หลักการแก้ปัญหา

***หลักการแก้ปัญหา***

***กับภาษาคอมพิวเตอร์***

***หลักการแก้ปัญหาและการโปรแกรม***

**1.หลักการแก้ปัญหา**

          ในชีวิตประจำวันทุกคนต้องเคยพบกับปัญหาต่างๆ ไม่ว่าจะเป็นปัญหาด้านการเรียน การงาน การเงิน หรือแม้แต่การเล่นเกม เมื่อพบกับปัญหา แต่ละคนมีวิธีที่จะจัดการหรือแก้ปัญหาเหล่านั้นแตกต่างกันไป ซึ่งแต่ละวิธีการอาจให้ผลลัพธ์ที่เหมือนหรือแตกต่างกันเล็กน้อย  ทั้งนี้ขึ้นอยู่กับความรู้  ความสามารถ และประสบการณ์ของบุคคลผู้นั้น  อย่างไรก็ตาม หากเรานำวิธีการแก้ปัญหาต่างวิธีนั้นมาวิเคราะห์ให้ดี  จะพบว่าสามารถสรุปวิธีการเหล่านั้นเป็นทฤษฎีซึ่งมีรูปแบบที่แน่นอนได้  และบางครั้งต้องอาศัยการเรียนรู้ในระดับสูงเพื่อแก้ปัญหาบางอย่างให้สมบูรณ์แบบ  แต่ก่อนที่เราจะศึกษาต่อไป  ลองพิจารณาปัญหาต่อไปนี้

**ตัวอย่างที่ 1**   เกมทายใจ

คือเกมให้ผู้เล่นทายตัวเลข 3 ตัว  ในการเล่นเกมต้องใช้ผู้เล่น 2 คน  คนที่หนึ่งคือ ผู้กำหนด เป็นคนกำหนดเลข 3 ตัวที่ไม่ซ้ำกัน โดยเลือกจากกลุ่มตัวเลข 1-9  และอีกคนหนึ่งคือผู้ทาย  เป็นผู้ทายตัวเลข 3 ตัวที่ไม่ซ้ำกันที่ผู้กำหนดได้กำหนดไว้แล้ว  หลังจากที่ผู้ทายทายเลขแต่ละครั้ง  ผู้กำหนดต้องให้รายละเอียดว่าตัวเลขที่ทายมานั้นถูกต้องกี่ตัว  และในกรณีที่ตัวเลขที่ทายมาถูกตำแหน่งด้วยก็ต้องบอกว่าถูกตำแหน่งกี่ตัว  เช่น  ถ้าตัวเลขที่กำหนดไว้เป็น 815 และผู้ทายทายว่า 123 ผู้กำหนดต้องแจ้งว่าตัวเลขที่ทายนั้นถูก 1 ตัว และไม่มีตัวใดถูกตำแหน่ง  ตารางที่ 1 เป็นตารางแสดงข้อมูลการเล่นเกม

**ตารางที่ 1** ข้อมูลการเล่นเกมทายใจ

|  |  |  |
| --- | --- | --- |
| เลขที่ทาย | จำนวนตัวเลขที่ถูก | จำนวนตำแหน่งที่ถูก |
| 123415425416715815 | 121123 | -21123 |

จะเห็นว่าการแก้ปัญหาดังกล่าวข้างต้น  นอกจากจะใช้วิธีลองผิดลองถูกในการทายครั้งแรกๆ แล้วยังมีการใช้เหตุผลประกอบการแก้ปัญหาซึ่งเราเรียกวิธีการดังกล่าวว่า *“วิธีขจัด”* *(method of elimination)*คือ จะแยกข้อมูลออกเป็นกรณีที่เป็นไปไม่ได้ทิ้ง จนเหลือกรณีที่เป็นไปได้ รูปแบบของการใช้เหตุผลประกอบการแก้ปัญหาอาจแตกต่างกันขึ้นอยู่กับเงื่อนไข ในบางปัญหาอาจจะขจัดให้เหลือกรณีเดียวไม่ได้ แต่อาจจะทำให้เหลือกรณีน้อยที่สุด

          นอกจากวิธีการแก้ปัญหาที่ยกตัวอย่างมาซึ่งได้แก่  วิธีการลองผิดลองถูก  การใช้เหตุผล  การใช้วิธีขจัด  ยังมีวิธีการแก้ปัญหาอีกมากมายที่ผู้แก้ปัญหาสามารถเลือกใช้ให้เข้ากับตัวปัญหาและประสบการณ์ของผู้แก้ปัญหาเอง  แต่อย่างไรก็ตาม  วิธีการเหล่านั้นล้วนมีขั้นตอนที่คล้ายคลึงกัน  และจากการศึกษาพฤติกรรมในการเรียนรู้และแก้ปัญหาของมนุษย์พบว่า โดยปกติมนุษย์มีกระบวนการในการแก้ปัญหา  ซึ่งประกอบด้วย 4 ขั้นตอน ดังนี้

**j การวิเคราะห์และกำหนดรายละเอียดของปัญหา**

          การวิเคราะห์และกำหนดรายละเอียดของปัญหา (State the problem) ขั้นตอนนี้เป็นขั้นตอนแรกสุดก่อนที่จะลงมือแก้ปัญหา แต่ผู้แก้ปัญหามักจะมองข้ามความสำคัญของขั้นตอนนี้อยู่เสมอ จุดประสงค์ของขั้นตอนนี้ คือการทำความเข้าใจกับปัญหาเพื่อแยกให้ออกว่าข้อมูลที่กำหนดมาในปัญหาหรือเงื่อนไขของปัญหาคืออะไร และสิ่งที่ต้องการคืออะไร อีกทั้งวิธีการที่ใช้ประมวลผล ในการวิเคราะห์ปัญหาใด กล่าวโดยสรุปมีองค์ประกอบในการวิเคราะห์ ดังนี้

**1.1  การระบุข้อมูลเข้า**  ได้แก่ การพิจารณาข้อมูลและเงื่อนไขที่กำหนดมาในปัญหา

**1.2  การระบุข้อมูลออก**  ได้แก่ การพิจารณาเป้าหมายหรือสิ่งที่ต้องหาคำตอบ

**1.3  การกำหนดวิธีประมวลผล**  ได้แก่ การพิจารณาขั้นตอนวิธีหาคำตอบหรือข้อมูลออก

**ตัวอย่างที่ 2** แสดงการวิเคราะห์และกำหนดรายละเอียดของการหาค่าเฉลี่ยของจำนวนเต็ม 5 จำนวน ได้แก่ 0 3 4 8 และ 12

          จากองค์ประกอบในการวิเคราะห์และกำหนดรายละเอียดของปัญหา

*(1)   การระบุข้อมูลเข้า*

ในที่นี้โจทย์กำหนดให้หาค่าเฉลี่ยของจำนวนเต็ม 5 จำนวน ดังนั้น ข้อมูลเข้าได้แก่ จำนวน 0 3 4 8 และ 12

*(2)   การระบุข้อมูลออก*

จากโจทย์สิ่งที่เป็นคำตอบของปัญหาคือ ค่าเฉลี่ย (X) ของจำนวนทั้งห้า

*(3)   การกำหนดวิธีประมวลผล*

จากสิ่งที่โจทย์ต้องการ “ค่าเฉลี่ย” หมายถึง ผลรวมของจำนวนทั้ง 5 หารด้วย 5 ดังนั้น ขั้นตอนของการประมวลผลประกอบด้วย

3.1) รับค่าจำนวนทั้ง 5 จำนวน

3.2) นำจำนวนเต็มทั้ง 5 มาบวกเข้าด้วยกัน

3.3) นำผลลัพธ์จากข้อ 3.2) มาหารด้วย 5

**ตัวอย่างที่ 3** แสดงการวิเคราะห์และกำหนดรายละเอียดของการหาค่า X เมื่อ X คือจำนวนเต็มจำนวนหนึ่งในกลุ่มจำนวนเต็ม 5 จำนวน ที่มีค่าเฉลี่ยเป็น 10 และจำนวนอีก 4 จำนวนได้แก่ 3 4 8 และ 12

          จากองค์ประกอบในการวิเคราะห์และกำหนดรายละเอียดของปัญหา

*(1)   การระบุข้อมูลเข้า*

จากโจทย์ข้อมูลเข้า ได้แก่

1.1)  จำนวนอีก 4 จำนวน คือ 3 4 8 12

1.2)  ค่าเฉลี่ยของจำนวนทั้ง 5 จำนวน คือ 10

*(2)   การระบุข้อมูลออก*

จากโจทย์สิ่งที่เป็นผลลัพธ์ คือ ค่า X

*(3)   การกำหนดวิธีประมวลผล*

จากโจทย์และความหมายของ “ค่าเฉลี่ย” เราสามารถสรุปขั้นตอนของการประมวลผลได้ดังนี้

3.1)  หาค่าผลรวมของจำนวนเต็มทั้ง 5 โดยนำค่าเฉลี่ยคูณด้วยจำนวนของเลขจำนวนเต็ม นั่นคือ 10 x 5 = 50

3.2)  จากความหมายของ “ผลรวม” จะได้ 3+4+8+12+X = 50

3.3)  แก้สมการ 27+X = 50 (จะได้ X = 23 ซึ่งคือผลลัพธ์)

**k การเลือกเครื่องมือและออกแบบขั้นตอนวิธี**

              การเลือกเครื่องมือและออกแบบขั้นตอนวิธี (Tools and Algorithm development) ขั้นตอนนี้เป็นขั้นตอนของการวางแผนในการแก้ปัญหาอย่างละเอียดถี่ถ้วน หลังจากที่เราทำความเข้าใจกับปัญหา พิจารณาข้อมูลและเงื่อนไขที่มีอยู่ และสิ่งที่ต้องการหาในขั้นตอนที่ 1 แล้ว เราสามารถคาดคะเนวิธีการที่จะใช้ในการแก้ปัญหา ขั้นตอนนี้จำเป็นต้องอาศัยประสบการณ์ของผู้แก้ปัญหาเป็นหลัก หากผู้แก้ปัญหาเคยพบกับปัญหาทำนองนี้มาแล้วก็สามารถดำเนินการตามแนวทางที่เคยปฏิบัติมา

              ขั้นตอนนี้จะเริ่มจากการเลือกเครื่องมือที่ใช้ในการแก้ปัญหา โดยพิจารณาความเหมาะสมระหว่างเครื่องมือกับเงื่อนไขต่างๆ ของปัญหาซึ่งหมายรวมถึงความสามารถของเครื่องมือในการแก้ปัญหาดังกล่าว และสิ่งที่สำคัญคือความคุ้นเคยในการใช้งานเครื่องมือนั้นๆ ของผู้แก้ปัญหา

              อีกสิ่งหนึ่งที่สำคัญในการแก้ปัญหา คือยุทธวิธีที่ใช้ในการแก้ปัญหาหรือที่เราเรียกว่า ***ขั้นตอนวิธี (algorithm)*** ในการแก้ปัญหา  หลังจากที่เราได้เครื่องมือช่วยแก้ปัญหาแล้ว ผู้แก้ปัญหาต้องวางแผนว่าจะใช้เครื่องมือดังกล่าวเพื่อให้ได้ผลลัพธ์ที่ถูกต้องและดีที่สุด การออกแบบขั้นตอนวิธีในการแก้ปัญหา ผู้แก้ปัญหาควรใช้แผนภาพหรือเครื่องมือในการแสดงขั้นตอนการทำงานเพื่อให้ง่ายต่อความเข้าใน เช่น ผังงาน (flowchart) ที่จำลองขั้นตอนวิธีการแก้ปัญหาในรูปของสัญลักษณ์  รหัสลำลอง (pseudo code) ซึ่งเป็นการจำลองขั้นตอนวิธีการแก้ปัญหาในรูปของคำบรรยาย การใช้เครื่องมือช่วยออกแบบดังกล่าวนอกจากแสดงกระบวนการที่ชัดเจนแล้ว ยังช่วยให้ผู้แก้ปัญหาสามารถหาข้อผิดพลาดของวิธีการที่ใช้ได้ง่ายและแก้ไขได้อย่างรวดเร็ว

**l การดำเนินการแก้ปัญหา**

              การดำเนินการแก้ปัญหา (Implementation) หลังจากที่ได้ออกแบบขั้นตอนวิธีเรียบร้อยแล้ว ขั้นตอนนี้เป็นขั้นตอนที่ต้องลงมือแก้ปัญหาโดยใช้เครื่องมือที่ได้เลือกไว้ หากการแก้ปัญหาดังกล่าวใช้คอมพิวเตอร์เข้ามาช่วยง่าน ขั้นตอนนี้ก็เป็นการใช้โปรแกรมสำเร็จหรือใช้ภาษาคอมพิวเตอร์เขียนโปรแกรมแก้ปัญหา ขั้นตอนนี้ต้องอาศัยความรู้เกี่ยวกับเครื่องมือที่เลือกใช้ซึ่งผู้แก้ปัญหาต้องศึกษาให้เข้าใจและเชี่ยวชาญ ในขณะที่ดำเนินการหากพบแนวทางที่ดีกว่าที่ออกแบบไว้ก็สามารถปรับเปลี่ยนได้

**m การตรวจสอบและปรับปรุง**

              การตรวจสอบและปรับปรุง (Refinement) หลังจากที่ลงมือแก้ปัญหาแล้ว ต้องตรวจสอบให้แน่ใจว่าวิธีการนี้ให้ผลลัพธ์ที่ถูกต้อง โดยผู้แก้ปัญหาต้องตรวจสอบว่าขั้นตอนวิธีที่สร้างขึ้นสอดคล้องกับรายละเอียดของปัญหา ซึ่งได้แก่ ข้อมูลเข้า และข้อมูลออก เพื่อให้มั่นใจว่าสามารถรองรับข้อมุเข้าได้ในทุกกรณีอย่างถูกต้องและสมบูรณ์ ในขณะเดียวกันก็ต้องปรับปรุงวิธีการเพื่อให้การแก้ปัญหานี้ได้ผลลัพธ์ที่ดีที่สุด

              ขั้นตอนทั้ง 4 ขั้นตอนดังกล่าวข้างต้น เป็นเสมือนขั้นบันได (stair) ที่ทำให้มนุษย์สามารถประสบความสำเร็จในการแก้ปัญหาต่างๆ ได้ รวมทั้งการเขียนหรือพัฒนาโปรแกรมคอมพิวเตอร์เพื่อแก้ปัญหาก็ต้องใช้กระบวนการตามขั้นตอนทั้ง 4 นี้เช่นกัน

**2. การจำลองความคิด**

          ขั้นตอนที่สำคัญในการแก้ปัญหาคือการวางแผน การวางแผนที่ดีจะช่วยให้การแก้ปัญหาเป็นไปได้โดยง่าย ผู้ที่สามารถวางแผนในการแก้ปัญหาได้ดีนอกจากจะต้องใช้ประสบการณ์ความรู้ และความมีเหตุผลแล้ว ยังควรรู้จักวางแผนให้เป็นขั้นตอนอย่างเป็นระเบียบด้วย

          การจำลองความคิดเป็นส่วนหนึ่งในขั้นตอนที่สองของการแก้ปัญหา การจำลองความคิดออกมาในลักษณะเป็นข้อความ หรือเป็นแผนภาพจะช่วยให้สามารถแก้ปัญหาได้ดี โดยเฉพาะปัญหาที่ยุ่งยากซับซ้อน การวางแผนจะเป็นแนวทางในการดำเนินการแก้ปัญหาต่อไป อีกทั้งเป็นการแสดงแบบเพื่อให้ผู้ที่เกี่ยวข้องได้เข้าใจและสามารถปฏิบัติตามในแนวทางเดียวกัน

          ทั้งนี้ก็ด้วยวัตถุประสงค์อย่างเดียวกับกลุ่มกิจการก่อสร้าง ซึ่งจำเป็นต้องมีแบบแปลนเป็นเครื่องมือติดต่อสื่อสารระหว่างผู้ออกแบบและผู้ก่อสร้าง แบบแปลนเหล่านั้นจะอยู่ในรูปลักษณะของการวาดภาพหรือแสดงเครื่องหมายซึ่งเป็นที่เข้าใจกันระหว่างผู้เกี่ยวข้อง แบบแปลนจะต้องจัดทำให้เสร็จก่อนที่จะลงมือก่อสร้าง โดยผ่านการตรวจสอบ ทบทวนและพิจารณาจากผู้เที่ยวข้องหลายฝ่าย เมื่อเห็นว่าเป็นที่ถูกต้องและพอใจของทุกฝ่ายแล้ว จึงก่อสร้างตามแบบนั้น แต่ถ้ายังไม่เป็นที่พอใจ ก็จะพิจารณาแก้ไขแบบแปลนส่วนนั้นๆ เสียก่อนจะได้ไม่ต้องรื้อถอนหรือทุบทิ้งภายหลัง และเมื่อต้องการซ่อมแซมหรือต่อเติมก็นำเอาแบบแปลนเดิมมาตรวจสอบและเพิ่มแบบแปลนในส่วนนั้นได้โดยง่าย การใช้แบบแปลนจึงเป็นสิ่งที่จำเป็นระหว่างช่างก่อสร้าง ผู้ออกแบบและผู้เกี่ยวข้องอื่นๆ เป็นอย่างมาก เพราะประหยัดเวลา ค่าใช้จ่ายและเข้าใจง่าย เมื่อสรุปรวมแล้วแบบแปลนเหล่านั้นก็คือข้อตกลงให้สร้างอาคารของผู้จ้างกับผู้รับจ้างที่อยู่ในรูปแบบกะทัดรัด แทนที่จะเขียนเป็นข้อความที่เป็นลายลักษณ์อักษรอย่างยืดยาว และยังเป็นเครื่องมือให้ช่างใช้ในการก่อสร้างอีกด้วย

          เครื่องมือที่ใช้ในการจำลองความคิดมักจะประกอบขึ้นด้วยเครื่องหมายที่แตกต่างกันหลายอย่าง แต่พอสรุปได้เป็น 2 ลักษณะ คือ

**j ข้อความหรือคำบรรยาย**

          เป็นการเขียนเค้าโครงด้วยการบรรยายเป็นภาษาที่มนุษย์ใช้สื่อสารกัน เพื่อให้ทราบถึงขั้นตอนการทำงานของการแก้ปัญหาแต่ละตอน ในบางครั้งอาจใช้คำสั่งของภาษาที่ใช้เขียนโปรแกรมก็ได้

**ตัวอย่างที่ 4** คำบรรยายแสดงขั้นตอนการเปลี่ยนยางรถเมื่อยางแตกขณะขับรถ

(1)   จอดรถหลบข้างทาง

(2)   คลายสกรูยึดล้อ

(3)   นำแม่แรงออกยกรถ

(4)   ถอดล้อออก นำยางอะไหล่มาเปลี่ยน

(5)   ขันสกรูเข้า เก็บยางที่ชำรุดเพื่อไปซ่อม

(6)   คลายแม่แรง เก็บแม่แรง

**k สัญลักษณ์**

          เครื่องหมายรูปแบบต่างๆ ซึ่งใช้สำหรับสื่อสารความหมายให้เข้าใจตรงกัน สถาบันมาตรฐานแห่งชาติอเมริกา (The American National Standard Institute, ANSI) ได้กำหนดสัญลักษณ์ไว้เป็นมาตรฐานแล้ว สามารถนำไปใช้ได้ตามความเหมาะสมต่อไป ซึ่งมีรายละเอียด รูปแบบและความหมายที่ควรทราบตามตารางต่อไปนี้

**ตารางที่ 2** ความหมายของสัญลักษณ์

|  |  |  |
| --- | --- | --- |
| **สัญลักษณ์** | **ชื่อเรียก** | **ความหมาย** |
| http://www.mwit.ac.th/~jeab/sheet40102/intro_files/image002.gif | การทำงานด้วยมือ(manual operation) | แทนจุดที่มีการทำงานด้วยแรงคน |
| http://www.mwit.ac.th/~jeab/sheet40102/intro_files/image003.gif | การนำข้อมูลเข้า – ออกโดยทั่วไป(general input/output) | แทนจุดที่จำนำข้อมูลเข้าหรือออกจากระบบคอมพิวเตอร์โดยไม่ระบุชนิดของอุปกรณ์ |
| http://www.mwit.ac.th/~jeab/sheet40102/intro_files/image004.gif | แถบบันทึกข้อมูล(magnetic tape) | แทนจุดที่นำข้อมูลเข้าหรือออกจากโปรแกรมด้วยแถบบันทึกข้อมูล |
| http://www.mwit.ac.th/~jeab/sheet40102/intro_files/image005.gif | จานบันทึกข้อมูล(magnetic disk) | แทนจุดที่นำข้อมูลเข้าหรือออกจากโปรแกรมด้วยจานบันทึกข้อมูล |
| http://www.mwit.ac.th/~jeab/sheet40102/intro_files/image006.gif | การนำข้อมูลเข้าด้วยมือ(manual input) | แทนจุดที่จะนำข้อมูลเข้าด้วยมือ |
| http://www.mwit.ac.th/~jeab/sheet40102/intro_files/image007.gif | การแสดงข้อมูล(display) | แทนจุดที่แสดงข้อมูลด้วยจอภาพ |
| http://www.mwit.ac.th/~jeab/sheet40102/intro_files/image008.gif | การทำเอกสาร(document) | แทนจุดที่มีข้อมูลเป็นเอกสารหรือแสดงข้อมูลด้วยเครื่องพิมพ์ |
| http://www.mwit.ac.th/~jeab/sheet40102/intro_files/image009.gif | การตัดสินใจ(decision) | แทนจุดที่จะต้องเลือกปฏิบัติอย่างใดอย่างหนึ่ง |
| http://www.mwit.ac.th/~jeab/sheet40102/intro_files/image010.gif | การปฏิบัติงาน(process) | แทนจุดที่มีการปฏิบัติงานอย่างใดอย่างหนึ่ง |
| http://www.mwit.ac.th/~jeab/sheet40102/intro_files/image011.gif | การเตรียมการ(preparation) | แทนจุดกำหนดชื่อข้อมูลหรือค่าเริ่มต้นต่างๆ |
| http://www.mwit.ac.th/~jeab/sheet40102/intro_files/image012.gif | การเรียกโปรแกรมภายนอก(external subroutine) | แทนจุดเรียกใช้โปรแกรมย่อยที่ไม่ได้อยู่ในโปรแกรมนั้น |
| http://www.mwit.ac.th/~jeab/sheet40102/intro_files/image013.gif | การเรียกโปรแกรมภายใน(internal subroutine) | แทนจุดเรียกใช้โปรแกรมย่อยที่อยู่ในโปรแกรมนั้น |
| http://www.mwit.ac.th/~jeab/sheet40102/intro_files/image014.gif | การเรียงข้อมูล(sort) | แทนจุดที่มีการเรียงข้อมูลใหม่ตามข้อกำหนด |
| http://www.mwit.ac.th/~jeab/sheet40102/intro_files/image015.gif | ทิศทาง(flow line) | แทนทิศทางขั้นตอนการดำเนินงานซึ่งจะปฏิบัติต่อเนื่องกันตามหัวลูกศรชี้ |
| http://www.mwit.ac.th/~jeab/sheet40102/intro_files/image016.gif | หมายเหตุ(annotation) | แทนจุดที่แสดงรายละเอียดเพิ่มเติมหรือหมายเหตุของจุดต่างๆ ที่แสดงในผังงานด้วยสัญลักษณ์ไม่ชัดเจน |
| http://www.mwit.ac.th/~jeab/sheet40102/intro_files/image017.gif | การติดต่อทางไกล(communication link) | แทนช่วงที่มีการติดต่อหรือย้ายข้อมูลด้วยระบบการติดต่อทางไกล |
| http://www.mwit.ac.th/~jeab/sheet40102/intro_files/image018.gif | จุดเชื่อมต่อ(connector) | แทนจุดเชื่อมต่อของผังงานเมื่อใช้สัญลักษณ์เพื่อให้ดูง่าย |
| http://www.mwit.ac.th/~jeab/sheet40102/intro_files/image019.gif | จุดเชื่อมต่อหน้ากระดาษ(off page connector) | แทนจุดเชื่อมต่อของผังงานที่อยู่คนละหน้ากระดาษ |
| http://www.mwit.ac.th/~jeab/sheet40102/intro_files/image020.gif | เริ่มต้นและลงท้าย(terminal) | แทนจุดเริ่มต้นและลงท้ายของผังงานของโปรแกรมหลักและโปรแกรมย่อย |

 **ตัวอย่างที่ 5** การวางแผนการไปโรงเรียน

          **การจำลองความคิดเป็นข้อความ**

                   เริ่มต้น

                             ตื่นนอน

                             อาบน้ำ

                             ไปโรงเรียน

                   จบ

          **การจำลองความคิดเป็นสัญลักษณ์**



**ตัวอย่างที่ 6** การจำลองความคิดในการหาผลบวกของ 1, 2, 3, …, 20 (นั่นคือ จะหาค่า 1+2+3+…+20)

          **การจำลองความคิดเป็นข้อความ**

                   เริ่มต้น

1.      กำหนดให้ N มีค่าเริ่มต้นเป็น 0

2.      กำหนดให้ K มีค่าเริ่มต้นเป็น 1

3.      นำค่า K มารวมกับค่า N เดิม ได้ผลลัพธ์เท่าไรไปเก็บไว้ที่ N

4.      นำค่า 1 มารวมกับค่า K เดิม ได้ผลลัพธ์เท่าไรไปเก็บไว้ที่ K

5.      เปรียบเทียบค่า K กับ 20 ถ้า K น้อยกว่าหรือเท่ากับ 20 ให้วนกลับไปทำในขั้นที่ 3 และทำคำสั่งถัดลงมาตามลำดับ แต่ถ้า K มากกว่า 20 ให้แสดงคำตอบ

                   จบ

          **การจำลองความคิดเป็นสัญลักษณ์**

 

**3. การเขียนโปรแกรม**

          จากการศึกษาหลักการขั้นตอนการแก้ปัญหาในหัวข้อที่ 1 และ 2 ที่ผ่านมา หลังจากที่เราสามารถวิเคราะห์ปัญหา และสร้างแบบจำลองความคิดเพื่อแสดงขั้นตอนในการแก้ปัญหาแล้ว ขั้นตอนต่อไปคือการลงมือแก้ปัญหาตามขั้นตอนที่ออกแบบไว้ โดยใช้เครื่องมือช่วยในการแก้ปัญหา ในที่นี้หากเครื่องมือที่นักเรียนเลือกคือภาษาคอมพิวเตอร์ ขั้นตอนในการลงมือแก้ปัญหาก็คือขั้นตอนของการเขียนโปรแกรมคอมพิวเตอร์ซึ่งถือได้ว่าเป็นขั้นตอนหนึ่งที่สำคัญในการแก้ปัญหาด้วยคอมพิวเตอร์

          การเขียนโปรแกรม (Programming) หมายถึง กระบวนการใช้ภาษาคอมพิวเตอร์เพื่อกำหนดโครงสร้างของข้อมูล และกำหนดขั้นตอนวิธีเพื่อใช้แก้ปัญหาตามที่ได้ออกแบบไว้ โดยอาศัยหลักเกณฑ์การเขียนโปรแกรมคอมพิวเตอร์ของแต่ละภาษา

          ก่อนการเขียนโปรแกรม ผู้พัฒนาโปรแกรมจะต้องเลือกภาษาคอมพิวเตอร์ที่จะนำมาใช้ช่วยงานโดยพิจารณาจากปัจจัยต่างๆ ในการทำงาน เช่น ลักษณะของปัญหา ความถนัดของผู้เขียนโปรแกรม สภาพแวดล้อมในการทำงานของระบบคอมพิวเตอร์ เป็นต้น เนื่องจากในปัจจุบันมีภาษาคอมพิวเตอร์ให้เลือกใช้ได้หลายภาษา เช่น ภาษาปาสคาล ภาษาซี ภาษาจาวา ภาษาเดลฟาย เป็นต้น ถึงแม้แต่ละภาษาจะมีรูปแบบและหลักการในการสร้างงานที่แตกต่างกัน แต่ทุกภาษาจะต้องมีโครงสร้างควบคุมหลักทั้ง 3 แบบ ได้แก่ โครงสร้างแบบลำดับ (sequential structure) โครงสร้างแบบมีทางเลือก (selection structure) และโครงสร้างแบบทำซ้ำ (repetition structure)

**j โครงสร้างแบบลำดับ (Sequential structure)**

          คือ โครงสร้างแสดงขั้นตอนการทำงานที่เป็นไปตามลำดับก่อนหลัง และแต่ละขั้นตอนจะถูกประมวลผลเพียงครั้งเดียวท่านั้น สามารถแสดงการทำงานของโครงสร้างนี้ โดยใช้ผังงานได้ดังรูปที่ 1



**รูปที่ 1** การทำงานของโครงสร้างแบบลำดับ

**k โครงสร้างแบบมีทางเลือก** **(Selection structure)**

          คือ โครงสร้างที่มีเงื่อนไข ขั้นตอนการทำงานบางขั้นตอนต้องมีการตัดสินใจเพื่อเลือกวิธีการประมวลผลขั้นต่อไป และจะมีบางขั้นตอนที่ไม่ได้รับการประมวลผล การตัดสินใจอาจมีทางเลือก 2 ทางหรือมากกว่าก็ได้ โครงสร้างที่มีทางเลือกเพียง 2 ทาง เราเรียกชื่อว่าโครงสร้างแบบ if…then…else และโครงสร้างที่มีทางเลือกมากกว่า 2 ทาง เราเรียกชื่อว่า โครงสร้างแบบ case ซึ่งสามารถแสดงการทำงานของโครงสร้างนี้โดยใช้ผังงานได้ดังรูปที่ 2 และ 3



**รูปที่ 2** การทำงานของโครงสร้างแบบมีทางเลือก if…then…else

|  |  |
| --- | --- |
|

|  |
| --- |
| กรณีที่ 4 |

  |



**รูปที่ 3** การทำงานของโครงสร้างแบบมีทางเลือก case

**l โครงสร้างแบบทำซ้ำ (Repetition structure)**

          คือ โครงสร้างที่ขั้นตอนการทำงานบางขั้นตอนได้รับการประมวลผลมากกว่า 1 ครั้ง ทั้งนี้ขึ้นอยู่กับเงื่อนไขบางประการ โครงสร้างแบบทำซ้ำนี้ต้องมีการตัดสินใจในการทำงานซ้ำ และลักษณะการทำงานของโครงสร้างแบบนี้มี 2 ลักษณะ ได้แก่

·   แบบที่มีการตรวจสอบเงื่อนไขในการทำซ้ำทุกครั้งก่อนดำเนินการกิจกรรมใดๆ ถ้าเงื่อนไขเป็นจริงจะทำงานซ้ำไปเรื่อยๆ และหยุดเมื่อเงื่อนไขเป็นเท็จ เรียกการทำงานลักษณะนี้ว่า **การทำซ้ำแบบ do while**

·   แบบที่ทำกิจกรรมซ้ำเรื่อยๆ จนกระทั่งเงื่อนไขที่กำหนดเป็นจริงแล้วจึงหยุดการทำงาน โดยแต่ละครั้งที่เสร็จสิ้นการดำเนินการแต่ละรอบจะต้องมีการตรวจสอบเงื่อนไข เรียกการทำซ้ำลักษณะนี้ว่า **การทำซ้ำแบบ do until**

ผังงานแสดงขั้นตอนการทำงานของโครงสร้างแบบทำซ้ำทั้งสองแบบ แสดงดังรูปที่ 4 และ 5



**รูปที่ 4** การทำงานของการทำซ้ำแบบ do while



**รูปที่ 5** การทำงานของการทำซ้ำแบบ do until

**ตัวอย่างที่ 7** แสดงผังงานที่จำลองขั้นตอนวิธีการหาค่าเฉลี่ยของจำนวนเต็ม 5 จำนวน ให้อยู่ในรูปของสัญลักษณ์

