

## Week 4,5: Function and Program structure

• การเขียน fn

ส่วนประกอบ	การเขียน fn	การเขียน fn
function header	int isEven (int value)	return - value - type (parameter-list)
function body	{ if (value % 2 == 0) return 1; else return 0; }	{ declarations statement statement }

• Function Prototype → function header 1 ครั้ง ;  
เช่น  
    int isEven (int value) ;

• Standard Library Function  
เช่น ฟังก์ชัน printf, scanf และ getch

7 21.71. 63

- คำว่า extern และ static → กำหนดให้ฟังก์ชันอยู่ใน memory
- คำว่า global และ static fn ถูกกำหนดให้เป็น extern โดย default

การกำหนด fn

- คำว่า local ถูกกำหนดโดย static ซึ่งอยู่ภายใน fn คำว่า global จะอยู่ใน memory

## C) สร้าง static library

gcc -c func.c

ar cr libfunc.a func.o

สร้าง func ในไฟล์

int foo()

{

return ส่วนที่เรากลับมา

}

→ เปิดอีกไฟล์

→ ทดสอบ int foo() ไปวาง

int foo();

→ เปิดอีกไฟล์

→ ทดสอบในโปรแกรม

→ ลงรายละเอียด ทดสอบ code foo.h

\* include "foo.h"

gcc -c foo.c

gcc -c main.c

gcc main.o foo.o -o main.exe

./main.exe

rand() → ช่วงค่า 0 - RAND\_MAX

rand() → เป็นค่าคงที่ที่กำหนดไว้แล้วใน <stdlib.h>  
srand() → ใช้ seed เพื่อไม่ให้ระบบ rand() วนซ้ำค่าเดิม

ข้อสอบ mid → ข้อสอบ 25 ข้อ  
ข้อสอบ 100 ข้อ  
ข้อสอบ 125 point

การสร้างฟังก์ชัน

```
int ชื่อ ( ) {  
    int ตัวแปรที่ใช้คำนวณในสูตร  
}
```

→ ทำอะไรก็ได้

return [ส่งอะไรกลับ?]

}

```
int main ( )
```

```
{
```

ใช้ฟังก์ชันที่สร้างไว้ (เรียกใช้)

```
}
```

- คลิกซ้ายที่ค่าที่พิมพ์ผิด แล้วกด Ctrl + d จะเลือกคำในทั้งบรรทัด  
แล้วสามารถลบทิ้งได้เลย

- เมื่อฟังก์ชันทำงานเสร็จ return จะส่งค่ากลับโดยที่ค่าจะหายไปจาก memory  
แล้วส่งค่ากลับไปยังตัวแปร

- ถ้าประกาศ int x; เดิมทีมันจะกำหนดให้อัตโนมัติว่า x = 0