



About Programming  
101!

# เครื่องทอผ้าของแจ็กการ์ด (1804)

1. **การควบคุมด้วยแผ่นกระดาษเจาะรู:** เครื่องทอผ้าของแจ็กการ์ดใช้แผ่นกระดาษเจาะรูเพื่อควบคุมลวดลายการทอผ้า เปรียบเสมือนการเขียนโปรแกรมด้วยชุดคำสั่ง โดยที่รูบนกระดาษเปรียบเสมือน "รหัส" ที่กำหนดว่าเส้นด้ายจะถูกทออย่างไร
2. **การทำงานโดยอัตโนมัติ:** เครื่องทอผ้าของแจ็กการ์ดสามารถทำงานโดยอัตโนมัติโดยไม่ต้องใช้คนควบคุม เปรียบเสมือนโปรแกรมคอมพิวเตอร์ที่ทำงานตามคำสั่งโดยไม่ต้องมีการแทรกแซงจากมนุษย์
3. **ความคิดสร้างสรรค์:** เครื่องทอผ้าของแจ็กการ์ดช่วยให้สามารถสร้างลวดลายผ้าที่ซับซ้อนและสวยงาม เปรียบเสมือนโปรแกรมคอมพิวเตอร์ที่สามารถสร้างสรรค์ผลงานต่างๆ ได้

เปรียบเทียบกับโปรแกรมคือ

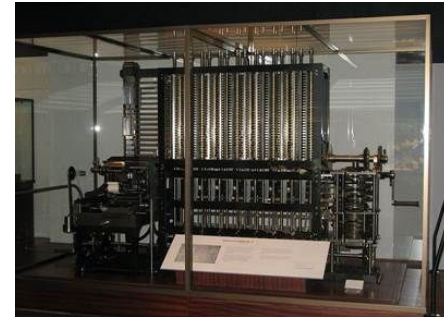
- โปรแกรมคือชุดคำสั่งที่บอกคอมพิวเตอร์ว่าต้องทำอะไร
- โปรแกรมช่วยให้คอมพิวเตอร์ทำงานโดยอัตโนมัติ
- โปรแกรมสามารถใช้สร้างสรรค์ผลงานต่างๆ ได้



## เครื่องวิเคราะห์ของชาร์ลส์ แบบเบจ (1822)

1. **เครื่องจักรกลที่ทำงานตามโปรแกรม:** เครื่องวิเคราะห์ของแบบเบจออกแบบมาเพื่อคำนวณตามชุดคำสั่งที่กำหนดไว้ล่วงหน้า เปรียบเสมือนคอมพิวเตอร์ยุคใหม่ที่ทำงานตามโปรแกรม
2. **ภาษาโปรแกรม:** Ada Lovelace บุตรสาวของ Lord Byron เขียนบันทึกเกี่ยวกับเครื่องวิเคราะห์ของแบบเบจ เธออธิบายวิธีการเขียนโปรแกรมสำหรับเครื่องวิเคราะห์ ถือเป็นการริเริ่มแนวคิดภาษาโปรแกรม
3. **ความซับซ้อน:** เครื่องวิเคราะห์ของแบบเบจออกแบบมาให้ทำงานได้ซับซ้อนกว่าเครื่องทอผ้าของแจ็กการ์ด แสดงให้เห็นถึงพัฒนาการของเทคโนโลยี

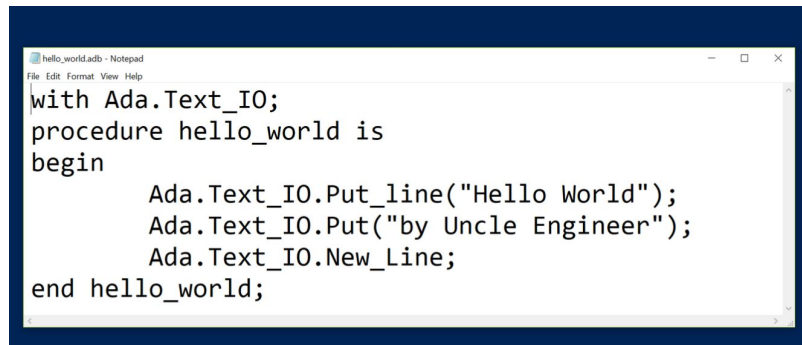
- มีหน่วยความจำจัดเก็บชุดข้อมูลตัวเลขได้



## เครื่องวิเคราะห์ของชาร์ลส์ แบบเบจ (1822)

1. **ภาษาโปรแกรมสำหรับเครื่องจักรกล:** ภาษา Ada พัฒนาขึ้นเพื่อใช้งานกับเครื่องวิเคราะห์ของชาร์ลส์ แบบเบจ แสดงให้เห็นถึงความพยายามในการสร้างภาษาที่สามารถสื่อสารกับเครื่องจักรได้
2. **การออกแบบโดย Ada Lovelace:** ภาษา Ada ออกแบบโดย Ada Lovelace บุตรสาวของ Lord Byron เธอได้รับการยกย่องว่าเป็น "โปรแกรมเมอร์คนแรกของโลก"
3. **อิทธิพลต่อภาษาโปรแกรมยุคใหม่:** ภาษา Ada มีอิทธิพลต่อภาษาโปรแกรมยุคใหม่หลายภาษา เช่น Pascal, C++, Java

เป็นภาษาที่ถูกยอมรับในวงการเครื่องบินและยานอวกาศ ใช้มาตั้งแต่ปี 1980



```
hello_world.adb - Notepad
File Edit Format View Help
with Ada.Text_IO;
procedure hello_world is
begin
    Ada.Text_IO.Put_line("Hello World");
    Ada.Text_IO.Put("by Uncle Engineer");
    Ada.Text_IO.New_Line;
end hello_world;
```

# ภาษาเครื่อง (Machine Language)

เป็นภาษาที่ใช้สื่อสารกับคอมพิวเตอร์โดยตรง เป็นภาษาที่คอมพิวเตอร์เข้าใจและทำงานตามได้โดยไม่ต้องผ่านตัวแปลภาษา (Compiler) หรือล่าม (Interpreter) ภาษาเครื่องจะประกอบด้วยรหัสเลขฐานสอง (Binary) คือ 0 และ 1 เท่านั้น

Binary Code							
A	100 0001	H	100 1000	O	100 1111	V	101 0110
B	100 0010	I	100 1001	P	101 0000	W	101 0111
C	100 0011	J	100 1010	Q	101 0001	X	101 1000
D	100 0100	K	100 1011	R	101 1010	Y	101 1001
E	100 0101	L	100 1100	S	101 0011	Z	101 1010
F	100 0110	M	100 1101	T	101 0100	a	110 0001
G	100 0111	N	100 1110	U	101 0101	b	110 0010

## Exercise

01001001 00100000 01100010 01101111 01110110  
01100111 หาค่าแปลของเลข Binary



# Compiler

Compiler คือ โปรแกรมคอมพิวเตอร์ที่ทำหน้าที่แปลงชุดคำสั่งภาษาคอมพิวเตอร์หนึ่ง ไปเป็นชุดคำสั่งที่มีความหมายเดียวกัน ในภาษาคอมพิวเตอร์อื่น

## หน้าที่หลักของ Compiler

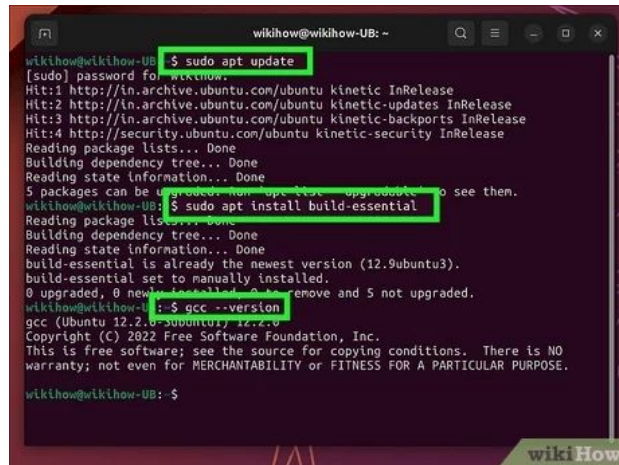
1. แปลงโค้ดภาษาโปรแกรมระดับสูง (High-Level Language) เช่น ภาษา C++ ภาษา Java ไปเป็นภาษาเครื่อง (Machine Language) ที่คอมพิวเตอร์เข้าใจ
2. ตรวจสอบข้อผิดพลาดในโค้ดภาษาโปรแกรม
3. แปลงโค้ดในโปรแกรม

## ประเภทของ Compiler

1. **Compiling Compiler:** แปลงโค้ดภาษาโปรแกรมทั้งหมดไปเป็นภาษาเครื่องก่อนรันโปรแกรม
2. **Interpreting Compiler:** แปลงโค้ดภาษาโปรแกรมทีละบรรทัดพร้อมรันโปรแกรมไปพร้อมกัน

## ตัวอย่าง Compiler

1. GCC (c, c++)
2. Javac (java)
3. Clang (c, c++)



```
wikihow@wikihow-UB: ~  
wikihow@wikihow-UB:~$ sudo apt update  
[sudo] password for wikihow:  
Hit:1 http://in.archive.ubuntu.com/ubuntu kinetic InRelease  
Hit:2 http://in.archive.ubuntu.com/ubuntu kinetic-updates InRelease  
Hit:3 http://in.archive.ubuntu.com/ubuntu kinetic-backports InRelease  
Hit:4 http://security.ubuntu.com/ubuntu kinetic-security InRelease  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
5 packages can be upgraded: non-free libc++1:amd64 upgradeable to see them.  
wikihow@wikihow-UB:~$ sudo apt install build-essential  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
build-essential is already the newest version (12.9ubuntu3).  
build-essential set to manually installed.  
0 upgraded, 0 newly installed, 0 to remove and 5 not upgraded.  
wikihow@wikihow-UB:~$ gcc --version  
gcc (Ubuntu 12.2.0-14ubuntu1) 12.2.0  
Copyright (C) 2022 Free Software Foundation, Inc.  
This is free software; see the source for copying conditions. There is NO  
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  
wikihow@wikihow-UB:~$
```

# ภาษาประกอบ (Assembly Language)

เป็นภาษาโปรแกรมระดับต่ำ (Low-level language) ที่อยู่ใกล้เคียงกับภาษาเครื่อง (Machine Language) มากที่สุด ภาษาประกอบถูกออกแบบมาเพื่อเขียนโปรแกรมให้คอมพิวเตอร์ทำงานโดยตรง โดยไม่ต้องผ่านตัวแปลภาษา (Compiler) หรือสาม (Interpreter)

1. เขียนด้วย mnemonics แทนรหัสเลขฐานสอง (Binary) ของภาษาเครื่อง
2. เข้าใจง่ายกว่าภาษาเครื่อง แต่ยากกว่าภาษาโปรแกรมระดับสูง
3. เขียนโปรแกรมได้รวดเร็วและมีประสิทธิภาพสูง
4. แต่ละเครื่องคอมพิวเตอร์มีภาษาประกอบเฉพาะของตัวเอง

## ตัวอย่างภาษาประกอบ

1. **x86 assembly:** ภาษาประกอบสำหรับคอมพิวเตอร์ที่ใช้สถาปัตยกรรม x86
2. **ARM assembly:** ภาษาประกอบสำหรับคอมพิวเตอร์ที่ใช้สถาปัตยกรรม ARM

```
proceed      P [=address] [number]
quit         Q
register     R [register]
search      S range list
trace       T [=address] [value]
unassemble  U [range]
write       W [address] [drive] [firstsector] [number]
allocate expanded memory  XA [mpages]
deallocate expanded memory XD [handle]
map expanded memory pages XM [Lpage] [Ppage] [handle]
display expanded memory status XS

-r
0X-0000 0X-0000 0X-0024 0X-0000 0X-FFFF 0X-0000 0X-0000 0X-0000
0X-239F 0X-239F 0X-239F 0X-239F 0X-239F 0X-0100 0X-0100 0X-0100
239F:0100 0X-0100 MOV DX,0100
-d
239F:0100 0X-0100 0X-0100 0X-0100 0X-0100 0X-0100 0X-0100 0X-0100 .....f..L..Well
239F:0110 0X-0110 0X-0110 0X-0110 0X-0110 0X-0110 0X-0110 0X-0110 o World, YouTube
239F:0120 0X-0120 0X-0120 0X-0120 0X-0120 0X-0120 0X-0120 0X-0120 f..$.
239F:0130 0X-0130 0X-0130 0X-0130 0X-0130 0X-0130 0X-0130 0X-0130 .....
239F:0140 0X-0140 0X-0140 0X-0140 0X-0140 0X-0140 0X-0140 0X-0140 .....
239F:0150 0X-0150 0X-0150 0X-0150 0X-0150 0X-0150 0X-0150 0X-0150 .....
239F:0160 0X-0160 0X-0160 0X-0160 0X-0160 0X-0160 0X-0160 0X-0160 .....
239F:0170 0X-0170 0X-0170 0X-0170 0X-0170 0X-0170 0X-0170 0X-0170 .....
-
```

# Assembly Language vs Machine Language

```
section      .text
global      _start
_start:
    mov edx, len
    mov ecx, msg

    mov ebx, 1

    mov eax, 4
    int 0x80

    mov eax, 1
    int 0x80

section      .data
msg          db "Hello world!"
len          equ $ -msg
```

```
#include <stdio.h>

int main()
{
    printf("Hello World!\n");
    return 0;
}
```