ใบความรู้ที่2.1 ภาษาคอมพิวเตอร์

1.ความหมายของภาษาคอมพิวเตอร์

ภาษาคอมพิวเตอร์มีอยู่เป็นจำนวนมาก แต่เครื่องคอมพิวเตอร์จะทำงานได้ภาษาเดียว คือ ภาษาเครื่อง..(Machine.language) ซึ่งเป็นภาษาที่บังคับการทำงานของเครื่องอย่าง แท้จริง ข้อเสียของภาษาเครื่อง คือ ภาษาของแต่ละเครื่องจะไม่เหมือนกัน ทั้งนี้แล้วแต่การ ออกแบบระบบเครื่องว่าจะเป็นแบบใด นอกจากนี้ เครื่องคอมพิวเตอร์แต่ละเครื่องที่สร้าง ขึ้นมานั้นส่วนใหญ่จะมีการแก้ไขปรับปรุงให้ดีขึ้นอยู่เสมอ ซึ่งจะทำให้ลักษณะของภาษาเครื่อง เปลี่ยนแปลงแตกต่างออกไปอีก แต่ภาษาที่ไม่ใช่ภาษาเครื่องอาจจะยังเหมือนเดิมได้ ดังนั้น.ในการสร้างเครื่องจึงมีความจำเป็นที่จะต้องคำนึงถึงประโยชน์ที่จะได้จาก การใช้ภาษาของเครื่องที่สร้างขึ้นมานั้นด้วย รวมทั้งต้องมีระบบการทำงานที่กว้างขวางพอที่จะ รับงานประเภทต่าง ๆ ได้อย่างมีประสิทธิภาพ

2. วิวัฒนาการของภาษาคอมพิวเตอร์

ภาษาคอมพิวเตอร์มีการพัฒนาหรือมีวิวัฒนาการโดยลำดับเช่นเดียวกับคอมพิวเตอร์ โดยจะสามารถแบ่งออกเป็นยุคหรือเป็นรุ่นของภาษา (Generation) ซึ่งในยุคหลัง ๆ จะมีการพัฒนาภาษาให้มีความสะดวกในการอ่านและเขียนง่ายขึ้นกว่าภาษาในยุคแรก ๆ เนื่องจากจะมีโครงสร้างภาษาใกล้เดียงกับภาษาอังกฤษ

สามารถแบ่งภาษาคอมพิวเตอร์ออกได้เป็น 5 ยุค

1) ภาษาเครื่อง (Machine Language)

2) ภาษาแอสเซมบลี (Assembly Language)

3) ภาษาชั้นสูง (High - level Language)

4) ภาษาชั้นสูงมาก (Very High - level Language)

5) ภาษาธรรมชาติ (Natural Language)

1) ภาษาเครื่อง

ในยุคแรก ๆ การใช้คอมพิวเตอร์ให้ทำงานตามต้องการนั้น ผู้เขียนโปรแกรมจะต้องเขียนคำสั่งด้วยภาษาของเครื่องคอมพิวเตอร์ซึ่งเรียกว่าภาษาเครื่อง คำสั่งของภาษาเครื่องนั้นจะประกอบด้วยกลุ่มของตัวเลขในระบบเลขฐานสอง เป็นภาษาเดียวเท่านั้นที่เครื่องคอมพิวเตอร์เข้าใจได้โดยตรง ลักษณะของภาษาเป็นภาษาที่ขึ้นอยู่กับฮาร์ดแวร์ของคอมพิวเตอร์

แต่ละระบบ โดยเขียนอยู่ในรูปของรหัสของระบบเลขฐานสอง ประกอบด้วย เลข 0 และเลข 1 ที่นำมาเขียนเรียงติดต่อกัน ประโยคคำสั่งของภาษาเครื่องจะประกอบด้วยส่วนที่ระบุ

ให้คอมพิวเตอร์ทำงานอะไรเช่น สั่งให้ทำการบวกเลข สั่งให้ทำการเคลื่อนย้ายข้อมูล เป็นต้น และอีกส่วนเพื่อ บอกแหล่งข้อมูลที่จะนำมาทำงานตามที่ระบุในตอนแรก

โครงสร้างของคำสั่งในภาษาเครื่อง

คำสั่งในภาษาเครื่องจะประกอบด้วย 2 ส่วนคือ

1.1 โอเปอเรชั่นโค้ด (Operation Code) เป็นคำสั่งที่สั่งให้เครื่องคอมพิวเตอร์ปฏิบัติการ เช่น การบวก (Addition) การลบ (Subtraction) เป็นต้น

1.2 โอเปอแรนด์ (Operands) เป็นตัวที่ระบุตำแหน่งที่เก็บของข้อมูลที่จะเข้าคอมพิวเตอร์เพื่อนำไปปฏิบัติการตามคำสั่งในโอเปอเรชันโค้ด

2) ภาษาระดับต่ำหรือภาษาแอสเซมบลี

เป็นภาษาที่มีการใช้สัญลักษณ์ข้อความ (Mnemonic codes) แทนกลุ่มของเลขฐานสอง เพื่อให้ง่ายต่อการเขียนและการจดจำมากกว่าภาษาเครื่อง ตัวอย่างเช่นมีการใช้สัญลักษณ์ต่อไปนี้

A ย่อมาจาก ADD หมายถึงการบวก

S ย่อมาจาก SUBTRACT หมายถึงการลบ

C ย่อมาจาก COMPLARE หมายถึงการเปรียบเทียบ

MP ย่อมาจาก MULTIPLY หมายถึงการคูณ

ST ย่อมาจาก SRORE หมายถึง การเก็บข้อมูลไว้ในหน่วยความจำ เป็นต้น

ถึงแม้ว่าสัญลักษณ์เหล่านี้จะไม่ใช่คำที่มีความหมายในภาษาอังกฤษแต่ก็ทำให้นักเขียนโปรแกรมสามารถเขียนโปรแกรมได้สะดวกสบายมากขึ้น เนื่องจากไม่ต้องสะดวกสบายมากขึ้น เนื่องจากไม่ต้องจดจำ 0 และ1 ของเลขฐานสองอีกนอกจากนี้ภาษาแอสเซมบลียังอนุญาตให้ผู้เขียนใช้ตัวแปรที่ตั้งขึ้นมาเองในการเก็บค่าข้อมูลใด ๆ เช่น X, Y, RATE หรือ TOTAL แทนการอ้างอิงถึงตำแหน่งที่เก็บข้อมูลจริงๆ ภายในหน่วยความจำดังได้กล่าวแล้วว่าเครื่องคอมพิวเตอร์จะรู้จักเฉพาะภาษาเครื่องเท่านั้นดังนั้นจึงจำเป็นที่จะต้องมีการแปลโปรแกรมภาษาแอสเซมบลีนั้นให้เป็นภาษาเครื่องเสียก่อนเพื่อให้คอมพิวเตอร์สามารถทำงานตามคำสั่งในโปรแกรมได้การแปลภาษาแอสเซมบลีเป็นภาษาเครื่องนั้นจะต้องมีตัวแปลภาษาแอสเซมบลีที่เรียกว่า "แอสเซมเบลอร์" (Assembler) เป็นตัวแปล ซึ่งภาษาแอสเซมบลี 1 คำสั่งจะสามารถแปลเป็นภาษาเครื่องได้ 1 คำสั่งเช่นกัน ดังนั้นเขียนโปรแกรมภาษาแอสเซมบลี 10 คำสั่ง ก็จะถูกแปลเป็นภาษาเครื่อง 10 คำสั่งเช่นกันจึงเห็นได้ว่าภาษาแอสเซมบลีจะมีลักษณะที่เหมือนกับภาษาเครื่องคือ เป็นภาษาที่ขึ้นอยู่กับเครื่อง กล่าวคือเราไม่สามารถนำโปรแกรมที่เขียนด้วยแอสเซมบลี

โปรแกรมเดียวกันไปใช้ในเครื่องต่างชนิดกันได้และนอกจากนี้ผู้ที่จะเขียนโปรแกรมภาษาแอสเซมบลีได้จะต้องมีความรู้ความเข้าใจในเรื่องของฮาร์ดแวร์เป็นอย่างดีเนื่องจากจะต้องยุ่งเกี่ยวกับการใช้งานหน่วยความจำที่เป็นงานหน่วยความจำที่เป็นรีจิสเตอร์ภายในตลอดดังนั้นจึงเหมาะที่จะใช้เขียนในงานที่ต้องการความเร็วในการทำงานสูง เช่น งานทางด้านกราฟิกหรืองานพัฒนาซอฟต์แวร์ระบบต่างๆ

3) ภาษาระดับสูง

ภาษาระดับสูงสามารถเรียกได้อีกอย่างว่าเป็นภาษารุ่นที่ 3 (3rd Generation Languagesหรือ 3GLS) เป็นภาษาที่ถูกสร้างขึ้นมาเพื่อให้สามารถเขียนและอ่านโปรแกรมได้ง่ายขึ้น เนื่องจากมีลักษณะเหมือนภาษาอังกฤษทั่วๆ ไป และที่สำคัญคือผู้เขียนโปรแกรมไม่จำเป็นต้องมีความรู้เกี่ยวกับระบบฮาร์ดแวร์แต่อย่างใด ภาษาระดับสูงมีดังนี้

- ภาษาปาสคาล (PASCAL)
- ภาษาฟอร์แทรน (FORTRAN)
- ภาษาอารีพีจี (RPG)
- ภาษาซี (C) - ภาษาโคบอล (COBOL)
- ภาษาเบสิก (BASIC) - ภาษาพีแอลวัน (PL/1)

ภาษาปาสคาล (PASCAI)

เป็นภาษาระดับสูงที่ง่ายต่อการเรียนรู้และง่ายต่อการใช้งานเหมาะสำหรับผู้ที่เริ่มเรียนภาษาระดับสูง และยังเป็นภาษาที่มีประสิทธิภาพสูง ภาษานี้เหมาะทั้งงานอเนกประสงค์ทั่วไปไม่ว่าจะเป็นการพิมพ์รายงาน การคำนวณและการสร้างรูปแบบการพิมพ์ รวมทั้งงานที่สามารถนำมาเขียนโปรแกรมประยุกต์เพื่อช่วยแก้ปัญหาหรือพัฒนาเกี่ยวกับระบบตัวเครื่องได้ ภาษาปาสคาลใช้ได้กับเครื่องไมโครคอมพิวเตอร์ทั่วไป ข้อดีของภาษาปาสคาล คือ เป็นภาษาที่มีลักษณะที่เป็นโครงสร้าง ดังนั้นจึงเป็นภาษาที่เหมาะสมที่จะนำมาใช้เป็นแม่แบบในการเรียนรู้รวมทั้งยังง่ายต่อการใช้งานอีกด้วย เพราะมีความสมบูรณ์ในตัวโปรแกรมเนื่องจากมีลักษณะโดรงสร้างที่เข้าใจง่าย และการตรวจสอบข้อผิดพลาดของโปรแกรมสามารถทำได้สะดวก

ภาษาซี (C)

เป็นภาษาระดับสูงแต่มีความอ่อนตัวหรือยืดหยุ่นเหมือนภาษาในระดับต่ำ(คล้ายภาษาแอสเซมบลี) ข้อดีของภาษาซี่ คือ มีความสั้นและกระชับ เป็นภาษาที่มีความเหมาะสมและมีประสิทธิภาพสูงในการใช้พัฒนาหรือเขียนโปรแกรมเกี่ยวกับระบบตัวเครื่องคอมพิวเตอร์รวมทั้งยังสามารถสามารถใช้งานเอนกประสงค์ทั่วไปหรือเขียนโปรแกรมประยุกต์เพื่อช่วย

แก้ปัญหาต่าง ๆทั่วไปได้ดีอีกด้วย ส่วนข้อเสีย คือ มีรูปแบบไวยากรณ์ (SYNTAX) ที่หลากหลายทำให้เกิดความสับสนและยุ่งยากแก่การจดจำ

ภาษาเบสิก (BASIC)

ย่อมาจาก Beginner's All - purpose Symbolic InstructionCode เป็นภาษาที่สามารถทำได้ทั้งงานด้านคำนวณ งานด้านธุรกิจต่าง ๆ เนื่องจากภาษานี้มีรูปแบบการทำงานดั้งเดิมเป็นแบบใช้ตัวแปลภาษาที่เรียกว่า อินเตอร์พรีเตอร์ (ตัวแปลคำสั่ง หรือ Interpreter เป็นโปรแกรมที่ใช้แปโปรแกรมที่เขียนขึ้นให้เป็นภาษาเครื่องลักษณะการแปลจะทำการแปลที่ละคำสั่งที่ได้รับให้เป็นกาษาเครื่องเมื่อแปลเสร็จหนึ่งคำสั่งเครื่องก็จะปฏิบัติตามคำสั่งทันทีและเมื่อมีข้อผิดพลาดก็จะรายงานออกมาให้ผู้ใช้ทราบในทันที เพื่อให้ผู้ใช้แก้ไขในส่วนที่ผิดพลาดเสียก่อนจึงจะทำการแปลคำสั่งต่อไปให้ การแปลแบบนี้จะทำให้เกิดการปฏิบัติการและทรายผลในทันที ดังนั้น ข้อดีของภาษานี้คือ ไม่มีความยุ่งยากในการเขียนโปรแกรมสามารถสามารถแก้ไขคำสั่งได้ทันที ทำให้เรียนรู้ได้ง่ายและเข้าใจได้อย่างรวดเร็ว ส่วนข้อเสียคือ ความสามารถในการทำงานมีจำกัดและการตรวจสอบความสัมพันธ์โดยรวมของโปรแกรมทำได้ยาก

ภาษาฟอร์แทรน (FORTRAN)

เป็นภาษาดั้งเดิมเหมาะกับงานทางด้านวิทยาศาสตร์คณิตศาสตร์ วิศวกรรมศาสตร์ ดังนั้น ข้อดีของภาษานี้คือ เหมาะกับงานที่มีข้อมูลเป็นตัวเลขมาก ๆใช้สูตรคำนวณหรือสมการที่ซับซ้อน มีความรวดเร็วและกระชับเหมาะสำหรับงานด้านคำนวณ จึงใช้ได้ดีกับงานค้นคว้าวิจัย ข้อเสียของฟอร์แทรน คือ มีลักษณะรูปแบบที่ยุ่งยากไม่เหมาะกับงานทางการพิมพ์และเก็บข้อมูล รวมทั้งไม่เหมาะต่อการนำมาใช้พัฒนาหรือเขียนโปรแกรมเกี่ยวกับระบบตัวเครื่องคอมพิวเตอร์

ภาษาโคบอล (COBOL)

ย่อมาจาก "Common Business Oriented Language" ได้รับการพัฒนาเพื่อจุดประสงค์ในการใช้เขียนโปรแกรมทางด้านธุรกิจเป็นส่วนใหญ่ ดังนั้น ข้อดีของภาษานี้ คือ สามารถทำงานกับข้อมูลทุกชนิดทั้งตัวเลขและตัวอักษร จึงเหมาะกับงานทางด้านประมวลผลทางบัญชี การสร้างฐานข้อมูลอันเป็นงานหลักของการประมวลผลทางธุรกิจโดยทั่ว ๆ ไป นอกจากนี้ ยังมีความสามารถด้านจัดรูปแบบการพิมพ์รายงานอีกด้วย และภาษานี้มีรูปแบบดล้ายคลึงกับคำในภาษาอังกฤษมากจึงสะดวกต่อความเข้าใจและจดจำ ข้อเสียของภาษาโคบอล คือ ไม่กระชับเนื่องจากมีรูปแบบของภาษายาว จึงเป็นเหตุให้ตัวโปรแกรมมีความยาวทำให้ต้องใช้เนื้อที่ในหน่วยความจำมากและเสียเวลาในการเขียนโปรแกรม

ภาษา (PL/1)

ย่อมาจาก Programming Language one เป็นภาษาที่ถูกพัฒนาขึ้นมาใช้กับเครื่องคอมพิวเตอร์ขนาดใหญ่ (แบบ MINI หรือ MAINFRAME) ดังนั้น ข้อดีของภาษานี้ คือ สามารถใช้ได้ดีทั้งงานด้านธุรกิจและวิทยาศาสตร์ ส่วนข้อเสียของภาษานี้คือ เท่าที่ผ่านมายังไม่สามารถใช้บนเครื่องไมโครคอมพิวเตอร์ ต้องใช้กับเครื่องคอมพิวเตอร์ขนาดใหญ่ที่มีขีดความสามารถสูงเท่านั้นและโปรแกรมของภาษานี้ต้องใช้เนื้อที่ในการเก็บโปรแกรมเป็นจำนวนมาก

ภาษาอาร์พีจี (RPG)

ย่อมาจาก Report Program Generator เป็นภาษาที่เหมาะกับงานด้านธุรกิจ ดังนั้น ข้อดีของภาษานี้ คือ นำมาใช้พิมพ์รายงานแบบธรรมดาจนถึงรายงานทางธุรกิจที่มีรูปแบบยุ่งยากซับซ้อนได้เป็นอย่างดี มีความสามารถในการสร้างฐานข้อมูลและควบคุมเกี่ยวกับไฟล์ต่าง ๆข้อเสีย คือ มีความยุ่งยากไม่อิสระในการเขียนโปรแกรมเนื่องจากลักษณะของโปรแกรมต้องอยู่ในแบบฟอร์มที่กำหนดไว้อย่างตายตัว และเหมาะที่จะใช้กับเครื่องคอมพิวเตอร์ขนาดใหญ่เท่านั้นอย่างไรก็ตามโปรแกรมที่ถูกเขียนด้วยภาษาประเภทนี้จะทำงานได้ ก็ต่อเมื่อมีการแปลงให้เป็นภาษาเครื่องเสียก่อน ซึ่งวิธีการแปลงจากภาษาชั้นสูงให้เป็นภาษาเครื่องนั้น จะทำได้โดยใช้โปรแกรมที่เรียกว่า คอมไพเลอร์ (Compiler) หรือ อินเตอร์พรีเตอร์ (Interpreter) อย่างใดอย่างหนึ่ง โดยภาษาระดับสูงแต่ภาษาจะมีตัวแปลภาษาเฉพาะเป็นของตัวเอง ดังนั้นจึงไม่สามารถนำตัวแปลภาษาที่เรียกว่าโคบอล คอมไพเลอร์ไม่สามารถนำคอมไพเลอร์ของภาษโคบอลนี้ไปใช้แปลภาษาปาสคาลได้ เป็นต้นสำหรับความแตกต่างระหว่างคอมไพเลอร์ และอินเตอร์พรีเตอร์ จะมีดังต่อไปนี้

คอมไพเลอร์ (Compiler)

จะทำการแปลโปรแกรมทั้งโปรแกรมให้เป็นภาษาเครื่องทีเดียว การแปลนี้จะเป็นการตรวจสอบไวยากรณ์ของภาษา ถ้ามีข้อผิดพลาดทางไวยากรณ์ของภาษาเกิดขึ้นก็จะแจ้งให้ทราบ เรียกข้อผิดพลาดทางไวยากรณ์ของภาษา (Syntax Error) นี้ได้ว่าเป็น ข้อความไดแอคนอสติค(Diagnostic Message) เพื่อให้ผู้เขียนโปรแกรมทำการแก้ไขให้ถูกต้อง แล้วจึงค่อยสั่งให้แปลใหม่โปรแกรมที่ยังไม่ผ่านการแปลจะเรียกว่า ซอร์สโปรแกรม (Source Program) หรือ ซอร์สโมดูล (Source module แต่ถ้าผ่านการแปลเรียบร้อยและไม่มีข้อผิดพลาดใด ๆ จะเรียกโปรแกรมส่วนนี้ว่าออปเจกต์โปรแกรม (Object Program) หรือออปเจกต์โมดูล (Object Module)ออปเจกต์โปรแกรมนี้ยังไม่สามารถทำงานได้ จะต้องผ่านการลิงค์ (link) หรือรวมเข้ากับไลบรารี (Library) ของระบบก่อนจึงจะเป็นโปรแกรมที่สามารถทำงานได้หรือเป็นภาษาเครื่องที่เรียกว่า เอ็กซ์ซีคิวท์โปรแกรม (Execute Program) หรือ

โหลดโมดูล (Load Module) ซึ่งโดยทั่วไปแล้วจะเป็นไฟล์ที่มีนามสกุลเป็น .exe หรือ.com และสามารถนำโปรแกรมนี้ไปใช้งานได้ตลอดโดยไม่ต้องสั่งแปลใหม่อีก แต่ถ้ามีการแก้ไขโปรแกรมแม้เพียงเล็กน้อยก็ต้องทำกรแปลใหม่หมดตั้งแต่ต้น

อินเตอร์พรีเตอร์ (Interpreter)

เป็นตัวแปลภาษาอีกตัวหนึ่งที่จะทำการแปลโปรแกรมภาษาชั้นสูงทีละคำสั่งให้เป็นภาษาเครื่องและทำการเอ็กซ์วีคิวท์ หรือทำงานคำสั่งนั้นทันทีทันใดเลยก่อนที่จะไปทำการแปลต่อในบรรทัดถัดไปถ้าในระหว่างการแปลเกิดพบข้อผิดพลาดที่บรรทัดใดก็จะฟ้องให้ทำการแก้ไขทีละบรรทัดนั้นทันทีอินเตอร์พรีเตอร์นี้เมื่อโปรแกรมเสร็จแล้วจะไม่สามารถเก็บเป็นเอ็กซ์ซีคิวท์โปรแกรม(Execute Program) ได้ซึ่งต่างกับคอมไพเลอร์ดังนั้นเมื่อจะเรียกใช้งานหรือรันโปรแกรมก็จะต้องทำการแปลหรือคอมไพล์โปรแกรมใหม่ทุกครั้งไปดังนั้นเมื่อจะเรียกใช้งานเอ็กซ์ซีคิวท์โปรแกรมย่อมจะทำงานได้เร็วกว่าการเรียกใช้งานโปรแกรมที่ต้องผ่านการแปลด้วยอินเตอร์พรีเตอร์แต่ประโยชน์ของภาษาที่ถูกแปลด้วยอินเตอร์พรีเตอร์ คือโปรแกรมจะมีโครงสร้างที่ง่ายต่อการพัฒนาตัวอย่างของภาษโปแกรมที่มีการใช้อินเตอร์พรีเตอร์เป็นตัวแปลภาษาได้แก่ ภาษาเบสิก ภาษาเพิร์ล เป็นต้น

4) ภาษาชั้นสูงมาก (Very high - Level Language)

สามารถเรียกได้อีกอย่างว่าภาษาในรุ่นที่ 4 (4GLs: Fourth Generation Languages) ภาษานี้เป็นภาษาที่อยู่ในระดับที่สูงกว่าภาษารุ่นที่ 3 มีลักษณะของภาษาในรุ่นที่เป็นธรรมชาติคล้าย ๆ กับภาษาพูดของมนุษย์จะช่วยในเรื่องของการสร้างแบบฟอร์มบนหน้าจอเพื่อจัดการเกี่ยวกับข้อมูลรวมไปถึงการออกรายงาน ซึ่งจะมีการจัดการที่ง่ายมากไม่ยุ่งยากเหมือนภาษารุ่นที่ 3 ตัวอย่างของภาษาในรุ่นที่ 4 ได้แก่ Informix - 4GL, Focus , Sybase , InGres เป็นต้น

4.1 ลักษณะของ 4GL มีดังต่อไปนี้

เป็นภาษาแบบ Nonprocedural ซึ่งหมายความว่าผู้ใช้เพียงแต่บอกว่าต้องการอะไรแต่ไม่ต้องบอกถึงรายละเอียดว่าต้องทำอย่างไร คอมพิวเตอร์จะเป็นผู้จัดการให้เองหมด ตัวอย่างเช่นถ้าต้องการสร้างแบบฟอร์มการรับข้อมูลจากผู้ใช้ผู้เขียนโปรแกรมเพียงแต่ทำการออกแบบหน้าตาของแบบฟอร์มนั้นบนโปรแกรมอิดิเตอร์ (Editor) ใด ๆ และเก็บเป็นไฟล์ไว้เมื่อจะเรียกใช้งานแบบฟอร์มนั้นเพียงแต่ใช้คำสั่งเปิดไฟล์นั้นขึ้นมาแสดงบนหน้าจอคอมพิวเตอร์ได้โดยทันทีซึ่งต่างจากภาษารุ่นที่ 3ซึ่งเป็นแบบ Procedural ผู้เขียนโปรแกรม จะต้องเขียนรายละเอียดของโปรแกรมทั้งหมดว่า ที่บรรทัดนี้คอลัมน์นี้จะให้แสดงข้อความหรือข้อมูลอะไรออกมา ซึ่งถ้าต่อไปจะมีการปรับเปลี่ยนหน้าตาของแบบฟอร์ม ก็จะเป็นเรื่องที่ยุ่งยากอย่างยิ่ง หรือในการสร้างรายงานด้วย 4GLs ก็สามารถทำได้อย่างง่ายดายเพียงแต่ระบุลงไปว่าต้องการรายงานอะไร มี

ข้อมูลใดที่จะนำมาแสดงบ้าง โดยไม่ต้องบอกถึงวิธีการสร้าง หรือการดึงข้อมูลแต่อย่างใด 4GLs จะจัดการให้เองหมด

ส่วนใหญ่จะพบว่า 4GLs มักจะอยู่ควบคู่กับระบบฐานข้อมูล โดยผู้ใช้ระบบฐานข้อมูลจะสามารถจัดการฐานข้อมูลได้โดยผ่านทาง 4GLs นี้

4.2 ส่วนประกอบของภาษา 4GLs

โดยทั่วไปแล้ว 4GLs จะประกอบด้วยส่วนสำคัญ 3 ส่วนดังต่อไปนี้

เครื่องช่วยสร้างรายงาน (Report Generators หรืออาจเรียกได้อีกอย่างว่าเครื่องมือช่วยเขียนรายงาน (Report Writer) เป็นโปรแกรมสำหรับผู้ใช้ (end - users) ให้สามารถสร้างรายงานอย่างง่ายได้ด้วยตนเอง โดยผู้ใช้สามารถกำหนดเงื่อนไขและข้อมูลที่จะออกมาพิมพ์ในรายงานรวมไปถึงรูปแบบ (format) ของการพิมพ์ไว้ โปรแกรมช่วยสร้างรายงานนี้จะทำการพิมพ์รายงานตามรูปแบบที่เรากำหนดไว้ให้ภาษาช่วยค้นหาข้อมูล (Query Languages) เป็นภาษาที่ช่วยในการค้นหาหรือดึงข้อมูลจากฐานข้อมูล ภาษานี้จะง่ายต่อการใช้งานมาก เนื่องจากจะอยู่ในรูปแบบที่ใกล้เคียงกับภาษาอังกฤษมาก ตัวอย่างของภาษาช่วยด้นหาข้อมูลนี้ได้แก่ ภาษา SOL (Structured QueryLanguage) ภาษา QBE (Query - By - Example) และ Intellect เป็นต้น

4.3 เครื่องมือช่วยสร้างโปรแกรม (Application Generators)

4GLS จะมีรูปแบบการเขียนโปรแกรมเฉพาะตัว และสามารถเรียกใช้เครื่องมือช่วยสร้างโปรแกรมนี้ทำการแปลง 4GLs ให้กลายเป็นโปรกรมในภาษารุ่นที่ 3 ได้ เช่น ภาษาโคบอล หรือภาษาซี เป็นต้น ซึ่งอาจนำภาษโคบอล หรือซีที่แปลงได้ไปพัฒนาต่อเพื่อใช้กับงานที่มีความซับซ้อนมากๆ ต่อไปได้

4.4 ประโยชน์ของ 4GL

เป็นภาษาที่ง่ายต่อการเรียนรู้ คำสั่งแต่ละคำสั่งสื่อความหมายได้อย่างชัดเจน ดังนั้นจึงสามารถใช้เวลาในการศึกษาสั้นกว่าภาษารุ่นที่ 3

1) ประหยัดเวลาในการเขียนโปรแกรมได้มาก เนื่องจาก 1 คำสั่งของ 4GL ถ้าต้อง

2) เขียนด้วยภาษารุ่นที่ 3 อาจต้องเขียนถึง 100 กว่าคำสั่งในการทำงานแบบเดียวกัน

3) สนับสนุนระบบจัดการฐานข้อมูล ทำให้สามารถจัดการกับข้อมูลได้อย่างสะดวกพอสมควรและรวดเร็ว สามารถสร้างแบบฟอร์มเพื่อจัดการกับข้อมูลในฐานข้อมูล และออกรายงานได้อย่างง่ายดาย ไม่ยุ่งยาก

4) มีเครื่องมือการใช้งานเพื่ออำนวยความสะดวกในการเขียนโปรแกรมมาก

5) สามารถทำงานได้ในลักษณะ Interactive คือมีการโต้ตอบกับผู้ใช้ได้ทันที

5) ภาษาธรรมชาติ

เป็นภาษาในยุคที่ 5 ที่มีรูปแบบเป็นแบบ Nonprocedural เช่นเดียวกับภาษารุ่นที่ 4 การที่เรียกว่า ภาษาธรรมชาติ เพราะจะสามารถสั่งงานคอมพิวเตอร์ได้โดยใช้ภาษามนุษย์โดยตรง ซึ่งโดยทั่วไปคำสั่งที่มนุษย์ป้อนเข้าไปในคอมพิวเตอร์จะอยู่ในรูปของภาษาพูดมนุษย์ ซึ่งอาจมีรูปแบบที่ไม่แน่นอนตายตัว แต่คอมพิวเตอร์ก็สามารถแปลคำสั่ง เหล่านั้นให้อยู่ในรูปแบบที่คอมพิวเตอร์เข้าใจได้ ถ้าตั้งคำถามใดไม่กระจ่างก็จะมีการถามกลับเพื่อให้เข้าใจคำถามได้อย่างถูกต้องภาษาธรรมชาตินี้ ถูกสร้างขึ้นมาจากเทคโนโลยีทางด้านระบบผู้เชี่ยวชาญ (Expert System) ซึ่งเป็นงานที่อยู่ในสาขาปัญญาประดิษฐ์(Artificial Intelligence) ในการที่พยายามทำให้คอมพิวเตอร์เปรียบเสมือนกับเป็นผู้เชี่ยวชาญคนหนึ่งที่สามารถคิดและตัดสินใจได้เช่นเดียวกับมนุษย์คอมพิวเตอร์สามารถตอบคำถามของมนุษย์ได้อย่างถูกต้องพร้อมทั้งมีข้อแนะนำต่าง ๆ เพื่อช่วยในการตัดสินใจของมนุษย์ได้อีกด้วย ระบบผู้เชี่ยวชาญนี้จะใช้กับงานเฉพาะด้านใดด้านหนึ่ง เช่นในการแพทย์ในการพยากรณ์อากาศ ในการวิเคราะห์ทางเคมี การลงทุน ฯลฯ ซึ่งในการนี้จะต้องมีการเก็บรวบรวมข้อมูลที่มีอยู่เป็นจำนวนมหาศาลและให้ผู้ใช้สามารถใช้ภาษาธรรมชาติในการดึงข้อมูลจากฐานความรู้นี้ได้ ดังนั้นเราจึงอาจเรียกระบบผู้เชี่ยวชาญนี้ได้อีกอย่างว่าเป็น ระบบฐานความรู้ (Knowledge Base System)อย่างไรก็ตามระบบผู้เชี่ยวชาญไม่สามารถนำมาแทนที่การทำงานของผู้เชี่ยวชาญที่เป็นมนุษย์ได้ เนื่องจากทั้งระบบผู้เชี่ยวชาญและมนุษย์จะต้องทำงานร่วมกัน โดยมนุษย์จะนำข้อมูลที่ได้จากระบบเชี่ยวชาญมาพิจารณาร่วมกับวิจารณญาณของตนเองเพื่อตัดสินปัญหาที่ซับซ้อนอีกที อย่างไรก็ตามระบบผู้เชี่ยวชาญนี้เป็นคลื่นแห่งอนาคตที่จะใช้เป็นเครื่องมือช่วยตัดสินใจการทำงานของมนุษย์ได้อย่างดีเยี่ยม

ใบความรู้เรื่อง ผังงาน

<u>ผังงาน (Flowchart)</u>

ความหมายของผังงาน

ผังงาน (Flowchart) คือ รูปภาพ (Image) หรือสัญลักษณ์(Symbol) ที่ใช้เขียนแทนขั้นตอน คำอธิบาย ข้อความหรือคำพูด ที่ใช้ในอัลกอริทึม (Algorithm) เพราะการนำเสนอขั้นตอนของงานให้เข้าใจตรงกัน ระหว่างผู้เกี่ยวข้อง ด้วยคำพูด หรือ ข้อความ ทำได้ยากกว่าเมื่อใช้รูปภาพ หรือสัญลักษณ์

ผังงานแบ่งได้ 2 ประเภท

1. ผังงานระบบ (System Flowchart) คือ ผังงานที่แสดงขั้นตอนการทำงานในระบบอย่างกว้างๆ แต่ไม่เจาะลงในระบบงานย่อย

2. ผังงานโปรแกรม (Program Flowchart) คือ ผังงานที่แสดงถึงขั้นตอนในการทำงานของโปรแกรม ตั้งแต่รับข้อมูล คำนวณ จนถึงแสดงผลลัพธ์

การเขียนผังงาน (Flowchart)

ผังงาน คือ แผนภาพที่มีการใช้สัญลักษณ์รูปภาพและลูกศรที่แสดงถึงขั้นตอนการทำงานของ

โปรแกรมหรือระบบทีละขั้นตอน รวมไปถึงทิศทางการไหลของข้อมูลตั้งแต่แรกจนได้ผลลัพธ์ตามที่ต้องการ

ประโยชน์ของผังงาน

1. ช่วยให้สามารถทำความเข้าใจลำดับขั้นตอนการทำงานของโปรแกรมได้อย่างรวดเร็ว
2. ช่วยแสดงลำดับขั้นตอนการทำงานของโปรแกรม ทำให้สามารถเขียนโปรแกรมได้อย่างเป็นระบบไม่สับสน
3. ช่วยให้เข้าใจลำดับและความสัมพันธ์ระหว่างขั้นตอนในการทำงานต่างๆ
4. ใช้เป็นสื่อกลางในการติดต่อประสานความคิดระหว่างผู้ที่เกี่ยวข้อง เช่น นักวิเคราะห์ระบบ (Systems Analyst) นักเขียนโปรแกรม (Programmer)
5. ช่วยในการตรวจสอบและแก้ไขโปรแกรมได้ง่าย เมื่อเกิดข้อผิดพลาด
6. ช่วยให้การดัดแปลง แก้ไข ทำได้อย่างสะดวกและรวดเร็ว
7. ช่วยให้ผู้อื่นสามารถศึกษาการทำงานของโปรแกรมได้อย่างง่าย และรวดเร็วมากขึ้น
8. สามารถเรียนรู้และเข้าใจได้ง่าย เพราะผังงานไม่ขึ้นอยู่กับภาษาคอมพิวเตอร์ภาษาใดภาษาหนึ่งโดยเฉพาะ
9. การบำรุงรักษาโปรแกรม เมื่อมีการเปลี่ยนแปลงแก้ไขโปรแกรม สามารถดูผังงานเพื่อแก้ไขคำสั่งในโปรแกรมก่อนแก้ไขได้

ลักษณะการเขียนผังงาน

ลักษณะของผังงานที่ดี ควรมีลักษณะดังนี้

1. ทุกผังงานต้องมีจุดเริ่มต้นและจุดสิ้นสุดเพียงอย่างละหนึ่งแห่งเท่านั้น
2. ทุกสัญลักษณ์ของผังงานต้องมีลูกศรชี้ทิศทางเข้า และลูกศรชี้ทิศทางออกอย่างละหนึ่ง ลูกศรยกเว้นสัญลักษณ์จุดเริ่มต้น สัญลักษณ์จุดสิ้นสุด สัญลักษณ์การตัดสินใจ และสัญลักษณ์จุดต่อ
3. สัญลักษณ์จุดเริ่มต้นมีเฉพาะลูกศรชี้ทิศทางออก สัญลักษณ์จุดสิ้นสุดมีเฉพาะลูกศรชี้ทิศทางเข้า
4. สัญลักษณ์การตัดสินใจมีลูกศรชี้ทิศทางเข้า 1 ทิศทาง มีลูกศรชี้ทิศทางออก 2 ทิศทาง คือ กรณีที่ผลที่ได้จากการตัดสินใจเป็นจริง และกรณีที่ผลที่ได้จากการตัดสินใจเป็นเท็จ
5. ทิศทางของลำดับขั้นตอนการทำงาน นิยมเขียนจากบนลงล่างหรือจากซ้ายไปขวา
6. เส้นของลูกศรที่ใช้บอกทิศทางของลำดับขั้นตอนวิธีการทำงาน ไม่ควรเขียนตัดกันหรือทับกัน
7. ไม่ควรเขียนเส้นของลูกศรเพื่อทำการเชื่อมโยงลำดับขั้นตอนที่อยู่ห่างกันมากหากจำเป็นควรใช้สัญลักษณ์จุดต่อแทน
8. การเขียนผังงานในส่วนของการกำหนดค่าหรือคำนวณค่า นิยมใช้เครื่องหมายลูกศรแทนการใช้เครื่องหมายเท่ากับ ส่วนเครื่องหมายเท่ากับนิยมใช้สำหรับการเปรียบเทียบค่า

โครงสร้างผังงานมี 3 ประเภท ได้แก่

1.โครงสร้างแบบเรียงลำดับ (Sequence Control Structure)

2.โครงสร้างแบบทางเลือก (Selection Structure)

3.โครงสร้างแบบทำซ้ำหรือวนรอบ (Iteration หรือ Looping Structure)

สัญลักษณ์	ความหมาย	ตัวอย่างการใช้งาน
Terminal Interrupt	เริ่มต้นและสิ้นสุดการทำงาน	Start Stop
Input/Output	รับข้อมูลและแสดงผลข้อมูล กรณีไม่กำหนดอุปกรณ์	Input base, high
Manual Input	การรับข้อมูลผ่านทางแป้นพิมพ์	Input A, B
Process	การกำหนดค่า การคำนวณ และการประมวลผล	Area ← 0.5*base*high
Decision	กำหนดเงื่อนไข ทางเลือก การเปรียบเทียบทาง ตรรกศาสตร์ เพื่อการตัดสินใจ	A<B NO YES
Display	การแสดงผลทางจอภาพ	Print Area
Document	การแสดงผลทางเครื่องพิมพ์	Print Area
Flow line	ลูกศรแสดงทิศทาง การดำเนินงาน	
Connector	จุดเชื่อมต่อภายในหน้าเดียวกัน	
Off-page connector	จุดเชื่อมต่อระหว่างหน้า	1 1

โครงสร้างผังงานมี 3 ประเภท ได้แก่

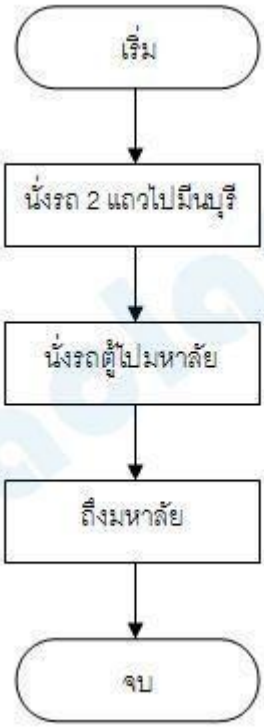
1.โครงสร้างแบบเรียงลำดับ (Sequence Control Structure)

2.โครงสร้างแบบทางเลือก (Selection Structure)

3.โครงสร้างแบบทำซ้ำหรือวนรอบ (Iteration หรือ Looping Structure)

ตัวอย่างผังงานที่มีโครงสร้างแบบเป็นลำดับ

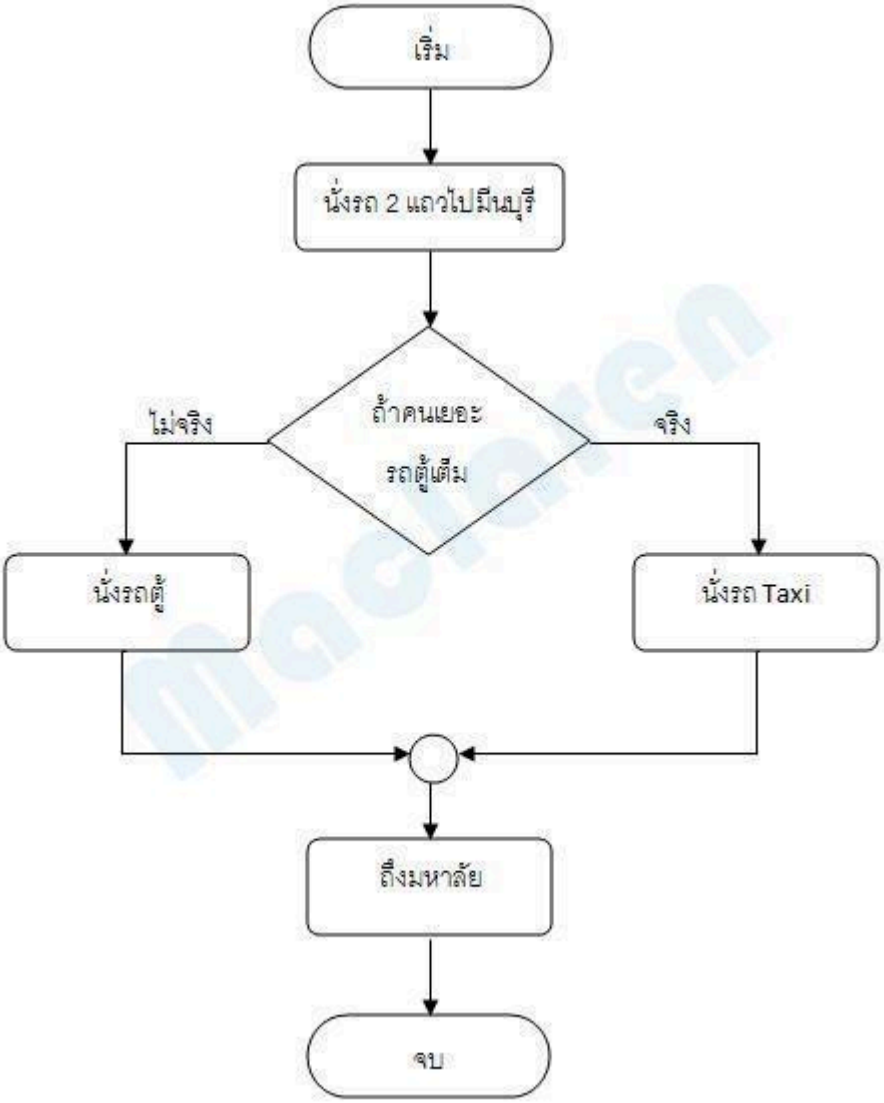
ผังงานวิธีการไปมหาลัย



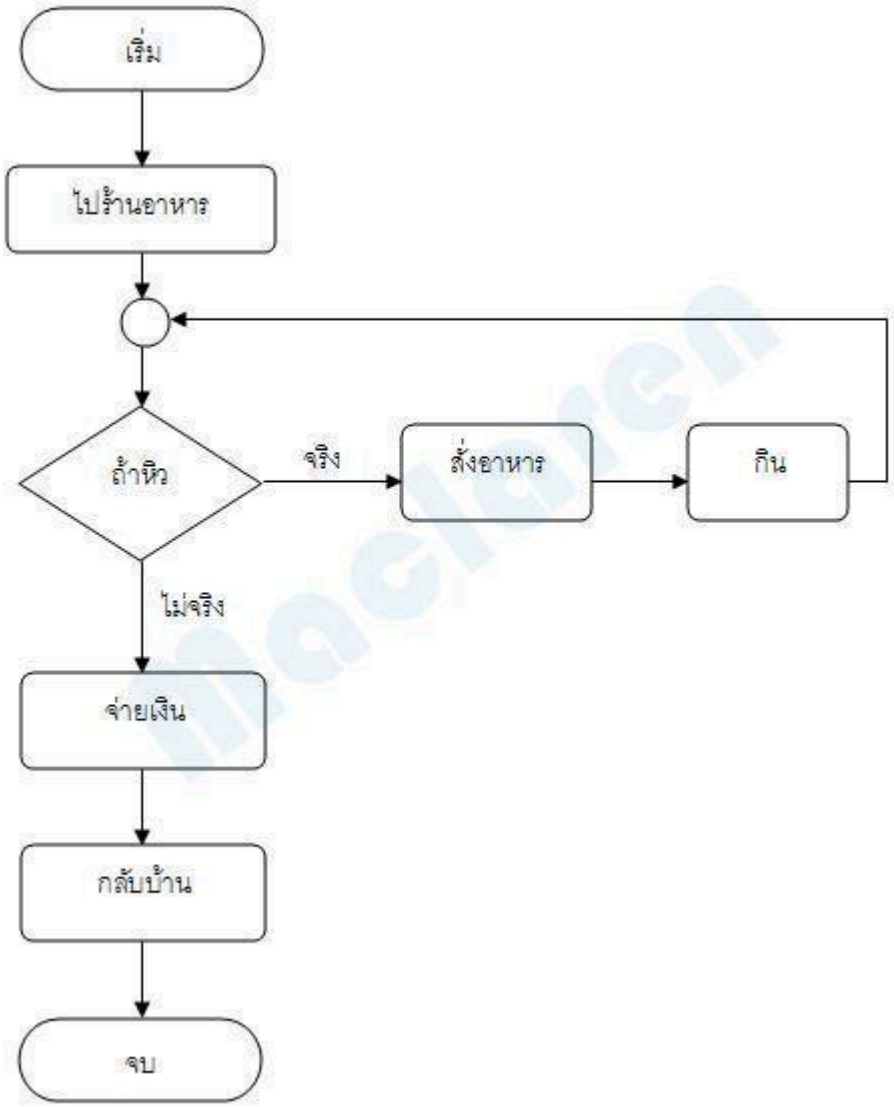
ตัวอย่างผังงานที่มีโครงสร้างแบบมีการเลือก

โครงสร้างการทำงานแบบมีการเลือกมีรูปแบบที่ซับซ้อนกว่าโครงสร้างแบบเป็นลำดับ รูปแบบที่ง่ายที่สุดของโครงสร้างแบบนี้คือ การเลือกแบบมีทางออก 2 ทาง ในการเลือกแบบมีทางออก 2 ทาง นี้จะมีทางออกจากสัญลักษณ์การตัดสินใจเพียง 2 ทาง คือ ใช่หรือไม่ใช่ เท่านั้น (แต่ระบบการเขียนผังงานระบบ อนุญาตให้มีทางออกจากการตัดสินใจได้มากกว่า 2 ทาง)

ผังงานวิธีการไปมหาลัย



ตัวอย่างผังงานที่มีโครงสร้างแบบทำซ้ำ (Iteration Structure)

โครงสร้างการทำงานแบบทำซ้ำ จะทำงานแบบเดียวกันซ้ำไปเรื่อย ๆ ในขณะที่ยังเป็นไปตามเงื่อนไขหรือเงื่อนไขเป็นจริง จนกระทั่งเงื่อนไขเป็นเท็จจึงทำงานอื่นต่อไป

ผังงานการทานอาหาร