

ประวัติความเป็นมาของภาษาซี
&
คอมไพเลอร์

โปรแกรมคืออะไร ?

- โปรแกรม หมายถึง ลำดับของคำสั่ง ที่ถูกเขียนขึ้นตามข้อกำหนดของ คำศัพท์และกลุ่มของกฎไวยากรณ์ซึ่งมีความหมายเพื่อให้เครื่องคอมพิวเตอร์หรือเครื่องจักรเข้าใจได้

ดังนั้นสิ่งที่ควรจะต้องรู้เกี่ยวกับการเขียนโปรแกรมภาษาคอมพิวเตอร์คือ

- คำศัพท์ของภาษาคอมพิวเตอร์
- กฎไวยากรณ์ของภาษาคอมพิวเตอร์
- นอกจากนี้ควรจะต้องตระหนักว่า โปรแกรมที่เขียนขึ้นมาจะมีวัตถุประสงค์เพื่อให้ดำเนินการ (executed) โดยเครื่องจักร ไม่ใช่มนุษย์ ดังนั้น คำศัพท์และกฎต่าง ๆ จะต้องกระชับและถูกต้อง

เราเขียนเขียนโปรแกรมเพื่อ ?

- เพราะอยู่ในหลักสูตรที่เรียนอยู่บังคับวิชานี้
- เพราะอาจารย์ให้เขียนโปรแกรมส่งอาทิตย์หน้า
- เพราะอยากเพิ่มพูนความรู้ด้านคอมพิวเตอร์
- เพราะต้องการเขียนโปรแกรมไว้ใช้เอง
- ฯลฯ

การเขียนโปรแกรม

- หมายถึง การเขียนชุดคำสั่งด้วยภาษาคอมพิวเตอร์
ภาษาใดภาษาหนึ่ง เพื่อให้คอมพิวเตอร์ทำงานตาม
โปรแกรมที่เราเขียนขึ้น



ภาษาคอมพิวเตอร์

- โปรแกรมรันบนวินโดวส์ C/C++, Visual Basic, Delphi
- โปรแกรมทำงานบนเว็บ PHP, ASP, JAVA, Perl, Tcl, C#
- โปรแกรมฐานข้อมูล Visual Basic
- โปรแกรมกราฟิก
- โปรแกรมเกม
- โปรแกรมคำนวณ

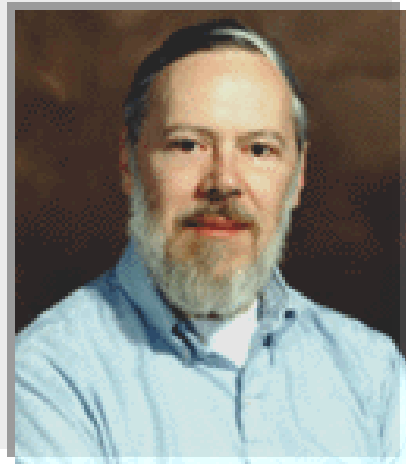


โปรแกรมที่เขียนขึ้นจะทำงานได้อย่างไร ?

โปรแกรมที่เขียนขึ้นมา ไม่ว่าจะเขียนโดยใช้เอดิเตอร์(editor) อะไรก็ตาม จะได้ซอร์สโค้ด (source code) ซึ่งจะเก็บในรูปแบบแฟ้มข้อมูล ซึ่งจะมีนามสกุลแตกต่างกันไปดังนี้

ภาษา	นามสกุล	ตัวอย่าง
c	.c	hello.c
c++	.cpp	hello.cpp
Pascal	.pas	Hello.pas
Perl	.pl	Hello.pl
PHP	.php	Hello.php
Java	.java	Hello.java

ความเป็นมาของภาษา C (1/3)



ภาษา C เป็นภาษาที่ออกแบบและพัฒนาโดย
เคนนิส ริทช์ (Dennis Ritchie) ที่ห้อง
ปฏิบัติการเบล (Bell Labs) ในปี ค.ศ. 1972

ซึ่งได้แนวคิดมาจากภาษา BCPL พัฒนาขึ้น โดย มาร์ติน ริชาร์ด
(Martin Richards) และภาษา B ที่เขียนขึ้นโดย เคน ทอมป์สัน
(Ken Thompson) เพื่อนำมาพัฒนาต่อจนได้ภาษาใหม่ที่มี
ประสิทธิภาพการทำงานสูง

ความเป็นมาของภาษา C (2/3)

หลังจากนั้นในปี ค.ศ.1978 ภาษา C จึงได้รับการเผยแพร่อย่างเป็นทางการโดย เคอร์นิกแฮน (Kernighan) และ เดนิส ริทชี โดยมีมุ่งให้เป็นภาษาสำหรับใช้เขียนโปรแกรมจัดการระบบงาน (Unix) กล่าวคือภาษา C เป็นภาษาซึ่งเหมาะสำหรับการเขียนโปรแกรมระบบ ภาษา C เป็นทั้งภาษาระดับต่ำและระดับสูง

ความเป็นมาของภาษา C (3/3)

แต่เนื่องจากภาษา C เป็นภาษาที่มีในทุก ๆ ระบบปฏิบัติการและได้รับการพัฒนาตัวแปลภาษาจากหลาย ๆ บริษัท ดังนั้นได้มีองค์มาตรฐานแห่งชาติสหรัฐอเมริกา ได้ทำการสร้างข้อตกลงที่เป็นมาตรฐานเดียวกันขึ้นของภาษา C ในปี ค.ศ. 1983 เรียกภาษา C นี้ในมาตรฐานนี้เรียกว่า ANIC

การพัฒนาโปรแกรมภาษา C

- ใช้ Text editor เพื่อสร้าง Source โปรแกรมภาษา C
- ใช้โปรแกรม Compiler ภาษา C แปล Source โปรแกรมที่สร้างขึ้นในตอนแรกให้เป็น object module
- ใช้โปรแกรม Liker ทำการ Like Object module ต่างๆที่ได้เข้าด้วยกันพร้อมกับฟังก์ชันต่างๆที่อยู่ใน Library ผลลัพธ์ที่ได้จะเป็นโปรแกรมที่สามารถทำงานได้



ข้อเด่นของภาษา C (1/4)

- ภาษาซี เป็นภาษาคอมพิวเตอร์ที่มีการพัฒนาขึ้นใช้งานเพื่อเป็นภาษามาตรฐานที่ไม่ขึ้นกับโปรแกรมหรือระบบปฏิบัติการ ระบบงาน หรือขึ้นกับฮาร์ดแวร์ ภาษาซีจึงเป็นภาษาคอมพิวเตอร์ตามอุดมการณ์ของนักคอมพิวเตอร์

ข้อเด่นของภาษา C (2/4)

- ภาษาซี เป็นภาษาคอมพิวเตอร์ที่อาศัยหลักการที่เรียกว่า โปรแกรมโครงสร้าง จึงเป็นภาษาที่เหมาะสมสำหรับงานพัฒนา
- เป็นคอมไพเลอร์ที่มีประสิทธิภาพสูง ให้รหัสออบเจกต์สั้น ทำงานได้รวดเร็ว เหมาะกับงานพัฒนาที่ต้องการความรวดเร็วเป็นสำคัญ

ข้อเด่นของภาษา C (3/4)

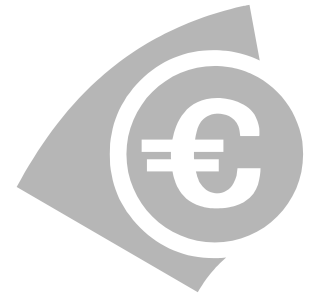
- มีความอ่อนตัวของภาษาคัดลอกภาษาแอสเซมบลี ภาษาซีสามารถเขียนแทนภาษาแอสเซมบลีได้ดี ค้นหาที่ผิดหรือแก้ไขตัวโปรแกรมได้ง่าย ภาษาซีจึงเป็นภาษาระดับสูงที่ใช้งานได้เหมือนภาษาระดับต่ำ

ข้อเด่นของภาษา C (4/4)

- ภาษาซี มีความคล่องตัวที่จะประยุกต์เข้ากับงานต่าง ๆ ได้เป็นอย่างดี การพัฒนาโปรแกรม เช่นเว็รค์โปรเซสซึ่งสเปรคชิต คาค้าเบส ฯลฯ มักใช้ภาษาซีเป็นภาษาสำหรับการพัฒนา
- เป็นภาษาซีที่มีอยู่เกือบทุกโปรแกรมจักระบบงาน มีบนเครื่องไมโครคอมพิวเตอร์ตั้งแต่ขนาด 8 บิต ไปจนถึงมินิ เมนเฟรมคอมพิวเตอร์ แม้แต่เครื่องซูเปอร์คอมพิวเตอร์ เช่น เกลย์วัน ก็ยังมีภาษาซี

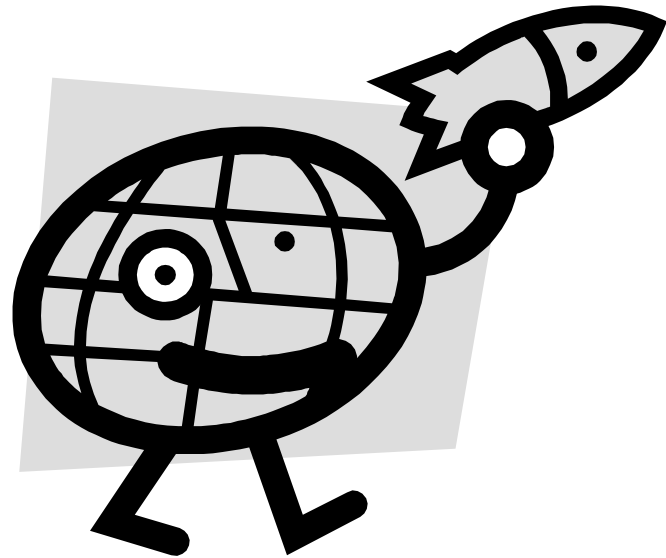
พัฒนาจากภาษา C สู่อ C++

- ภาษา C ได้มีการพัฒนาต่อ โดยใช้แนวคิด โปรแกรมเชิงวัตถุ หรือ OOP (Object Oriented Programming)
- เกิดภาษาใหม่เรียกว่า “ซี พลัส พลัส” (C++)
- ภาษาซียังเป็นต้นฉบับให้กับอีกหลายๆ ภาษาในปัจจุบัน เช่น Java, C# (อ่านว่าซีชาร์ป)
- C# คือภาษาที่ออกแบบมาเพื่อทำงานบนแพลตฟอร์ม .NET

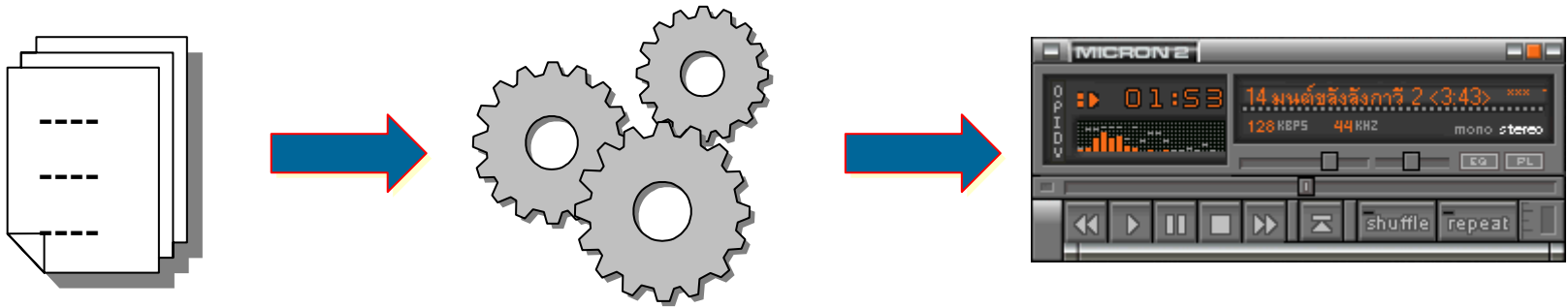


พัฒนาการของคอมพิวเตอร์ภาษา C

- **DOS >> Turbo C**
- **Windows >>**
 - Microsoft Visual C++
 - Borland C++
 - Borland C++ Builder
 - Symantec C/C++



กระบวนการแปลภาษา



ซอร์สโค้ด

กระบวนการแปลโปรแกรม

โปรแกรมที่สามารถทำงานได้
โดยไม่ต้องมี **source code**

Compiler และ Linker (1/2)

การเขียนโปรแกรมในรูปของรหัสเลขฐานสองเป็นงานที่ยุ่งยากและซับซ้อน นอกจากนี้การอ่านความหมายของคำสั่งยังมีความยุ่งยากมากขึ้นอีก

```
int Num1;  
int Sum ;  
  
Sum = Num1 + 5 ;
```

ภาษา C

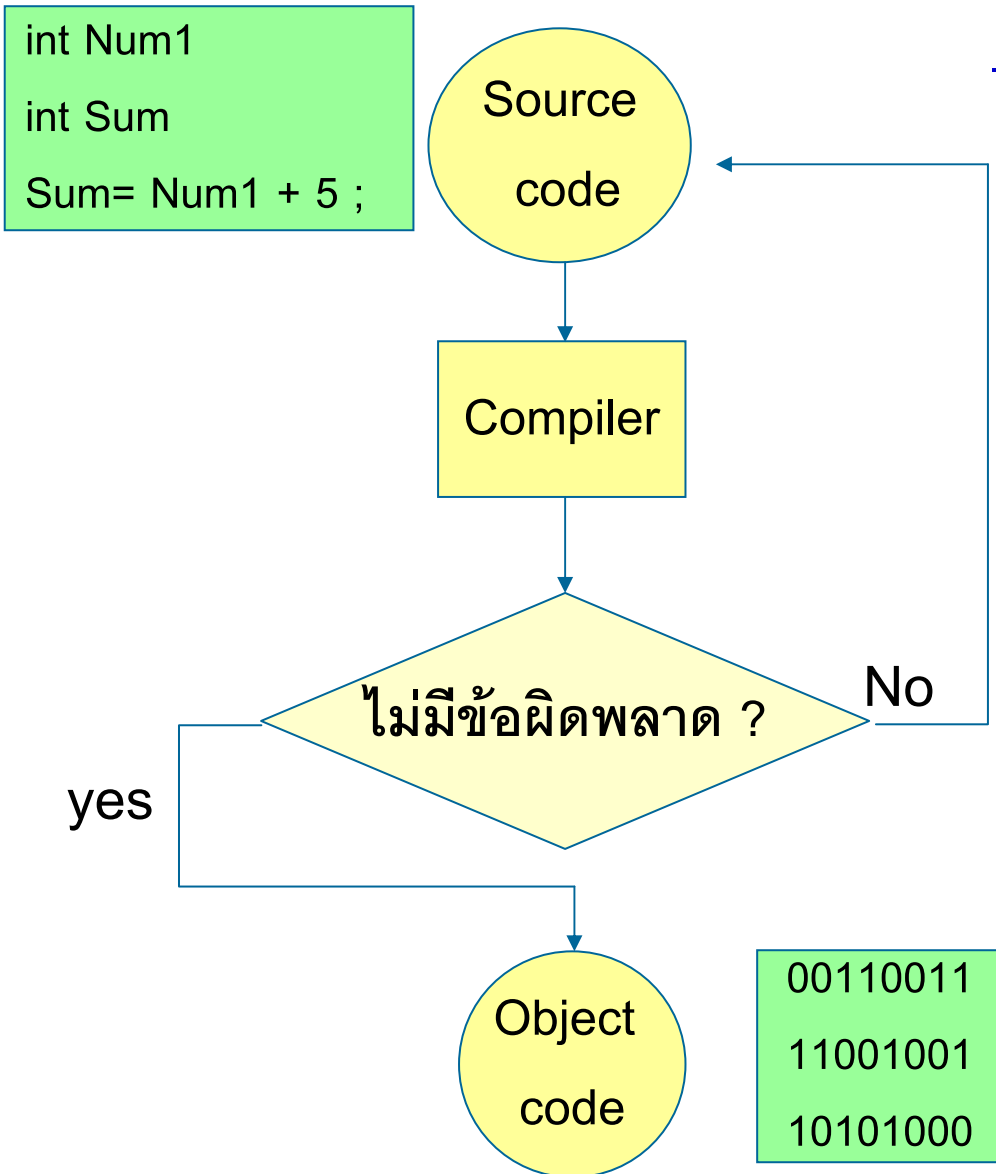
```
010001000000000100000000000001010  
01101000001000010000000000000010  
010011000010000100000000000010100
```

ภาษา Machine

Compiler และ Linker(2/2)

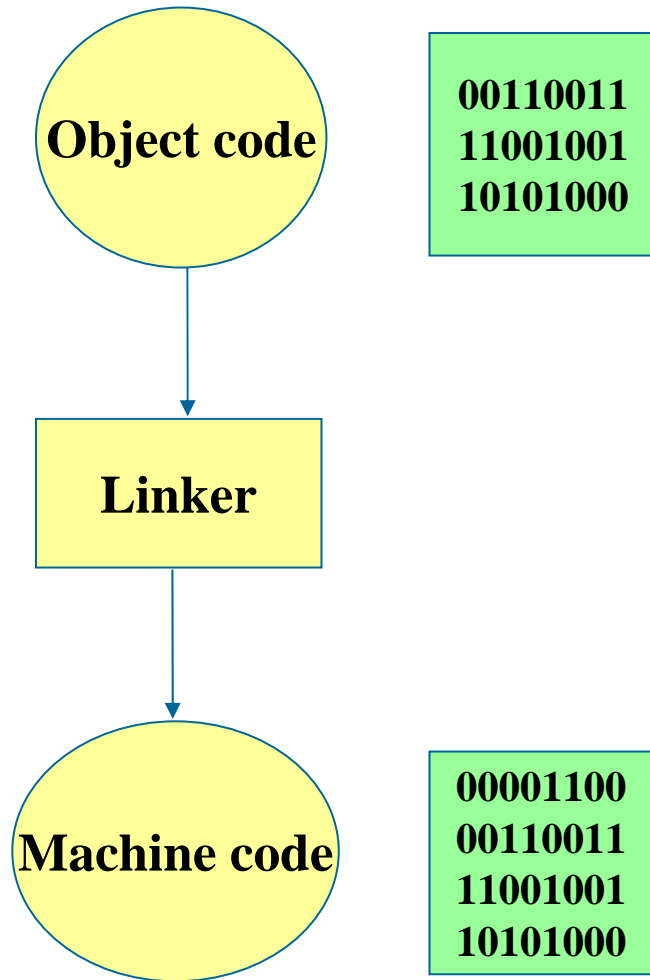
เนื่องจาก CPU เข้าใจได้เฉพาะรหัสคำสั่งที่อยู่ในรูปเลขฐานสองเท่านั้น จึงต้องมีตัวที่ทำหน้าที่ในการแปลหรือแปลงรหัสต้นแบบ(Source Code) ของภาษา C ให้อยู่ในรูปของรหัสเลขฐานสอง เรียกตัวนี้ว่า ตัวแปลภาษา (Compiler)

การทำงานของ compiler

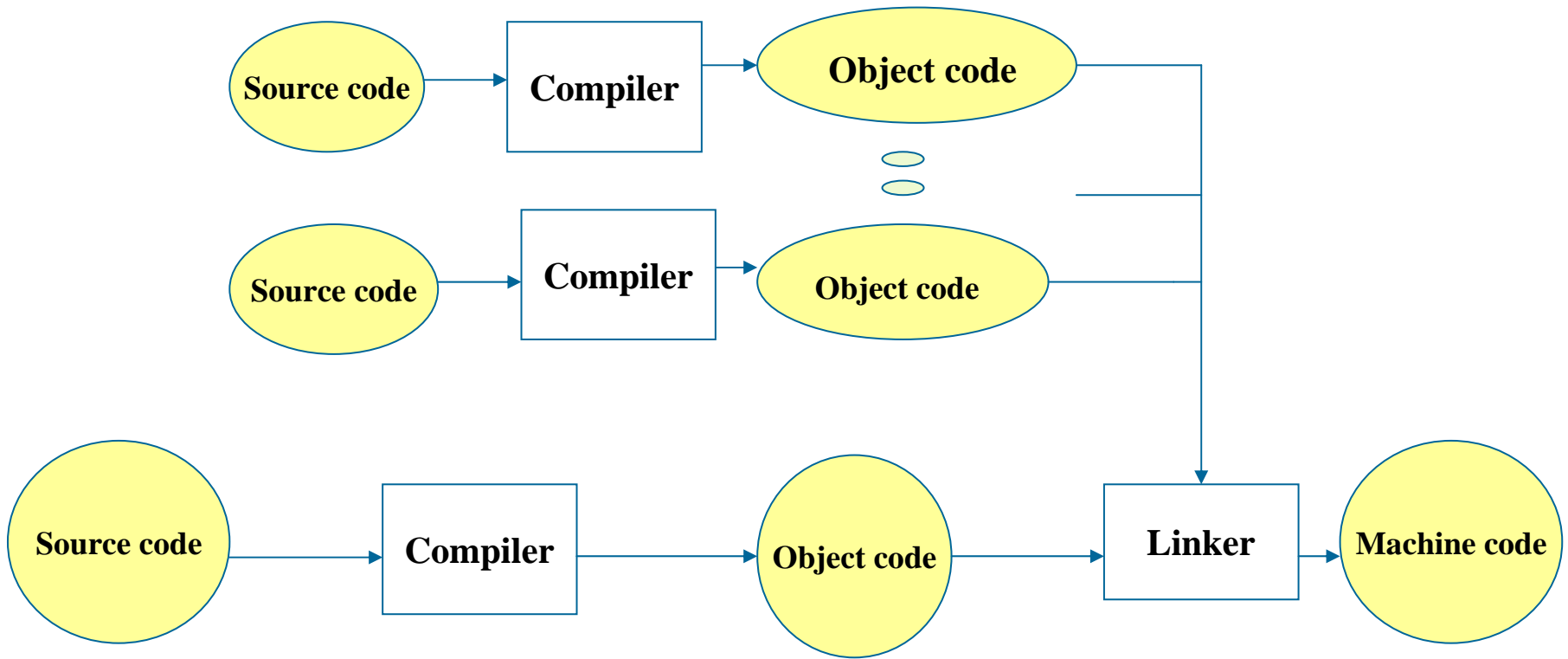


- ตรวจสอบและตรวจจับข้อผิดพลาดของโปรแกรม
- ถ้าโปรแกรมไม่มีข้อผิดพลาดแล้วจะถูกแปลจาก **source code** ไปเป็นรหัสจุดหมาย (**object code**)
- object code มีลักษณะคล้ายกับ รหัสภาษาเครื่อง(**machine code**) แต่ไม่สามารถดำเนินการได้ (executable)

ตัวเชื่อมโยง(linker)



linker จะเปลี่ยน
object code ไปเป็น
รหัสภาษาเครื่องที่
สามารถดำเนินการได้
ด้วยตัวเอง



```

int Num
Num = 10 ;
printf('%d', Num) ;
  
```

```

00110011
11001001
10101000
  
```

```

00001100
00110011
11001001
10101000
10100010
  
```

Executable program

linker จะทำการรวม object codes อื่น เพื่อสร้างรูปแบบของโปรแกรมที่สามารถดำเนินการได้ ในกรณีนี้ที่ source code มีบางส่วนของคำสั่ง ถูกประกาศไว้ในโปรแกรมอื่น จากตัวอย่างข้างบน คำสั่ง printf ซึ่งพิมพ์ค่าของ Num บนจอภาพ จะถูกกำหนดไว้ในโปรแกรมอื่น linker จะรวม object code ที่เกี่ยวข้องกันกับ object code หลักเพื่อสร้างโปรแกรมที่สามารถ execute ได้

เครื่องมือในการพัฒนาโปรแกรม

- สิ่งที่จะช่วยในการพัฒนาโปรแกรมเรียกว่า IDE-Integrated development environment ซึ่งสามารถทำงานได้ ดังนี้
- แก้ไขโปรแกรม หรือเขียนส่วนของข้อความที่เก็บคำสั่งของโปรแกรม โดยการใช้ editor ในตัวมันเอง
- การแปลหรือตีความคำสั่งของโปรแกรม จะแสดงข้อผิดพลาดที่เกิดขึ้นและช่วยให้ผู้ใช้แก้ไขโปรแกรมให้ถูกต้อง

เครื่องมือในการพัฒนาโปรแกรม

- เชื่อมโยงโปรแกรมหรือรหัสจุดหมายหลาย ๆ ตัว
- ดำเนินการตามคำสั่งของโปรแกรมหรือ Run โปรแกรม
- บันทึกโปรแกรมใน floppy หรือ hard disc
- Debugging โปรแกรม ปกติแล้วตัวช่วยในการพัฒนาโปรแกรมจะมาพร้อมกับตัวแปลภาษาหรือคอมไพเลอร์ ตัวอย่างของ IDE เช่น

Microsoft → Visual C++

Borland → Turbo C, Turbo C++