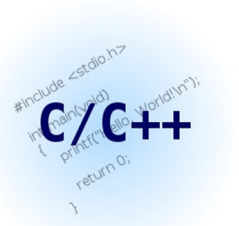


ใบความรู้ที่ 1

เรื่อง ประวัติและขั้นตอนการพัฒนาโปรแกรมภาษาซี



ประวัติความเป็นมาภาษาซี

ภาษาซีเกิดขึ้นในปี ค.ศ.1972 ผู้คิดค้นคือ Dennis Ritchie โดยพัฒนามาจากภาษา B และภาษา BCPL แต่ยังไม่มีการใช้งานอย่างกว้างขวางนัก ในปี ค.ศ.1978 Brian Kernighan ได้พัฒนามาตรฐานของภาษาซี เรียกว่า K&R ทำให้มีผู้สนใจเกี่ยวกับภาษาซีมากขึ้น จึงเกิดเป็นภาษาซีอีกหลายรูปแบบเพราะยังไม่มี การกำหนดรูปแบบภาษาซีที่เป็นมาตรฐาน และในปี 1988 Ritchie จึงได้กำหนดมาตรฐานของภาษาซี เรียกว่า ANSI C เพื่อใช้เป็นตัวกำหนดมาตรฐานในการสร้างภาษารุ่นต่อไป ภาษาซีเป็นภาษาระดับกลาง เหมาะ สำหรับการเขียนโปรแกรมแบบโครงสร้าง เป็นภาษาที่มีความยืดหยุ่นมาก คือใช้งานได้กับเครื่องต่าง ๆ ได้ และปัจจุบันภาษาซีเป็นภาษาพื้นฐานของภาษาโปรแกรมรุ่นใหม่ ๆ เช่น C++

ขั้นตอนการพัฒนาโปรแกรมภาษาซี

ขั้นตอนที่ 1 เขียนโปรแกรม (source code)

ใช้ editor เขียนโปรแกรมภาษาซี และทำการบันทึกไฟล์ให้มีนามสกุลเป็น .c เช่น work.c เป็นต้น editor คือ โปรแกรมที่ใช้สำหรับการเขียนโปรแกรม ได้แก่ Notepad, TextPad และ EditPlus เป็นต้น ผู้ใช้ สามารถเลือกใช้โปรแกรมใดก็ได้แล้วแต่ความถนัดของแต่ละบุคคล

ขั้นตอนที่ 2 คอมไพล์โปรแกรม (compile)

นำ source code จากขั้นตอนที่ 1 มาทำการคอมไพล์ เพื่อแปลจากภาษาซีที่มนุษย์เข้าใจเป็น ภาษาเครื่องที่คอมพิวเตอร์เข้าใจได้ ในขั้นนี้คอมไพล์จะทำการตรวจสอบ source code ว่าเกิดข้อผิดพลาด หรือไม่

* หากเกิดข้อผิดพลาด จะแจ้งให้ผู้เขียนโปรแกรมทราบ ผู้เขียนโปรแกรมต้องกลับไปแก้ไขโปรแกรม และทำการคอมไพล์โปรแกรมใหม่อีกครั้ง

* หากไม่พบข้อผิดพลาด คอมไพล์จะแปลไฟล์ source code จากภาษาซีเป็นภาษาเครื่อง (ไฟล์ นามสกุล .obj) เช่นถ้า source code ชื่อ work.c ก็จะถูกแปลไปเป็นไฟล์ work.obj ซึ่งเก็บภาษาเครื่องไว้ เป็นต้น

Compile เป็นตัวแปลภาษารูปแบบหนึ่ง มีหน้าที่หลักคือการแปลภาษาโปรแกรมที่มนุษย์เขียนขึ้นไป เป็นภาษาเครื่อง โดยคอมไพเลอร์ ของภาษาซี คือ C compiler ซึ่งหลักการที่คอมไพล์ใช้ เรียกว่า คอมไพล์ (compile) โดยจะทำการอ่านโปรแกรมภาษาซีทั้งหมดตั้งแต่ต้นจนจบ แล้วทำการแปลผลที่เดียว

นอกจากคอมไพเลอร์แล้ว ยังมีตัวแปลภาษาอีกรูปแบบหนึ่งที่เรียกว่า **อินเตอร์พรีเตอร์** การอ่านและ **แปลโปรแกรมทีละบรรทัด** เมื่อแปลผลบรรทัดหนึ่งเสร็จก็จะทำงานตามคำสั่งในบรรทัดนั้น แล้วจึงทำการ แปลผลตามคำสั่งในบรรทัดถัดไป หลักการที่อินเตอร์พรีเตอร์ใช้ เรียกว่า **อินเตอร์พรีต (Interpret)**

ข้อดีและข้อเสียของตัวแปลภาษาทั้งสองแบบมีดังนี้

	ข้อดี	ข้อเสีย
คอมไพเลอร์	<ul style="list-style-type: none"> * ทำงานได้เร็ว เนื่องจากทำการแปลผลทีเดียว แล้วจึงทำงานตามคำสั่งของโปรแกรมในภายหลัง * เมื่อทำการแปลผลแล้ว ในครั้งต่อไปไม่จำเป็นต้องแปลผลใหม่อีก เนื่องจากภาษาเครื่องที่แปลได้ถูกเก็บไว้ที่หน่วยความจำ สามารถเรียกใช้งานได้ทันที 	<ul style="list-style-type: none"> * เมื่อเกิดข้อผิดพลาดขึ้นกับโปรแกรม จะตรวจสอบหาข้อผิดพลาดได้ยาก เพราะทำการแปลผลทีเดียวทั้งโปรแกรม
อินเทอร์พรีเตอร์	<ul style="list-style-type: none"> * หาข้อผิดพลาดของโปรแกรมได้ง่าย เนื่องจากทำการแปลผลทีละบรรทัด * เนื่องจากทำงานทีละบรรทัด ดังนั้นจึงสั่งให้โปรแกรมทำงานตามคำสั่งเฉพาะจุดที่ต้องการได้ * ไม่เสียเวลารอการแปลโปรแกรมเป็นเวลานาน 	<ul style="list-style-type: none"> * ช้า เนื่องจากทำงานทีละบรรทัด

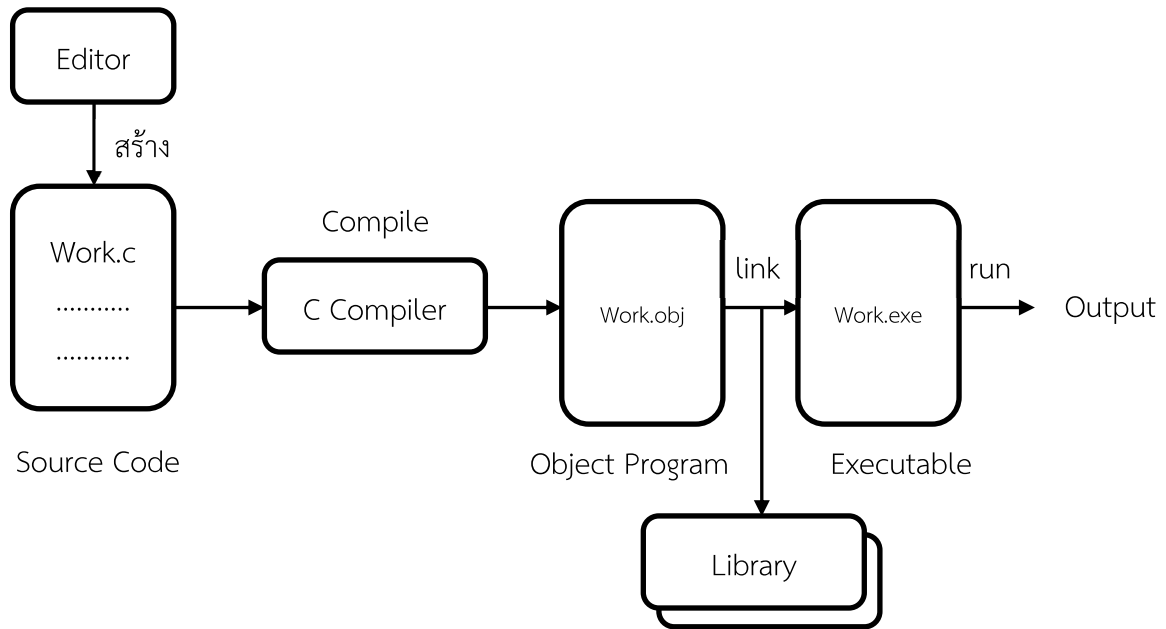
ขั้นตอนที่ 3 เชื่อมโยงโปรแกรม (link)

การเขียนโปรแกรมภาษาซีนั้นผู้เขียนโปรแกรมไม่จำเป็นต้องเขียนคำสั่งต่าง ๆ ขึ้นใช้งานเอง เนื่องจากภาษาซีมีฟังก์ชันมาตรฐานมาให้ผู้เขียนโปรแกรมสามารถเรียกใช้งานได้ เช่น การเขียนโปรแกรมแสดงข้อความ "Bokru Wittaya School" ออกทางหน้าจอ ผู้เขียนสามารถเรียกใช้ฟังก์ชัน printf() ซึ่งเป็นฟังก์ชันมาตรฐานของภาษาซีมาใช้งานได้ โดยส่วนการประกาศ (declaration) ของฟังก์ชันมาตรฐานต่าง ๆ จะถูกจัดเก็บอยู่ในเฮดเดอร์ไฟล์แต่ละตัว แตกต่างกันไปตามลักษณะการใช้งาน

ด้วยเหตุนี้ภาษาเครื่องที่ได้จากขั้นตอนที่ 2 จึงยังไม่สามารถนำไปใช้งานได้ แต่ต้องนำมาเชื่อมโยงเข้ากับ library ก่อน ซึ่งผลจากการเชื่อมโยงจะทำให้ได้ executable program (ไฟล์นามสกุล .exe เช่น work.exe) ที่สามารถนำไปใช้งานได้

ขั้นตอนที่ 4 ประมวลผล (run)

เมื่อนำ executable program จากขั้นตอนที่ 3 มาประมวลผลก็จะได้ผลลัพธ์ (output) ของโปรแกรมออกมา (ถ้ามี)



รูปที่ 1 ขั้นตอนการพัฒนาโปรแกรมด้วยภาษาซี