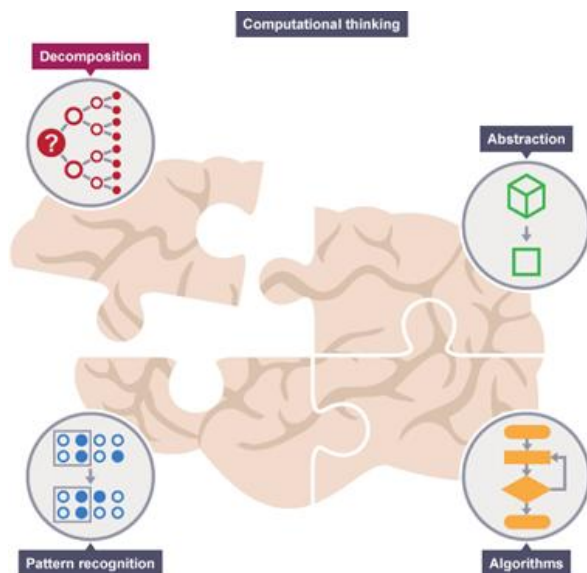


ใบความรู้ที่ 1 เรื่อง แนวคิดเชิงคำนวณ
หน่วยการเรียนรู้ที่ 1 เรื่อง การแก้ปัญหา
แผนการจัดการเรียนรู้ที่ 4 เรื่อง อัลกอริธึมหรือขั้นตอนวิธี
รายวิชา วิทยาการคำนวณ รหัสวิชา ว22104 ภาคเรียนที่ 1 ชั้นมัธยมศึกษาปีที่ 2

แนวคิดเชิงคำนวณ (Computational Thinking)

แนวคิดเชิงคำนวณ (Computational Thinking) เป็นกระบวนการวิเคราะห์ปัญหา เพื่อให้ได้แนวทางหาคำตอบอย่างเป็นขั้นตอนที่สามารถนำไปปฏิบัติได้โดยบุคคลหรือคอมพิวเตอร์อย่างถูกต้องแนวคิดเชิงคำนวณจะช่วยให้เข้าใจปัญหาที่ซับซ้อนสามารถได้ง่ายขึ้นเป็นทักษะที่เป็นประโยชน์อย่างยิ่งต่อทุกๆ สาขาวิชาและปัญหาในชีวิตประจำวันซึ่งไม่ได้จำกัดอยู่เพียงการคิดให้เหมือนคอมพิวเตอร์แต่เป็นกระบวนการคิดแก้ปัญหาของมนุษย์เพื่อสั่งให้คอมพิวเตอร์ทำงานและช่วยแก้ปัญหาตามที่เรากำลังต้องการได้อย่างมีประสิทธิภาพ

แนวคิดเชิงคำนวณมีองค์ประกอบที่สำคัญ 4 ส่วน ได้แก่ การแบ่งปัญหาใหญ่เป็นปัญหาย่อย (Decomposition) การพิจารณารูปแบบ (Pattern Recognition) การคิดเชิงนามธรรม (Abstraction) การออกแบบอัลกอริทึม (Algorithm)



ภาพที่ 1 แนวคิดเชิงคำนวณ

ที่มา <https://www.bbc.co.uk/bitesize/guides/zqqfyrd/revision/1>, BBC

1. การแบ่งปัญหาใหญ่เป็นปัญหาย่อย (Decomposition)

เป็นการแยกส่วนประกอบเป็นวิธีคิดรูปแบบหนึ่งของแนวคิดเชิงคำนวณเป็นการพิจารณาเพื่อแบ่งปัญหาหรืองานออกเป็นส่วนย่อย ทำให้สามารถจัดการกับปัญหาหรืองานได้ง่ายขึ้นการแตกปัญหาที่ซับซ้อนให้เป็นปัญหาย่อยที่มีขนาดเล็กลงและซับซ้อนน้อยลง เพื่อช่วยให้การวิเคราะห์และออกแบบวิธีการแก้ปัญหาทำได้ง่ายขึ้น เช่น รถยนต์สตาร์ทไม่ติด

กรณีตัวอย่างการแบ่งปัญหาใหญ่เป็นปัญหาย่อย (Decomposition)



คุณครูกำลังจะเดินทางไปโรงเรียนในเช้าของวันหนึ่งแต่ไม่สามารถสตาร์ทรถยนต์ที่ใช้ทุกวันให้ติดได้ ช่วยครูในการหาวิธีการแก้ไขปัญหาลงต้นเนื่องจากคุณครูอาจไม่มีความชำนาญในการซ่อมบำรุงรถยนต์ การแบ่งปัญหาใหญ่เป็นปัญหาย่อยช่วยให้การออกแบบขั้นตอนการแก้ปัญหาทำได้เป็นระบบมากขึ้น การแบ่งปัญหาใหญ่เป็นปัญหาย่อยของปัญหา รถยนต์สตาร์ทไม่ติด อาจได้ดังนี้

1. แบตเตอรี่หมดหรือเสื่อมสภาพ
2. ไดชาร์จมีปัญหา
3. ปัญหาที่ระบบไฟฟ้าอื่นๆ
4. ระบบเซ็นเซอร์มีปัญหา
5. น้ำมันหมด
6. เกียร์อยู่ในสถานะที่พร้อมสำหรับการสตาร์ท

การแบ่งปัญหาใหญ่ให้เป็นปัญหาย่อยจะทำให้มีความซับซ้อนของปัญหาลดลงช่วยให้การวิเคราะห์และพิจารณารายละเอียดของปัญหาทำได้อย่างถี่ถ้วนส่งผลให้สามารถออกแบบขั้นตอนการแก้ปัญหาย่อยแต่ละปัญหาได้ง่ายขึ้น

2. การพิจารณารูปแบบ (Pattern Recognition)

เป็นการหารูปแบบซึ่งเป็นทักษะการหาความสัมพันธ์ที่เกี่ยวข้อง แนวโน้ม และลักษณะทั่วไปของสิ่งต่าง ๆ โดยพิจารณาว่าเคยพบปัญหาลักษณะนี้มาก่อนหรือไม่ หากมีรูปแบบของปัญหาที่คล้ายกันสามารถนำวิธีการแก้ปัญหานั้นมาประยุกต์ใช้ และพิจารณารูปแบบปัญหาย่อยซึ่งอยู่ภายในปัญหาเดียวกันว่ามีส่วนใดที่เหมือนกันเพื่อใช้วิธีการแก้ปัญหาเดียวกันได้ ทำให้จัดการกับปัญหาได้ง่ายขึ้น และการทำงานมีประสิทธิภาพเพิ่มขึ้น

กรณีตัวอย่างการพิจารณารูปแบบ (Pattern Recognition)



เมื่อพิจารณาจากปัญหาข้างต้นอาจพบว่า มีบางปัญหาที่เราสามารถตรวจสอบเบื้องต้นได้ เช่น เกียร์ในโหมดที่พร้อมสตาร์ท น้ำมัน และแบตเตอรี่และนอกนั้นเป็นส่วนที่ต้องอาศัยผู้เชี่ยวชาญในการตรวจสอบซึ่งการพิจารณาของแต่ละคนอาจแตกต่างกันออกไป

3. การคิดเชิงนามธรรม (Abstraction)

การคิดเชิงนามธรรม(Abstraction) เป็นองค์ประกอบหนึ่งของแนวคิดเชิงคำนวณซึ่งใช้กระบวนการคัดแยกคุณลักษณะที่สำคัญออกจากรายละเอียดปลีกย่อยในปัญหาหรืองานที่กำลังพิจารณาเพื่อให้ได้ข้อมูลที่จำเป็นและเพียงพอในการแก้ปัญหาเป็นการแยกรายละเอียดที่สำคัญและจำเป็นต่อการแก้ปัญหาออกจากรายละเอียดที่ไม่จำเป็นซึ่งรวมไปถึงการแทนกลุ่มของปัญหาขั้นตอนหรือกระบวนการที่มีรายละเอียดปลีกย่อยหลายขั้นตอนด้วยขั้นตอนใหม่เพียงขั้นตอนเดียว



สาระสำคัญในการแก้ไขปัญหาเบื้องต้นที่สามารถทำได้ด้วยตนเอง คือ ตรวจสอบน้ำมัน ตรวจสอบการเข้าเกียร์ในโหมดที่พร้อมสตาร์ท ตรวจสอบวันติดตั้งแบตเตอรี่เพื่อประมาณการอายุการใช้งาน

4. การออกแบบอัลกอริทึม (Algorithm)

เป็นการพัฒนากระบวนการหาคำตอบให้เป็นขั้นตอนที่บุคคลหรือคอมพิวเตอร์สามารถนำไปปฏิบัติตามเพื่อแก้ปัญหาได้ อีกทั้งเป็นการพัฒนาแนวทางแก้ปัญหาอย่างเป็นขั้นเป็นตอนเพื่อดำเนินตามทีละขั้นตอนในการแก้ไขปัญหา เช่นเมื่อเราต้องการสั่งคอมพิวเตอร์ให้ทำงานบางอย่างเราจะต้องเขียนโปรแกรมคำสั่งเพื่อให้คอมพิวเตอร์ทำงานไปตามขั้นตอน ตามแนวทางการแก้ปัญหาเพื่อให้คอมพิวเตอร์ทำงานตอบสนองความต้องการของเรา วิธีคิดนี้ที่เรียกว่าวิธีคิดแบบอัลกอริทึม คอมพิวเตอร์จะทำงานได้ดีเพียงใดนั้นขึ้นอยู่กับชุดคำสั่งอัลกอริทึมที่เราออกแบบให้มันทำงานนั่นเองการออกแบบอัลกอริทึมยังเป็นประโยชน์ต่อการคำนวณการประมวลผลข้อมูลและการวางระบบอัตโนมัติต่าง ๆ

กรณีตัวอย่างการออกแบบอัลกอริทึม (Algorithm) รถยนต์สตาร์ทไม่ติด

1. เริ่มต้น

2. สตาร์ทรถ

ถ้า รถสตาร์ทไม่ติด ทำ

2.1 ถ้า เกียร์ไม่พร้อมสำหรับการสตาร์ท ทำ

2.1.1 เปลี่ยนเกียร์ให้พร้อม

2.2 ถ้า น้ำมันหมด ทำ

2.2.1 เติมน้ำมัน

2.3 ถ้า แบตเตอรี่มีอายุใช้งานเกิน 2 ปี

2.3.1 โทรเรียกช่างแบตเตอรี่

ไม่เช่นนั้น

2.4 โทรปรึกษาช่างซ่อม พร้อมบอกปัญหาเบื้องต้น

3. จบการทำงาน

กรณีศึกษาการใช้แนวคิดเชิงคำนวณในการแก้ปัญหา

ตัวอย่างต่อไปนี้จะใช้แนวคิดเชิงคำนวณในการแก้ปัญหา บางปัญหาอาจไม่ได้ใช้ครบทุกองค์ประกอบ ขึ้นอยู่กับลักษณะของปัญหาแต่ทุกปัญหาจะต้องได้อันกอธิมในการแก้ไขปัญหามันถูกต้องรวดเร็วและมีประสิทธิภาพ

ตัวอย่าง การบอกเพื่อนเรียงลำดับตามความสูงจากมากไปน้อย



สมมติว่านักเรียนต้องการบอกเพื่อนให้เรียงลำดับความสูงจากมากไปน้อยให้นักเรียนออกแบบขั้นตอนการเรียงลำดับให้เพื่อนสามารถปฏิบัติตามได้ง่ายไม่ว่าจะมีจำนวนคนเท่าใดและมีลำดับเริ่มต้นแบบใดก็ได้

การแบ่งปัญหาใหญ่เป็นปัญหาย่อย

การพยายามจัดความสูงจากมากไปน้อยของเพื่อนทั้งห้องนั้นอาจเกิดความยุ่งยากการแบ่งปัญหาใหญ่เป็นปัญหาย่อยช่วยให้การออกแบบขั้นตอนการแก้ปัญหาทำได้เป็นระบบมากขึ้นโดยอาจแบ่งเป็นปัญหาย่อยได้ดังต่อไปนี้

เพื่อนกลุ่มใดควรจัดไว้เป็นลำดับแรก
เพื่อนกลุ่มที่เหลือ กลุ่มใดควรเลือกออกมาเป็นกลุ่มอยู่ในลำดับที่สอง
เพื่อนกลุ่มที่เหลือ กลุ่มใดควรเลือกออกมาเป็นกลุ่มอยู่ในลำดับที่สาม

จะเห็นได้ว่าปัญหาในการจัดความสูงของเพื่อนทั้งห้องสามารถแบ่งเป็นปัญหาย่อยได้โดยคัดเลือกกลุ่มเพื่อนที่สูงที่สุดจากเพื่อนทั้งห้อง (สมมติว่ากลุ่มเพื่อนมี n กลุ่ม) ทำให้ขนาดของกลุ่มเพื่อนลดลงเหลือ $n - 1$ กลุ่มปัญหาย่อยในที่นี้คือการจัดความสูงในห้องที่มี $n - 1$ กลุ่มซึ่งเป็นปัญหาในรูปแบบเดิมที่มีความซับซ้อนน้อยลง

การแตกปัญหาในการจัดความสูงจากมากไปน้อยของเพื่อนทั้งห้องจะได้ผลลัพธ์เป็นปัญหาย่อยดังนี้

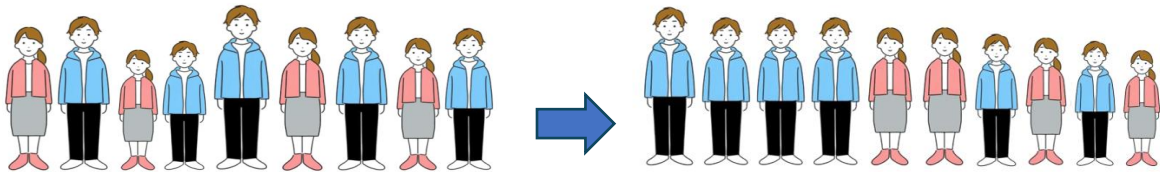
ปัญหาย่อยที่ 1 เพื่อนคนใดควรจัดไว้เป็นลำดับแรก

ปัญหาย่อยที่ 2 เพื่อนที่เหลือ คนใดควรเลือกออกมาเป็นอยู่ในลำดับที่สอง

ปัญหาย่อยที่ 3 เพื่อนคนที่เหลือ คนใดควรเลือกออกมาเป็นอยู่ในลำดับที่สาม

:

การพิจารณารูปแบบในการจัดความสูงจากมากไปน้อยของเพื่อนทั้งห้อง



จากการแก้ปัญหาในตัวอย่าง จัดความสูงจากมากไปน้อยของเพื่อนทั้งห้องนักเรียนอ่านอธิบายคำตอบของปัญหาย่อยได้ดังนี้

ปัญหาย่อยที่ 1 เพื่อนคนใดควรจัดไว้เป็นลำดับแรก

คำตอบ เพื่อนคนที่มีความสูงมากที่สุด

ปัญหาย่อยที่ 2 เพื่อนที่เหลือ คนใดควรเลือกออกมาอยู่ในลำดับที่สอง

คำตอบ เพื่อนที่มีความสูงมากที่สุดในกองที่เหลือ

ปัญหาย่อยที่ 3 เพื่อนที่เหลือ คนใดควรเลือกออกมาอยู่ในลำดับที่สาม

คำตอบ เพื่อนกลุ่มที่มีความสูงมากที่สุดในกองที่เหลือ

:

ปัญหาย่อยสุดท้ายที่มีขนาด 1

เห็นได้ว่าแต่ละปัญหาย่อยนั้นต่างก็มีรูปแบบเดียวกันและมุ่งหาคำตอบในลักษณะเดียวกัน

คือ ปัญหาย่อย ในห้องเพื่อนคนที่เหลือ คนใดควรเลือกออกมาอยู่ลำดับถัดไป คำตอบ เพื่อนคนที่มีความสูงมากที่สุดที่เหลืออยู่

การคิดเชิงนามธรรมในปัญหาจัดความสูงจากมากไปน้อยของเพื่อนทั้งห้อง

เนื่องจากขั้นตอนที่นำไปปฏิบัติตามต้องการเพียงการจัดความสูงจากมากไปน้อยของเพื่อนทั้งห้องรายละเอียดที่จำเป็นเพื่อใช้ในการตัดสินใจเลือกหนังสือจึงมีเพียงความสูงของเพื่อนในขณะที่เพศและน้ำหนักของเพื่อน ๆ ในห้องนั้นถือเป็นรายละเอียดที่ไม่จำเป็นจึงสามารถตัดออกไปในการออกแบบกระบวนการการแก้ปัญหา



ถ้านักเรียนต้องการใช้เพียงความสูงของเพื่อนแต่ละคนเพื่อใช้พิจารณาการจัดความสูงนักเรียนสามารถใช้ตัวเลขหนึ่งจำนวนแทนความสูงของกลุ่มเพื่อนแต่ละกลุ่มเพื่อใช้ในการออกแบบอัลกอริทึมโดยจากปัญหาย่อยที่เคยตั้งเอาไว้ก่อนหน้านี้ว่า

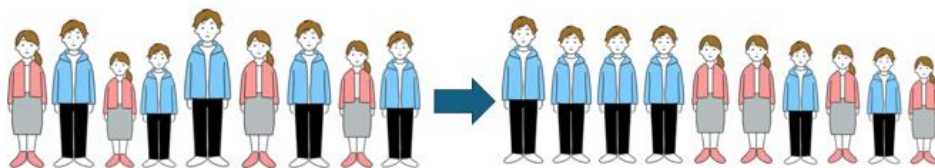
ในห้องเพื่อนที่เหลือคนใดควรเลือกออกมาอยู่ลำดับถัดไป

มีกระบวนการการแก้ปัญหาแบบเดียวกันกับปัญหาที่ระบุว่า

ในชุดจำนวนที่พิจารณาอยู่จำนวนใดมีค่ามากที่สุด

อัลกอริทึมสำหรับจัดความสูงจากมากไปน้อยของเพื่อนทั้งห้อง

กระบวนการที่ผ่านมาสามารถนำมาออกแบบเป็นอัลกอริทึมสำหรับให้นักเรียนปฏิบัติตามได้ดังนี้



ขั้นตอนหลัก

1. ทำขั้นตอนต่อไปนี้อ้างจนกระทั่งไม่มีกลุ่มเพื่อนเหลืออยู่ในห้อง
 - 1.1 เลือกเพื่อนที่มีความสูงมากที่สุดในห้อง
 - 1.2 นำเพื่อนที่เลือกจากขั้นตอนที่ 1.1

เรียงไว้หน้าห้องเรียนโดยเรียงไว้ถัดจากเพื่อนที่ยืนไว้แล้วก่อนหน้านี้ถ้ายังไม่มีให้นำเพื่อนคนนี้ไว้เป็นคนแรก