสียง) 2) Game, Controller, Info, Variable และ Math (เมนูหลักในการออกแบบการเล่นเกม) 3) Logic และ Loops (คำสั่งที่ต้องตัดสินใจและเกิดการวนซ้ำ)

2.2.1) Sprite and Scene

1. เลือก เมนูบล็อค



2. เมนู Sprite



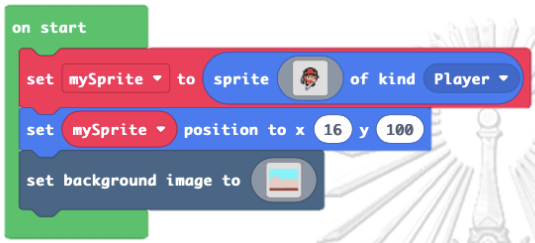
3. เมนู Scene

4.

บล็อกที่ 1 ออกแบบตัวละครได้ตามต้องการ หรือ เลือกจาก gallery

บล็อกที่ 2 เลือกตำแหน่งของตัวละคร

บล็อกที่ 3 ออกแบบพื้นหลัง

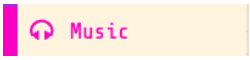


5.

Output

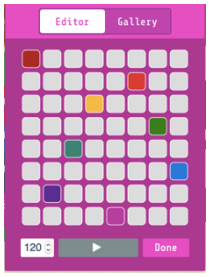
จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

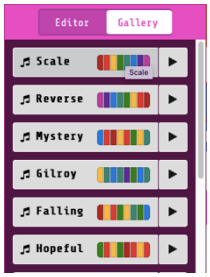
6. หากต้องการใส่เสียงในเกมสามารถเลือกใส่ได้จากเมนูบล็อก Music



7. เมนู melody

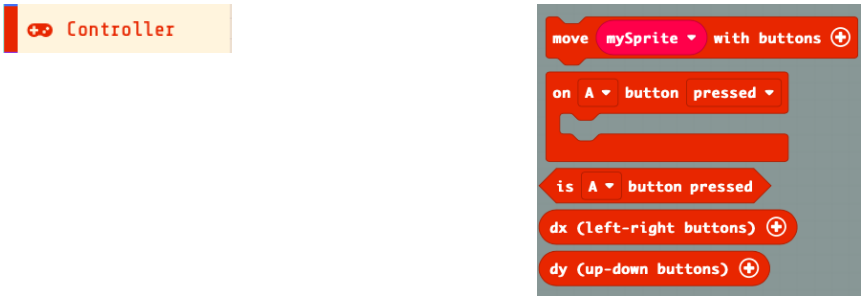
8. ออกแบบเสียงได้เองหรือเลือกได้จาก Gallery





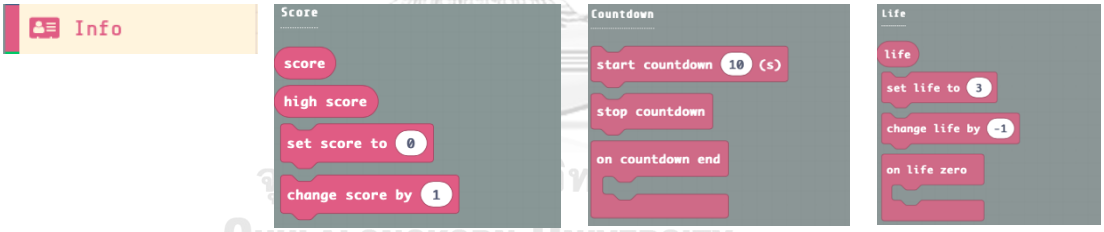
2.2.2) Game, Controller, Info, Variable และ Math

1. เมนูควบคุมตัวละครจากเครื่องเล่นเกม



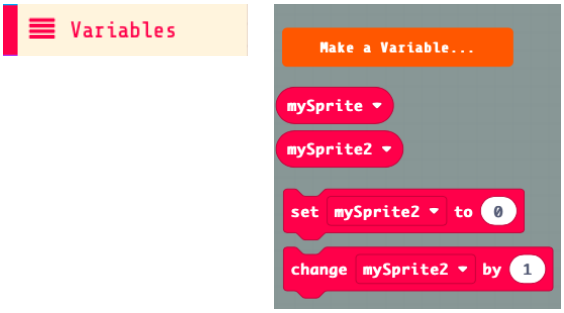
2. เมนูคำสั่งในการทำงานของเกม

3. เมนูเกี่ยวกับข้อมูลตัวละคร

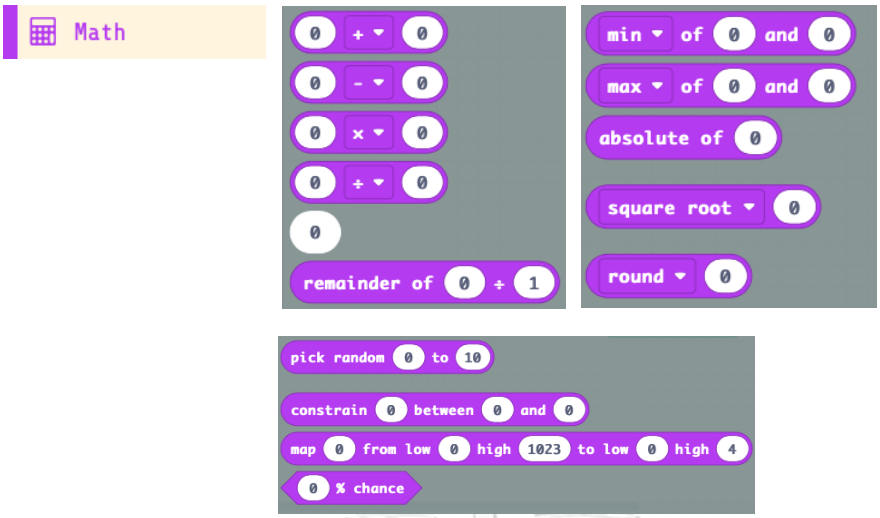


CHULALONGKORN UNIVERSITY

4. เมนูสร้างตัวแปรอื่น ๆ ขึ้นมาใหม่เอง

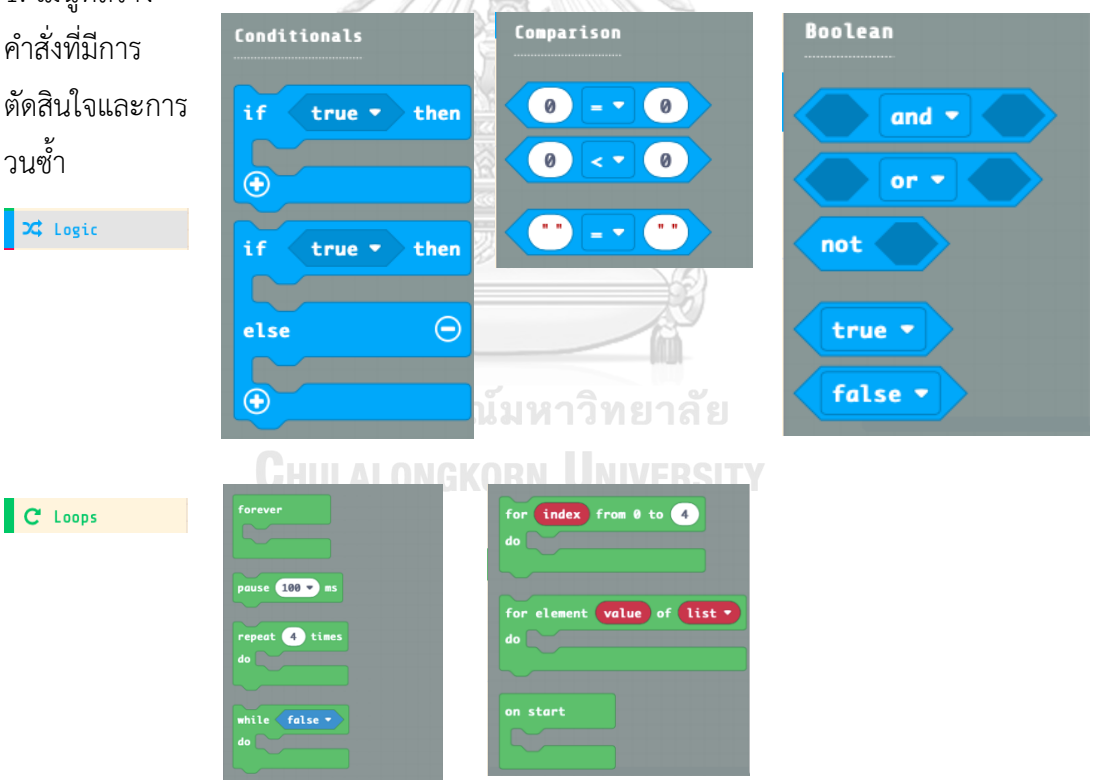


5. เมนูที่สร้างคำสั่งเกี่ยวกับการดำเนินการทางคณิตศาสตร์ต่าง ๆ



2.2.3) Logic และ Loops

1. เมนูที่สร้างคำสั่งที่มีการตัดสินใจและการวนซ้ำ

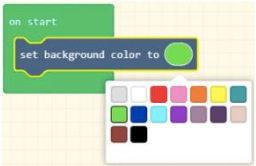


3) สร้างเกมอย่างง่าย: Chase the pizza

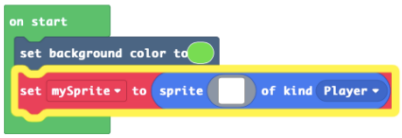
เกมที่ให้ตัวละครเคลื่อนที่ไปกิน pizza ที่เกิดขึ้นอย่างสุ่ม ให้ได้จำนวนครั้งมากที่สุดภายในเวลาที่กำหนด

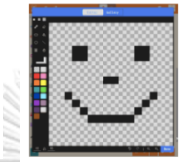
3.1) ลำดับการสร้าง

1. สร้างพื้นหลัง

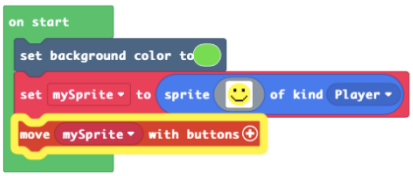


2. สร้างตัวละคร

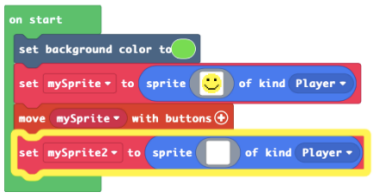


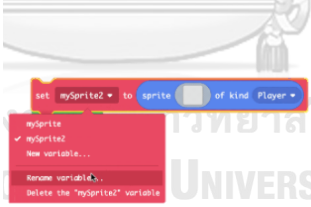


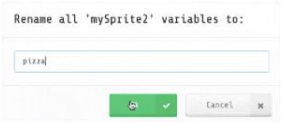
3. สร้างคำสั่งให้ตัวละครเคลื่อนที่โดยการใช้ปุ่มบนจอย

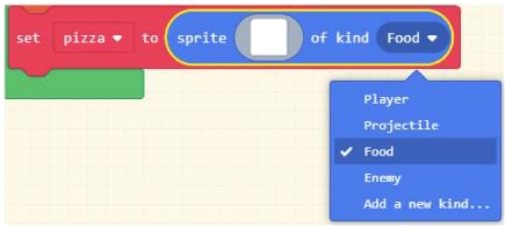


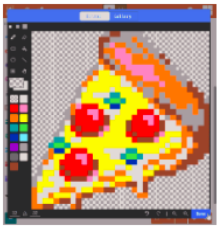
4. สร้างตัวละครที่ 2 (pizza)



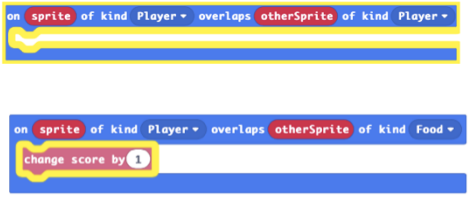




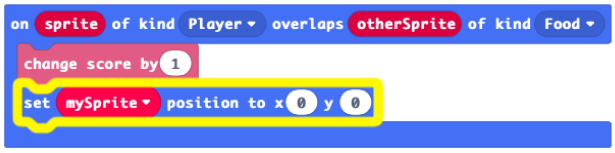




5. กำหนดเงื่อนไขของเกม โดยคะแนนจะเพิ่มขึ้น 1 เมื่อตัวละครเคลื่อนที่มาชนกับ pizza



6. กำหนดให้ pizza เกิดขึ้นอย่างสุ่มบนจอเครื่องเล่นเกม



7. กำหนดให้แต่ละครั้งมีเวลา 10 วินาทีในการบังคับให้ตัวละครไปชนกับ pizza หากหมดเวลาเกมจะจบลง

CHULALONGKORN UNIVERSITY

เมื่อนำเนื้อหาที่ได้ออกแบบไว้มากำหนดเป็นเนื้อหาจุลภาคเพื่อกำหนดเป็นบทเรียนย่อยในโมดูล จะได้เนื้อหาจุลภาคดังตารางต่อไปนี้

ตาราง 4.6 เนื้อหาจุลภาคของโมดูลที่ 2 ภาษา Scratch บน Microsoft MakeCode Arcade

หัวข้อ	เนื้อหาจุลภาค
การสร้างตัวละครและพื้นหลัง	การตกแต่งตัวละคร พื้นหลัง เสียง
การเคลื่อนที่	การเคลื่อนที่ของตัวละคร, การบังคับตัวละครด้วยปุ่ม
การแสดงข้อมูลบนจอ	คะแนนของเกม ค่าพลังชีวิต การจับเวลา
การจัดการลำดับของโปรแกรม	คำสั่งเงื่อนไข : เมื่อ.....ให้ทำ.....คำสั่งการทำงานพื้นฐานในการสร้างเกม : การปล่อยพลัง การสร้างด่าน การเก็บไอเทม เป็นต้น
ตรรกศาสตร์และการตัดสินใจ	คำสั่ง if : กำหนดเงื่อนไขการแพ้ หรือ ชนะในเกม, และการให้แสดงผลที่แตกต่างกันเมื่อมีเงื่อนไข
	ตัวแปร Boolean
Project เกมอย่างง่าย	การสร้างเกมอย่างง่ายโดยใช้คำสั่งพื้นฐาน

โมดูลที่ 3 ภาษา Python

3.1) พื้นฐานภาษา Python

ภาษา Python เป็นภาษาคอมพิวเตอร์ที่นิยมมากในปัจจุบัน เนื่องจาก 1) เป็นภาษาที่เขียนง่าย มีโครงสร้าง สัญลักษณ์ ตัวเชื่อมที่ไม่มากเป็น ภาษาที่ไม่ซับซ้อนเหมาะสำหรับมือใหม่ 2) ฟรีและเป็น open source ที่สามารถปรับปรุง แก้ไข และนำไปใช้ได้อย่างอิสระ 3) มี libraries จำนวนมากที่ใช้กับงานที่เฉพาะเจาะจง จึงสามารถเขียนโปรแกรมได้ง่ายและรวดเร็ว 4) สามารถทำงานบนแพลตฟอร์มได้หลากหลาย เช่น macOS, Windows และ PlayStation

3.2) ประเภทของข้อมูล

1) จำนวนเต็ม (Integer) เช่น -3,-2,-1,0,1,2,3 2)ทศนิยม (Float) เช่น -1.25, 0, 1.25, 1.5e5 (หมายถึง 1.5×10^5) 3) ข้อความ (String) เช่น ชื่อนักเรียน “สุธี” ชื่อวิชา “คณิตศาสตร์” 4) บูลีน (Boolean) เช่น “สุธีได้เกรดเฉลี่ยมากกว่า 3.5” ถ้าเงื่อนไขนี้เป็นจริง ข้อมูลนี้จะมีค่าเป็น “จริง (True)” ถ้าเงื่อนไขนี้ไม่เป็นจริง ข้อมูลนี้จะมีค่าเป็น “เท็จ (False)”

3.2.1) ตัวแปร เป็นเครื่องมือพื้นฐานสำหรับเขียนโปรแกรม โดยเปรียบเสมือนป้ายที่ติดไว้กับข้อมูลเพื่อบอกว่าข้อมูลนั้นเรียกว่าอะไรเพื่อให้สามารถเรียกมาใช้งานได้ง่ายขึ้น เช่น

student = 30	(student คือตัวแปร, 30 คือข้อมูลประเภทจำนวนเต็ม)
print(student)	(คำสั่งให้แสดงข้อมูลที่มีชื่อว่า student)
30	(ข้อมูลที่ชื่อว่า student)

3.2.2) การดำเนินการ

การดำเนินการประกอบด้วย 3 แบบได้แก่ 1) การดำเนินการทางคณิตศาสตร์ ยกตัวอย่างเช่น บวก(+) ลบ(-) คูณ(*) หาร(/) ยกกำลัง(**) หารปัดเศษ(//) เศษจากการหาร(%) 2) การดำเนินการของข้อความ ยกตัวอย่างเช่น ข้อความบวกกัน หมายถึง ข้อความต่อกัน เช่น “วิชา”+“ คอมพิวเตอร์” = “วิชา คอมพิวเตอร์” ข้อความคูณด้วยจำนวนเต็ม หมายถึง การแสดงข้อความซ้ำ เช่น ‘สวัสดี ’*3 = ‘สวัสดี สวัสดี สวัสดี ’ และ 3) การดำเนินการทางตรรกศาสตร์ ยกตัวอย่างเช่น เครื่องหมายเท่ากับ (==) ไม่เท่ากับ (==!) มากกว่า (>) มากกว่าหรือเท่ากับ (>=) น้อยกว่า (<) น้อยกว่าหรือเท่ากับ (<=)

คำเชื่อม ประกอบด้วย และ (and) หรือ (or) นิเสธ (not) โดยมีตัวอย่างโค้ดดังนี้

```
>>> (student >30) and (teacher<10)
True            (เงื่อนไขเป็นจริงเพราะ ทั้ง 2 ส่วนเป็นจริงทั้งคู่)
>>> (student >30) or (teacher<10)
True            (เงื่อนไขเป็นจริงเพราะ มีอย่างน้อย 1 ส่วนที่จริง)
>>> not(student==30)
False           (เงื่อนไขเป็นเท็จเพราะ student เท่ากับ 30 เป็นจริง)
```

3.3) การกำหนดเงื่อนไข

การเขียนโค้ดแบบกำหนดเงื่อนไข ประกอบด้วยเงื่อนไขทั้งหมด 3 แบบดังต่อไปนี้ 1) เงื่อนไขแบบ if 2) การวนซ้ำ และ 3) การวนซ้ำจนกว่าเงื่อนไขจะเป็นจริง (ไม่ทราบว่าต้องวนซ้ำกี่ครั้ง)

ตัวอย่างการเขียนโค้ด เช่น

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

คำสั่ง if แบบ 1 เงื่อนไข

```
if score >= 5:
    print(‘You’re passed’)
```

คำสั่ง if แบบ 2 เงื่อนไข เช่น

```
if score >= 5:
    print(‘You’re passed’)
else:
    print(‘You’re failed’)
```

คำสั่ง if แบบ มากกว่า 2 เงื่อนไข

```
score = 7
if score >= 8:
        print('Excellent')
elif score >=5
        print('Good')
else:
        print('Try again')
```

การวนซ้ำตามจำนวนครั้งที่กำหนด

```
>>>     for i in range(1,3):            (วนซ้ำ 2 ครั้ง)
                print(i)
>>>     room_1={"Alex", "Adam", "Bay"}
        for student in room_1:          (วนซ้ำ 3 ครั้ง ตามรายชื่อใน room_1)
                print(student)
```

การวนซ้ำจนกว่าเงื่อนไขจะเป็นจริง (ไม่ทราบว่าต้องวนซ้ำกี่ครั้ง)

```
>>>     number = 1
        while number >10:
                number = number + 1
                return(number)
```

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

3.3) ฟังก์ชัน

การสร้างฟังก์ชัน

```
        def power(a,x):         ชื่อฟังก์ชัน = power, พารามิเตอร์ = a,x
                y=a**x
                return y        การคืนค่า
```

ตาราง 4.7 ตัวอย่างฟังก์ชัน (built-in function)

x="learn Python"	**Output**
print(x.upper())	LEARN PYTHON
print(x.lower())	learn python
print(x.split(" "))	['learn', 'Python']
print(x.count("P"))	1
print(len(x))	12

ตาราง 4.8 ตัวอย่างฟังก์ชันอื่น ๆ

Syntax	**Output**
random โมดูลที่ใช้ในการสุ่ม	
random.randint(0, 100)	5
random.random()	0.28650
random.choice('abcdefghij')	'c'
random.choice([11, 22, 33, 55, 44])	11
random.choice(['Boy', 'Cat', 'Ton', 'Pop', 'You'])	'Ton'
items = [1, 2, 3, 4, 5, 6, 7] random.shuffle(items)	[7, 3, 2, 5, 6, 4, 1]
turtle ใมดูลที่สร้างหุ่นยนต์เต่าที่สามารถเคลื่อนที่ได้	
turtle.forward()	ลากเส้นตรงไปข้างหน้ามีหน่วยเป็นพิกเซล
turtle.left()	หันไปทางขวา ทำมุมตามองศาที่กำหนดจากมุมเดิม
turtle.right()	หันไปทางซ้าย ทำมุมตามองศาที่กำหนดจากมุมเดิม
turtle.color()	เป็นคำสั่งกำหนดสีให้กับเส้น (ชื่อสีในภาษาอังกฤษ)
turtle.mainloop()	เป็นคำสั่งลูปค้างหน้าจอไว้สำหรับรอวาดรูปต่อ

เมื่อนำเนื้อหาที่ได้ออกแบบไว้มากำหนดเป็นเนื้อหาจุลภาคเพื่อกำหนดเป็นบทเรียนย่อยในโมดูล จะได้เนื้อหาจุลภาคดังตารางต่อไปนี้

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

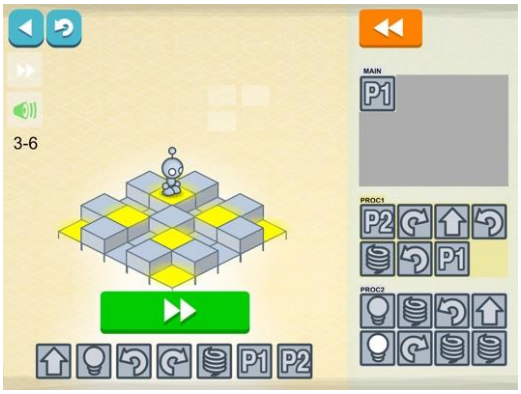
ตาราง 4.9 เนื้อหาจุลภาคของโมดูลที่ 3 ภาษา Python

หัวข้อ	เนื้อหาจุลภาค
ประเภทของตัวแปร	Integer, Float, String, Tuple, List, Dictionary, Boolean การประกาศตัวแปร การเปลี่ยนค่าตัวแปร(คำสั่ง เช่น +=, *=) คำสั่ง type() คำสั่งเปลี่ยนประเภทของตัวแปรเช่น int() float() str()
Number (integer, float)	บวก (+) ลบ (-) คูณ (*) ยกกำลัง (**) หาร (/) ปัดเศษ (//) เศษจากการหาร (%) ฟังก์ชันทางคณิตศาสตร์ (import math)
Condition Statement	if, elif, else statements
Loops	for loops, while loops
Function and Libraries	การกำหนดฟังก์ชัน, built-in module (Random) built-in module (Turtle)

2.2) ผลของการออกแบบกิจกรรมจุลภาค ในแต่ละโมดูลจะมีลักษณะของกิจกรรมการเรียนรู้ และสื่อการเรียนรู้ที่แตกต่างกัน เพื่อให้เหมาะสมกับเนื้อหาจุลภาคที่มีลักษณะและความยากง่ายแตกต่างกัน โดยแต่ละโมดูลจะประกอบไปด้วยกิจกรรมจุลภาคดังต่อไปนี้

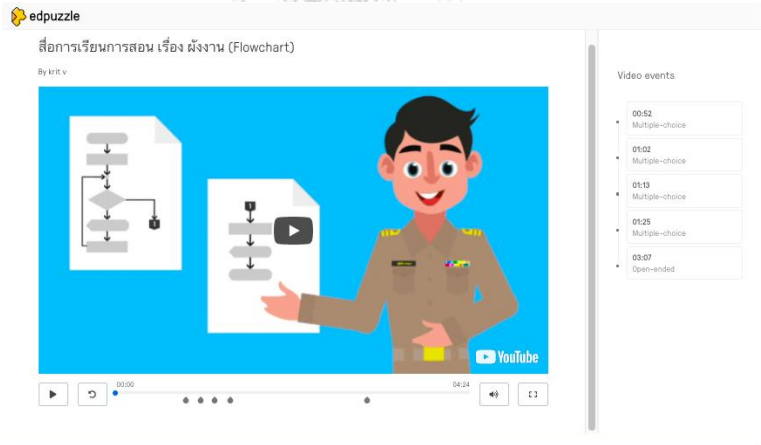
โมดูลที่ 1 การเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ (การแก้ปัญหา)

1. บทเรียนขั้นตอนวิธี ประกอบด้วย 1) ข้อความบรรยาย 2) เกมบนแท็บเล็ตและมือถือ และ 3) วิดีโอ ตัวอย่างของเกมที่ใช้เป็นสื่อการเรียนรู้คือ เกม lightbot hour เป็นเกมที่ใช้ฝึกฝนการออกแบบขั้นตอนวิธีในการทำงานผ่านการเล่นเกมโดยให้ระบุขั้นตอนเป็นลำดับในการเดิน กำหนดทิศทาง หรือกระโดด เพื่อให้สามารถเปิดไฟในตำแหน่งที่ระบุให้ครบ โดยระดับของด่านจะยากขึ้นเรื่อย ๆ เริ่มจากพื้นฐาน การวนซ้ำ การเรียกใช้คำสั่ง



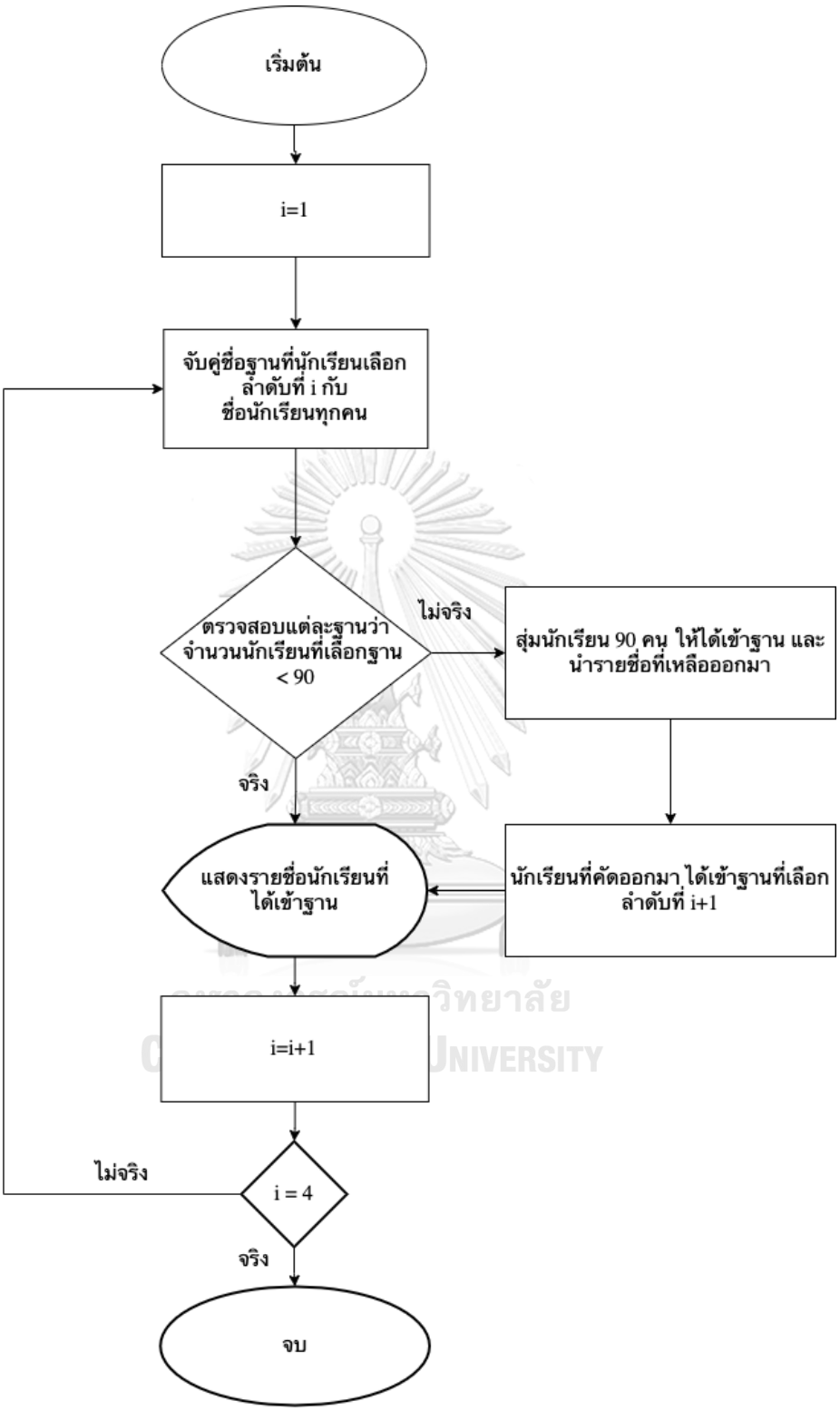
ภาพ 4.13 กิจกรรมเกม lightbot hour

2. บทเรียนผังงาน ประกอบด้วย 1) ข้อความบรรยาย 2) interactive video 3) interactive slide ตัวอย่างของ interactive video เป็นวิดีโอที่สร้างขึ้นบทแพลตฟอร์ม edpuzzle โดยคัดเลือกวิดีโอที่เกี่ยวข้องกับบทเรียนจากแหล่งการเรียนรู้ออนไลน์เพื่อนำมาดัดแปลงโดยการแทรกคำถามเข้าไปในแต่ละช่วงของวิดีโอเพื่อให้กลุ่มทดลองมีปฏิสัมพันธ์กับบทเรียนโดยการตอบคำถามแบบเลือกตอบ และการตอบสั้น



ภาพ 4.14 วิดีโอเชิงโต้ตอบ เรื่อง ผังงาน

3. บทเรียนขั้นตอนวิธี ประกอบด้วย 1) ข้อความบรรยาย 2) interactive slide 3) เว็บไซต์สร้าง Flowchart (https://www.draw.io) ตัวอย่างของผังงานที่สร้างขึ้นบทเว็บไซต์ www.draw.io เป็นเว็บไซต์ที่สามารถมีเครื่องมือเพื่อให้สามารถสร้างผังงานได้อย่างสะดวกและรวดเร็ว

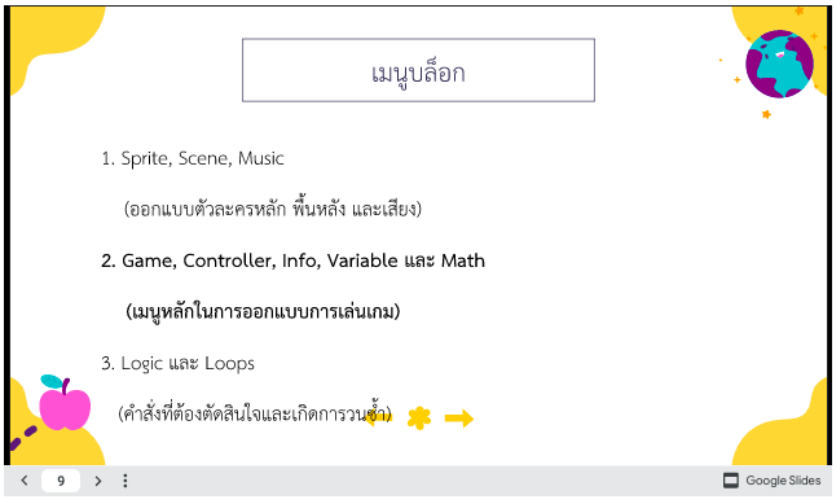


ภาพ 4.15 ผังงาน

โมดูลที่ 2 ภาษา Scratch บน Microsoft MakeCode Arcade

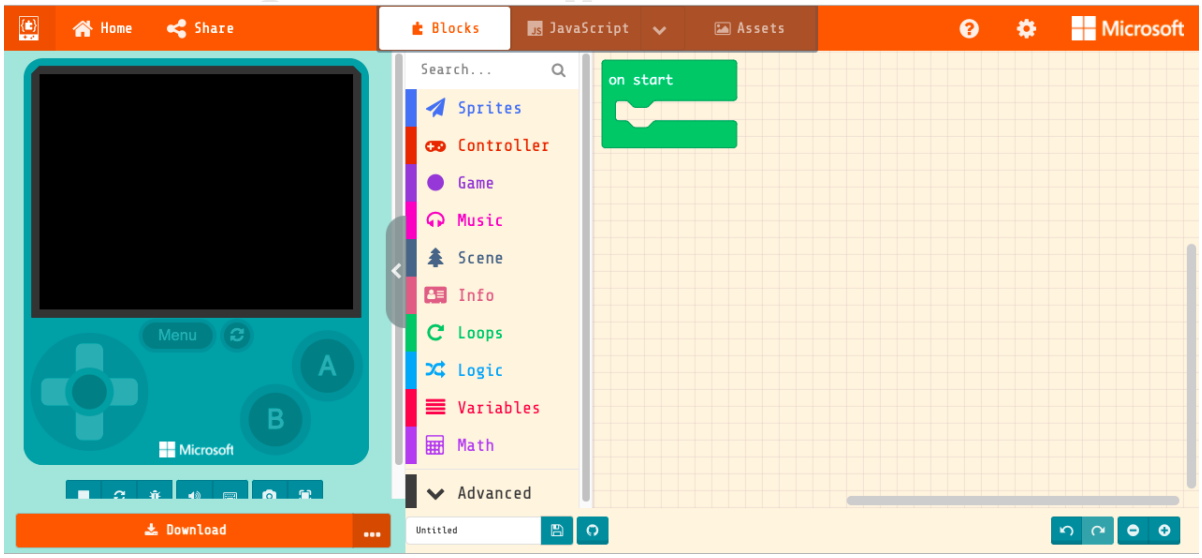
บทเรียนทั้งหมดได้แก่ พื้นฐานของ Scratch เมนูบล็อก และเกม ประกอบด้วย ข้อความบรรยาย interactive slide และ เว็บไซต์ Microsoft MakeCode Arcade

1. เอกสารเชิงโต้ตอบสร้างขึ้นบน google slide เป็นเอกสารที่ผู้เรียนสามารถมีปฏิสัมพันธ์กับเอกสารได้จาก transition และ animation ของเอกสาร ทำให้ผู้เรียนสามารถทราบได้ถึงจุดเน้นทำความเข้าใจได้เพิ่มขึ้นจากการเรียนซ้ำ ๆ



ภาพ 4.16 เอกสารเชิงโต้ตอบ เรื่อง ภาษา Scratch

2. เว็บไซต์ Microsoft Makecode Arcade เป็นเว็บไซต์ที่สามารถใช้เป็นแหล่งการเรียนรู้ออนไลน์ที่ฝึกฝน และทำความเข้าใจในการเขียนโค้ดด้วยภาษาบล็อก เว็บไซต์มีหน้าต่างแสดงผลเป็นหน้าจอเกม arcade และบล็อกคำสั่งจะเกี่ยวข้องกับการสร้างเกม arcade ซึ่งการเรียนรู้ผ่านเกมเป็นส่วนหนึ่งของบริบทที่เหมาะสมกับตัวอย่างวิจัยที่เป็นนักศึกษาในวัยรุ่น



ภาพ 4.17 เว็บไซต์ ww.arcade.makecode.com

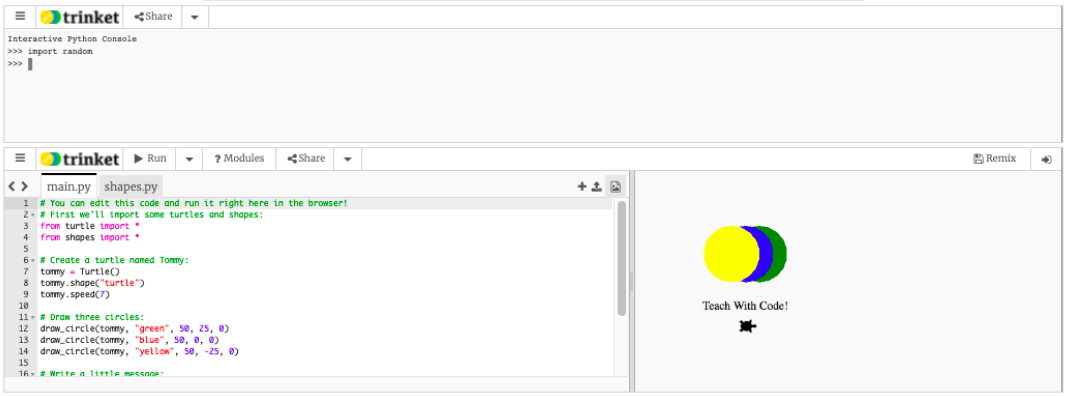
โมดูลที่ 3 ภาษา Python

บทเรียนทั้งหมดได้แก่ 1) พื้นฐานของภาษา Python 2) ประเภทข้อมูล ตัวแปร และการดำเนินการ 3) การวางเงื่อนไข และ 4) ฟังก์ชันและไลบรารี ประกอบด้วย ข้อความบรรยาย interactive slide และ interactive Python Console

หน้าต่างการเขียนโค้ดภาษา Python เชิงโต้ตอบ จะมี 2 ลักษณะ ดังนี้

1. หน้าต่างการเขียนโค้ดภาษา Python ที่มีแค่ console เป็นหน้าต่างที่ให้ผู้เรียนสามารถทดลองเขียนโค้ด ตรวจสอบโค้ดที่เรียนว่าให้ผลลัพธ์อย่าง

2. หน้าต่างการเขียนโค้ดภาษา Python ที่มี หน้าโค้ด และหน้าแสดงแผนภาพ ซึ่งใช้เพื่อให้ผู้เรียนได้เรียนรู้การเขียนโค้ดที่ซับซ้อนขึ้น โดยสามารถกำหนดสิ่งที่อยากให้เรียนไว้ก่อนหรือโค้ดที่แสดงไว้ เพื่อให้ผู้เรียนได้เห็นผลลัพธ์ได้เลย โดยไม่ต้องเขียนโค้ดเองใหม่ทั้งหมด อีกทั้งยังสามารถแก้ไขเพิ่มเติม โค้ดเพื่อทดสอบความเข้าใจ และฝึกฝนได้อีกด้วย



ภาพ 4.18 หน้าต่างการเขียนโค้ดภาษา Python เชิงโต้ตอบ

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ตอนที่ 3 การเปรียบเทียบโปรไฟล์และความสามารถด้านการเขียนโค้ดระหว่างได้รับบทเรียนออนไลน์ด้านการเขียนโค้ดที่ออกแบบตามการเรียนรู้แบบจุลภาคที่มีรูปแบบแตกต่างกันของนักศึกษาครูที่มีภูมิหลังแตกต่างกัน

ตัวอย่างวิจัยที่ได้รับการทดลองแบบปรับเหมาะจำนวน 31 คน เป็นนิสิตคณะครุศาสตร์ที่ได้มาด้วยการเลือกตามสะดวก ซึ่งมีโปรไฟล์ที่ประกอบด้วยตัวแปรภูมิหลังและความสามารถด้านการเขียนโค้ดในแต่ละองค์ประกอบก่อนเรียนแตกต่างกัน ทำให้ได้รับบทเรียนที่แตกต่างกัน

ตัวอย่างวิจัยเป็นผู้ชาย 18 คน (ร้อยละ 58.1) ชั้นปีที่ 4 มากที่สุดจำนวน 12 คน (ร้อยละ 38.7) วิชาเอกคณิตศาสตร์มากที่สุดจำนวน 14 คน (ร้อยละ 48.3) สาขามัธยมศึกษาวิทยาศาสตร์มากที่สุดจำนวน 20 คน (ร้อยละ 64.5) ไม่เคยเรียนเขียนโค้ดจำนวน 9 คน (ร้อยละ 29) ภาษาที่เคยเรียน

มากที่สุดคือ ภาษา C จำนวน 7 คน (ร้อยละ 22.6) รองลงมาคือภาษา Python จำนวน 5 คน (ร้อยละ 16.1) ความพึงพอใจต่อการเรียนรู้แบบจุลภาคส่วนมากมีความพึงพอใจต่อการศึกษาเรื่องต่าง ๆ ด้วยตนเอง ระยะเวลาในการเรียนครั้งละ 10-15 นาที เรียนเนื้อหาตวามที่มีการจัดไว้แล้ว และเรียนรู้จากภาพเคลื่อนไหว วิดีโอ รายละเอียดดังตารางต่อไปนี้

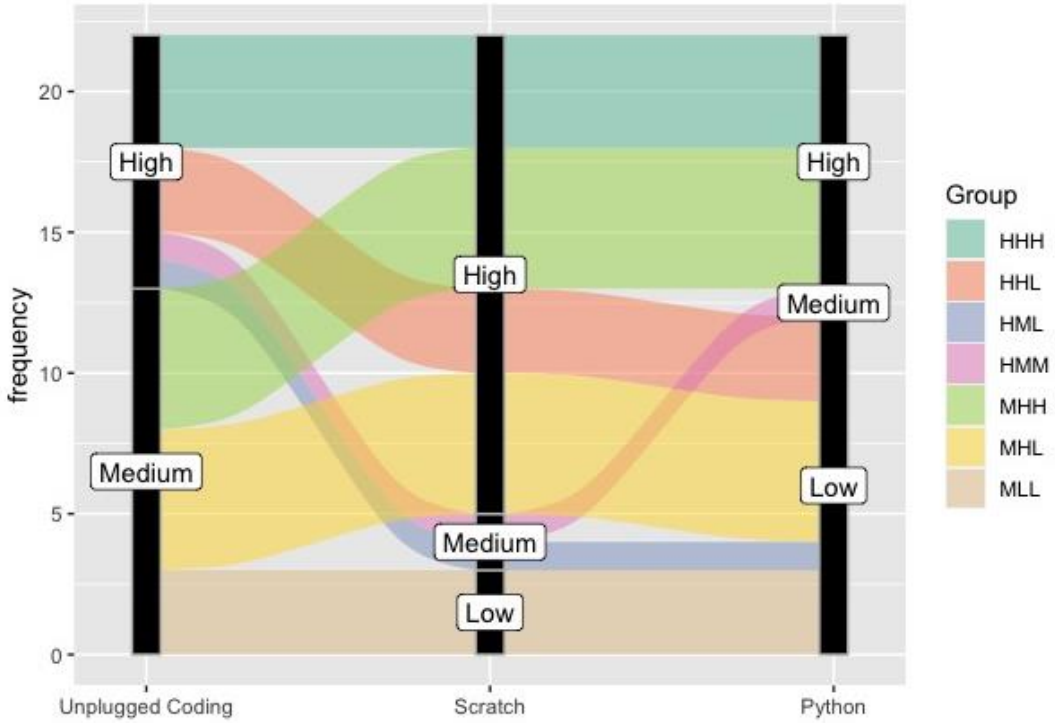
ตาราง 4.10 ความถี่ ร้อยละ ของตัวอย่างวิจัยจำแนกตามเพศ ชั้นปี สาขาวิชา และวิชาเอก

	เพศ					
	ชาย		หญิง		รวม	
	n	%	*n*	%	*n*	%
ชั้นปี						
1	5	27.78	3	23.08	8	25.81
2	4	22.22	2	15.38	6	19.35
3	3	16.67	2	15.38	5	16.13
4	6	33.33	6	46.15	12	38.71
สาขาวิชา						
มัธยมศึกษา (วิทย์)	12	66.67	8	61.54	20	64.52
เทคโนโลยีการศึกษา	4	22.22	4	30.77	8	25.81
อื่น ๆ	2	11.11	1	7.70	3	9.67
วิชาเอก						
คณิตศาสตร์	6	33.33	8	61.54	14	45.16
คอมพิวเตอร์	3	16.67	2	15.38	5	16.14
วิทยาศาสตร์	6	33.33	0	0.00	6	19.35
ภาษา	1	5.56	1	7.70	2	6.45
อื่น ๆ	2	11.11	2	15.38	4	12.90
รวม	18	100.00	13	100.00	31	100.00

หลังจากการตรวจสอบข้อมูลพบว่ามีตัวอย่างที่ให้ข้อมูลครบถ้วนจำนวน 22 คน พื้นฐานความรู้จากการประเมินก่อนได้รับบทเรียนในแต่ละโมดูลจำแนกออกได้เป็น 7 กลุ่มที่แตกต่างกัน ทำให้แต่ละกลุ่มได้รับบทเรียนตามเส้นทางการเรียนรู้แตกต่างกัน โดย โมดูลที่ 1 เกี่ยวข้องกับการ

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

เขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ โมดูลที่ 2 ภาษา Scratch และ โมดูลที่ 3 ภาษา Python ซึ่งสามารถจำแนกเป็นเส้นทางของแต่ละกลุ่มได้ดังแผนภาพต่อไปนี้



ภาพ 4.19 เส้นทางการเรียนบทเรียนออนไลน์ด้านการเขียนโค้ด

ความสามารถด้านการเขียนโค้ดระหว่างกลุ่มที่ได้รับบทเรียนการเรียนรู้ที่แตกต่างกัน ได้รับการประเมินจากบทเรียนย่อย ๆ ในแต่ละโมดูลหลังได้รับบทเรียน กำหนดให้ L หมายถึงกลุ่มต่ำ M หมายถึงกลุ่มกลาง และ H หมายถึงกลุ่มสูง เรียงลำดับตามโมดูล 1 – 3 เรียงจากซ้ายไปขวาตามลำดับ

กลุ่ม HHH หมายถึง กลุ่มที่มีพื้นฐานความรู้ในทุก ๆ ด้าน ในระดับสูง

กลุ่ม HMM หมายถึง กลุ่มที่มีพื้นฐานความรู้ด้านการเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ในระดับสูง และมีความรู้เรื่อง Scratch กับ Python ในระดับปานกลาง

กลุ่ม MHH หมายถึง กลุ่มที่มีพื้นฐานความรู้ด้านการเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ในระดับปานกลาง และมีความรู้เรื่อง Scratch กับ Python ในระดับสูง

กลุ่ม HHL หมายถึง กลุ่มที่มีพื้นฐานความรู้ด้านการเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์และภาษา Scratch ในระดับสูง และมีความรู้เรื่อง Python ในระดับต่ำ

กลุ่ม HML หมายถึง กลุ่มที่มีพื้นฐานฐานความรู้ด้านการเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ในระดับสูง Scratch ระดับปานกลาง และ Python ระดับต่ำ

กลุ่ม MHL หมายถึง กลุ่มที่มีพื้นฐานฐานความรู้ด้านการเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์

ในระดับปานกลาง Scratch ระดับสูง และ Python ระดับต่ำ

กลุ่ม MLL หมายถึง กลุ่มที่มีพื้นฐานความรู้ด้านการเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ใน ระดับปานกลาง และมีความรู้เรื่อง Scratch กับ Python ในระดับต่ำ

เมื่อพิจารณาทั้ง 7 กลุ่มจำแนกตามระดับความสามารถของแต่ละโมดูลพบว่า โมดูล 1 การเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ มี 3 กลุ่มที่อยู่กลุ่มกลาง และ 4 กลุ่มที่อยู่กลุ่มสูง ดังนั้น ไม่มีกลุ่มที่ไม่มีความรู้พื้นฐานเกี่ยวกับการเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ โมดูล 2 ภาษา Scratch มี 1 กลุ่มที่อยู่กลุ่มต่ำ 2 กลุ่มที่อยู่กลุ่มกลาง และ 4 กลุ่มที่อยู่กลุ่มสูง ดังนั้น ส่วนใหญ่มีความรู้พื้นฐานเกี่ยวกับภาษา Scratch และ โมดูล 3 ภาษา Python มี 4 กลุ่มที่อยู่กลุ่มต่ำ 1 กลุ่มที่อยู่กลาง และ 2 กลุ่มที่อยู่กลุ่มสูง ดังนั้น ส่วนใหญ่ไม่มีความรู้พื้นฐานเกี่ยวกับภาษา Python และหากจำแนกกลุ่มตามระดับความสามารถจะพบว่า 3 กลุ่มแรกได้แก่ กลุ่ม HHH, HMM และ MHH เป็นกลุ่มที่ได้คะแนนก่อนเรียนปานกลางถึงสูง และกลุ่มที่เหลือคือกลุ่ม HHL, HML, MHL และ MLL เป็นกลุ่มที่ได้คะแนนก่อนเรียนต่ำอย่างน้อย 1 โมดูล

ในแต่ละกลุ่มเมื่อจำแนกตามตัวแปรภูมิหลัง ได้แก่ เพศ ชั้นปี สาขาวิชา วิชาเอก และประสบการณ์ด้านการเขียนโค้ด พบว่า เพศชายส่วนใหญ่อยู่กลุ่ม MHH เพศหญิงส่วนใหญ่อยู่กลุ่ม MHL ชั้นปีที่ 1 ส่วนใหญ่อยู่กลุ่ม HHH ชั้นปีที่ 4 ส่วนใหญ่อยู่กลุ่ม MHL แล้วชั้นปีที่ 2 และ 3 มีการกระจายตัวอยู่ในหลากหลายกลุ่ม สาขาวิชามัธยมศึกษาวิทยาศาสตร์กระจายตัวอยู่ในหลากหลายกลุ่ม สาขาวิชาเทคโนโลยีการศึกษาส่วนใหญ่อยู่กลุ่มสูง และสาขาอื่น ๆ อยู่กลุ่มต่ำ วิชาเอกคอมพิวเตอร์ทั้งหมดอยู่กลุ่มสูง วิชาเอกคณิตศาสตร์และวิทยาศาสตร์มีการกระจายอยู่ในหลากหลายกลุ่ม วิชาเอกภาษาและอื่น ๆ ทั้งหมดอยู่กลุ่มต่ำ และทุกคนที่อยู่กลุ่มสูงเคยเรียนการเขียนโค้ดมาก่อน รายละเอียดดังตารางต่อไปนี้

ตาราง 4.11 สาขาวิชา วิชาเอก และประสบการณ์ด้านการเขียนโค้ดจำแนกตามเส้นทางการเรียนรู้

ตัวแปร	เส้นทางการเรียนรู้						
	กลุ่มสูง			กลุ่มต่ำ			
	HHH	HMM	MHH	HHL	HML	MHL	MLL
สาขาวิชา							
เทคโนโลยีการศึกษา	3	0	2	0	0	1	0
อื่น ๆ	1	1	3	3	1	4	3
วิชาเอก							
คอมพิวเตอร์	3	0	2	0	0	0	0
อื่น ๆ	1	1	3	3	1	5	3

ประสบการณ์ด้านการเขียนโค้ด							
เคยเรียน	4	1	5	2	0	3	2
ไม่เคยเรียน	0	0	0	1	1	2	1
รวม	4	1	5	3	1	5	3

การวิเคราะห์เพื่อเปรียบเทียบความสามารถด้านการเขียนโค้ด

การวิเคราะห์เพื่อเปรียบเทียบความสามารถด้านการเขียนโค้ดแบ่งออกเป็น 2 ลักษณะได้แก่ การเปรียบเทียบความสามารถด้านการเขียนโค้ดก่อนเรียนกับหลังเรียน และการเปรียบเทียบความสามารถด้านการเขียนโค้ดระหว่างเรียน

การเปรียบเทียบความสามารถด้านการเขียนโค้ดก่อนเรียนกับหลังเรียน ใช้การเปรียบเทียบคะแนนสัมพัทธ์ที่คำนวณได้จากสูตร $\frac{\text{คะแนนหลังเรียน} - \text{คะแนนก่อนเรียน}}{\text{คะแนนเต็ม} - \text{คะแนนก่อนเรียน}}$ เปรียบเทียบระหว่างกลุ่มที่ได้รับบทเรียนแตกต่างกัน พบว่า กลุ่ม MLL ได้คะแนนสูงที่สุดเท่ากับ .57 รองลงมาคือกลุ่ม MHL ได้คะแนนเท่ากับ .55 น้อยที่สุดคือ กลุ่ม HML ได้คะแนนเท่ากับ .00 รองลงมาคือกลุ่ม HHH ได้คะแนนเท่ากับ .15 และกลุ่มที่ไม่เคยเรียนเขียนโค้ดได้คะแนนสัมพัทธ์สูงกว่ากลุ่มที่เคยเรียนเขียนโค้ด รายละเอียดดังตารางต่อไปนี้

ตาราง 4.12 คะแนนก่อนเรียน คะแนนหลังเรียน คะแนนสัมพัทธ์ และร้อยละคะแนนสัมพัทธ์จำแนกตามกลุ่มที่ได้รับบทเรียนแตกต่างกัน และประสบการณ์เรียนรู้

	จำนวน (คน)	**คะแนนก่อนเรียน**	**คะแนนหลังเรียน**	**คะแนนสัมพัทธ์**	**ร้อยละคะแนนสัมพัทธ์**
เส้นทางการเรียนรู้					
MLL	3	0.74	6.00	.57	57
MHL	5	2.50	6.60	.55	55
HHL	3	3.00	6.67	.52	52
MHH	5	4.28	6.60	.40	40
HMM	1	2.50	5.00	.33	33
HHH	4	4.47	5.25	.15	15
HML	1	2.00	2.00	.00	0
ประสบการณ์ด้านการเขียนโค้ด					
ไม่เคยเรียน	5	2.08	5.4	.42	42
เคยเรียน	17	3.36	6.2	.40	40

CHULALONGKORN UNIVERSITY

การเปรียบเทียบความสามารถด้านการเขียนโค้ดระหว่างเรียน เก็บรวบรวมข้อมูลจากแบบทดสอบในแต่ละบทเรียนย่อยของแต่ละโมดูลซึ่งแต่ละกลุ่มจะได้รับจำนวนบทเรียนที่ไม่เท่ากัน กล่าวคือ กลุ่มต่ำ จะได้รับบทเรียนพื้นฐาน บทเรียนปกติ และบทเรียนท้าทาย ไล่เรียงเป็นลำดับ กลุ่มกลางได้รับการประเมินก่อนเรียนแล้วว่ามีพื้นฐานเพียงพอจึงได้รับบทเรียนปกติและบทเรียนท้าท้าย ส่วนกลุ่มสูงจะได้รับการประเมินแล้วว่ามีพื้นฐานที่มากเพียงพอ จึงได้รับบทเรียนท้าท้าย ซึ่งคะแนนของบทเรียนในแต่ละกลุ่มเส้นทางการเรียนรู้มีรายละเอียดดังตารางต่อไปนี้

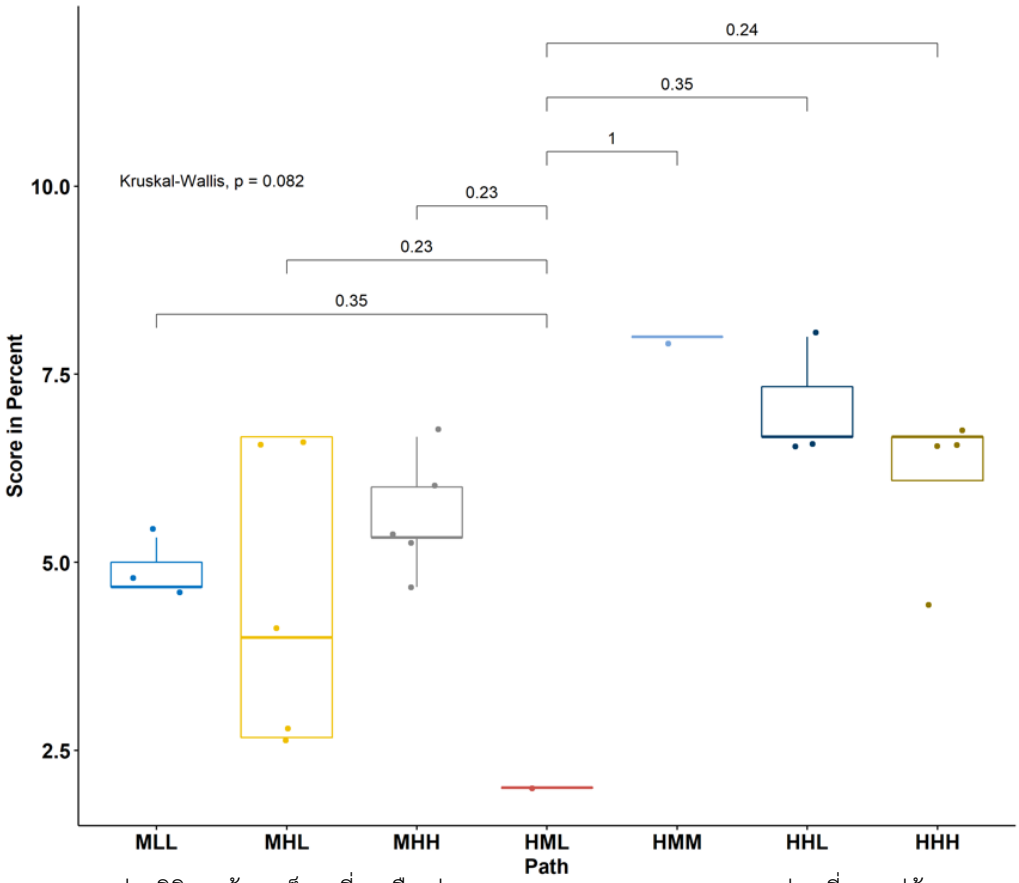
ตาราง 4.13 ค่าเฉลี่ยคะแนนก่อนเรียน หลังเรียนในโมดูลที่ 1 การเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ จำแนกตามกลุ่มที่ได้รับบทเรียนแตกต่างกัน

กลุ่ม	**การเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์**			
	คะแนนก่อนเรียน (เต็ม 4 คะแนน)	**lightbot (เต็ม 6 คะแนน)**	**flowchart (เต็ม 4 คะแนน)**	**algorithm (เต็ม 8 คะแนน)**
MLL (n=3)	1.47	-	4.00	4.89
MHL (n=5)	1.60	-	3.80	4.54
MHH (n=5)	1.55	-	3.80	5.60
HML (n=1)	2.00	-	-	2.00
HMM (n=1)	2.00	-	-	8.00
HHL (n=3)	2.67	-	-	7.11
HHH (n=4)	2.69	-	-	6.09

หมายเหตุ: - หมายถึงกลุ่มดังกล่าวไม่ได้รับบทเรียนเนื่องจากการปรับเหมาะตามระดับความสามารถพื้นฐาน

CHULALONGKORN UNIVERSITY

จากตารางข้างต้น พบว่าทุกคนได้คะแนนสูงกว่าเกณฑ์สำหรับกลุ่มต่ำจึงทำให้ไม่มีกลุ่มใดได้รับบทเรียนพื้นฐานสำหรับกลุ่มต่ำเลย และมีกลุ่มกลาง 3 กลุ่มที่ได้รับบทเรียนปกติและบทเรียนท้าท้าย และมีกลุ่มสูง 4 กลุ่มที่ได้รับบทเรียนท้าท้าย หากพิจารณาคะแนนของบทเรียนท้าทายเท่านั้นจะพบว่า โมดูลที่ 1 การเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ กลุ่ม HMM ได้คะแนนสูงที่สุด 8 คะแนน คิดเป็นร้อยละ 100 และ HML ได้คะแนนต่ำที่สุด 2 คะแนน คิดเป็นร้อยละ 25



หมายเหตุ: ค่าสถิติบนเส้นวงเล็บเหลี่ยม คือ ค่า exact p-value จากการทดสอบค่าเฉลี่ยรายคู่ด้วย Wilcoxon test

ภาพ 4.20 แผนภาพแสดงคะแนนหลังเรียนจากโมดูลที่ 1 การเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์

ในแผนภาพ 2.20 – 2.22 จุด หมายถึงคะแนนของผู้เรียนในแต่ละกลุ่ม ซึ่งนำเสนอร่วมกับแผนภาพกล่องเพื่อให้แสดงถึงการกระจายของคะแนนในแต่ละกลุ่ม ค่าสถิติในแผนภาพนำมาจากการวิเคราะห์เปรียบเทียบค่าเฉลี่ยภาพรวม 7 กลุ่มด้วย Kruskal–Wallis test การเปรียบเทียบรายคู่ใช้กลุ่ม HML เป็นกลุ่มฐาน เนื่องจากมีพฤติกรรมการเรียนที่ไม่ปกติ โดยมีพฤติกรรมที่เริ่มต้นที่ดี แต่เมื่อพิจารณาจากระยะเวลาในการทำแบบทดสอบและเรียนบทเรียนพบว่าใช้ระยะเวลาในการเรียนต่ำกว่าค่าเฉลี่ย จึงอาจหมายถึงการไม่ตั้งใจแม้ว่าผลการเปรียบเทียบรายคู่ส่วนใหญ่จะไม่พบนัยสำคัญทางสถิติ แต่ก็ทำให้ค้นพบรูปแบบของคะแนนบางอย่างดังต่อไปนี้

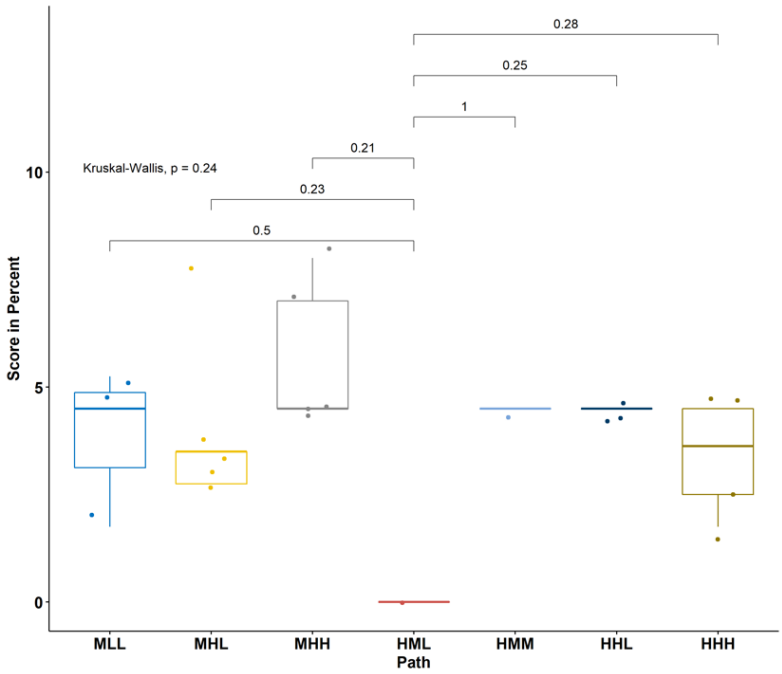
กลุ่มที่ขึ้นต้น H 3 กลุ่มด้านขวา ได้คะแนนรวมสูงกว่ากลุ่มฝั่งซ้าย กลุ่มที่ขึ้นต้น M มี 2 คนในกลุ่ม MHL ที่ทำให้คะแนนกลุ่มต่ำกว่ากลุ่ม M ที่เหลือ กลุ่ม MHH ค่อนข้าง มีคะแนนทับซ้อนกับ กลุ่ม HHH และกลุ่ม HHH มี 1 คน ที่ได้คะแนนต่ำกว่าอีก 3 คนที่เหลือ

ตาราง 4.14 ค่าเฉลี่ยคะแนนก่อนเรียน หลังเรียนในโมดูลที่ 2 ภาษา Scratch จำแนกตามกลุ่มที่ได้รับบทเรียนแตกต่างกัน

กลุ่ม	การเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์			
	คะแนนก่อนเรียน (เต็ม 4 คะแนน)	introduction (เต็ม 2 คะแนน)	menu block (เต็ม 2 คะแนน)	game (เต็ม 8 คะแนน)
MLL (n=3)	0.00	2	2	3.83
MHL (n=5)	3.40	-	-	4.10
MHH (n=5)	3.40	-	-	5.70
HML (n=1)	2.00	-	2	0.00
HMM (n=1)	1.00	-	2	4.50
HHL (n=3)	3.33	-	-	4.50
HHH (n=4)	3.25	-	-	3.38

หมายเหตุ: - หมายถึงกลุ่มดังกล่าวไม่ได้รับบทเรียนเนื่องจากการปรับเหมาะตามระดับความสามารถพื้นฐานของผู้เรียน

จากตารางข้างต้น พบว่า มี 1 กลุ่มที่ได้บทเรียนพื้นฐาน ปกติ และท้าทายเนื่องจากได้คะแนนก่อนเรียนต่ำ มีกลุ่มกลาง 2 กลุ่มที่ได้รับบทเรียนปกติและบทเรียนท้าท้าย และมีกลุ่มสูง 4 กลุ่มได้รับบทเรียนท้าท้าย หากพิจารณาคะแนนของบทเรียนท้าทายเท่านั้น จะพบว่า โมดูลที่ 2 ภาษา Scratch กลุ่ม MHH ได้คะแนนสูงที่สุด 5.70 คะแนน คิดเป็นร้อยละ 71.25 และ HML ได้คะแนนต่ำที่สุด 0 คะแนน คิดเป็นร้อยละ 0



หมายเหตุ: ค่าสถิติบนเส้นวงเล็บเหลี่ยม คือ ค่า exact p-value จากการทดสอบค่าเฉลี่ยรายคู่ด้วย Wilcoxon test

ภาพ 4.21 แผนภาพแสดงคะแนนหลังเรียนจากโมดูลที่ 2 ภาษา Scratch

จากแผนภาพข้างต้น ส่วนใหญ่จะได้คะแนนใกล้เคียงกันทั้ง 6 กลุ่ม ซึ่งแต่ละกลุ่มมีจะคนที่ได้คะแนนสูงกว่าหรือต่ำกว่าคะแนนส่วนใหญ่อย่างเห็นได้ชัด ได้แก่ กลุ่ม MLL มี 1 คน ได้คะแนนต่ำ กลุ่ม MHL มี 1 คนได้คะแนนสูง กลุ่ม MHH และ HHH แบ่งออกเป็น 2 กลุ่มกลุ่มละ 2 คน อย่างชัดเจน

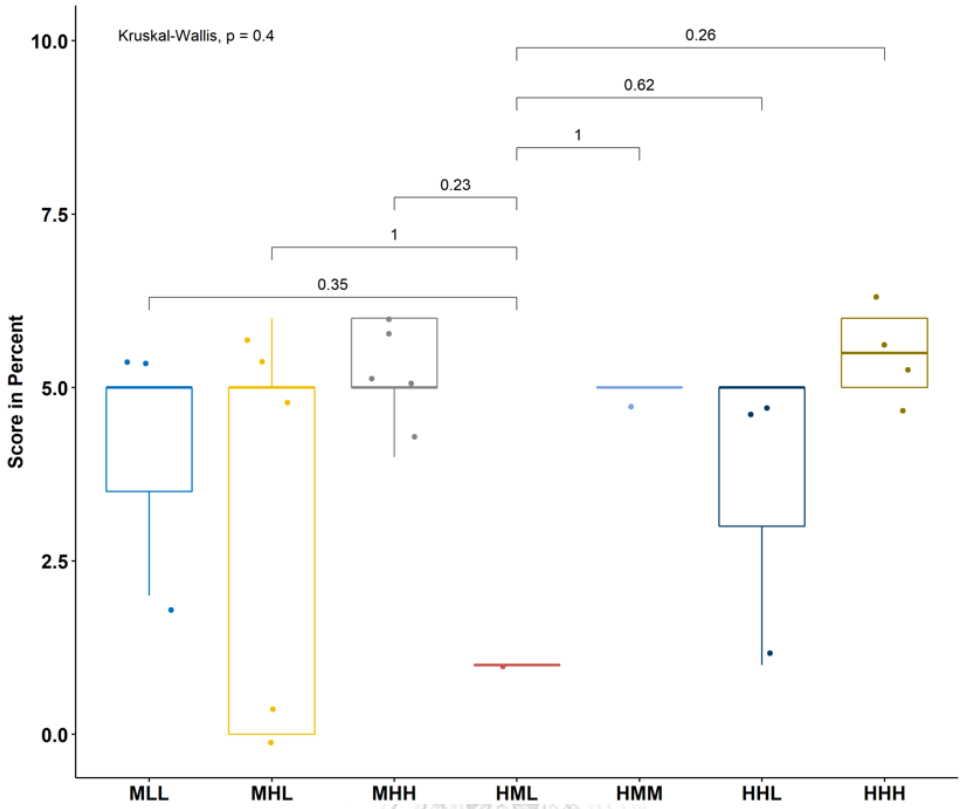
ตาราง 4.15 ค่าเฉลี่ยคะแนนก่อนเรียน หลังเรียนในโมดูลที่ 3 ภาษา Python จำแนกตามกลุ่มที่ได้รับบทเรียนแตกต่างกัน

กลุ่ม	**การเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์**			
	คะแนนก่อนเรียน (เต็ม 4 คะแนน)	**data type (เต็ม 5 คะแนน)**	**condition (เต็ม 3 คะแนน)**	**function (เต็ม 8 คะแนน)**
ML**L** (n=3)	0.00	1.00	1.00	4.00
MH**L** (n=5)	0.00	2.40	2.00	3.20
MH**H** (n=5)	3.60	-	-	5.00
HM**L** (n=1)	0.00	0.00	1.00	1.00
HM**M** (n=1)	2.00	-	2.00	5.00
HH**L** (n=3)	0.00	2.00	1.67	3.67
HH**H** (n=4)	3.00	-	-	5.50

หมายเหตุ: - หมายถึงกลุ่มดังกล่าวไม่ได้รับบทเรียนเนื่องจากการปรับเหมาะตามระดับความสามารถพื้นฐานของผู้เรียน

CHULALONGKORN UNIVERSITY

จากตารางข้างต้น พบว่า มี 4 กลุ่ม ได้รับบทเรียนพื้นฐาน ปกติ และท้าทาย มี 1 กลุ่มได้รับบทเรียนปกติและท้าท้าย และมี 2 กลุ่มได้รับบทเรียนท้าทาย หากพิจารณาคะแนนของบทเรียนท้าทายเท่านั้น จะพบว่า โมดูลที่ 3 ภาษา Python กลุ่ม HHH ได้คะแนนสูงที่สุด 5.50 คะแนน คิดเป็นร้อยละ 68.75 และ HML ได้คะแนนต่ำที่สุด 1 คะแนน คิดเป็นร้อยละ 12.50



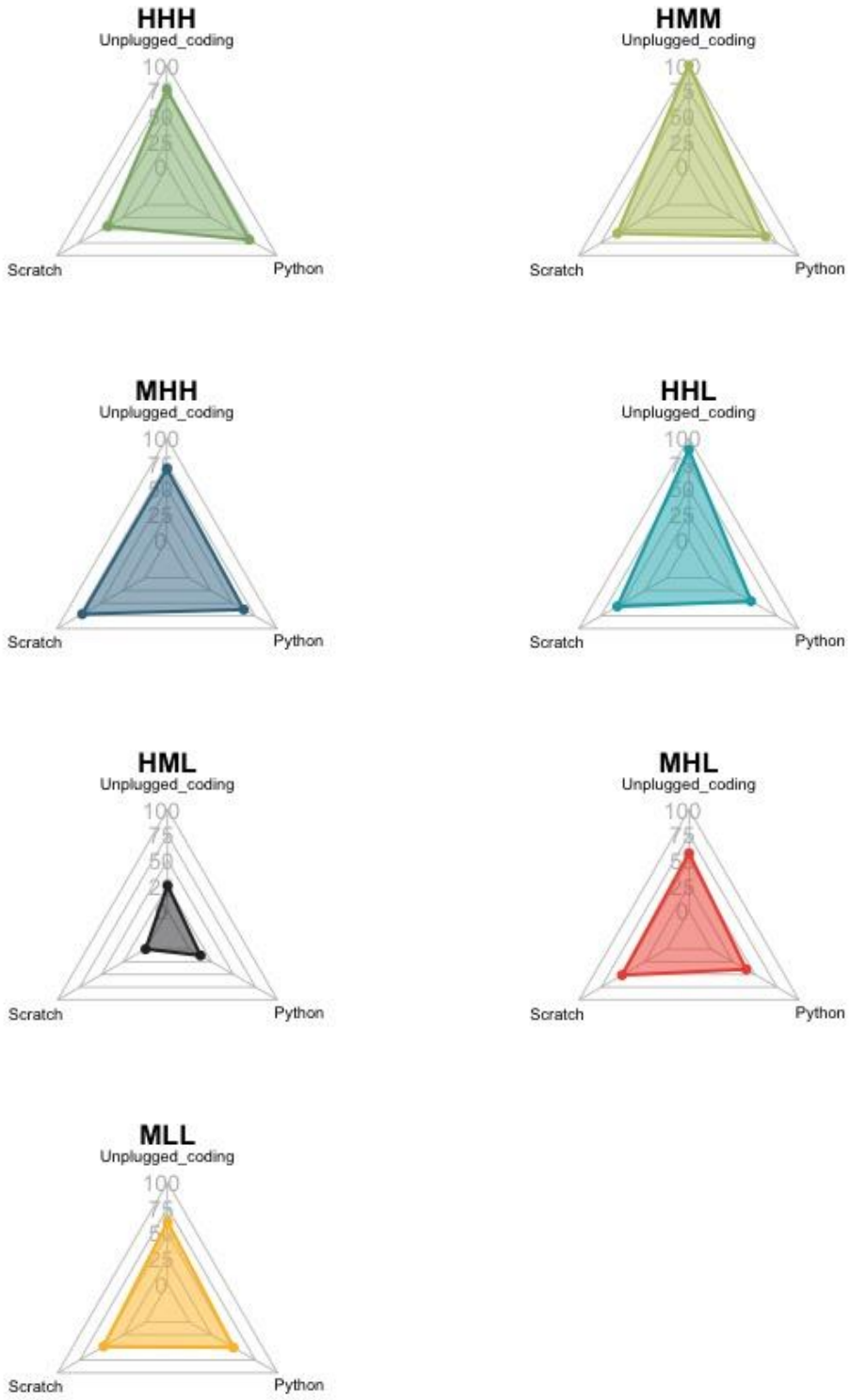
หมายเหตุ: ค่าสถิติบนเส้นวงเล็บเหลี่ยม คือ ค่า exact p-value จากการทดสอบค่าเฉลี่ยรายคู่ด้วย Wilcoxon test

ภาพ 4.22 แผนภาพแสดงคะแนนหลังเรียนจากโมดูลที่ 3 ภาษา Python

จากแผนภาพข้างต้น ส่วนใหญ่จะได้คะแนนใกล้เคียงกันทั้ง 6 กลุ่ม ซึ่งแต่ละกลุ่มมีจะคนที่ได้คะแนนสูงกว่าหรือต่ำกว่าคะแนนส่วนใหญ่อย่างเห็นได้ชัด ได้แก่ กลุ่ม MLL, MHL และ HHL มีคนที่ได้คะแนนต่ำกว่าคนอื่นอย่างชัดเจน 1 คน 2 คน และ 1 คน ตามลำดับ และกลุ่ม MHH มี 1 คนที่ได้คะแนนต่ำกว่าคนอื่น ๆ ภายในกลุ่ม

การเปรียบเทียบความสามารถด้านการเขียนโค้ดของกลุ่มที่ได้รับบทเรียนแตกต่างกันในแต่ละเส้นทาง เมื่อพิจารณาเปรียบจากคะแนนที่แปลงเป็นร้อยละ ประกอบด้วย คะแนนระหว่างเรียนในโมดูล การเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ (Unplugged coding) โมดูล Scratch และโมดูล Python ได้ผลลัพธ์ดังแผนภาพต่อไปนี้

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University



ภาพ 4.23 แผนภาพระดับความสามารถด้านการเขียนโค้ดระหว่างเรียน

บทที่ 5
สรุปผล อภิปรายผล และข้อเสนอแนะการวิจัย

การวิจัยเรื่อง “การพัฒนาบทเรียนออนไลน์เพื่อส่งเสริมความสามารถด้านการเขียนโค้ดโดยใช้การเรียนรู้แบบจุลภาคสำหรับนักศึกษาครู : การวิจัยเชิงทดลองแบบปรับเหมาะ” มีวัตถุประสงค์ 3 ข้อ ได้แก่ 1) เพื่อเปรียบเทียบความสามารถด้านการเขียนโค้ดของนักศึกษาครูที่มีภูมิหลังแตกต่างกัน 2) เพื่อออกแบบและพัฒนาบทเรียนออนไลน์ด้านการเขียนโค้ดโดยใช้การเรียนรู้แบบจุลภาค 3) เพื่อเปรียบเทียบโปรไฟล์และความสามารถด้านการเขียนโค้ดระหว่างได้รับบทเรียนออนไลน์ด้านการเขียนโค้ดที่ออกแบบตามการเรียนรู้แบบจุลภาคที่มีรูปแบบแตกต่างกันของนักศึกษาครูที่มีภูมิหลังแตกต่างกัน

ระยะที่ 1 การพัฒนาบทเรียนการเรียนรู้แบบจุลภาคด้านการเขียนโค้ด ประกอบด้วย 2 ส่วน คือการออกแบบโครงสร้างของบทเรียนและการออกแบบบทเรียน ซึ่งโครงสร้างของบทเรียนประกอบด้วย 3 ขั้นตอนหลักได้แก่ การแนะนำบทเรียน การศึกษาบทเรียน และการประเมินผล การออกแบบบทเรียนประกอบด้วย 2 ขั้นตอนหลักได้แก่ การออกแบบเนื้อหาจุลภาคที่ต้องออกแบบเนื้อหาวิชาเกี่ยวกับการเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ ภาษา Scratch และ ภาษา Python รวมถึงการคัดเลือกแพลตฟอร์มการเรียนรู้ และการออกแบบกิจกรรมจุลภาค ตามหลักการออกแบบโดย Buchem และ Hemelmann

ระยะที่ 2 การศึกษาผลของการใช้บทเรียนออนไลน์ด้านการเขียนโค้ดที่ออกแบบโดยแนวคิดการเรียนรู้แบบจุลภาค วิเคราะห์ความสามารถด้านการเขียนโค้ดและความพึงพอใจต่อการเรียนรู้แบบจุลภาคเปรียบเทียบตามตัวแปรภูมิหลัง และทำการทดลองกับตัวอย่างวิจัยโดยตัวอย่างวิจัยจะได้รับบทเรียนที่แตกต่างกันขึ้นอยู่กับความสามารถพื้นฐาน โดยบทเรียนที่ได้รับประกอบด้วย 3 โมดูลหลัก ได้แก่ โมดูลการเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ โมดูล ภาษา Scratch และ โมดูลภาษา Python หลังจากนั้นประเมินผลด้วยแบบทดสอบความสามารถด้านการเขียนโค้ดหลังเรียนเพื่อใช้ในการเปรียบเทียบความสามารถด้านการเขียนโค้ดทั้งระหว่างเรียน กับก่อนและหลังเรียน

สรุปผลการวิจัย

1. ความสามารถด้านการเขียนโค้ดของนักศึกษาครูที่มีภูมิหลังแตกต่างกัน

คะแนนความสามารถด้านการเขียนโค้ดรวม และคะแนนด้านการเขียนโค้ดด้านโปรแกรมต่ำกว่าร้อยละ 50 ส่วนคะแนนด้านการแก้ไขปัญหามากกว่าร้อยละ 50 ของคะแนนเต็ม เมื่อวิเคราะห์ความแตกต่างของค่าเฉลี่ยของคะแนนรวม และคะแนนในแต่ละองค์ประกอบ ระหว่างกลุ่มที่เคยเรียน

การเขียนโค้ดกับกลุ่มที่ไม่เคยเรียนการเขียนโค้ด โดยใช้สถิติ independent sample t-test พบว่า คะแนนรวมระหว่างกลุ่มที่เคยเรียนการเขียนโค้ดกับกลุ่มที่ไม่เคยเรียนการเขียนโค้ดแตกต่าง คะแนนองค์ประกอบด้านโปรแกรมระหว่างกลุ่มที่เคยเรียนการเขียนโค้ดกับกลุ่มที่ไม่เคยเรียนการเขียนโค้ดแตกต่างกัน และคะแนนรวม และคะแนนองค์ประกอบด้านโปรแกรมระหว่างวิชาเอกคอมพิวเตอร์กับวิชาเอกอื่น ๆ แตกต่างกัน และระหว่างวิชาเอกวิชาเอกคณิตศาสตร์-วิทยาศาสตร์กับวิชาเอกอื่น ๆ แตกต่างกัน

2. บทเรียนออนไลน์ด้านการเขียนโค้ดที่ออกแบบโดยแนวคิดการเรียนรู้แบบจุลภาค

โครงสร้างของบทเรียนประกอบด้วย 3 ขั้นตอนได้แก่ 1) การแนะนำบทเรียน 2) การศึกษาบทเรียน และ 3) การประเมินผล

1) แนะนำบทเรียนเป็นโมดูลเริ่มต้นประกอบด้วย แบบสำรวจความพึงพอใจต่อการเรียนรู้แบบจุลภาคและความสามารถด้านการเขียนโค้ด บทเรียนพื้นฐานเกี่ยวกับการเขียนโค้ด และการประเมินผล

2) การศึกษาบทเรียนประกอบด้วย 3 โมดูลได้แก่ โมดูลการเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ โมดูล ภาษา Scratch และ โมดูลภาษา Python ซึ่งแต่ละโมดูลก่อนเรียนจะได้รับการประเมินความสามารถด้านการเขียนโค้ดว่าอยู่ในระดับใดเพื่อที่จะได้รับบทเรียนที่ตอบสนองความต้องการของผู้เรียน ดังนี้ ความสามารถระดับต่ำ จะได้รับบทเรียนพื้นฐาน บทเรียนปกติ และบทเรียนท้าทายตามลำดับ ความสามารถระดับปานกลาง จะได้รับบทเรียนปกติ และบทเรียนท้าทาย และความสามารถระดับสูง จะได้รับบทเรียนท้าทาย โดยโมดูลทั้ง 3 จะประกอบด้วยบทเรียนย่อยดังต่อไปนี้ 2.1) โมดูลการเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ ประกอบด้วย บทเรียนขั้นตอนวิธีผ่าน lightbot hour ผังงาน และขั้นตอนวิธีในบริบทโรงเรียน 2.2) ภาษา Scratch ประกอบด้วย บทเรียนพื้นฐาน เมนูบล็อก และเกม 2.3) ภาษา Python ประกอบด้วย บทเรียนพื้นฐาน เงื่อนไข และฟังก์ชัน เมื่อได้ศึกษาในแต่ละบทเรียนแล้วจะได้รับการประเมินความสามารถในบทเรียนนั้น ๆ

3) การประเมินผลใช้แบบวัดความสามารถด้านการเขียนโค้ดหลังเรียนที่สร้างขึ้นโดยอิงกับองค์ประกอบของความสามารถด้านการเขียนโค้ดทั้ง 2 องค์ประกอบได้แก่ องค์ประกอบด้านแก้ไขปัญหา และองค์ประกอบด้านโปรแกรม จำแนกตามเนื้อหาที่ประกอบด้วย 3 บทเรียนได้แก่การเขียนโค้ดแบบไม่ใช้คอมพิวเตอร์ โมดูล ภาษา Scratch และ โมดูลภาษา Python

บทเรียนออนไลน์ด้านการเขียนโค้ดออกแบบขั้นบทแพลตฟอร์มการเรียนรู้ออนไลน์ Canvas learning management system (Canvas lms) โดยมีสื่อการเรียนรู้ที่สร้างขึ้นตามแนวคิดการเรียนรู้แบบจุลภาค ประกอบด้วย ข้อความบรรยาย เกม วิดีโอ วิดีโอเชิงโต้ตอบ (interactive video) เอกสารเชิงโต้ตอบ (interactive slide) โปรแกรมเขียนผังงานบนเว็บไซต์ draw.io เว็บเขียนภาษา

Scratch Microsoft MakeCode arcade และหน้าต่างการเขียนโค้ดเชิงโต้ตอบบนเว็บ (interactive Python Console)

3. โปรไฟล์และความสามารถด้านการเขียนโค้ดระหว่างได้รับบทเรียนออนไลน์ด้านการเขียนโค้ดที่ออกแบบตามการเรียนรู้แบบจุลภาคที่มีรูปแบบแตกต่างกันของนักศึกษาครูที่มีภูมิหลังแตกต่างกัน

3.1 การเปรียบเทียบความสามารถด้านการเขียนโค้ดก่อนเรียนกับหลังเรียน ใช้การเปรียบเทียบคะแนนสัมพัทธ์ สรุปได้ว่ากลุ่ม MLL MHL HHL ซึ่งไม่มีความรู้พื้นฐานเกี่ยวกับภาษา Python จะสามารถพัฒนาความสามารถด้านการเขียนโค้ดได้สูงกว่ากลุ่ม MHH HMM HHH ซึ่งมีความสามารถด้านภาษาโปรแกรมมาแล้ว และ กลุ่ม HML ไม่มีการพัฒนาความสามารถด้านการเขียนโค้ด

3.2 การเปรียบเทียบความสามารถด้านการเขียนโค้ดระหว่างเรียน เมื่อพิจารณาจากบทเรียนท้าท้ายของแต่ละโมดูล สรุปได้ว่า กลุ่มที่มีพื้นฐานของแต่ละโมดูลนั้น ๆ สูง จะมีคะแนนหลังเรียนที่สูงกว่ากลุ่มที่ไม่มีพื้นฐาน ยกเว้นกลุ่ม HML ที่ได้คะแนนต่ำในทุกโมดูล และเมื่อเปรียบเทียบความสามารถด้านการเขียนโค้ดในแต่ละโมดูลระหว่างกลุ่มที่มีความพึงพอใจต่อการเรียนรู้แบบจุลภาคแตกต่างกันไม่พบนัยสำคัญทางสถิติ

อภิปรายผลการวิจัย

ประเด็นการอภิปรายผลการวิจัย จำแนกออกเป็น 3 ประเด็นได้แก่ 1) ความสามารถด้านการเขียนโค้ดระหว่างกลุ่มที่มีภูมิหลังต่างกัน 2) บทเรียนออนไลน์ด้านการเขียนโค้ดที่สร้างขึ้นตามแนวคิดการเรียนรู้แบบจุลภาค 3) เส้นทางการเรียนรู้บทเรียนออนไลน์ด้านการเขียนโค้ดระหว่างกลุ่มที่มีภูมิหลังแตกต่างกัน

1. ความสามารถด้านการเขียนโค้ดระหว่างกลุ่มที่มีภูมิหลังต่างกัน

ความรู้ด้านการเขียนโค้ดประกอบด้วย 2 องค์ประกอบคือ ด้านการแก้ไขปัญหา และด้านโปรแกรม ซึ่งจากผลของคะแนนความสามารถระหว่างกลุ่มที่มีสาขาวิชา และวิชาเอกแตกต่างกันนั้น กลุ่มที่มีความสามารถด้านการเขียนโค้ดสูงจะเป็นกลุ่มที่อยู่สาขาวิชาเทคโนโลยีการศึกษา และวิชาเอกเกี่ยวกับคอมพิวเตอร์ เนื่องจากมีพื้นฐานความรู้และได้เคยเรียนการเขียนโค้ดมากก่อน ประกอบกับความสามารถด้านการเขียนโค้ดเป็นส่วนหนึ่งของความสามารถที่ตรงกับความสามารถที่สาขาวิชาและวิชาเอกนี้มุ่งเน้น นอกจากนี้ กลุ่มสาขาวิชา และวิชาเอก คณิตศาสตร์ วิทยาศาสตร์ จะมีทั้งคนที่ได้คะแนนต่ำไปจนกระทั่งถึงสูง ถึงแม้ว่าวิชาเอกและสาขาวิชาทั้งสองไม่ได้มีการเรียนเกี่ยวกับด้านการเขียนโค้ดโดยตรง แต่เนื้อหาวิชาด้านคณิตศาสตร์ที่เกี่ยวกับตรรกะการคิดนั้น ก็เป็นส่วนหนึ่งในองค์ประกอบด้านการแก้ไขปัญหา ร่วมถึงเป็นพื้นฐานในการทำความเข้าใจภาษาโปรแกรม จึงพบว่ามี

คนกลุ่มนึงที่มีคะแนนในระดับที่สูงได้ นอกจากนี้กลุ่มวิชาอื่น ๆ ส่วนใหญ่จะมีความสามารถด้านการเขียนโค้ดในระดับต่ำ ด้วยเนื้อหาและความเกี่ยวข้องของสาขาวิชาและวิชาเอกอื่น ๆ ไม่ได้มีการศึกษาที่ใกล้เคียงกับการเขียนโค้ด จึงพบว่าไม่มีคนในกลุ่มวิชาอื่น ๆ ที่ได้คะแนนในระดับสูงเลย

ความสามารถด้านการเขียนโค้ดระหว่างกลุ่มที่เคยเรียนและไม่เคยเขียนโค้ดก่อนทดลองพบว่า ทั้งสองกลุ่มมีความสามารถในองค์ประกอบด้านการแก้ไขปัญหาใกล้เคียงกัน แต่กลุ่มที่ไม่เคยเรียนเขียนโค้ดมีคะแนนองค์ประกอบด้านโปรแกรมต่ำกว่ามาก แต่เมื่อทำการทดลองและเปรียบเทียบคะแนนก่อนเรียนและหลังเรียนพบว่า ความแตกต่างระหว่างคะแนนหลังเรียนนั้นใกล้เคียงกันมากยิ่งขึ้น และเมื่อพิจารณาตามคะแนนสัมพัทธ์พบว่า กลุ่มที่ไม่เคยเรียนเขียนโค้ดกลับมีคะแนนสัมพัทธ์สูงกว่ากลุ่มที่เคยเรียนเขียนโค้ด ดังนั้น บทเรียนออนไลน์ด้านการเขียนโค้ดที่พัฒนาขึ้นนี้สามารถทำให้กลุ่มที่ไม่เคยเรียนเขียนโค้ดมีความสามารถด้านการเขียนโค้ดได้ดียิ่งขึ้น

การใช้สถิติทดสอบกับตัวอย่างวิจัยที่ได้มาโดยการเลือกตามสะดวกและมีขนาดเล็กนั้น เมื่อทดสอบข้อตกลงเบื้องต้นเกี่ยวกับการกระจายพบว่าเป็นไปตามข้อตกลงเบื้องต้นจึงสามารถใช้สถิติทดสอบได้แม้ขนาดตัวอย่างจะมีขนาดเล็กก็ตาม สำหรับตัวอย่างขนาดเล็กยังสามารถใช้เทคนิคการสุ่มซ้ำหรือ bootstrap ซึ่งเป็นเทคนิคทางสถิติเพื่อใช้ดูผลลัพธ์ประกอบกันว่าผลที่ได้สอดคล้องกันหรือขัดแย้งกันหรือไม่ ในส่วนที่ไม่เป็นไปตามข้อตกลงเบื้องต้นสามารถใช้สถิติ non-parametric แทนเพื่อให้ถูกต้องตามหลักสถิติและเพื่อให้ได้ผลลัพธ์ที่สมเหตุสมผลตามหลักวิชามากยิ่งขึ้น

2. บทเรียนออนไลน์ด้านการเขียนโค้ดที่สร้างขึ้นตามแนวคิดการเรียนรู้แบบจุลภาค

การได้มาซึ่งบทเรียนออนไลน์ด้านการเขียนโค้ดในแต่ละโมดูลนั้นจะประกอบไปด้วยเนื้อหาจุลภาคที่ได้วิเคราะห์ไว้แล้ว และกิจกรรมจุลภาคของแต่ละบทเรียนจะสังเกตได้ว่ามีความแตกต่างกันและแต่ละบทเรียนไม่ได้มีกิจกรรมจุลภาคที่หลากหลาย กล่าวคือ แต่ละบทเรียนจะมีรูปแบบการเรียนรู้แบบเดียว ซึ่งในส่วนนี้ได้มีการจับคู่ระหว่างลักษณะเนื้อหากับลักษณะของกิจกรรมที่มีความเหมาะสมกัน ดังนี้ เนื้อหาพื้นฐานที่เน้นความเข้าใจ จะใช้ข้อความบรรยายสั้น ๆ วิดีโอเชิงโต้ตอบ เอกสารเชิงโต้ตอบ เพื่อที่สามารถอ่านซ้ำและเรียนซ้ำได้หลาย ๆ ครั้ง ส่วนเนื้อหาที่ต้องใช้การฝึกฝน เช่น การเขียนภาษาโปรแกรม เลือกใช้เว็บไซต์สำเร็จรูป และ หน้าต่างการเขียนโค้ดเชิงโต้ตอบ ที่สามารถตรวจสอบความถูกต้องของโค้ดได้ตลอดเวลา เพื่อให้สามารถทำความเข้าใจได้ดียิ่งขึ้น

3. เส้นทางการเรียนรู้ได้จำแนกออกตามระดับความรู้พื้นฐานของแต่ละโมดูลกล่าวคือ กลุ่มที่มีพื้นฐานการแก้ไขปัญหาแบบไม่ใช้คอมพิวเตอร์ ซึ่งเป็นกลุ่มที่ไม่จำเป็นต้องรู้เรื่องโปรแกรมเลยก็สามารถได้คะแนนในระดับสูงได้ ซึ่งไม่พบกลุ่มที่ได้คะแนนในระดับต่ำเลย ดังนั้น กลุ่มทดลองนี้มีความรู้พื้นฐาน ความเข้าใจในการแก้ไขปัญหาอยู่ในระดับหนึ่งแล้ว ส่วนกลุ่มที่มีพื้นฐานด้านโปรแกรมจะมีทั้งส่วนที่มีความรู้พื้นฐานในภาษา Scratch และภาษา Python โดยปกติแล้วภาษา Scratch เป็นภาษาบล็อคที่คนส่วนมากเริ่มเรียนก่อนภาษาอื่น ๆ และทำความเข้าใจได้ง่ายกว่า ซึ่งจากลักษณะ

ของกลุ่มที่พบทั้งหมด จะมีระดับความรู้พื้นฐานของภาษา Scratch สูงกว่าหรือเท่ากับ ภาษา Python หากพิจารณากลุ่มทั้ง 7 กลุ่มแล้วจะพบว่า มีทั้งกลุ่มที่มีความรู้ด้านโปรแกรมและไม่มีความรู้ด้านโปรแกรมเลย สังเกตได้จากกลุ่ม HHH และ MLL ซึ่งทั้งสองกลุ่มนี้นั้นก็มคะแนนสัมพัทธ์ที่แตกต่างกันมากซึ่งสามารถสรุปได้ว่า บทเรียนออนไลน์ด้านการเขียนโค้ดที่ออกแบบขึ้นนั้นมีแนวโน้มที่จะสามารถพัฒนาความสามารถด้านการเขียนโค้ดของกลุ่มที่มีพื้นฐานด้านการแก้ปัญหาในระดับหนึ่งแต่ไม่มีความรู้ด้านโปรแกรมได้ดีกว่ากลุ่มอื่น ๆ

กลุ่ม HML ที่มีคะแนนต่ำในทุกโมดูลและมี 1 ครั้งที่ได้คะแนนเท่ากับ 0 นั้นประกอบด้วยตัวอย่างวิจัยเพียง 1 คน และเมื่อตรวจสอบเกี่ยวกับเวลาที่ใช้ในการทำแบบทดสอบและเรียนบทเรียนพบว่า ใช้ระยะเวลาต่ำกว่าเวลาเฉลี่ยค่อนข้างมาก ซึ่งอาจจะส่งผลต่อประสิทธิภาพในการเรียนและการทำแบบประเมินที่ไม่เต็มความสามารถ

ข้อจำกัดของการวิจัย

การดำเนินการเก็บรวบรวมข้อมูลในระยะการทดลอง ได้ตัวอย่างวิจัยมาในจำนวนที่น้อยกว่าที่ได้คาดไว้ เนื่องจาก สถานการณ์การแผ่ระบาดของไวรัส COVID-19 ที่ให้มีการเรียนออนไลน์ ซึ่งประชากรในการวิจัยส่วนใหญ่ไม่ได้อยู่ในมหาวิทยาลัย ทำให้ไม่สามารถเข้าถึงประชากรในการวิจัยครั้งนี้ได้อย่างเต็มประสิทธิภาพ จึงเลือกใช้การติดต่อผ่านทางออนไลน์ซึ่งได้รับการตอบรับในสัดส่วนที่ต่ำอาจเนื่องจาก การทดลองต้องใช้ระยะเวลาในการทดลองค่อนข้างนาน และต้องใช้ทรัพยากรอื่น ๆ เช่น คอมพิวเตอร์ แท็บเล็ต และการสื่อสารที่มีข้อจำกัดหลายอย่าง

ตัวอย่างวิจัยที่ได้จำนวนต่ำกว่าที่คาดไว้นั้นส่งผลต่อขอบเขตในการวิเคราะห์ข้อมูลเนื่องจากจำนวนข้อมูลในแต่ละกลุ่มน้อยเกินกว่าที่จะใช้สถิติอ้างอิงบางสถิติได้ อาจทำให้สารสนเทศที่ได้จากการวิเคราะห์นั้นมีขอบเขตที่แคบลง

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ข้อเสนอแนะจากการวิจัย

1. ข้อเสนอแนะในการนำผลการวิจัยไปใช้

1.1 แบบวัดความสามารถด้านการเขียนโค้ดสามารถนำไปใช้กับนิสิต นักศึกษาคณะครุศาสตร์/ศึกษาศาสตร์ หรือครูจบใหม่ จะมีความเหมาะสมมากกว่ากลุ่มอื่น ๆ เนื่องจากแบบวัดสร้างขึ้นโดยมีการอ้างอิงบริบทของการศึกษา โรงเรียน และบริบทที่เกี่ยวข้องกับช่วงวัย เช่น ค่ายแนะแนวการสอบเข้าศึกษาระดับอุดมศึกษา เกม เป็นต้น แบบวัดความสามารถด้านการเขียนโค้ดมีวัตถุประสงค์เพื่อวัดระดับความสามารถด้านการเขียนโค้ดในองค์ประกอบด้านการแก้ปัญหา ซึ่งเป็นการแก้ปัญหาที่ไม่ใช้คอมพิวเตอร์ และองค์ประกอบด้านโปรแกรม ซึ่งเป็นพื้นฐานเกี่ยวกับภาษา Scratch และ Python ในการนำไปใช้อาจจะต้องคำนึงถึงรูปแบบของแบบวัดที่สร้างขึ้นนั้นต้อง

สามารถตอบโดยการเขียนเป็นภาษาโปรแกรม ข้อคำถามที่ใช้รูปภาพผู้ตอบต้องเห็นอย่างชัดเจน และควรอยู่ในหน้าต่างเดียวกับคำถามและตัวเลือก ในส่วนของข้อสอบที่เป็นข้อสอบแบบตอบสั้น และบรรยาย จะต้องมีการกำหนดเกณฑ์การให้คะแนน เช่น ตัวอย่างคำตอบ หรือ รูบริก

1.2 บทเรียนการเรียนรู้ออนไลน์ด้านการเขียนโค้ดที่พัฒนาขึ้นมีเป้าหมายโดยตรงเพื่อให้คณะครุศาสตร์/ศึกษาศาสตร์ โรงเรียน หรือหน่วยงานที่เกี่ยวข้องกับการพัฒนาครู สามารถนำบทเรียนนี้ไปใช้เพื่อพัฒนาความสามารถด้านการเขียนโค้ดให้กับนิสิต นักศึกษาคณะครุศาสตร์ศึกษาศาสตร์ หรือครูจบใหม่ โดยการเข้าถึงบทเรียนการเรียนรู้ออนไลน์ด้านการเขียนโค้ด สามารถเข้าถึงได้ด้วยการให้ผู้ดูแลระบบของหน่วยงานนั้นเพิ่มนักเรียนเข้าบัญชี Canvas LMS ผ่านทางอีเมล ในส่วนของการเรียนสามารถทำได้ทั้งบนคอมพิวเตอร์ มือถือ และแท็บเล็ตผ่านเว็บไซต์ www.canvas.instructure.com และแอปพลิเคชัน Canvas Student โดยผู้ดูแลระบบหรือครูผู้สอนสามารถสังเกตการณ์และให้ความช่วยเหลือผ่านทางเว็บไซต์ และ แอปพลิเคชัน Canvas Teacher

นอกจากนี้การปรับเหมาะบทเรียนที่ผู้เรียนจะได้รับต่อไปนั้น จะเกิดขึ้นหลังได้รับการประเมินผลแล้วซึ่งประกอบด้วยระบบอัติโนมัติ และการประเมินจากผู้ดูแลหรือครูผู้สอนในส่วนนี้อาจจะใช้เวลาระยะหนึ่งผู้เรียนถึงจะได้รับบทเรียนในส่วนนี้ และควรระวังเรื่องการปรับแต่งเส้นทางการเรียนรู้ใด ๆ หากเกิดขึ้นแล้วจะทำให้ผู้ที่เรียนก่อนและหลังปรับแต่งมีเส้นทางการเรียนรู้ไม่เหมือนกัน แก้ไขปัญหาได้โดยการลบและสร้างบัญชี Canvas LMS ใหม่

1.3 จากผลการวิจัย เมื่อพิจารณาจากประสบการณ์ด้านการเขียนโค้ด และระดับความสามารถด้านการเขียนโค้ดของตัวอย่างวิจัย พบว่า ความสามารถด้านการเขียนโค้ดในองค์ประกอบการแก้ไขปัญหาในระดับปานกลางถึงสูง และความสามารถด้านการเขียนโค้ดในองค์ประกอบด้านโปรแกรมค่อนข้างต่ำ และมีบางคนที่ไม่เคยเรียนการเขียนโค้ดภาษาใด ๆ มาก่อน

ดังนั้น คณะครุศาสตร์/ศึกษาศาสตร์ โรงเรียน และหน่วยงานการศึกษาที่เกี่ยวข้อง สามารถปรับปรุงวิชาเกี่ยวกับการคิดแก้ไขปัญหา ให้มีความเข้มข้นขึ้นหรือสอดแทรกการแก้ไขปัญหาที่เชื่อมโยงกับคอมพิวเตอร์เช่น ขั้นตอนวิธี ได้ เนื่องจากนิสิต นักศึกษาครูมีพื้นฐานที่ค่อนข้างดีอยู่แล้ว นอกจากนี้อาจเพิ่มเติมรายวิชาที่เกี่ยวกับโปรแกรม ภาษาโปรแกรมทั้งภาษาบล็อกและภาษาข้อความให้กับนิสิต นักศึกษาครูได้รู้จักและพัฒนาพื้นฐานความรู้ในระดับปริญญาบัณฑิตก่อนที่จะเข้าไปปฏิบัติหน้าที่เป็นครูในโรงเรียน

เมื่อพิจารณาตามกลุ่มสาขาเทคโนโลยีการศึกษา และกลุ่มสาขาวิชาอื่น ๆ พบว่า มีประสบการณ์ด้านการเขียนโค้ด และระดับความสามารถด้านการเขียนโค้ดแที่แตกต่างกัน คณะครุศาสตร์/ศึกษาศาสตร์ โรงเรียน และหน่วยงานการศึกษาที่เกี่ยวข้อง สามารถพัฒนาความสามารถด้านการเขียนโค้ดของ 2 กลุ่มได้แตกต่างกัน ดังต่อไปนี้

1) กลุ่มสาขาเทคโนโลยีการศึกษา ส่วนใหญ่มีประสบการณ์ด้านการเขียนโค้ดมาก่อน และมีระดับความสามารถด้านการเขียนโค้ดทั้งการแก้ไขปัญหา ภาษาบล็อก และภาษาแบบข้อความ ในระดับที่สูงกว่ากลุ่มสาขาอื่น ๆ ดังนั้น การพัฒนาส่งเสริมความสามารถด้านการเขียนโค้ดสามารถเน้นความรู้ในระดับที่สูงกว่าระดับพื้นฐาน เน้นการต่อยอดจากพื้นฐานความรู้ และความสามารถที่ได้รับจากบทเรียนการเรียนรู้ออนไลน์ด้านการเขียนโค้ดในงานวิจัยครั้งนี้

2) กลุ่มสาขาวิชาอื่น ๆ ส่วนใหญ่มีประสบการณ์ด้านการเขียนโค้ด แต่มีระดับความสามารถด้านการเขียนโค้ดที่ค่อนข้างต่ำ ดังนั้น การพัฒนาส่งเสริมอาจเป็นลักษณะของการทบทวนความรู้เดิมก่อนเพื่อให้กลุ่มนี้มีพื้นฐานเพียงพอที่จะสามารถพัฒนาต่อยอด และเพิ่มเติมพื้นฐานด้านโปรแกรมด้วยภาษาบล็อกก่อนที่จะพัฒนาความสามารถและความรู้เกี่ยวกับภาษาข้อความต่อไป

2. ข้อเสนอแนะในการวิจัยครั้งต่อไป

2.1 การพัฒนาและออกแบบบทเรียนออนไลน์ด้านการเขียนโค้ดในการวิจัยครั้งนี้สร้างขึ้นบนแพลตฟอร์มออนไลน์ที่มีอยู่แล้ว จึงมีข้อจำกัดต่าง ๆ อันเนื่องมากจากแพลตฟอร์ม เพื่อให้สามารถสร้างสรรค์วิธีการหรือรูปแบบของการปรับเหมาะที่ซับซ้อนขึ้นได้ จึงควรมีการศึกษา วิจัยและพัฒนาแพลตฟอร์มออนไลน์ที่สามารถออกแบบการปรับเหมาะได้อย่างมีประสิทธิภาพมากยิ่งขึ้น

2.2 เส้นทางการเรียนรู้ที่พิจารณาจากระดับความสามารถด้านการเขียนโค้ดของแต่ละโมดูลซึ่งประกอบด้วย โมดูลการแก้ไขปัญหาแบบไม่ใช้คอมพิวเตอร์ ภาษา Scratch และ ภาษา Python นั้นจากงานวิจัยนี้พบเพียง 7 กลุ่ม จากความเป็นไปได้ทั้งหมด 27 แบบ อาจเนื่องมาจากจำนวนตัวอย่างวิจัยที่น้อย จึงควรมีการวิจัยครั้งต่อไปไม่ว่าจะเป็นประชากรกลุ่มเดิมหรือประชากรกลุ่มใหม่ด้วยจำนวนตัวอย่างที่มากขึ้น เพื่อศึกษาเกี่ยวกับรูปแบบอื่น ๆ ของเส้นทางการเรียนรู้ที่เกิดขึ้น

2.3 ตัวอย่างวิจัยในกลุ่ม HML ที่มีคะแนนหลังเรียนต่ำในทุกบทเรียนนั้นเมื่อพิจารณาจากระยะเวลาที่ใช้ในการเรียนรู้บทเรียนที่ใช้ระยะเวลาน้อยมากนั้น อาจเป็นตัวบ่งชี้ถึงความตั้งใจในการเรียน ดังนั้น ในการวิจัยครั้งต่อไปหากพัฒนาระบบที่สามารถแจ้งเตือนได้หากใช้เวลาในการเรียนน้อยเกินไป หรือศึกษาเกี่ยวกับตัวแปรความตั้งใจโดยเพิ่มเข้ามาในการออกแบบการทดลอง เพื่อให้ได้ผลลัพธ์ที่แสดงถึงความสามารถที่แท้จริงของผู้เรียน

2.4 การใช้สถิติทดสอบหากตัวอย่างวิจัยมีขนาดเล็กสามารถใช้เทคนิคการสุ่มซ้ำหรือ bootstrap เข้ามาช่วยเพื่อให้สามารถใช้สถิติทดสอบได้อย่างมีประสิทธิภาพมากยิ่งขึ้น และทดสอบข้อตกลงเบื้องต้นก่อนใช้สถิติพาราเมตริก หากข้อมูลที่มีไม่เป็นไปตามข้อตกลงเบื้องต้นสามารถเลือกใช้สถิตินอนพาราเมตริกแทน

บรรณานุกรม

ภาษาไทย

จิระศักดิ์ สุวรรณโณ และคณะฯ. (2561). *สนุก Kids สนุก Code กับ KidBright ฉบับStudent Handbook*. มหาวิทยาลัยธรรมศาสตร์.ปทุมธานี.

รวิวรรณ เทนอิสสระ และคณะฯ. (2560). *คู่มือการใช้หลักสูตรรายวิชาพื้นฐานวิทยาศาสตร์ สาระเทคโนโลยี (วิทยาการคำนวณ) ระดับประถมศึกษาและมัธยมศึกษา กลุ่มสาระการเรียนรู้วิทยาศาสตร์ ตามหลักสูตรแกนกลางการศึกษาขั้นพื้นฐาน พุทธศักราช 2551 (ฉบับปรับปรุง พ.ศ. 2560)*. อักษรเจริญทัศน์ อจท. กรุงเทพมหานคร.

อภิภา ปรัชญพฤทธิ์. (2561). *การพัฒนารูปแบบการผลิตครูเพื่อรองรับการศึกษายุค 4.0*. รายงานการวิจัย, คณะครุศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, กรุงเทพมหานคร.

ภาษาอังกฤษ

Alqurashi, E. (2017). Microlearning: A pedagogical approach for technology integration [Special issue]. In A. Isman, J. Willis, & J. A. Donaldson (Eds.). *The Turkish Online Journal of Educational Technology*. *International Educational Technology Conference and International Teacher Education Conference* (pp. 942–947).

Buchem, I., & Hamelmann, H. (2010). Microlearning: a strategy for ongoing professional development. *eLearning Papers, 21*(7), 1–15.

Collins, L. M., Murphy, S. A., & Strecher, V. (2007). The multiphase optimization strategy and the sequential multiple assignment randomized trial: New methods for more potent eHealth interventions. *American journal of preventive medicine*, *32*(5), S112-S118.

Collins, L. M., Nahum-Shani, I., & Almirall, D. (2014). Optimization of behavioral dynamic treatment regimens based on the sequential, multiple assignment, randomized trial. *Clinical Trials*, *11*(4), 426-434.

Chaplot, D.S., Rhim, E., & Kim, J. (2016). Personalized adaptive learning using neural networks. *Proceedings of 3rd ACM Conference on Learning @ Scale, UK*, 165–168.https://doi.org/10.1145/2876034.2893397

Chisholm, L. (2005). Micro-learning in the lifelong learning context (foreword). In T. Hug, M. Lindnem, & P. Bruck (Eds.), Microlearning: Emerging concepts, practices and technologies after e-Learning: Microlearning 2005 Learning & Working in New Media (pp. 5–6). Innsbruck.

Crow, D. (2020, March 13). *Why every child should learn to code*. The Guardian. https://www.theguardian.com/technology/2014/feb/07/year-of-code-dan-crow-songkick

Raadt, M. D. (2008). *Teaching programming strategies explicitly to novice programmers.* [Doctoral dissertation, University of Southern Queensland]. USQ ePrints. https://eprints.usq.edu.au/4827

Evans, D. (2011). *Programming*. In D. Evans (Ed.), *CS 1120*: *Introduction to computing (pp.35–52)*. University of Virginia.

Gabrielli, S., Kimani, S., & Catarci, T. (2006, June 23–24). The design of microlearning experiences: A research agenda [Paper presentation]. Microlearning 2005 conference, Innsbruck, Austria.

Green, D. P., (2020, April 10). *10 things to know about adaptive experimental design*. https://egap.org/methods-guides/10-things-adaptiveexperiments

Hug, T. (Ed.). (2007). *Didactics of microlearning concepts, discourses and examples.* Waxmann.

Jacobson, L. (2020, March 13). *Never too young to code*. School Library Journal. http://www.slj.com/2016/04/technology/never-too-young-tocode/#_

Jomah, O., Masoud, A. K., Kishore, X. P., & Aurelia, S. (2016). Microlearning: A modernized education system. *Broad Research in Artificial Intelligence and Neuroscience*, *7*(1), 103–110.

Kalelioglu, F. (2015) A new way of teaching programming skills to K–12 students. *Computers in Human Behavior, 52,* 200–210.https://doi.org/10.1016/j.chb.2015.05.047

Kamilali, D., & Sofianopoulou, C. (2015). Microlearning as innovative pedagogy for mobile learning in MOOCs. *Proceeding of International Association for Development of the Information Society, Portugal*, 127–131

Kapp, K. M., & Defelice, R. A. (2019). *Microlearning: Short and Sweet*. American Society for Training and Development.

Kelleher, C., & Pausch, R. (2005). Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. *ACM Computing Surveys, 37*(2), 83–137. https://doi.org/10.1145/1089733.1089734

Kerres, M. (2007). Microlearning as a Challenge for Instructional Design. In T. Hug (Ed.) *Didactics of microlearning concepts, discourses and examples* (pp. 98–109), Waxmann.

Koulouri, T., Lauria, S., & Macredie, R. D. (2014). Teaching introductory programming: A quantitative evaluation of different approaches. *ACM Transaction on Computing Education. Edu. 14*(4). 1–28. https://doi.org/10.1145/2662412

Langreiter, C., & Bolka, A. (2005). *Snips & spaces: Managing microlearning*. Na.

Li, X., & Prasad, C. (2005). Effectively teaching coding standards in programming.*Proceedings of the 6th conference on Information technology education*, USA,239-244. https://doi.org/10.1145/1095714.1095770

Lindner, M. (2006). Use these tools, your mind will follow: Learning in immersive micromedia and microknowledge environments. *proceedings of the 13th ALT-C conference, UK*. 41–49.

Mayer R.E. (1992) Teaching for transfer of problem-solving skills to computer programming. In E. De Corte, M.C. Linn, H. Mandl, & L. Verschaffel (Eds.), *NATOASI Series: Vol. 84. Computer-Based Learning Environments and Problem Solving* (pp. 193-206). Springer. https://doi.org/10.1007/978-3-642-77228-3_9

McGill, T. J., & Volet, S. E. (1997). A conceptual framework for analyzing students' knowledge of programming. *Journal of research on Computing in Education*, *29*(3), 276-297. https://doi.org/10.1080/08886504.1997.10782199

Meerbaum-Salant, O., Armoni, M., & Ben-Ari, M. (2010). Learning computer science concepts with scratch. *Proceedings of the 6th international workshop on Computing education research*, *Denmark*, *23*(3), 69–76. https://doi.org/10.1080/08993408.2013.832022

Nguyen, H. D., Pham, V. T., Tran, D. A., & Le, T. T. (2019). Intelligent tutoring chatbot for solving mathematical problems in high-school. *Proceedings of 1 1 th International Conference on Knowledge and Systems Engineering (KSE), Vietnam,* 1-6. https://doi.org/10.1109/KSE.2019.8919396

Rodrigo, M. J. (2016). Quality of implementation in evidence-based positive parenting programs in Spain. *Psychosocial Intervention, 25*(2), 63-68. https://doi.org/ 10.1016/j.psi.2016.02.004

Park, Y., & Kim, Y. (2018). A design and development of micro-learning content in e learning system. *International Journal on Advanced Science, Engineering and Information Technology*, *8*(1), 56-61. https://doi.org/10.18517/ijaseit.8.1.2698

Robins, A., Rountree, J., & Rountree, N. (2003). Learning and teaching programming: A review and discussion. *Computer science education, (13)*137-172. https://doi.org/10.1076/csed.13.2.137.14200

Rushkoff, D. (2010). *Program or be programmed: Ten commands for a digital age*. ORBooks.

Sanz, A. (2020, March 13). *Why teaching and learning how to code in schools.* EdTech Review. http://edtechreview.in/trends-insights/insights/1934why-teaching-and-learning-how-to-code-in-schools

Schmidt, A. (2007). *Microlearning and the knowledge maturing process: Towards conceptual foundations for work-integrated microlearning support.* Na.

Skalka J., & Drlík, M. (2018). Conceptual framework of microlearning-based training mobile application for improving programming skills. In M. Auer, & T. Tsiatsos (Eds.), *Advances in intelligent systems and computing: Vol. 725. Interactive mobile communication technologies and learning* (pp. 213–224). Springer. https://doi.org/10.1007/978-3-319-75175-7_22

Soloway, E., & Spohrer, J. (2013). *Studying the novice programmer*. Psychology Press.

Somyurek, S. (2009). Student modeling: Recognizing the individual needs of users in e-learning environments. *International Journal of Human Sciences, 6*(2), 429–450.

Su, X., Zhu, G., Liu, X., & Yuan, W. (2005). Presentation of programming domain knowledge with ontology. *Proceeding of 2005 First International Conference on Semantics, Knowledge and Grid*, Shanghai, 131–132.

Thalmann, S. (2014). Adaptation criteria for the personalised delivery of learning materials: A multi-stage empirical investigation. *Australasian Journal of Educational Technology, 30*(1), 45-60. https://doi.org/10.14742/ajet.235

Winslow, L.E. (1996). Programming pedagogy: A psychological overview. *ACM SIGCSE Bull*. *28*(3), 17–22.

Yang, T. C., Hwang, G. J., & Yang, S. J. H. (2013). Development of an adaptive learning system with multiple perspectives based on students' learning styles and cognitive styles. *Journal of Educational Technology & Society*, *16*(4), 185-200.

Zyl, V. M., Wong, F., Guerrero, A. G., Duffy, M. (Eds.). (2020). *Beginner's step-by-step coding course*. Dorling Kindersley.

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ภาคผนวก

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ภาคผนวก ก
รายชื่อผู้ทรงคุณวุฒิในการตรวจสอบคุณภาพเครื่องมือวิจัย

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

รายชื่อผู้ทรงคุณวุฒิในการตรวจสอบคุณภาพเครื่องมือวิจัย

รายชื่อผู้ทรงคุณวุฒิ	**หน่วยงาน/สถาบัน**
รองศาสตราจารย์ ดร.ชูพันธุ์ รัตนโภคา	ภาควิชาเทคโนโลยีวิศวกรรมอิเล็กทรอนิกส์ วิทยาลัยเทคโนโลยีอุตสาหกรรม มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ
ผู้ช่วยศาสตราจารย์ ดร.กนิษฐ์ ศรีเคลือบ	สาขาวิชาวิธีวิทยาการวิจัยการศึกษา คณะครุศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ผู้ช่วยศาสตราจารย์ ดร.ณัฐพล แจ้งอักษร	ภาควิชาพื้นฐานและการพัฒนาการศึกษา คณะศึกษาศาสตร์ มหาวิทยาลัยเชียงใหม่

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ภาคผนวก ข

เครื่องมือที่ใช้ในการวิจัย

- แบบสำรวจเกี่ยวกับการเขียนโค้ดสำหรับนิสิตระดับปริญญาบัณฑิต คณะครุศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
- แบบทดสอบความสามารถด้านการเขียนโค้ดหลังเรียน

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

แบบสำรวจเกี่ยวกับการเขียนโค้ดสำหรับนิสิตระดับปริญญาบัณฑิต คณะครุศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ส่วนที่ 1 ข้อมูลพื้นฐาน

คำชี้แจง โปรดทำเครื่องหมาย ✓ ลงในข้อที่ตรงกับท่านมากที่สุด

1. เพศ ☐ ชาย ☐ หญิง

2. ชั้นปีที่ ☐ 1 ☐ 2 ☐ 3 ☐ 4

3. สาขาวิชา

☐ 1. การศึกษาปฐมวัย ☐ 6. ศิลปศึกษา

☐ 2. ประถมศึกษา ☐ 7. ดนตรีศึกษา

☐ 3. มัธยมศึกษา ☐ 8. การศึกษานอกระบบโรงเรียน

☐ 4. สุขศึกษาและพลศึกษา ☐ 9. ธุรกิจศึกษา

☐ 5. เทคโนโลยีการศึกษา ☐ 10. จิตวิทยาการปรึกษา แนะแนว การศึกษาพิเศษ

4. วิชาเอก (ตอบเฉพาะสาขาวิชาที่ 3, 4, 5, 7, 10)

☐ 1. ภาษาไทย ☐ 10. วิทยาศาสตร์ทั่วไป ☐ 19. การศึกษาพิเศษ

☐ 2. ภาษาอังกฤษ ☐ 11. คณิตศาสตร์ ☐ 20. จิตวิทยาการปรึกษาและการแนะแนว

☐ 3. ภาษาจีน ☐ 12. พลศึกษา

☐ 4. ภาษาฝรั่งเศส ☐ 13. สุขศึกษา

☐ 5. ภาษาเยอรมัน ☐ 14. เทคโนโลยีการศึกษา

☐ 6. สังคมศึกษา ☐ 15. คอมพิวเตอร์ศึกษา

☐ 7. ฟิสิกส์ ☐ 16. ดนตรีศึกษา

☐ 8. เคมี ☐ 17. การสอนดนตรีไทย

☐ 9. ชีววิทยา ☐ 18. การสอนดนตรีสากล

5. ท่านชอบศึกษาหาความรู้แบบใด

☐ ศึกษาเรื่องต่าง ๆ ด้วยตนเอง ☐ ศึกษาหาความรู้ร่วมกับเพื่อน ๆ

6. ท่านชอบการเรียนโดยมีระยะเวลาในการเรียนแบบใด

☐ 5-10 นาที ☐ 10-15 นาที ☐ 15-20 นาที

7. ท่านชอบสื่อการเรียนรู้แบบใด

☐ เอกสารที่มีคำอธิบาย ข้อความ ☐ ภาพเคลื่อนไหว วีดิโอ

8. ท่านชอบลำดับเนื้อหาในการเรียนแบบใด

☐ เลือกเรียนเนื้อหาด้วยตนเอง ☐ เรียนเนื้อหาตามที่มีการจัดลำดับไว้ให้แล้ว

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ส่วนที่ 2 ข้อมูลเกี่ยวกับการเขียนโค้ด

คำชี้แจง โปรดทำเครื่องหมาย ✓ ลงในข้อที่ตรงกับท่านมากที่สุด

นิสิตเคยเรียนเขียนโปรแกรมมาก่อนหรือไม่

☐ ไม่เคย

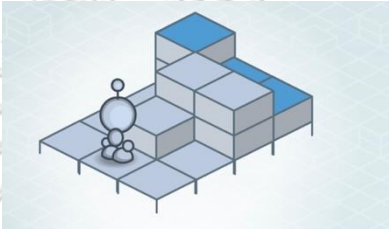
☐ เคย ได้แก่ ☐ C ☐ C# ☐ C++ ☐ Java

☐ JavaScript ☐ Python ☐ PHP ☐ Ruby

☐ R ☐ อื่น ๆ โปรดระบุ.....

คำชี้แจง โปรดทำเครื่องหมาย ✕ ลงหน้าข้อที่ถูกต้องมากที่สุด

1.



จากภาพข้างต้น จงเลือกคำสั่งจากข้อ 1-5 (ใช้ซ้ำได้)เพื่อแสดงขั้นตอนให้หุ่นยนต์ไปเปิดไฟในตำแหน่งที่กำหนด

1. เดินไปด้านหน้าของหุ่นยนต์
2. หันไปทางซ้ายของหุ่นยนต์
3. หันไปทางขวาของหุ่นยนต์
4. กระโดดขึ้น/ลงไปข้างหน้า (ถ้าพื้นด้านหน้าอยู่ระดับเดียวกันหุ่นยนต์จะกระโดดอยู่กับที่)
5. เปิดไฟที่พื้น

1) จงเขียนอธิบายขั้นตอนในการทำงานของหุ่นยนต์อย่างละเอียด สามารถเขียนเป็นภาษาของท่านเองได้ (เช่น กระโดนขึ้นไปด้านหน้า หันขวา กระโดดลง เดินไปด้านหน้า เปิดไฟ)

.....

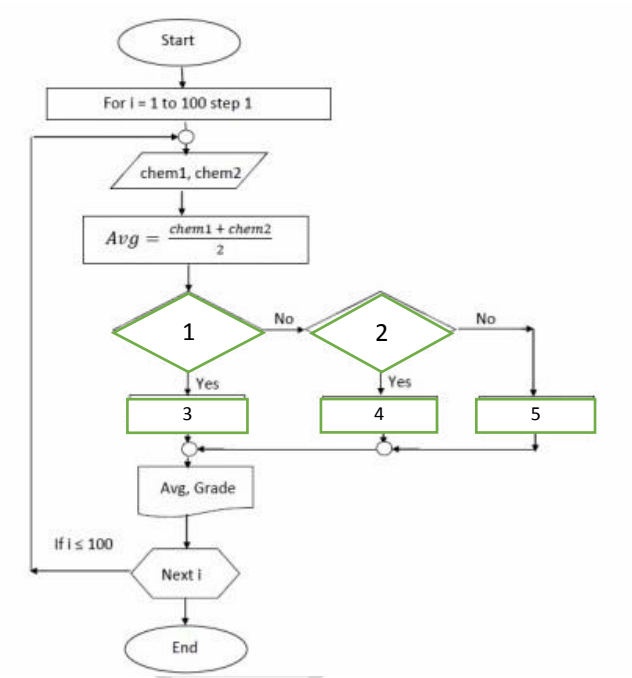
.....

2) จงเรียงลำดับโดยใช้ตัวเลขแทนคำสั่ง (เช่น เริ่ม 1 2 3 4 1 4 1 5 จบ เป็นต้น)

.....

.....

2. Flowchart ต่อไปนี้แสดงขั้นตอนวิธีในการแสดงเกรดของนักเรียนจำนวน 100 คน โดย เกณฑ์ในการให้คะแนนคือ คะแนนเฉลี่ย (Avg) ที่ได้มีค่ามากกว่าหรือเท่ากับ 70 จะได้ A ถ้าต่ำกว่า 70 แต่มากกว่าหรือเท่ากับ 40 ได้ B และถ้าต่ำกว่า 40 จะได้ C ข้อใดต่อไปนี้ถูกต้อง



ก. หมายเลข 1 คือ if avg < 40
หมายเลข 2 คือ if avg ≥ 70
หมายเลข 5 คือ Grade = “A”

ข. หมายเลข 1 คือ if avg ≥ 70
หมายเลข 2 คือ if avg > 40
หมายเลข 4 คือ Grade = “B”

ค. หมายเลข 1 คือ if avg < 40
หมายเลข 2 คือ if avg < 70
หมายเลข 5 คือ Grade = “B”

ง. หมายเลข 1 คือ if avg ≥ 70
หมายเลข 2 คือ if avg ≥ 40
หมายเลข 4 คือ Grade = “B”

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

3. หากนิสิตได้รับมอบหมายให้คัดเลือกนักเรียนที่สมัครเข้าร่วมค่ายแนะแนวการศึกษาของคณะครุศาสตร์ นิสิตจะมีขั้นตอน เกณฑ์ในการคัดเลือก/จำแนกนักเรียนออกเป็นกลุ่มที่ได้รับคัดเลือกและกลุ่มที่ไม่ได้รับคัดเลือกอย่างไร โดยในใบสมัครมีข้อมูลดังนี้

ใบสมัคร

ข้อมูลส่วนตัว

ชื่อ.....นามสกุล.....

โรงเรียน.....ห้อง.....เกรดเฉลี่ย.....

อาชีพที่ใฝ่ฝัน.....วิชาที่ชอบ.....

ที่อยู่ปัจจุบัน.....

คำถาม

1. น้อง ๆ อยากเข้าค่ายแนะแนวการศึกษาในครั้งนี้เพราะอะไร

.....

.....

2. หากน้อง ๆ ได้รับการคัดเลือกแต่เพื่อนสนิทไม่ได้รับการคัดเลือก น้องจะยังมาเข้าร่วมค่ายหรือไม่ เพราะอะไร

.....

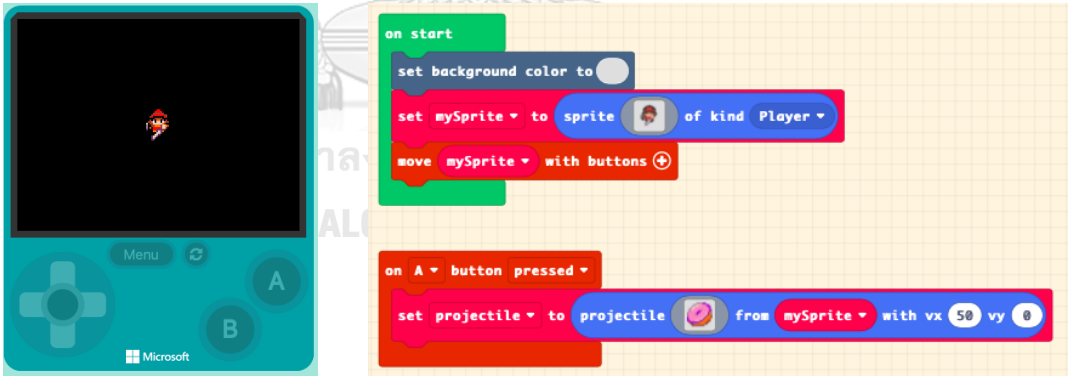
.....

จงเขียนแสดงขั้นตอนเป็นผังงานหรือขั้นตอนวิธี

.....

.....

4.



จงอธิบายการทำงานของบล็อกคำสั่งด้านขวาของภาพให้ได้มากที่สุด

บล็อกที่ 1

.....

บล็อกที่ 2

.....

5. จงเติมคำในช่องว่างให้ถูกต้อง ภาษาที่ใช้เป็นภาษา Python

input: รับจำนวนจริง 1 จำนวนจากแป้นพิมพ์ เก็บใน a

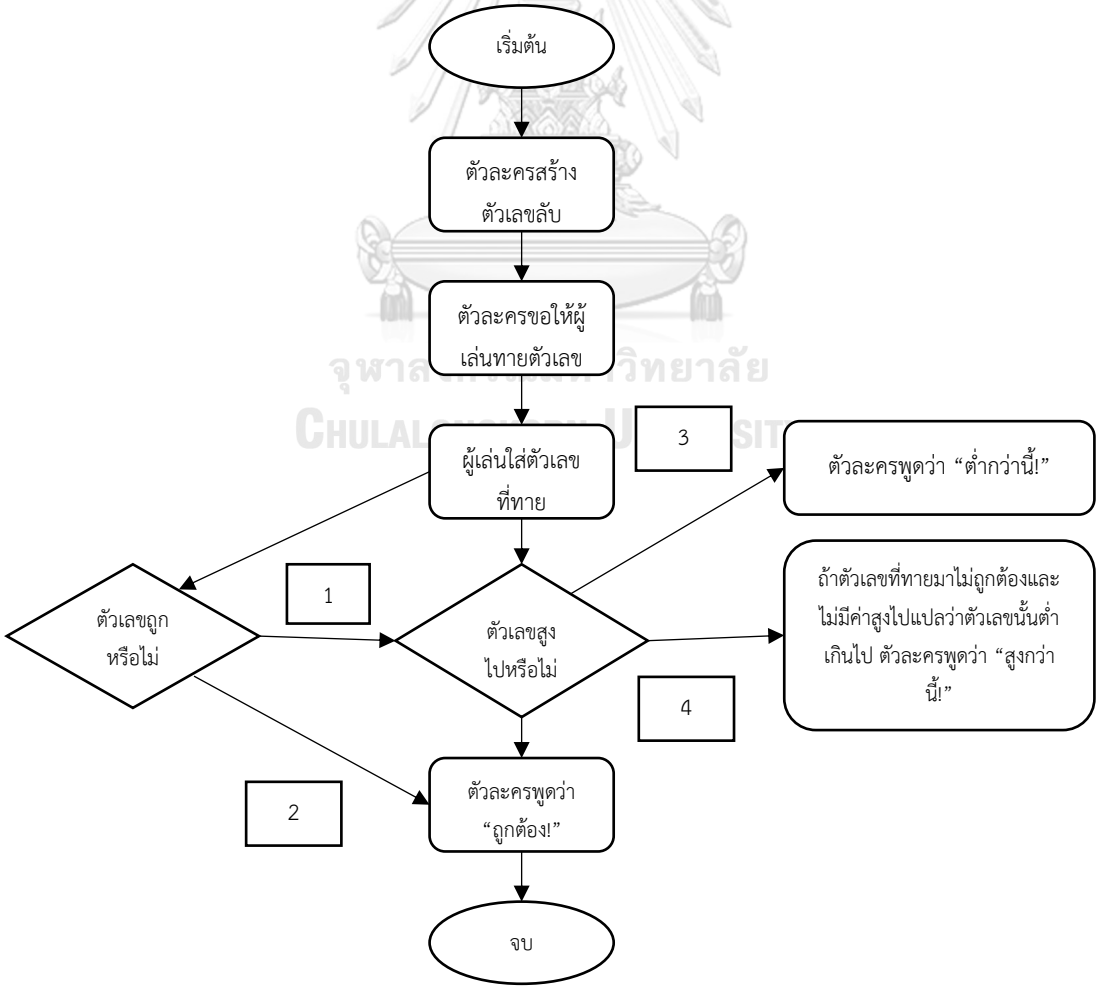
process: ให้ x มีค่าเป็น 1 จากนั้นทำคำสั่ง x=(x+a/x)/2 จำนวน 4 ครั้ง

output: ค่า x ที่ได้จากการทำงาน

คำตอบ

แบบทดสอบความสามารถด้านการเขียนโค้ดหลังเรียน

จากผังงานต่อไปนี้จงตอบคำถามข้อที่ 1 – 2



1. การทำงานมีข้อผิดพลาดตรงส่วนใด และสามารถแก้ไขได้อย่างไร จงอธิบาย

.....

.....

2. พิจารณาเส้นทางการทำงาน 1 – 4 เงื่อนไขที่ทำให้คำสั่งในผังงานทำงานได้อย่างถูกต้องคือข้อใด ตามลำดับ

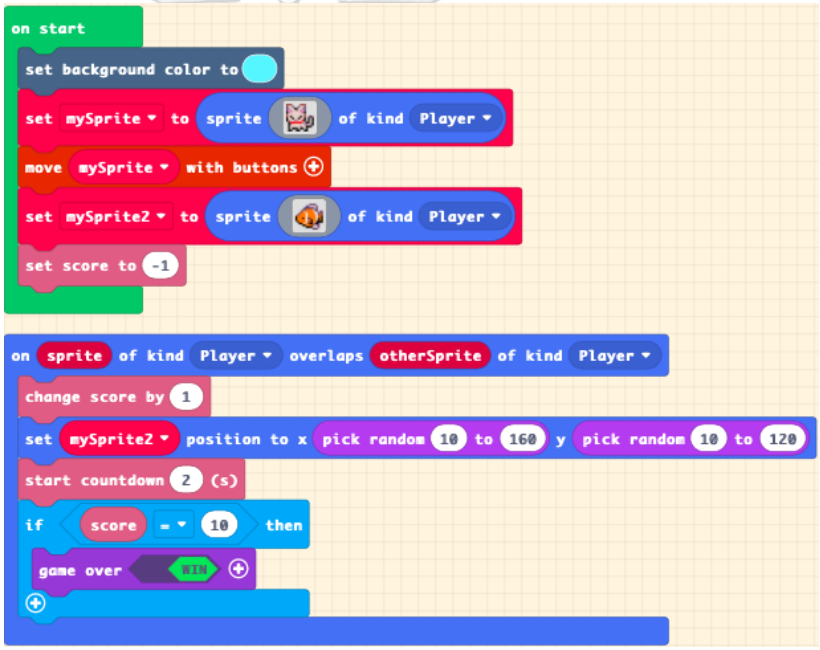
ก. ใช่ ไม่ ใช่ ไม่

ข. ไม่ ใช่ ไม่ ใช่

ค. ใช่ ไม่ ไม่ ใช่

ง. ไม่ ใช่ ใช่ ไม่

จากบล็อกคำสั่งต่อไปนี้จงตอบคำถามข้อที่ 3 – 5



3. การทำงานของเกม เป็นดังนี้

i. แสดงตัวละครแมวและปลาอย่างสุ่มบนจอ

ii. ใช้ปุ่มบังคับทิศทางในการเลื่อนแมว

iii. หากแมวและปลาสัมผัสกันตัวละครทั้งสองจะถูกสุ่มเกิดใหม่บนจอ

จากคำกล่าวข้างต้นประโยคใดบ้างตรงกับคำสั่งบล็อกที่ระบุไว้

ก. i

ข. ii*

ค. i,iii

ง. I,ii,iii

4. คำสั่งข้างต้นเขียนขึ้นได้เหมาะสมกับกฎการแพ้ ชนะเกมในข้อใดมากที่สุด

ก. ชนะ เมื่อ ผู้เล่นบังคับให้ตัวละครแมวชนกับตัวละครปลาทั้งหมด 10 ครั้ง ภายในเวลา 20 วินาที
ข. ชนะ เมื่อ ผู้เล่นบังคับให้ตัวละครแมวชนตัวละครปลาภายในเวลา 2 วินาที ทั้งหมด 10 ครั้ง *
ค. แพ้ เมื่อ ผู้เล่นไม่สามารถบังคับให้ตัวละครแมวชนกับปลา10 ครั้ง ภายในเวลา 20 วินาที
ง. แพ้ เมื่อ ผู้เล่นไม่สามารถบังคับให้ตัวละครแมวชนกับปลาภายในเวลา 2 วินาที

5. จงสร้างฟังก์ชันที่สามารถคำนวณการหาผลบวกของจำนวนนับที่ติดกันตั้งแต่ 1 ถึง n มาอย่างน้อย 2 ฟังก์ชัน พร้อมทั้งอธิบายแนวคิดในการสร้างฟังก์ชันนั้น ๆ (สร้างสรรค์)

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ภาคผนวก ค

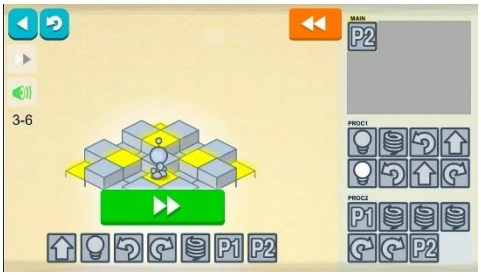
ผลงานของกลุ่มทดลองในการเรียนบทเรียนออนไลน์ด้านการเขียนโค้ด

- ตัวอย่างกิจกรรมเกม lightbot hour
- ตัวอย่างกิจกรรมการสร้างตัวละครด้วยภาษา Scratch บน Microsoft MakeCode Arcade
- ตัวอย่างกิจกรรมการสร้างเกมด้วยภาษา Scratch บน Microsoft MakeCode Arcade

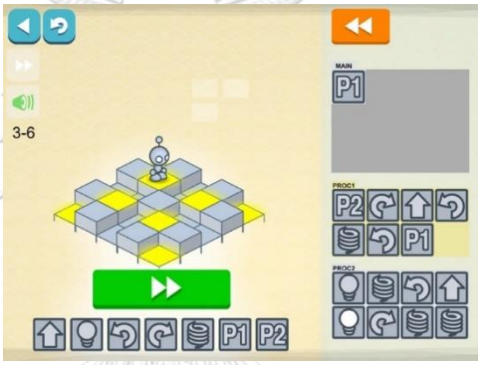
จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ตัวอย่างกิจกรรมเกม lightbot hour

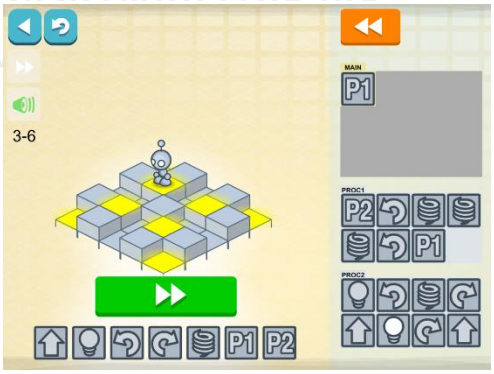
ภาพของขั้นตอนวิธีในการพิชิตด่านสุดท้ายในเกม lightbot hour ที่กลุ่มทดลองได้ทำไว้



รูปแบบที่ 1



รูปแบบที่ 2

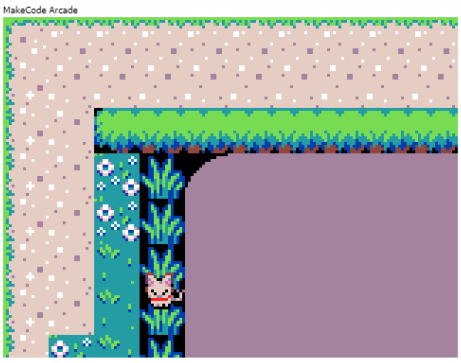


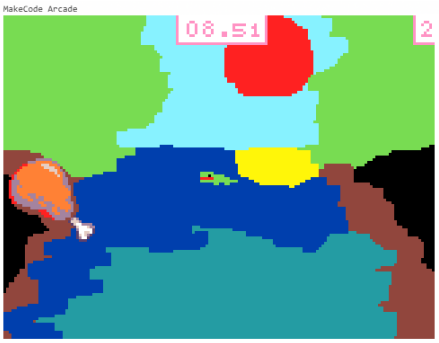
รูปแบบที่ 3

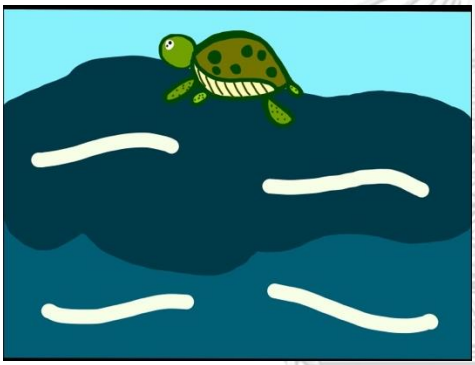
ตัวอย่างกิจกรรมการสร้างตัวละครด้วยภาษา Scratch

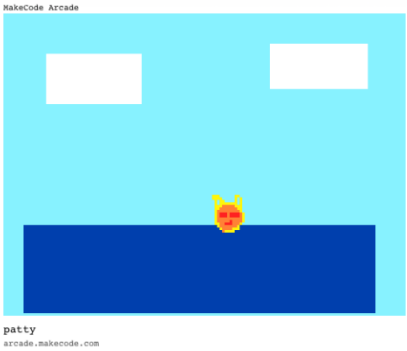
บน Microsoft MakeCode Arcade

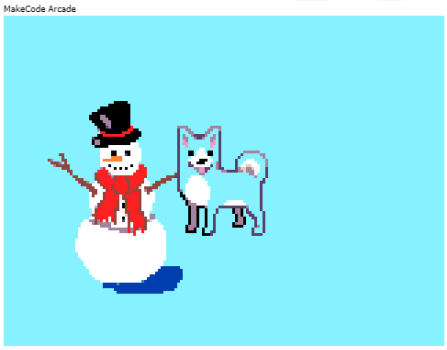
กิจกรรมนี้เป็นการฝึกใช้เมนูบล็อกเพื่อสร้างตัวละครและพื้นหลังตามเงื่อนไขที่กำหนดไว้

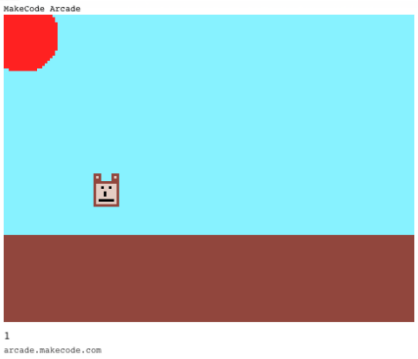








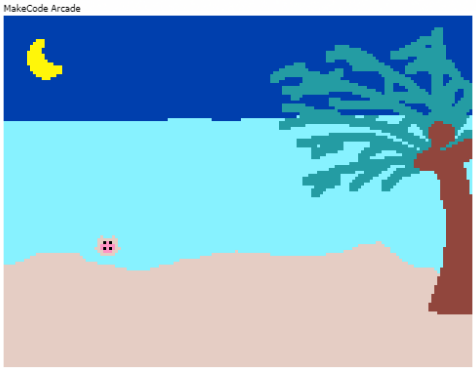




MakeCode Arcade

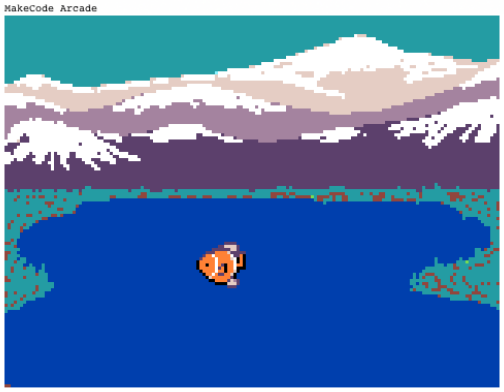
Kong
arcade.makecode.com

MakeCode Arcade



mini pj
arcade.makecode.com

MakeCode Arcade

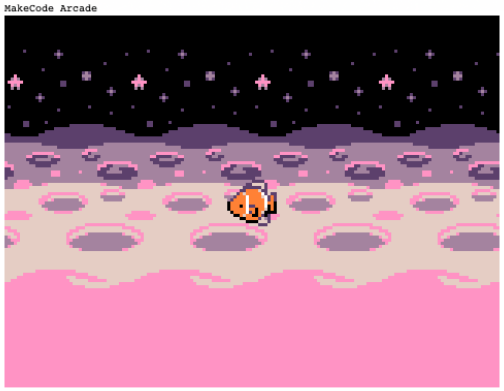


test
arcade.makecode.com

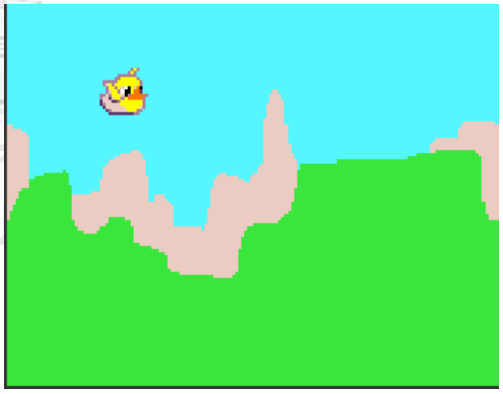
MakeCode Arcade

My Gane
arcade.makecode.com

MakeCode Arcade



best
arcade.makecode.com



MakeCode Arcade

Untitled
arcade.makecode.com

MakeCode Arcade

Untitled
arcade.makecode.com

ตัวอย่างกิจกรรมการสร้างเกมด้วยภาษา Scratch บน Microsoft MakeCode Arcade

กิจกรรมนี้เป็นการฝึกใช้เมนูบล็อกเพื่อสร้างเกมตามเงื่อนไขที่กำหนดไว้ ผลงานของกลุ่มทดลองต่อไปนี้จะเป็นการแสดง หน้าต่างเกม และโค้ดภาษา Scratch บท Microsoft MakeCode Arcade(E, 2017)

ลิงค์การเข้าถึง https://makecode.com/_iuu8ihRR45er

ลิงค์การเข้าถึง https://makecode.com/_EVM0M0cqcEbt

ลิงค์การเข้าถึง https://makecode.com/_5ka4rjUW1R9D

ประวัติผู้เขียน

ชื่อ-สกุล	นายกฤษณ์ วิริยะสิทธารถ
วัน เดือน ปี เกิด	11 มีนาคม 2539
สถานที่เกิด	จันทบุรี
วุฒิการศึกษา	ครุศาสตรบัณฑิต (เกียรตินิยมอันดับหนึ่ง) สาขามัธยมศึกษา (วิทยาศาสตร์) จุฬาลงกรณ์มหาวิทยาลัย
ที่อยู่ปัจจุบัน	84/5 หมู่ 10 ต.ท่าช้าง อ.เมืองจันทบุรี จ.จันทบุรี 22000

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY