EDUPSU Journal of Education, Prince of Songkla University, Pattani Campus
Vol.31 No.3 : September - December 2020, pp. 1-14

Pedagogical Content Knowledge (PCK) for Abstraction of Computational Thinking

Artorn Nokkaew
Ph.D. (Science and Technology Education), Lecturer
Department of Education, Faculty of Education, Naresuan University
Suparat Chuechote
M.Sc. (Applied Mathematics), Lecturer
Department of Education, Faculty of Education, Naresuan University

Received: January 19, 2020/ **Revised:** August 7, 2020/ **Accepted:** September 3, 2020

Abstract

. As we entered the age of technological disruption, teaching in schools has also evolved. There comes the scenario that teachers must teach things that they have no experience. And, computational thinking is one such thing. Computational thinking is not a new skill, yet has been in educational focus since past ten years. Thinking about the computational thinking skill in the past, we mostly associated this with people who work with computers or software development. However, as teachers become aware that this is the fundamental skill for students since primary levels, the challenge in teaching this skill has arisen; in particular, the cultivation of abstraction, which is considered to be the core part of computational thinking. To develop abstraction, teachers must considerately choose learning tools or activities to create learning and thinking experience. To assist teachers with the goal for computational thinking enhancement, this article presents the pedagogical content knowledge for abstraction of computational thinking, which links learning theory to pedagogical framework to promote computational thinking.

Keywords: Abstraction, Computational Thinking, Pedagogical Content Knowledge, Neo-Piagetian, Boolean Logic

EDUPSU วารสารศึกษาศาสตร์ มหาวิทยาลัยสงขลานครินทร์ วิทยาเขตปัตตานี
ปีที่ 31 ฉบับที่ 3 เดือนกันยายน – ธันวาคม 2563, หน้า 1-14

ความรู้เนื้อหาผสานวิธีสอนการคิดเชิงนามธรรม สำหรับการคิดเชิงคำนวณ

อาทร นกแก้ว
ปร.ด. (วิทยาศาสตร์และเทคโนโลยีศึกษา), อาจารย์
ภาควิชาการศึกษา คณะศึกษาศาสตร์ มหาวิทยาลัยนเรศวร
สุภารัตน์ เชื้อโชติ
M.Sc. (Applied Mathematics), อาจารย์
ภาควิชาการศึกษา คณะศึกษาศาสตร์ มหาวิทยาลัยนเรศวร

วันรับบทความ: 19 มกราคม 2563/ **วันแก้ไขบทความ:** 7 สิงหาคม 2563/ **วันตอบรับบทความ:** 3 กันยายน 2563

บทคัดย่อ

นับเป็นยุคใหม่ของการจัดการเรียนรู้ ที่ครูต้องสอนในสิ่งที่ไม่เคยเรียนและไม่เคยมีประสบการณ์มาก่อน การคิดเชิงคำนวณไม่ใช่ทักษะใหม่ หากแต่เพิ่งได้รับความสนใจในแวดวงการศึกษาในช่วงสิบกว่าปีที่ผ่านมา ในอดีตการคิดเชิงคำนวณเป็นทักษะเฉพาะของคนในแวดวงที่ทำงานเกี่ยวกับการใช้คอมพิวเตอร์ในการแก้ปัญหาต่าง ๆ แต่ในปัจจุบันทักษะนี้นับว่าเป็นทักษะพื้นฐานสำหรับนักเรียนทุกคนตั้งแต่ระดับประถมศึกษา การเปลี่ยนแปลงนี้นำมาซึ่งความท้าทายในการสอนของครู โดยเฉพาะอย่างยิ่งการสอนการคิดเชิงนามธรรมซึ่งเป็นหัวใจของทักษะการคิดเชิงคำนวณ การพัฒนาการคิดเชิงนามธรรมนั้นต้องอาศัยการฝึกฝนผ่านประสบการณ์การเรียนรู้ที่หลากหลาย ครูเป็นส่วนสำคัญที่ช่วยพัฒนาทักษะดังกล่าวให้เกิดขึ้นกับผู้เรียน ครูต้องใช้วิจารณญาณในการเลือกสื่อและกิจกรรมที่มีอยู่มากมายอย่างเหมาะสมเพื่อสร้างสิ่งแวดล้อมในการเรียนรู้ที่ส่งเสริมการคิดเชิงนามธรรม บทความนี้นำเสนอความรู้เนื้อหาผสานวิธีสอนสำหรับครูในการพัฒนาการคิดเชิงนามธรรมสำหรับการคิดเชิงคำนวณ ซึ่งได้บูรณาการการใช้ทฤษฎีการเรียนรู้กับความเข้าใจทางวิทยาการคำนวณเพื่อสังเคราะห์แนวทางปฏิบัติในการจัดการเรียนรู้ ซึ่งจะช่วยครูสามารถจัดการเรียนรู้โดยมีเป้าหมายที่ชัดเจนและนำไปสู่การพัฒนาทักษะการคิดเชิงนามธรรมสำหรับการคิดเชิงคำนวณ

คำสำคัญ: การคิดเชิงนามธรรม, การคิดเชิงคำนวณ, ความรู้เนื้อหาผสานวิธีสอน, นีโอเพียเจต์, ตรรกศาสตร์บูลลีน

บทนำ

"the mind is helpless without some mechanical rule."

กล่าวโดยฟรานซิส เบคอน (Jevons, 1870)

ความสามารถในการพัฒนากลไกทางความคิดทำให้มนุษย์สามารถแก้ปัญหาต่าง ๆ ได้ นอกจากนี้ มนุษย์ยังสามารถพัฒนากลไกทางความคิดไปสู่การใช้สิ่งต่าง ๆ เป็นเครื่องมือเพื่ออำนวยความสะดวกในการคิดหรือแก้ปัญหา เช่น การใช้ก้อนหินเพื่อช่วยนับจำนวน รวมไปถึงการพัฒนาไปสู่การสร้างเครื่องมือสำหรับช่วยในการคำนวณ เช่น ลูกคิด (ในช่วง 2,500 ปีก่อนคริสตกาล) และ กระดูกนาเปียร์ (ช่วงปี ค.ศ. 1617) เป็นต้น นับตั้งแต่ศตวรรษที่ 19 มนุษย์สามารถถ่ายทอดความคิดของมนุษย์ให้กับเครื่องจักรที่ไม่มีชีวิตและความคิดได้ และใช้มันเป็นตัวแทนในการแก้ปัญหา ตัวอย่างที่โดดเด่นก็คือการทำงานของเครื่องจักรในอุตสาหกรรมสิ่งทอ Jacquard machine (ในปี ค.ศ. 1804) ซึ่งเป็นเครื่องจักรทอผ้า เครื่องจักรนี้ใช้บัตรเจาะรูควบคุมลำดับของการดำเนินการทำให้เกิดลวดลายของผ้า และสามารถเปลี่ยนลายผ้าได้โดยการเปลี่ยนบัตรเจาะรู ซึ่งสิ่งนี้ถือเป็นการพัฒนาก้าวสำคัญของเครื่องจักรที่สามารถทำงานตามชุดคำสั่งและเปลี่ยนชุดคำสั่งควบคุมได้ และยังเป็นต้นแบบในการพัฒนาเครื่องคอมพิวเตอร์ในยุคต่อ ๆ มา การเปลี่ยนแปลงบทบาทของเทคโนโลยีจากเครื่องอำนวยความสะดวกมาเป็นเครื่องจักรที่ทำงานแทนมนุษย์นี้ ทำให้มนุษย์ต้องพัฒนาทักษะการคิดที่มีลักษณะแตกต่างจากทักษะการคิดในรูปแบบเดิม และการคิดในรูปแบบใหม่นี้เรียกว่า *การคิดเชิงคำนวณ*

ในอดีตทักษะการคิดเชิงคำนวณเป็นทักษะเฉพาะสำหรับคนที่อยู่ในแวดวงที่มีการใช้คอมพิวเตอร์ช่วยแก้ปัญหาที่ซับซ้อน แต่ในปัจจุบันเมื่อคอมพิวเตอร์เข้ามามีบทบาทมากขึ้น แม้ว่ามนุษย์จะไม่ได้ใช้คอมพิวเตอร์ในการแก้ปัญหาโดยตรง แต่เราก็ได้รับผลกระทบจากโลกที่มีคอมพิวเตอร์เข้ามาเกี่ยวข้องในทุก ๆ ส่วนของชีวิต ทักษะการคิดเชิงคำนวณจึงได้รับการให้ความสำคัญในฐานะทักษะพื้นฐานของทุกคน และถูกบรรจุให้อยู่ในหลักสูตรการเรียนรู้ในหลายประเทศ และยังเป็นทักษะที่องค์กรเพื่อความร่วมมือทางเศรษฐกิจและการพัฒนา หรือ OECD จัดให้มีการประเมินในการทดสอบ PISA 2021 ในหมวดการรู้เรื่องคณิตศาสตร์ แม้ว่านักวิชาการหลายคนจะเห็นพ้องถึงความสำคัญของทักษะนี้ แต่ยังมีความเห็นที่แตกต่างหลากหลายเกี่ยวกับการให้คำจำกัดความ รวมถึงบทบาทและความสำคัญของการคิดเชิงคำนวณ (Denning, 2017) ความไม่ชัดเจนนี้อาจส่งผลต่อการจัดการเรียนการสอน โดยเฉพาะอย่างยิ่งเมื่อครูส่วนใหญ่ไม่เคยมีประสบการณ์ในการใช้คอมพิวเตอร์เพื่อแก้ปัญหาที่มีความซับซ้อน ความไม่ชัดเจนในคำนิยามและเป้าหมายอาจจะทำให้การปฏิบัติเกิดความคลาดเคลื่อน ที่ไม่เพียงแต่จะส่งผลต่อความล้มเหลวทางนโยบายการศึกษา การสูญเสียงบประมาณ แต่ยังอาจรวมถึงการที่นักเรียนขาดโอกาสในการพัฒนา เกิดการรับรู้ที่คลาดเคลื่อนต่อทักษะและกระบวนการใหม่ ๆ จึงนับเป็นความท้าทายของการขับเคลื่อนการเรียนการสอนวิทยาการคำนวณในปัจจุบัน เพราะการจัดการเรียนรู้จะมีประสิทธิภาพได้ก็ต่อเมื่อครูสามารถผสานความรู้ในเนื้อหากับความรู้เชิงกระบวนการจัดการเรียนรู้ได้ (Shulman, 1986) บทความนี้จึงนำเสนอกรอบในการจัดการเรียนรู้ที่มุ่งเน้นการพัฒนาการคิดเชิงนามธรรมสำหรับการคิดเชิงคำนวณเพื่อเป็นแนวทางการจัดการเรียนรู้โดยเฉพาะสำหรับครูที่ไม่มีพื้นฐาน นอกจากนี้บทความยังสังเคราะห์แนวทางการพัฒนาความคิดเชิงนามธรรมสำหรับการคิดเชิงคำนวณจากทฤษฎีการเรียนรู้

การคิดเชิงคำนวณ (Computational thinking)

แม้ว่าทักษะการคิดเชิงคำนวณเป็นทักษะที่จำเป็นในแวดวงของคนที่ใช้คอมพิวเตอร์ในการแก้ปัญหามายาวนาน อย่างไรก็ตาม ตามที่ปรากฏเป็นหลักฐานอ้างอิง คำว่า "การคิดเชิงคำนวณ" หรือ "Computational thinking" ถูกใช้ในวงการ

เรียนรู้ครั้งแรกโดย Papert (1980) เพื่ออธิบายการสร้างสรรค์งานในโลกเสมือน (โปรแกรม) ที่จำลองโดยคอมพิวเตอร์ ในโลกจำลองนั้นนักเรียนสามารถควบคุมตัวละครได้โดยใช้คำสั่งที่กำหนดไว้ให้ อย่างไรก็ตาม Papert (1980) ไม่ได้ขยายความของความหมายคำนี้ไว้ ต่อมาคำนี้ได้รับความสนใจและถูกพูดถึงในวงกว้างภายหลังจากที่ Wing (2006) เสนอแนวคิดว่าทักษะการคิดเชิงคำนวณเป็นทักษะยุคใหม่ที่สำคัญสำหรับเด็กทุกคน นอกเหนือจากการอ่าน (Reading) การเขียน (Writing) และ การคำนวณเชิงตัวเลข (Arithmetic) ในปัจจุบันมีการให้คำจำกัดความที่หลากหลายเกี่ยวกับการคิดเชิงคำนวณ ทั้งนี้การอภิปรายคำจำกัดความที่หลากหลายดังกล่าวอยู่นอกขอบเขตของบทความนี้

บทความนี้อ้างอิงคำนิยาม การคิดเชิงคำนวณของศาสตราจารย์ด้านวิทยาการคอมพิวเตอร์จาก Carnegie Mellon University และ Columbia University ซึ่งให้นิยามการคิดเชิงคำนวณไว้ดังนี้

> *Computational thinking is the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively carried out by an information-processing agent. (Wing, 2011)*
>
> *Computational thinking is the thought processes involved in formulating problems so their solutions can be represented as computational steps and algorithms. (Aho, 2012)*

ทั้ง Wing (2011) และ Aho (2012) ให้คำจำกัดความที่สอดคล้องกัน กล่าวคือ การคิดเชิงคำนวณเป็นกระบวนการคิดที่เกี่ยวกับการแปลงปัญหาและการออกแบบวิธีการแก้ปัญหาที่เครื่องจักรสามารถดำเนินการได้ เครื่องจักรในที่นี้หมายความถึงสิ่งที่มีกลไกการทำงานที่แม่นยำและเที่ยงตรง ซึ่งอาจจะเป็นเครื่องจักรที่เป็นนามธรรม เช่น กลไกความคิดเชิงตรรกะของมนุษย์ หรือเครื่องจักรที่จับต้องได้ในเชิงกายภาพที่มีกลไกที่แน่นอนชัดเจน ตามนิยามของ Wing (2011) เครื่องจักร หมายถึงตัวประมวลผลข้อมูล (An information-processing agent) ในบทความนี้จะใช้คำว่า “คอมพิวเตอร์” เป็นตัวแทนของตัวประมวลผลข้อมูลที่สามารถใส่ชุดคำสั่งเพื่อให้ดำเนินการได้ และกระบวนการคิดแก้ปัญหาเพื่อให้คอมพิวเตอร์เป็นตัวแทนในการดำเนินการนี้เองที่ทำให้กระบวนการคิดเชิงคำนวณแตกต่างจากการคิดแบบอื่น ๆ

> *The constraints of the underlying computing device force computer scientists to think computationally, not just mathematically.* (Wing, 2006,)

การคิดเชิงคำนวณ ไม่ใช่การคิดแบบเดียวกับคอมพิวเตอร์ เพราะคอมพิวเตอร์เป็นเครื่องจักรประมวลผลข้อมูลที่ทำงานตามคำสั่ง แต่การคิดเชิงคำนวณเป็นการคิดเพื่อใช้ความสามารถของคอมพิวเตอร์ (ภายใต้ข้อจำกัดทางกายภาพ เช่น มีการรับรู้ที่จำกัด มีความสามารถที่จำกัด มีหน่วยความจำที่จำกัด เป็นต้น) มาดำเนินการแก้ปัญหา ซึ่งการสร้างสรรค์งานเช่นนี้ได้จำเป็นจะต้องผสานความรู้ ทักษะ และลักษณะนิสัย (Brennan & Resnick, 2012; Korkmaz, Çakir, & Özden, 2017) ที่มีความจำเพาะ ดังแสดงในตาราง 1

ตาราง 1

องค์ประกอบของการพัฒนาการคิดเชิงคำนวณ

ความรู้	ทักษะ	ลักษณะนิสัย
● ความรู้เกี่ยวกับระบบเครื่องจักรที่ใช้ input process output กลไกภายในของเครื่องจักร ● การควบคุมโปรแกรม ● โครงสร้างข้อมูล ● รู้จักปัญหา และธรรมชาติของปัญหา เป็นต้น	● การแปลงปัญหา ● การแปลงข้อมูล ● การออกแบบวิธีการแก้ปัญหา ● การเขียนโปรแกรม ● การตรวจสอบและแก้ไขข้อผิดพลาด ● ทักษะการคิดเป็นระบบ ● ทักษะการคิดวิเคราะห์ เป็นต้น	● ความมุ่งมั่นในการทำงานที่ยากให้สำเร็จ ● การพัฒนาวิธีการให้มีประสิทธิภาพ ● ความละเอียดรอบคอบในการตรวจสอบความถูกต้อง ● การเรียนรู้จากข้อผิดพลาด ● การทำงานเป็นทีม เป็นต้น

การคิดเชิงนามธรรม: ห้องเครื่องของการคิดเชิงคำนวณ

คำจำกัดความของการคิดเชิงคำนวณข้างต้นได้นำเสนอแนวคิดที่ว่าความคิดของมนุษย์สามารถถูกแปรรูปให้เป็นกระบวนการทำงานของเครื่องจักรเพื่อให้เครื่องจักรทำงานตามความต้องการมนุษย์ได้ โดยการที่มนุษย์สามารถใส่ความคิดลงไปในเครื่องจักรและทำให้เครื่องจักรทำงานตามที่เราต้องการได้นั้นต้องอาศัยทักษะการคิดในระดับนามธรรม

การคิดเชิงนามธรรม คือ กระบวนการคิดที่ซ่อนส่วนที่ไม่ใช่สาระและให้ความสนใจกับส่วนที่เป็นสาระในการทำงานในแต่ละขั้น (Curzon, Bell, Waite, & Dorling, 2019) นักวิชาการหลายท่านเห็นตรงกันว่า การคิดเชิงนามธรรมเป็นสิ่งที่สำคัญที่สุดของการคิดเชิงคำนวณ (Aho, 2012; Priami, 2007; Wing, 2008, 2011) และยังเป็นเครื่องมือในการจัดการกับปัญหาที่ซับซ้อน เพื่อแตกย่อยปัญหาหารูปแบบ ทำให้อยู่ในรูปทั่วไป ประเมินประสิทธิภาพของลำดับขั้นตอนการดำเนินการ และเขียนให้อยู่ในรูปแบบภาษาคอมพิวเตอร์ที่เครื่องจักรเข้าใจและนำไปดำเนินการตามได้ อย่างไรก็ดี การคิดเชิงนามธรรมในการคิดเชิงคำนวณอาจจะมีลักษณะที่แตกต่างจากการคิดเชิงนามธรรมที่คนทั่ว ๆ ไปเข้าใจ รวมถึงแตกต่างจากการคิดเชิงนามธรรมในคณิตศาสตร์ (Wing, 2006)

การคิดเชิงคำนวณมีลักษณะสำคัญร่วมกันระหว่างการคิดเชิงคณิตศาสตร์ที่มีการเชื่อมโยงโลกนามธรรมที่ปราศจากข้อจำกัดทางกายภาพเข้ากับการคิดเชิงวิศวกรรมที่คิดอยู่ภายใต้ข้อจำกัดและกลไกทางกายภาพ (Wing, 2011) การคิดเชิงคำนวณจึงประกอบด้วยการคิดเชิงนามธรรมหลายระดับ (Wing, 2008) ตั้งแต่ความเป็นนามธรรมของปัญหา วิธีการแก้ปัญหา การถ่ายทอดวิธีการแก้ปัญหาจนถึงความเป็นนามธรรมที่ฝังในเครื่องจักรสำหรับใช้ในการแก้ปัญหา โดยการคิดในแต่ละขั้นเรามักจะละรายละเอียดของขั้นที่ต่ำกว่าไว้เพื่อทำให้เราจัดการงานในแต่ละขั้นได้ ด้วยเหตุนี้เองมนุษย์เราจึงสามารถแก้ปัญหาที่มีความซับซ้อนได้

เครื่องมือทางความคิดที่เป็นตัวเชื่อมต่อการคิดเชิงนามธรรมในแต่ละขั้น และทำให้เราสร้างกลไกเชิงนามธรรมที่เครื่องจักรดำเนินการได้ คือ ระบบตรรกะของบูล (Boolean logic) ดังภาพประกอบ 1 ระบบตรรกะของบูลเป็นระบบตรรกะที่หลุดออกจากข้อจำกัดในโลกกายภาพและสามารถสร้างขึ้นบนสิ่งที่ไม่มีอยู่จริงได้ ซึ่งแตกต่างจากระบบตรรกะของอริสโตเติลที่อ้างถึงสิ่งที่ปรากฏอยู่จริงซึ่งทำให้เกิดข้อจำกัดในการนำไปใช้ (Jevons, 1870) ระบบตรรกะของบูลทำให้เราสามารถเชื่อมต่อกับโลกอื่นที่มีความสอดคล้องกันได้ ดังนั้น มนุษย์จึงสามารถถ่ายทอดความคิดไปยังเครื่องจักรที่มีระบบกลไกที่ชัดเจนและแม่นยำเพื่อดำเนินการงานต่าง ๆ แทนมนุษย์ได้ ภาพประกอบ 2 แสดงการคิดเชิง

คำนวณเพื่อสร้างสรรค์วิธีการบวกเลขโดยใช้ตรรกะของบูล จากนั้นจึงแปลงวิธีการดังกล่าวไปสู่การทำงานในรูปแบบที่หลากหลาย เราจึงได้เห็นเครื่องจักรที่บวกเลขในรูปแบบต่าง ๆ ตัวอย่างนี้ยังได้บ่งถึงการใช้จินตนาการเพื่อสร้างสรรค์การใช้ระบบตรรกะที่เป็นตัวแทนทางความคิดและวิธีการที่สามารถถ่ายทอดไปสู่เครื่องจักรอื่น ๆ เพื่อให้ทำงานให้ได้ผลลัพธ์ที่ต้องการ

ภาพประกอบ 1

เปรียบเทียบระบบการคิดเชิงตรรกะของอริสโตเติล และบูล

All fresh are lazy.

John is a freshman.

John is lazy.

ระบบตรรกะของอริสโตเติล

A	*B*	*AND*	*OR*	*XOR*	*NOT B*
0	*0*	*0*	*0*	*0*	*1*
0	*1*	*0*	*1*	*1*	*0*
1	*0*	*0*	*1*	*1*	*1*
1	*1*	*1*	*1*	*0*	*0*

ระบบตรรกะของบูล

ภาพประกอบ 2

การถ่ายทอดความคิดของมนุษย์เพื่อให้เครื่องจักรโดยอาศัยตรรกะ

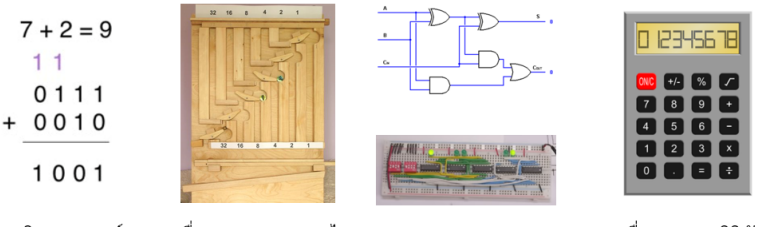
การคิดเชิงนามธรรมสำหรับการคิดเชิงคำนวณ

A	*B*	*AND*	*OR*	*XOR*	*NOT B*
0	*0*	*0*	*0*	*0*	*1*
0	*1*	*0*	*1*	*1*	*0*
1	*0*	*0*	*1*	*1*	*1*
1	*1*	*1*	*1*	*0*	*0*

ออกแบบ

ข้อมูลนำเข้า (Inputs)		ผลลัพธ์ (Outputs)	
A	B	ตัวทดเข้า	ตัวทดออก	ผลรวม
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

การสร้างสรรค์กลไกทางตรรกะสำหรับการบวกเลขด้วย Boolean logic



การคิดของมนุษย์ เครื่องบวกเลขแบบกลไก วงจรบวกเลข เครื่องบวกเลขดิจิทัล

ตัวอย่างข้างต้นแสดงให้เห็นว่า การใช้การคิดเชิงคำนวณในการแก้ปัญหาหรือทำงานหนึ่ง ๆ จะเกี่ยวข้องกับการคิดเชิงนามธรรมในหลายระดับ (Curzon et al., 2019; Wing, 2008) อาทิ ระดับปัญหา (กลไกการแก้ปัญหา) ระดับซอฟต์แวร์ (การถ่ายทอดกลไกให้เครื่องจักร) หรือระดับฮาร์ดแวร์ (การสร้างเครื่องจักรที่ดำเนินการตามกลไก) ในอีกแง่หนึ่งการคิดเชิงนามธรรมอาจถูกแสดงออกได้หลายรูปแบบ ไม่ว่าจะเป็นขั้นตอนการดำเนินการ สมการทางคณิตศาสตร์ วงจรไฟฟ้า หรือกลไกของเครื่องจักร กล่าวโดยสรุป การคิดเชิงคำนวณเกี่ยวข้องกับการคิดในเชิงนามธรรมในหลากหลายระดับ และรวมถึงการเชื่อมโยงนามธรรมในแต่ละระดับเข้าด้วยกัน (Curzon et al., 2019; Wing, 2008)

ความรู้เนื้อหาผสานวิธีสอนการคิดเชิงนามธรรมสำหรับการคิดเชิงคำนวณ

การทำงานกับคอมพิวเตอร์ต้องอาศัยการคิดเชิงนามธรรมซึ่งมีความแตกต่างจากการคิดเชิงนามธรรมทั่ว ๆ ไป หรือแม้กระทั่งการคิดเชิงนามธรรมที่คุ้นเคยในวิชาคณิตศาสตร์ ดังนั้น แนวคิดเชิงนามธรรมสำหรับการคิดเชิงคำนวณจึงเป็นสิ่งที่นักเรียนควรได้รับการพัฒนาอย่างมีประสิทธิภาพ โดยผ่านการจัดประสบการณ์การเรียนรู้ที่เหมาะสม ซึ่งจะส่งผลให้ผู้เรียนสามารถใช้คอมพิวเตอร์ในการแก้ปัญหาที่ซับซ้อนได้มากยิ่งขึ้น ในส่วนนี้ของบทความจะเป็นการผสานความรู้ทางวิทยาการคำนวณกับทฤษฎีการเรียนรู้เพื่อสังเคราะห์แนวทางปฏิบัติในการจัดการเรียนรู้เพื่อส่งเสริมการคิดเชิงนามธรรมสำหรับการคิดเชิงคำนวณ

สภาพแวดล้อมการพัฒนาการคิดเชิงนามธรรมสำหรับการคิดเชิงคำนวณ

การพัฒนาการคิดเชิงนามธรรมสำหรับการคิดเชิงคำนวณสามารถทำได้หลายวิธี แต่ละวิธีมีจุดเด่นแตกต่างกัน ในบทความนี้จัดประเภทกิจกรรมการพัฒนาการคิดเชิงนามธรรมสำหรับการคิดเชิงคำนวณออกเป็น 3 กลุ่ม

1. กิจกรรมถอดปลั๊ก หรือ Unplugged activity

กิจกรรมถอดปลั๊ก คือ เทคนิคการจัดการเรียนรู้ที่เน้นพัฒนาทักษะการคิดและความเข้าใจโดยตัดส่วนที่เกี่ยวข้องกับการสั่งงานคอมพิวเตอร์ออกไป (Curzon et al., 2019) ดังนั้น กิจกรรมนี้มักจะไม่มีการใช้คอมพิวเตอร์ (หรือถอดปลั๊ก) โดยส่วนใหญ่จะเป็นกิจกรรมที่มีอุปกรณ์ที่เป็นรูปธรรมจับต้องได้ซึ่งจะเป็นประโยชน์ช่วยให้นักเรียนพัฒนาทักษะการคิดเชิงนามธรรมได้ง่ายขึ้นผ่านกิจกรรม อย่างไรก็ตาม เป้าหมายของการจัดกิจกรรมควรมุ่งเน้นที่การพัฒนาทักษะการคิด และการสกัดความคิดรวบยอดซึ่งเป็นการคิดเชิงนามธรรม และกิจกรรมส่วนใหญ่มีวัตถุประสงค์เพื่อรวบยอดเกี่ยวกับการแก้ปัญหาใด ๆ ที่จะนำไปสู่แนวคิดแก้ปัญหาโดยใช้คอมพิวเตอร์ ตัวอย่างกิจกรรมถอดปลั๊ก เช่น กิจกรรม Move it Move it (Code.org, 2019) ที่เป็นกิจกรรมที่ผู้เรียนสามารถมีส่วนร่วมได้อย่างสนุกสนานพร้อมทั้งได้ฝึกฝนทักษะการออกคำสั่งที่มีความชัดเจนและเป็นลำดับ ซึ่งเป็นทักษะที่สำคัญสำหรับการคิดเชิงคำนวณ ในปัจจุบันอาจจะมีการนำกิจกรรมถอดปลั๊กไปอยู่ในรูปแบบเกมบนแท็บเล็ต หรือคอมพิวเตอร์ Sorted game (Phongsasithorn, Laosinchai, & Nokkaew, A., 2019) เป็นเกมที่ผู้เรียนจะได้ฝึกฝนการสร้างขั้นตอนวิธีการเรียงลำดับที่สามารถต่อยอดไปสร้างขั้นตอนวิธีให้คอมพิวเตอร์สามารถเรียงลำดับข้อมูลได้

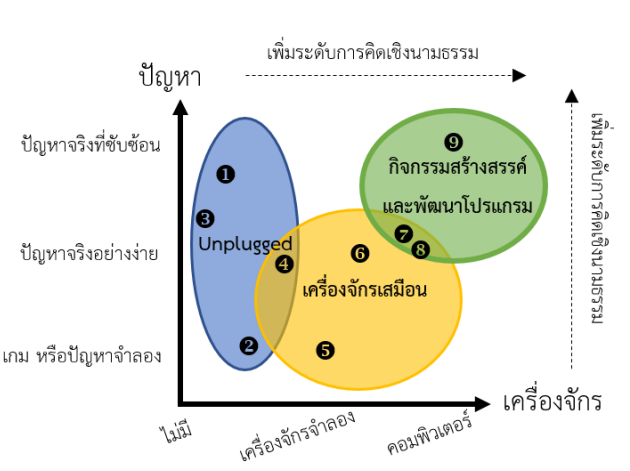
2. กิจกรรมแก้ปัญหาหรือสร้างสรรค์ชิ้นงานในเครื่องจักรเสมือน

เครื่องจักรเสมือนทำหน้าที่สร้างสภาพแวดล้อมที่จำลองการทำงานของคอมพิวเตอร์ แต่ถูกออกแบบให้อยู่ในรูปแบบที่ง่ายต่อการสร้างสรรค์วิธีในการแก้ปัญหาของนักเรียนโดยใช้ชุดคำสั่งที่ถูกกำหนดไว้ โดยคำสั่งมักจะอยู่ในรูปแบบบล็อกคำสั่ง (Blockly code) ทำให้นักเรียนไม่จำเป็นต้องเรียนรู้ไวยากรณ์เฉพาะของภาษา คอมพิวเตอร์ สิ่งที่แตกต่างจากกิจกรรมถอดปลั๊กคือในเครื่องจักรเสมือนมีเครื่องมือที่แปลและแสดงผลบล็อกคำสั่งซึ่งช่วยให้นักเรียนสามารถตรวจสอบชุดคำสั่งที่สร้างขึ้นได้ เครื่องมือนี้มักถูกออกแบบมาให้มีสภาพแวดล้อมที่เอื้อสำหรับนักเรียนในการแก้ปัญหา หรือสร้างสรรค์ชิ้นงาน ตัวอย่างเครื่องจักรเสมือน เช่น scratch หรือ Code with Google บางเครื่องมือมีการออกแบบภาระกิจเป็นลำดับขั้นโดยเริ่มจากสภาพแวดล้อมอย่างง่าย และค่อย ๆ เพิ่มองค์ประกอบและคำสั่งที่เหมือนจริงมากขึ้น เช่น code.org เครื่องจักรเสมือนสร้างสภาพแวดล้อม ที่ใกล้เคียงกับคอมพิวเตอร์มากกว่ากิจกรรมถอดปลั๊ก และเป็นเครื่องมือที่นักเรียนสามารถเรียนรู้ได้ด้วยตนเอง ซึ่งทำให้นักเรียนได้มีประสบการณ์การทำงานร่วมกับคอมพิวเตอร์ได้ใกล้เคียงมากยิ่งขึ้น รวมถึงเป็นโอกาสในการพัฒนาทัศนคติต่อการทำงานร่วมกับคอมพิวเตอร์ แต่เนื่องด้วยเครื่องจักรเสมือนถูกสร้างด้วยสมมติฐานของการใช้งานที่มีความจำกัด เช่น ชุดคำสั่งมีจำกัด หรือสร้างเพื่อจุดประสงค์เฉพาะเจาะจง จึงทำให้การใช้เครื่องจักรเสมือนไม่สะดวกและไม่เอื้อต่อการแก้ปัญหาที่มีความซับซ้อนมาก ๆ

3. กิจกรรมสร้างสรรค์และพัฒนาโปรแกรม เป็นรูปแบบกิจกรรมที่เรียนรู้ในสภาพจริงที่นักเรียนจะได้รับประสบการณ์ที่เหมือนกับนักพัฒนาโปรแกรมที่ทำงานจริงในการใช้คอมพิวเตอร์เป็นเครื่องมือในการแก้ปัญหา นักเรียนจะใช้เครื่องมือจริง ใช้ภาษาคอมพิวเตอร์ในการสร้างสรรค์ชิ้นงาน ซึ่งต้องอาศัยความรู้และทักษะที่มากกว่าสองกิจกรรมที่กล่าวมา ในกิจกรรมการเรียนรู้นี้นักเรียนจะได้ฝึกฝนทักษะที่ใกล้เคียงกับการนำคอมพิวเตอร์มาใช้งานจริงมากที่สุด นักเรียนจะต้องแปลงปัญหาจากโลกความจริงให้อยู่ในรูปที่คอมพิวเตอร์สามารถดำเนินการได้ ต้องออกแบบการจัดการกับข้อมูล ออกแบบวิธีการลำดับขั้นตอนในการแก้ปัญหา ได้เรียนรู้ความสามารถและข้อจำกัดที่แท้จริงของคอมพิวเตอร์ ได้ฝึกฝนการสื่อสารแนวคิด การเขียนโค้ดที่ง่ายต่อการอ่าน และแลกเปลี่ยนกับทีม การทำโครงงานที่คอมพิวเตอร์ในการแก้ปัญหาในชีวิตจริง เป็นตัวอย่างหนึ่งของกิจกรรมสร้างสรรค์และพัฒนาโปรแกรมนี้

ภาพประกอบ 3

แผนภาพแสดงขอบเขตการใช้กิจกรรมการเรียนรู้เพื่อพัฒนาการคิดเชิงนามธรรม



กิจกรรมพัฒนาการคิดเชิงคำนวณ

1. Machine learning (SupaDaow, 2019, December 15)
2. Move It Move It (Code.org, 2019)
3. Sorting Network (Bell, Alexander, Freeman, & Grimley, 2009)
4. Sorted game (Phongsasithorn, Laosinchai, & Nokkaew, 2019)
5. หลักสูตร Code.org (https://code.org/)
6. Scratch (https://scratch.mit.edu/)
7. MIT App Inventor (Keereerat, 2019)
8. Tinkercad (https://www.tinkercad.com/)
9. โครงงานการใช้คอมพิวเตอร์ในการแก้ปัญหาในชีวิตจริง

กิจกรรมทั้งสามประเภทมีจุดเด่นและข้อจำกัดที่แตกต่างกันไป ภาพประกอบ 3 แสดงขอบเขตสมมติของกิจกรรมแต่ละประเภทเพื่อแสดงให้เห็นว่าแต่ละกิจกรรมมีขอบเขตและจุดเด่นที่แตกต่างกัน นอกจากนี้ ยังแสดงให้เห็นถึงโอกาสในการเลือกใช้เครื่องมือเพื่อสร้างสภาพแวดล้อมการเรียนรู้ที่เอื้อกับการพัฒนาการคิดเชิงนามธรรมสำหรับการคิดเชิงคำนวณ จากภาพประกอบ 3 เมื่อความซับซ้อนของปัญหา หรือเครื่องจักรที่ใช้ในการดำเนินการมากขึ้น จะยิ่งต้องการระดับการคิดเชิงนามธรรมที่สูงมากขึ้นด้วย ในการจัดการเรียนรู้ครูควรคาดหวังให้นักเรียนสามารถพัฒนาไปถึงการใช้คอมพิวเตอร์จริงเป็นเครื่องมือในการแก้ปัญหาที่มีความซับซ้อนได้ (พื้นที่สีเขียว) แต่อย่างไรก็ตาม นักเรียนมีความรู้และพื้นฐานที่แตกต่างกัน ดังนั้นครูควรใช้วิจารณญาณในการเลือกใช้กิจกรรมที่ช่วยนักเรียนในการพัฒนา ซึ่งจะกล่าวในส่วนต่อไป

การพัฒนาการคิดเชิงนามธรรมสำหรับการคิดเชิงคำนวณ

หัวใจของการคิดเชิงคำนวณ คือ ความสามารถในการจัดการกับความคิดในระดับนามธรรมและการเชื่อมโยงนามธรรมในแต่ละระดับเข้าด้วยกันได้ ดังนั้น ครูควรจะต้องเข้าใจการพัฒนาการคิดเชิงนามธรรมของผู้เรียนเพื่อเลือกกิจกรรมและจัดการเรียนรู้ได้อย่างเหมาะสม เพียเจต์ (Piaget) แบ่งระดับของการพัฒนาของมนุษย์ออกเป็น 4 ระดับ

ตามช่วงอายุ Sensorimotor (แรกเกิด-2 ปี), Preoperational (2-7 ปี), Concrete operational (7-11 ปี) และ Formal operational (สูงกว่า 11 ปี) (Ginsburg & Opper, 1988) ทฤษฎีของเพียเจต์ทำให้เราเข้าใจพัฒนาการทางความคิดของเด็ก และสามารถจัดการเรียนรู้ที่เหมาะสมสำหรับนักเรียนในแต่ละช่วงวัย ต่อมาภายหลังนักทฤษฎีการเรียนรู้ที่เรียกตัวเองว่า *กลุ่มนีโอเพียเจต์* ซึ่งเห็นพ้องกับลำดับพัฒนาการมนุษย์ตามที่เพียเจต์เสนอ แย้งว่าระดับการพัฒนาอาจจะไม่ได้ขึ้นกับอายุ หากแต่ขึ้นกับประสบการณ์ในเรื่องนั้น ๆ และคนเราอาจจะมีระดับการคิดเชิงนามธรรมที่แตกต่างกันแม้จะเป็นเรื่องเดียวกัน (Beilin & Pufall, 2013; Case, 1992; Lister, 2011)

หากพิจารณาการเรียนรู้วิทยาการคำนวณตามแนวคิดของนีโอเพียเจต์ เราสามารถอธิบายพัฒนาการของการคิดเชิงนามธรรมเพื่อการเรียนรู้การคิดเชิงคำนวณได้ โดยในที่นี้จะอธิบายพัฒนาการตามแนวคิดของนีโอเพียเจต์ในสองบริบทการเรียนรู้ คือ การเรียนรู้ในโลกเสมือน (code.org) และการเรียนรู้ผ่านกิจกรรมถอดปลั๊ก ดังตาราง 2

แนวคิดของนีโอเพียเจต์ดังกล่าวสามารถใช้เป็นแนวทางในการจัดประสบการณ์การเรียนรู้และการประเมินผลผู้เรียนได้ ประการแรก การพัฒนาแนวคิดเชิงนามธรรมนั้นต้องอาศัยประสบการณ์ที่หลากหลายของผู้เรียนเอง ดังนั้นการจัดการเรียนรู้ไม่ใช่การบอกให้นักเรียนจดจำขั้นตอนหรือวิธีการ แต่นักเรียนต้องพัฒนาแนวคิดและวิธีคิดของตนเอง ประการที่สอง แนวคิดของนีโอเพียเจต์แนะนำว่าระดับความสามารถในการคิดไม่ขึ้นกับช่วงอายุโดยตรง หากแต่ขึ้นกับประสบการณ์ในเนื้อหานั้น ๆ ด้วย ดังนั้น ในชั้นเรียนจะมีนักเรียนที่มีหลากหลายความสามารถทางการคิด แม้ว่าจะมีอายุที่ใกล้เคียงกัน ครูจึงต้องจัดกิจกรรมการเรียนรู้ที่คำนึงถึงความหลากหลายของประสบการณ์ของผู้เรียน และทำให้นักเรียนทุกคนสามารถมีส่วนร่วมได้ อาจจะเริ่มจากแนวคิดง่าย ๆ หรือตรงไปตรงมา มีความเป็นรูปธรรมก่อนในช่วงแรก หากปัญหามีความซับซ้อนให้แตกออกเป็นปัญหาย่อย ๆ ที่เหมาะกับระดับการคิดของผู้เรียน เพื่อฝึกให้ผู้เรียนได้เรียนรู้และแก้ทีละส่วน บทบาทครูที่สำคัญมากอีกอย่างคือการช่วยให้นักเรียนสะท้อนคิดในวิธีการ เพื่อพัฒนาแนวคิดให้มีประสิทธิภาพมากยิ่งขึ้น หรือเอื้อให้เกิดการแลกเปลี่ยนระหว่างนักเรียนกันเองเพื่อให้เห็นความคิดที่หลากหลายจะช่วยให้นักเรียนสามารถพัฒนาได้รวดเร็วยิ่งขึ้น (Vygotsky, 1978)

ตาราง 2

พัฒนาการของการคิดเชิงนามธรรมสำหรับการคิดคำนวณตามทฤษฎีนีโอเพียเจต์

ลำดับขั้นการพัฒนาของ Piaget (Ginsburg & Opper, 1988; Lister, 2011)	**การคาดการณ์พฤติกรรมนักเรียน**	
	Unplugged activity	**โลกเสมือน เช่น code.org**
1. Sensorimotor - พัฒนาการปฏิสัมพันธ์กับสภาพแวดล้อมใหม่ ผู้เรียนลองผิดลองถูกกับสภาพแวดล้อมใหม่ และเรียนรู้จากผลสะท้อนกลับของสิ่งที่อยู่ตรงหน้า	ลองเล่นกับอุปกรณ์ หรือ วัตถุต่าง ๆ ที่มีในกิจกรรม และเรียนรู้ผลสะท้อนกลับโดยที่ยังไม่สามารถสร้างคำตอบได้	ลองผิดลองถูกกับการใช้แต่ละคำสั่ง หรือ ปุ่มต่าง ๆ และเรียนรู้ผลสะท้อนกลับ โดยที่ยังไม่สามารถเอาคำสั่งมาเรียงกันเพื่อทำงานได้
2. Preoperational – เริ่มมีการให้ความหมายกับภาษาหรือวัตถุที่เกี่ยวข้อง และใช้จากมุมมองของตนเอง แต่ไม่สามารถเข้าใจจากมุมมองของคนอื่น ตัดสินใจโดยใช้สหัชญาณ	เข้าใจความหมายของวัตถุต่าง ๆ แต่นำมาใช้ในการแก้ปัญหาจากมุมมองและการตีความของตนเอง ยังไม่แก้ปัญหาให้สอดคล้องกับเงื่อนไขที่กำหนดได้ ทำให้ยังคงไม่สามารถนำวัตถุเหล่านั้นมาดำเนินการเพื่อการแก้ปัญหาได้เอง	เข้าใจคำสั่งแต่ล่ะคำสั่ง หรือ บล็อก สามารถเรียงบล็อกตามตัวอย่างได้ แต่ไม่สามารถถ่ายทอดความคิดของตัวเองเป็นภาษาหรือบล็อกคำสั่งของเครื่องจักรที่ใช้งานอยู่ได้ ทำให้ไม่สามารถนำคำสั่ง ต่าง ๆ มาเรียงต่อกันเพื่อให้งานสำเร็จได้เอง

ตาราง 2 (ต่อ)

ลำดับขั้นการพัฒนาของ Piaget (Ginsburg & Opper, 1988; Lister, 2011)	การคาดการณ์พฤติกรรมนักเรียน	
	Unplugged activity	โลกเสมือน เช่น code.org
3. Concrete operational – เริ่มมีการใช้เหตุผลในการกระทำ เข้าใจมุมมองอื่น ๆ นอกจากตัวเอง ยังต้องอาศัยสิ่งของที่จับต้องได้เพื่อช่วยในการคิด	ลองผิดลองถูก ปรับวิธีการไปเรื่อย ๆ จนได้คำตอบ ในขั้นนี้นักเรียนอาจจะทำงานได้สำเร็จ แต่ส่วนมากวิธีที่ใช้เป็นวิธีการแก้ปัญหาที่มีประสิทธิภาพน้อย	ลองเอาคำสั่งมาวางเรียงและทดสอบเพื่อดูผลลัพธ์ แล้วจึงปรับแก้เป็นส่วน ๆ นักเรียนอาจจะทำงานสำเร็จได้แต่เกิดจากการค่อย ๆ ลอง และ ปรับแก้ไปเรื่อย ๆ ยังไม่สามารถสร้างความคิดรวบยอดได้
4. Formal operational - คิดในระดับนามธรรมโดยใช้ความคิดรวบยอดได้ ไม่จำเป็นต้องอาศัยวัตถุ หรือ สิ่งที่เป็นรูปธรรมเพื่อช่วยในการคิด ใช้ความสัมพันธ์ของสิ่งต่าง ๆ เพื่อแก้ปัญหาอย่างเป็นระบบ	ไม่จำเป็นต้องพึ่งวัตถุในการคิด สามารถสร้างคำตอบได้โดยวิธีการที่มีประสิทธิภาพในเวลาอันรวดเร็ว นักเรียนสามารถคิดเป็นนามธรรม เห็นภาพรวมในการแก้ปัญหาและสามารถสร้างวิธีการแก้ปัญหาได้	ไม่จำเป็นต้องพึ่งการนำคำสั่งมาเรียงต่อกันและเพื่อทดสอบ แต่สามารถออกแบบขั้นตอนการแก้ปัญหาได้ในความคิดได้

ในการเพิ่มประสบการณ์การคิดเชิงนามธรรม ครูอาจจะพิจารณาใช้สภาพแวดล้อมการเรียนรู้ที่หลากหลายและแตกต่างกัน ซึ่งแต่ละวิธีมีจุดเด่นที่แตกต่างกัน ดังภาพประกอบ 3 ครูอาจจะพิจารณาเพิ่มระดับความซับซ้อนของปัญหา หรือความซับซ้อนของเครื่องจักร กิจกรรมใน code.org เป็นตัวอย่างของการออกแบบปัญหาที่เริ่มจากปัญหาที่มีความตรงไปตรงมาไปสู่ปัญหาที่ต้องอาศัยแนวคิดเชิงนามธรรมมากขึ้น เครื่องมือ รวมทั้งชุดคำสั่งที่มีให้ใช้ก็จะเพิ่มขึ้นไปเรื่อย ๆ ด้วยเช่นกัน ซึ่งเป็นสภาพแวดล้อมที่นักเรียนจะได้เรียนรู้จากง่ายไปสู่ยากโดยการแตกการเรียนรู้ออกเป็นส่วนย่อย ๆ

ในด้านการวัดและประเมินผลผู้เรียนแนวคิดของนีโอเพียเจต์ตามตาราง 2 ให้แนวทางในการวัดและประเมินผลการคิดเชิงนามธรรมของผู้เรียนไว้ว่าการประเมินโดยที่ดูผลลัพธ์ที่ตัวชิ้นงานสุดท้ายหรือการที่นักเรียนแก้ปัญหาได้หรือทำงานสำเร็จไม่ได้หมายความว่านักเรียนได้พัฒนาถึงขั้นสูงของความคิดเชิงนามธรรมแล้ว นักเรียนอาจจะได้วิธีการแก้ปัญหามาจากการลองผิดลองถูกและปรับแก้มาเรื่อย ๆ จนได้วิธีการที่ใช้ได้ และทำงานที่กำหนดให้จนสำเร็จ ดังนั้น การวัดและประเมินที่ผลลัพธ์ปลายทางอาจจะไม่เพียงพอที่จะวัดระดับการคิดเชิงนามธรรม อีกทั้งยังไม่ได้ช่วยให้ผู้เรียนเกิดการพัฒนา การประเมินผลควรจะต้องประเมินวิธีคิดที่ผู้เรียนใช้ในการหาคำตอบ การประเมินผลระหว่างทางมีความสำคัญในการให้ผลสะท้อนกลับและช่วยพัฒนาผู้เรียนในการคิดการแก้ปัญหาในเชิงนามธรรมให้ดียิ่งขึ้น

แนวการปฏิบัติสำหรับการจัดการเรียนรู้เพื่อพัฒนาการคิดเชิงนามธรรมสำหรับการคิดเชิงคำนวณ

การส่งเสริมให้เกิดการพัฒนาแนวคิดเชิงนามธรรมขึ้นกับสภาพแวดล้อมและประสบการณ์การเรียนรู้ และครูเป็นส่วนสำคัญ ดังนั้น การจัดการเรียนรู้จึงไม่ใช่เพียงการให้นักเรียนเล่นกิจกรรม แต่ครูควรมีบทบาททั้งการเลือกกิจกรรมการเรียนรู้ การกระตุ้นให้เกิดการพัฒนาแนวคิดในระหว่างกิจกรรมและการนำอภิปรายหลังจากที่นักเรียนทำกิจกรรมดังกล่าวเพื่อนำไปสู่การสร้างแนวคิด บทความนี้ขอเสนอแนวการปฏิบัติสำหรับการจัดการเรียนรู้เพื่อพัฒนาการคิดเชิงนามธรรมสำหรับการคิดเชิงคำนวณ โดยดัดแปลงมาจากแนวปฏิบัติในการจัดการเรียนรู้คณิตศาสตร์ของ Stein et al. (2008) และ Nabb et al. (2018) ดังนี้

1. เลือกกิจกรรมการเรียนรู้

กิจกรรมการเรียนรู้ที่ใช้ควรเป็นกิจกรรมที่เหมาะสมและท้าทายระดับการคิดของผู้เรียน ไม่ง่ายหรือ ยากจนเกินไป ในกรณีที่การเรียนรู้นั้นมุ่งหวัง

พัฒนาแนวคิดของผู้เรียน อาจจะพิจารณาใช้กิจกรรมที่มีความท้าทายแต่มีความเป็นรูปธรรม มีวัตถุที่จับต้องได้ เพราะจะเปิดโอกาสให้ผู้เรียนทุกคนสามารถร่วมคิดและร่วมอภิปรายในกิจกรรมนั้นได้ หากกิจกรรมมีความเป็นนามธรรมเกินไปอาจจะทำให้ผู้เรียนบางส่วนไม่สามารถมีส่วนร่วมได้และเกิดทัศนคติที่ไม่ดีต่อการเรียนรู้การคิดเชิงคำนวณ กิจกรรมการเรียนรู้การคิดเชิงนามธรรมสำหรับการคิดเชิงคำนวณที่ดีควรเป็นกิจกรรมที่นักเรียนสามารถหาคำตอบได้หลากหลาย กล่าวคือเด็กที่เรียนรู้ช้าก็สามารถมีส่วนร่วมในการคิดวิธีการหาคำตอบได้แม้อาจจะไม่ถูกทั้งหมด หรือเป็นคำตอบที่ตรงไปตรงมา ส่วนเด็กเก่งก็สามารถคิดวิธีการแก้ปัญหาที่ซับซ้อนมากขึ้นได้ ความหลากหลายที่เกิดขึ้นในห้องเรียนจะเป็นจุดเริ่มต้นที่สำคัญสำหรับพัฒนาทักษะการคิดเชิงนามธรรมของนักเรียนทุกคนในห้องเรียน

2. มอบหมายงาน

การพัฒนาแนวคิดเชิงนามธรรมในการคิดเชิงคำนวณจะเกิดขึ้นเมื่อนักเรียนมีส่วนร่วมในการคิด กลยุทธ์ในการมอบหมายงานเป็นอีกเครื่องมือหนึ่งที่จะกระตุ้นให้นักเรียนมีส่วนร่วมคิดแก้ปัญหาในกิจกรรมการเรียนรู้ ครูสามารถใช้กลยุทธในการมอบหมายงานได้อย่างผสมผสาน เช่น การมอบหมายให้ทำเป็นกลุ่ม การทำงานเป็นคู่ และการทำงานเดี่ยว ทั้งนี้ แต่ละกลยุทธ์มีข้อดีและข้อจำกัดแตกต่างกัน สำหรับงานที่ไม่ยากมากนัก หรือ มีอุปกรณ์ที่เป็นรูปธรรมสำหรับช่วยการคิด ครูอาจจะมอบหมายเป็นงานเดี่ยวหรืองานคู่ และอาจจะให้สลับกันตรวจสอบวิธีคิดของเพื่อนโดยใช้อุปกรณ์ที่กำหนดให้ แต่สำหรับงานที่มีความยาก ครูอาจจะมอบหมายให้ทำเป็นกลุ่ม ทั้งนี้ จำนวนสมาชิกในกลุ่มไม่ควรมีจำนวนมากเกินไป เพราะจะทำให้มีนักเรียนส่วนหนึ่งไม่ได้มีส่วนร่วม นอกจากนี้ การมอบหมายงานควรมอบหมายงานโดยใช้คำสั่งที่เฉพาะเจาะจง เพราะจะเป็นการช่วยนักเรียนโดยเฉพาะอย่างยิ่งนักเรียนที่อาจจะยังไม่คุ้นกับวิธีการคิดเชิงคำนวณ ครูควรมีการมอบหมายงานที่ระบุความคาดหวังจากนักเรียนที่ชัดเจน เช่น คิดวิธีการและเล่าให้เพื่อนฟังถึงวิธีการของตัวเองโดยใช้อุปกรณ์ที่กำหนดให้ ตรวจสอบวิธีการคิดโดยแสดงการตรวจสอบโดยใช้อุปกรณ์ เป็นต้น และควรกำหนดกรอบเวลาที่มากพอที่จะเห็นแนวคิดที่มีความแตกต่างกันเกิดขึ้นในห้องเรียน ซึ่งความแตกต่างนี้จะนำไปสู่การอภิปราย แต่ก็ไม่ควรให้เวลามากจนเกินไปและครูอาจจะไม่จำเป็นต้องรอให้ทุกคนคิดวิธีการที่สมบูรณ์ได้ ทั้งนี้การปรับกรอบเวลาขึ้นอยู่กับพื้นฐานของนักเรียน และความหลากหลายของระดับความสามารถและความคิดของนักเรียน

3. ติดตามการดำเนินการ

จากการที่ครูมอบหมายงานที่จะทำให้เกิดวิธีการคิดที่หลากหลายในชั้นเรียนแล้ว ครูควรติดตามดูว่าวิธีการที่นักเรียนใช้มีอะไรบ้าง ครูควรสนใจทั้งวิธีการที่ถูกต้องและวิธีการที่ยังไม่สมบูรณ์ ในการเดินดูครูอาจจะสอบถามวิธีการคิดของนักเรียน แต่ในขั้นนี้ครูไม่ควรแทรกแซงหรือชี้นำการคิดของนักเรียน ครูไม่ควรตัดสินหรือบอกว่าวิธีการไหนที่ดีหรือไม่ดี แต่พยายามให้นักเรียนอธิบายวิธีการคิดด้วยเหตุผลและคำพูดของนักเรียนเอง ครูอาจจะปล่อยให้นักเรียนทำผิดได้บ้างและตระหนักรู้ได้ด้วยตัวเอง แต่ครูควรตั้งคำถามให้นักเรียนสังเกตผลลัพธ์ที่เกิดจากวิธีการที่ผิดนั้นเพื่อปรับปรุงวิธีการแก้ปัญหาใหม่ เหตุผลสำคัญที่ครูควรติดตามการดำเนินการของนักเรียนคือครูจะได้รู้ว่ามีวิธีการที่แตกต่างกันกี่แบบ เพื่อที่จะสามารถเลือกวิธีการเหล่านั้นมาเพื่อเชื่อมโยงและนำไปสู่การพัฒนาแนวคิดเชิงนามธรรมได้ในขั้นต่อไป

4. จัดลำดับผลงานนำเสนอ

เมื่อครูได้สำรวจความหลากหลายของวิธีคิดในห้องเรียนแล้ว ครูอาจจะเลือกนักเรียนให้ออกมาอธิบายแนวคิดของตน โดยครูสามารถแจ้งนักเรียนล่วงหน้าในระหว่างที่เดินดูในขั้นติดตามการดำเนินงาน หลักในการเลือกคือ ไม่ควรเลือกนักเรียนที่มีแนวคิดเดียวกันมานำเสนอ เพราะจะทำให้เวลาในการเรียนรู้ถูกใช้ไปโดยที่ไม่ได้เกิดการพัฒนาหรือการต่อยอดในแนวคิด ครูควรเลือกนำเสนอแนวคิดที่มีความแตกต่างกัน โดยที่ครูจะต้องมองถึงการเชื่อมโยงวิธีการที่นำเสนอ เพื่อนำไปสู่การอภิปรายและแนวคิดที่ได้จากงานชิ้นนี้ หลักในการจัดลำดับ

การนำเสนอคือ อาจจะเลือกนำเสนอแนวคิดที่มีความซับซ้อนน้อยที่สุดหรือตรงไปตรงมามากที่สุดก่อน เพราะนักเรียนที่ช้าในห้องจะสามารถมีส่วนร่วมในสร้างการเชื่อมโยงได้ ครูพึงตระหนักว่ากิจกรรมการเรียนรู้ไม่ได้จบที่การนำเสนอ แต่ครูควรเป็นผู้นำการอภิปรายเพื่อให้นักเรียนทุกคน หรือนักเรียนส่วนใหญ่สามารถเชื่อมโยงไปสู่วิธีการอื่น ๆ ที่อาจมีประสิทธิภาพมากกว่า เพื่อพัฒนาการคิดของนักเรียนโดยอาศัยผลลัพธ์ที่เกิดขึ้นจากนักเรียนเอง

5. นำสู่การอภิปรายและพัฒนาแนวคิด

ขั้นอภิปรายและพัฒนาแนวคิดคือ ขั้นที่จะพัฒนาแนวคิดของนักเรียน เครื่องมือที่สำคัญคือการใช้คำถามโดยมีจุดประสงค์เพื่อให้ผู้เรียนสะท้อนผลลัพธ์และวิธีคิดของตัวเอง คำถามที่มักจะถูกใช้เสมอในการจัดการเรียนรู้การคิดเชิงคำนวณ คือ “วิธีการเหล่านี้เหมือนหรือแตกต่างกัน และอย่างไร” “วิธีการนี้ให้คำตอบที่ถูกหรือไม่ หากไม่ ส่วนใดที่ผิด” “เรามีวิธีการตรวจสอบความถูกต้องได้อย่างไร” “สามารถทำให้วิธีการดำเนินการมีประสิทธิภาพมากกว่านี้ได้หรือไม่” “มีส่วนใดในการแก้ปัญหาที่ทำงานลักษณะซ้ำ ๆ หรือมีส่วนใดที่ไม่จำเป็นต้องมี” “เราพบว่ามีวิธีการที่แก้ปัญหาได้หลากหลาย นักเรียนควรเลือกใช้วิธีการใดเพราะเหตุใด” ซึ่งตัวอย่างคำถามอาจเป็นได้ดังภาพ ประกอบ 4

ภาพประกอบ 4

ตัวอย่างการใช้คำถามเพื่อพัฒนาการคิดเชิงนามธรรมและนำสู่การอภิปราย ในกิจกรรม code.org

6. ปรับปรุงคำตอบ

เมื่อถามคำถาม ควรให้เวลานักเรียนได้คิดและปรับปรุงลำดับขั้นตอนในการแก้ปัญหาของตนเอง เมื่อได้วิธีการใหม่ก็จะวนเข้าวงจรของการอภิปรายคำตอบที่ได้ร่วมกัน

7. ฝึกฝน

เมื่อแนวคิดของผู้เรียนได้รับการพัฒนาแล้ว อีกสิ่งที่สำคัญคือ การฝึกฝนการนำไปใช้ในสถานการณ์ที่อาจจะแตกต่างออกไป ในส่วนของการฝึกฝนการใช้โปรแกรมโลกเสมือน เช่น code.org จะเป็นประโยชน์ เพราะระบบถูกออกแบบมีลักษณะ

คล้ายเกมที่สามารถสร้างแรงจูงใจให้กับนักเรียนได้ลองเล่น และยังมีงานและปัญหาที่หลากหลายที่นักเรียนจะได้ฝึกฝนจากง่ายไปยาก มีการให้ผลสะท้อนกลับที่ช่วยนักเรียนในการพัฒนาวิธีการแก้ปัญหาและช่วยในการเรียนรู้ สำหรับนักเรียนที่พร้อมสำหรับการสร้างชิ้นงานของตัวเอง ครูอาจจะมอบหมายให้นักเรียนทำชิ้นงาน ซึ่งจะทำให้นักเรียนได้มีประสบการณ์และเรียนรู้เพิ่มเติม

บทสรุป

การคิดเชิงนามธรรมเป็นทักษะที่สำคัญสำหรับการคิดเชิงคำนวณและการใช้คอมพิวเตอร์ในการแก้ปัญหาที่มีความซับซ้อน กิจกรรมการเรียนรู้ที่ใช้พัฒนาการคิดเชิงนามธรรมการสำหรับการคิดเชิงคำนวณมีหลากหลายรูปแบบ แต่ละกิจกรรมมีจุดเด่นและข้อจำกัดแตกต่างกัน การเลือกใช้กิจกรรมที่เหมาะสมจะช่วยสร้างสภาพแวดล้อมการเรียนรู้ที่เอื้ออำนวยต่อการพัฒนาการคิดเชิงนามธรรมสำหรับการคิดเชิงคำนวณ ไม่เพียงเท่านั้น การจัดการเรียนรู้ของครูเองเป็นส่วนที่สำคัญเช่นกัน ครูจึงควรมีความรู้และทักษะในการผสานความรู้เนื้อหาและกระบวนวิธีการสอนเพื่อทำให้ผู้เรียนสามารถพัฒนาได้อย่างเต็มศักยภาพ กระบวนการสอนที่พัฒนาการคิดเชิงนามธรรมควรเน้นการสร้างประสบการณ์การคิดที่หลากหลาย เริ่มจากปัญหาเล็กไปสู่ปัญหาใหญ่ที่ต้องอาศัยการแตกย่อย ครูนำอภิปรายให้เกิดแนวคิด ครูยังสามารถเพิ่มความซับซ้อนของปัญหาและทักษะการคิดเชิงนามธรรมให้เกิดความท้าทาย การจัดการเรียนรู้ควรผนวกกับกระบวนการสะท้อนคิด การอภิปราย การตั้งคำถามที่มีความเชื่อมโยงไปสู่เป้าหมายในการแก้ปัญหาอยู่เสมอ

เอกสารอ้างอิง

Aho, A. V. (2012). Computation and computational thinking. *Computer Journal*, *55*(7), 833–835.

Beilin, H., & Pufall, P. B. (2013). *Piaget's theory: prospects and possibilities*. Psychology Press.

Bell, T., Alexander, J., Freeman, I., & Grimley, M. (2009). Computer Science Unplugged: school students doing real computing without computers. *The New Zealand Journal of Applied Computing and Information Technology*, *13*(1), 20–29.

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. *AERA*, 1-25.

Case, R. (1992). Neo-Piagetian theories of child development. In R. J. Sternberg & C. A. Berg (Eds.), *Intellectual Development* (pp. 161–196).

Curzon, P., Bell, T., Waite, J., & Dorling, M. (2019). Computational Thinking. In S. A. Fincher & A. V. Robins (Eds.), *The Cambridge Handbook of Computing Education Research* (pp. 513–546).

Denning, P. J. (2017). Remaining trouble spots with computational thinking. *Communications of the ACM*, *60*(6), 33–39.

Ginsburg, H. P., & Opper, S. (1988). Piaget's Theory of Intellectual Development. In *Education*. International Psychotherapy Institute.

Jevons, W. S. (1870). XXII. On the mechanical performance of logical inference. *Philosophical transactions of the Royal Society of London*, (*160*), 497-518.

Keereerat, C. (2019). Using the Problem-Solving and App Inventor to Develop Computational Thinking Skill for High School Students. *Journal of Education Studies*, *2*, 31–47.

Korkmaz, Ö., Çakir, R., & Özden, M. Y. (2017). A validity and reliability study of the computational thinking scales (CTS). *Computers in Human Behavior*, *72*.

Lister, R. (2011, January). Concrete and other neo-Piagetian forms of reasoning in the novice programmer. In *Proceedings of the Thirteenth Australasian Computing Education Conference-Volume 114* (pp. 9-18). Australian Computer Society, Inc.

Nabb, K., Hofacker, E. B., Ernie, K., & Ahrendt, S. (2018). Using the 5 Practices in Mathematics Teaching. *The Mathematics Teacher*, *111*(5), 366–373.

Papert, S. (1980). *Mindstroms: children, computers, and powerful ideas*. New York: Basic Books, Inc.

Phongsasithorn, A., Laosinchai, P., & Nokkaew, A. (2019). Sorted: An Educational Digital Game for Learning Sorting Algorithms. In *Proceedings of the International Symposium on Education and Psychology 2019.* Knowledge Association of Taiwan.

Priami, C. (2007). Computational thinking in biology. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *4780 LNBI*, 63–76.

Shulman, L. S. (1986). Those Who Understand: Knowledge Growth in Teaching. *Educational Researcher*, *15*(2), 4–14.

Stein, M. K., Engle, R. A., Smith, M. S., & Hughes, E. K. (2008). Orchestrating Productive Mathematical Discussions: Five Practices for Helping Teachers Move Beyond Show and Tell. *Mathematical Thinking and Learning*, *10*(4), 313-340.

SupaDaow. (2019). *Unplugged Data Science for High School เข้าใจวิทยาศาสตร์ข้อมูลก่อนพุ่งตัวไปที่ AI*. Retrieved from HACK yourself website: https://everyhackilearn.wordpress.com/2019/11/30/unplugged-data-science-for-high-school-เข้าใจวิทยาศาสตร์/

Vygotsky, L. S. (1978). Mind in Society: The Development of Higher Psychological Processes . *American Anthropologist,* 81. https://doi.org/10.1525/aa.1979.81.4.02a00580

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

Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *366*(1881), 3717–3725. https://doi.org/10.1098/rsta.2008.0118

Wing, J. M. (2011). Research notebook: Computational thinking—What and why? *The Link Magazine*, June 23, 2015. Retrieved from http://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why