

Computer Engineering

วิศวกรรมคอมพิวเตอร์



บทที่ 2 ชนิดข้อมูล การเขียนโปรแกรมเบื้องต้น^{ชั้นป.ตรี} และคำสั่งแสดงผลบนจอภาพ

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

- รู้จักชนิดข้อมูลในภาษาซี
- สามารถเขียนโปรแกรมภาษาซีอย่างง่าย ๆ ได้
- สามารถแสดงข้อความอ(okทางจอภาพ)ได้
- สามารถแสดงข้อมูลชนิดต่างๆ อ(okทางจอภาพ)ได้
- สามารถกำหนดรูปแบบการแสดงข้อมูลอ(okทางจอภาพ)ได้

โครงสร้างโปรแกรมภาษาซี

01006012 Computer Programming

```
#include<file.h>
type function_name(type);
type variable;
int main()
{
    type variable;    // ตัวแปรชนิดโลคอล
    statement-1;
    ...
    statement-n;
    return 0;
}
type function_name(type variable)
{
    statement-1;
    ...
    statement-n;
    return(var);
}
```

พรีprocessor ไดเรคทีฟ
ฟังก์ชันโปรแกรม
ตัวแปรชนิด กอบอล

ตัวแปรชนิด โลคอล

คำสั่ง

ฟังก์ชันหลัก

ส่วนหัวโปรแกรม

ส่วนตัวโปรแกรม

ฟังก์ชันย่อย

โครงสร้างโปรแกรมภาษาซี

01006012 Computer Programming

- ส่วนหัวโปรแกรม (Header Program)
 - พรีprocessor directive (Preprocessor Directive)

#include

#define

- ฟังก์ชันโปรแกรม (Function Prototype Declaration)
- ตัวแปรชนิดโภบด (Global Variable Declaration)

2.1 โครงสร้างโปรแกรมภาษาซี

01006012 Computer Programming

- ส่วนตัวโปรแกรม (Body Program)
 - พื้นก์ชั้นหลัก (Main Function)
 - ตัวแปรชนิด局部 (Local Variable Declaration)
 - คำสั่ง (Statement)
 - พื้นก์ชั้นอื่น ๆ (Functions)
- คำอธิบายโปรแกรม (Comment Program)

2.2 พรีprocessor ไนเดเร็คทีฟ

01006012 Computer Programming

- ส่วนประมวลผลก่อน เป็นส่วนที่ส่งให้คอมไพล์อร์เตรียมการทำงานที่กำหนดไว้ ก่อนที่จะทำงานในฟังก์ชันหลัก

```
#include<file-name.h>
```

ให้โปรแกรมดึงคำสั่งจาก *file-name.h* ใน Include Directory

```
#include"file-name.h"
```

ให้โปรแกรมดึงคำสั่งจาก *file-name.h* ใน Current Directory

file-name คือชื่อไฟล์นามสกุล h

ตัวอย่างคำสั่งฟรีโปรเซเชอร์ไดเร็คทีฟ

01006012 Computer Programming

#include<stdio.h> standard input output

ทำการเรียกไฟล์ stdio.h เพื่อทำให้สามารถใช้คำสั่งทั่วไปได้

#include<conio.h> console input output

ทำการเรียกไฟล์ conio.h เพื่อทำให้สามารถใช้คำสั่งจัดการหน้าจอ และคำสั่งรับและแสดงผลเพิ่มเติม

#include<math.h>

ทำการเรียกไฟล์ math.h เพื่อทำให้สามารถใช้คำสั่งเกี่ยวกับคณิตศาสตร์ได้

#include<string.h>

ทำการเรียกไฟล์ string.h เพื่อทำให้สามารถใช้คำสั่งจัดการเกี่ยวกับข้อความได้

- ฟังก์ชันหลักเป็นฟังก์ชันที่โปรแกรมภาษาซีต้องมีอยู่เสมอ เพราะคอมไพล์อร์ของภาษาซีจะเริ่มต้นการทำงานจากฟังก์ชันหลัก
- ตัวอย่างการเขียนฟังก์ชันหลักที่ไม่มีการทำงานใดๆ

```
int main(void)
{
    return(0);
}
```

```
int main()
{
    return 0;
}
```

กฎเกณฑ์การใช้คำสั่งในภาษาซี

01006012 Computer Programming

- ใช้เครื่องหมาย semi colon ; เป็นจุดสิ้นสุดคำสั่ง
- ใช้อักษรตัวเล็กสำหรับเรียกใช้คำสั่ง (statement)
- ใช้เครื่องหมาย comma , สำหรับคืนตัวแปร และพารามิเตอร์
- หากคำสั่งใดมีคำสั่งส่วนย่อยภายในหลาย ๆ คำสั่ง ให้ใช้เครื่องหมายปีกกา { } สำหรับกำหนดขอบเขต

2.3 ชนิดของข้อมูลในภาษาซี

01006012 Computer Programming

- ภาษาซี รู้จักข้อมูล 4 ชนิดหลัก

int

ข้อมูลชนิดจำนวนเต็ม

float

ข้อมูลชนิดจำนวนทศนิยม

char

ข้อมูลชนิดอักขระ

void

ข้อมูลชนิดว่างเปล่า

2.3.1 ข้อมูลชนิดจำนวนเต็ม / integer

01006012 Computer Programming

int เป็นพารามิเตอร์หลักที่ใช้กับข้อมูลชนิดจำนวนเต็ม

unsigned int ข้อมูลชนิดจำนวนเต็มไม่คิดเครื่องหมายขนาด 2 bytes
หรือ 4 byte

short int ข้อมูลชนิดจำนวนเต็มขนาด 2 bytes

int ข้อมูลชนิดจำนวนเต็มขนาด 2 bytes หรือ 4 byte

(ในคอมไพร์ 32 bit ตัวแปร int มีขนาด 4 byte แต่ในคอมไพร์ 16 bit ตัวแปร int มีขนาด 2 byte)

unsigned long ข้อมูลชนิดจำนวนเต็มไม่คิดเครื่องหมายขนาด 4 bytes

long ข้อมูลชนิดจำนวนเต็มขนาด 4 bytes

2.3.2 ข้อมูลชนิดจำนวนทศนิยม / float

01006012 Computer Programming

float เป็นพารามิเตอร์หลักที่ใช้กับข้อมูลชนิดจำนวนทศนิยม โดยมีการใช้งาน 3 รูปแบบดังนี้

float	ข้อมูลชนิดจำนวนทศนิยมขนาด 4 byte
double	ข้อมูลชนิดจำนวนทศนิยมขนาด 8 byte
long double	ข้อมูลชนิดจำนวนทศนิยมขนาด 10 byte

2.3.3 ข้อมูลชนิดอักขระ / character

01006012 Computer Programming

char เป็นพารามิเตอร์ที่ใช้งานเกี่ยวกับตัวอักษร และข้อความในภาษาซี โดยมีการกำหนดค่าอักขระโดยให้อยู่ในเครื่องหมาย single quote (' . . . ') เช่น 'C' , 'o' , 'm' , '1' โดยมีรูปแบบการใช้งาน 2 รูปแบบ

unsigned char ข้อมูลชนิดอักขระไม่คิดเครื่องหมาย

char ข้อมูลชนิดอักขระปกติ

2.3.4 ข้อมูลชนิดว่างเปล่า / void

01006012 Computer Programming

void เป็นพารามิเตอร์ที่ใช้งานในส่วนของ พังก์ชัน โปรแกรม
ไทย การสร้าง และใช้งานพังก์ชัน เพื่อแสดงให้รู้ว่าพังก์ชันที่
สร้างขึ้นมาไม่มีการส่ง หรือรับค่าจากการเรียกใช้งานพังก์ชัน

2.3.5 ตารางแสดงขนาดและขอบเขตชนิดของข้อมูล

01006012 Computer Programming

ชนิดของตัวแปร	ขนาด	ค่าต่ำสุด	ค่าสูงสุด
<code>unsigned char</code>	8 bit	0	255
<code>char</code>	8 bit	-128	127
<code>unsigned int</code>	16 bit	0	65,535
<code>short int</code>	16 bit	-32,768	32,767
<code>int</code>	16 bit	-32,768	32,767
<code>unsigned long</code>	32 bit	0	4,294,967,295
<code>long</code>	32 bit	-2,147,483,648	2,147,483,647
<code>float</code>	32 bit	$\pm 3.4 \times 10^{-38}$	$\pm 3.4 \times 10^{38}$
<code>double</code>	64 bit	$\pm 1.7 \times 10^{-308}$	$\pm 1.7 \times 10^{308}$
<code>long double</code>	80 bit	$\pm 3.4 \times 10^{-4932}$	$\pm 3.4 \times 10^{4932}$

ความละเอียดต่ำสุดที่รองรับ



ความละเอียดสูงสุดที่รองรับ 15

2.4 การแสดงผลด้วยคำสั่ง printf

01006012 Computer Programming

- การเขียนโปรแกรมจำเป็นต้องมีการแสดงผล เพื่อให้ผู้ใช้งานทราบว่าโปรแกรมสามารถทำงานอะไร จำเป็นต้องป้อนค่าอะไรบ้าง และเมื่อโปรแกรมทำงานเสร็จ ผลลัพธ์ที่ได้เป็นอย่างไร

รูปแบบของคำสั่ง printf

01006012 Computer Programming

printf (format-string, data-list);

format-string

คือรูปแบบของข้อความซึ่งจะประกอบด้วย
ข้อความธรรมดា, ค่ารหัส ASCII และ ส่วน
แสดงชนิดข้อมูล โดย format-string จะอยู่ใน
เครื่องหมาย Double quote ""

data-list

คือชื่อข้อมูล หรือตัวแปรที่จะทำการแสดงผล
ตามส่วนแสดงชนิดข้อมูล ใน format-string

หมายเหตุ คำสั่ง printf ต้องเรียกใช้ Preprocessor Directive `#include<stdio.h>`

การใช้งานคำสั่ง printf โดยไม่ include preprocessor directive

ming

Solution Explorer - Solution 'T...' X

helloworld.cpp Start Page

(Global Scope)

```
1 int main()
2 {
3     printf("hello world!");
4     return 0;
5 }
```

Output

Show output from: Build

```
1>Compiling...
1>helloworld.cpp
1>d:\temp\test\test\helloworld.cpp(3) : error C3861: 'printf': identifier not found
1>Build log was saved at "file:///d:/Temp/Test/Debug/Test/Log.htm"
1>Test - 1 error(s), 0 warning(s)
===== Build: 0 succeeded, 1 failed, 0 up-to-date, 0 skipped =====
```

ก็ printf
มีอยู่ในนั้น
หมุน 90 องศา

include stdio.h

Find Results 1 Find Symbol Results Output Error List

18

http://www rabang

Build Failed

Ln 8

การใช้งานคำสั่ง printf ต้อง include stdio.h

01006012 Computer Programming

```
#include<stdio.h>
int main()
{
    printf ("Hello World!");
    return 0;
}
```

Hello World!

Format string - รูปแบบของข้อความในคำสั่ง printf

01006012 Computer Programming

- ข้อความธรรมดा เป็นส่วนที่แสดงตัวอักษร ตัวเลขออกโดยตรง
- ค่ารหัส ASCII เป็นส่วนที่ควบคุมรูปแบบการแสดงผล เช่น การจัดย่อหน้า การขึ้นต้นบรรทัดใหม่ เป็นต้น
- ส่วนแสดงชนิดข้อมูล เป็นการกำหนดรูปแบบของชนิดข้อมูลที่จะแสดงผลข้อมูล หรือตัวแปร
 - ส่วนแสดงชนิดข้อมูลแบบปกติ
 - ส่วนแสดงชนิดข้อมูลที่กำหนดรูปแบบการแสดงผล

ข้อความธรรมดा

01006012 Computer Programming

- การแสดงข้อความธรรมดานี้เป็นการใช้คำสั่ง printf ให้แสดงข้อความที่ต้องการอักขระจากภาพ โดยจะอยู่ในส่วนของ format-string ในคำสั่ง (อยู่ในเครื่องหมาย Double quote " ")

```
printf (format-string, data-list);
```

- เช่น คำสั่งในตัวอย่างโปรแกรมที่ผ่านมา

```
printf ("Hello World!");
```

จะทำการแสดงข้อความ Hello World! อักขระจากภาพหลังจากสั่ง Run โปรแกรม



ตัวอย่างการใช้งานคำสั่ง printf หลายๆ คำสั่ง

01006012 Computer Programming

```
#include<stdio.h>
#include<conio.h>
int main()
{
    printf ("Hello World!");
    printf ("This is my first Program.");
    printf ("I am a programmer.");
    return 0;
}
```

ค่ารหัส ASCII สำหรับอักขระพิเศษ

01006012 Computer Programming

- ค่ารหัส ASCII สำหรับอักขระพิเศษที่ใช้งานในภาษาซี ซึ่งใช้ควบคุณการแสดงผลในคำสั่ง printf ในส่วนของ format-string

ค่ารหัส ASCII	การใช้งาน
\t	เว้นช่องว่างทุก 1 แท็บ (8 ช่องตัวอักษร)
\n	ขึ้นบรรทัดใหม่
\0	เป็นอักขระว่าง
\ '	แสดง single quote ออกรหัสอักษร
\ "	แสดง double quote ออกรหัสอักษร
\ \	แสดง backslash ออกรหัสอักษร

ตัวอย่างการใช้งานคำสั่ง printf โดยใช้ค่ารหัส ASCII

01006012 Computer Programming

```
#include<stdio.h>
int main()
{
    printf ("Hello World! \n");
    printf ("This is my first Program.");
    printf ("\n\tI'm a programmer.");
    return 0;
}
```

จงแสดงชื่อ นามสกุลและเบอร์โทรศัพท์ของตนเองบนจอภาพ โดยให้ข้อมูลที่แสดงอยู่คณะบรรทัดกัน
01006012 Computer Programming

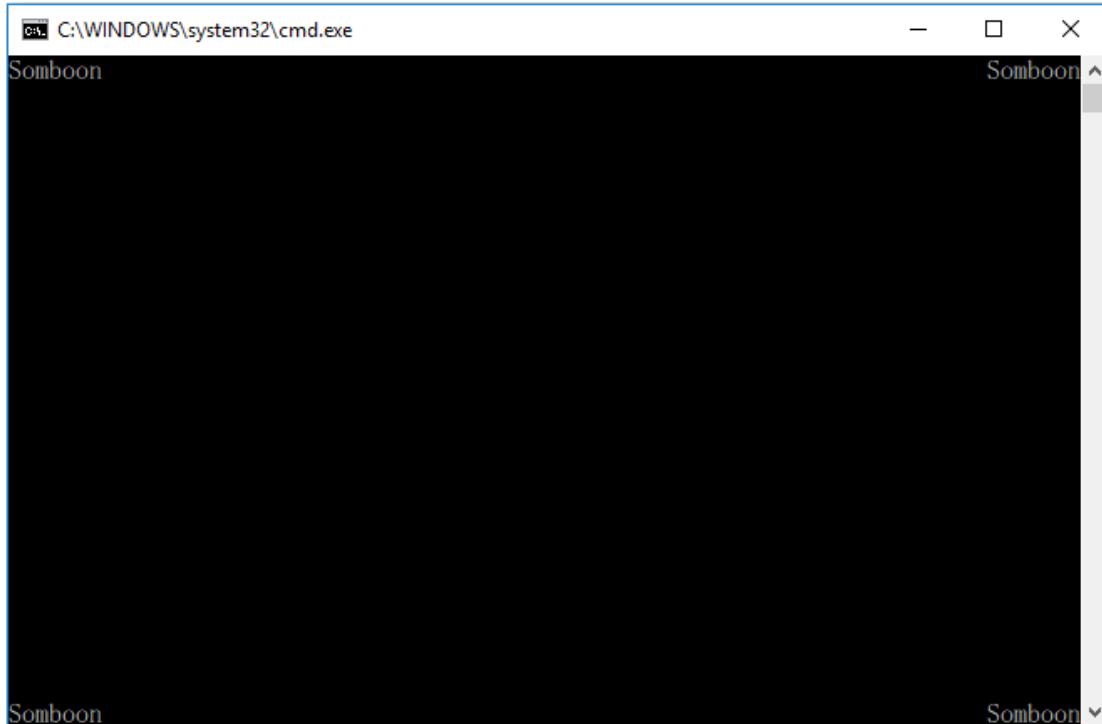
```
#include<stdio.h>
int main()
{
    printf ("My name is : John \n");
    printf ("My surname is : Smith\n");
    printf ("My telephone number is : 0819999999\n");
    return 0;
}
```

My name is : John

My surname is : Smith

My telephone number is : 0819999999

หากโปรแกรมต้องการให้แสดงผลชื่อของตัวเองปรากฏที่มุมทึ้ง 4 ของ
จอภาพ (จอภาพ 80 คอลัมน์ 25 บรรทัด)



ทำอย่างไร ???

ส่วนแสดงชนิดข้อมูล

01006012 Computer Programming

- ในการเขียนโปรแกรมที่มีการประยุกต์ขึ้น เช่น โปรแกรมคำนวณเลข จะต้องมีการแสดงผลลัพธ์ของค่าที่ทำการคำนวณ ซึ่งคำสั่ง printf สามารถแสดงผลค่าตัวแปรได้ แต่จะต้องเขียนโปรแกรมให้มีความสัมพันธ์กันในส่วนของ format-string และ data-list

```
printf (format-string, data-list) ;
```

- โดยที่ใน format-string จะมีส่วนแสดงชนิดข้อมูล
- และใน data-list จะมีข้อมูล หรือตัวแปรที่จะแสดงผล

การกำหนดส่วนแสดงชนิดข้อมูลแบบต่างๆ

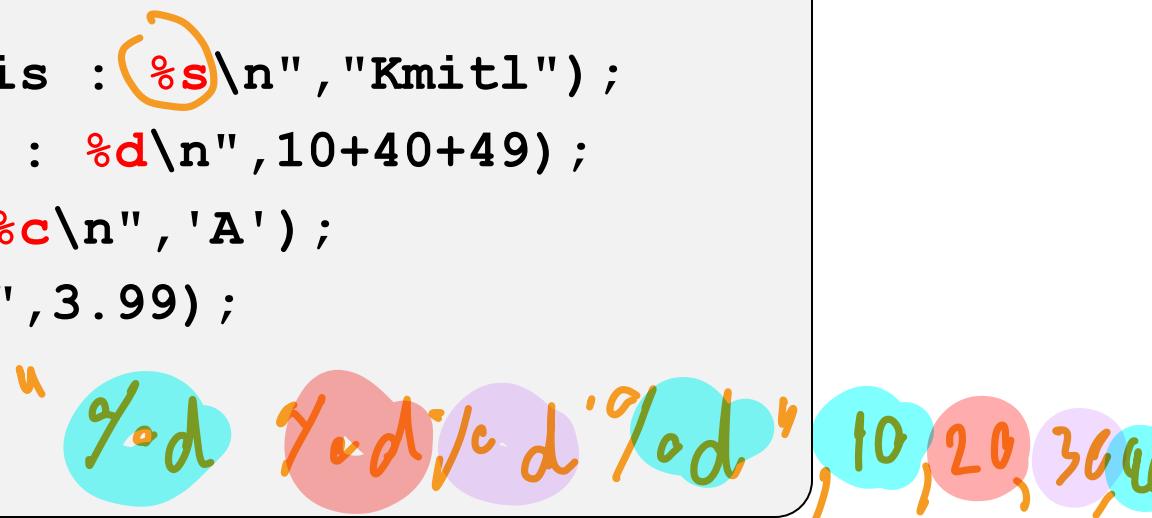
01006012 Computer Programming

ส่วนแสดงชนิดข้อมูล	การใช้งาน
%d 1	แสดงผลข้อมูลชนิดจำนวนเต็ม
%b	แสดงผลข้อมูลชนิดจำนวนเต็มบวก (ไม่คิดเครื่องหมาย)
%o	แสดงผลข้อมูลเป็นเลขฐานแปด
%x	แสดงผลข้อมูลเป็นเลขฐานสิบหก
%f 2	แสดงผลข้อมูลชนิดจำนวนทศนิยม (6 ตำแหน่ง)
%e	แสดงผลข้อมูลเป็นจำนวนทศนิยมและอยู่ในรูปยกกำลัง
%c 3	แสดงผลข้อมูลชนิดอักขระ
%s	แสดงผลข้อมูลชนิดข้อความ
%p	แสดงผลข้อมูลชนิดตัวชี้ตำแหน่ง

ตัวอย่างการใช้งานคำสั่ง printf แสดงผลข้อมูล

01006012 Computer Programming

```
#include<stdio.h>
int main()
{
    printf ("My name is : %s\n","Kmitl");
    printf ("My point : %d\n",10+40+49);
    printf ("Grade : %c\n",'A');
    printf ("GPA : %f",3.99);
    return 0;
}
```



การกำหนดรายชื่อตัวแปรในคำสั่ง printf

01006012 Computer Programming

- การแสดงผลตัวแปรโดยใช้คำสั่ง printf สามารถใช้งานโดยการใส่ข้อมูล หรือชื่อตัวแปรในส่วนของ data-list
- หากไม่ต้องการแสดงผลตัวแปร ไม่ต้องมีส่วนของ data-list
- หากต้องการแสดงผลข้อมูล หรือตัวแปรมากกว่าหนึ่งตัวในคำสั่งให้ใช้เครื่องหมาย comma , สำหรับคั่นชื่อตัวแปร โดยจะทำการแสดงผลตามลำดับตัวแปร และส่วนการแสดงชนิดข้อมูล

รายชื่อตัวแปรในคำสั่ง printf

01006012 Computer Programming

```
printf ("...%?...", data);
```

...data...

ตัวอย่าง

```
printf ("%d", 19);
```

...19...

```
printf ("%? %? ... %?", data-1, data-2, ..., data-n);
```

data-1 data-2 ... data-n

ตัวอย่าง

```
printf ("%d-%d-%d", 19, 1, 1980);
```

19-1-1980

ตัวอย่างการใช้งานคำสั่ง printf แสดงผลข้อมูลหลายตัว-1



01006012 Computer Programming

```
#include<stdio.h>
int main()
{
    printf ("Age = %d, GPA = %f\n",17,3.75);
    printf ("Programming: %f\nMechanics: %f",4.0,3.5);
    return 0;
}
```

ตัวอย่างการใช้งานคำสั่ง printf แสดงผลข้อมูลหลายตัว-2



01006012 Computer Programming

```
#include<stdio.h>
int main()
{
    printf ("Subject : %s(%d)\n", "Programming", 2552);
    printf ("Point    : %d\nGrade    : %c", 99, 'A');
    return 0;
}
```

Sub : Programming (2552)
Point : 99
Grade : A

ส่วนแสดงชนิดข้อมูลที่กำหนดรูปแบบการแสดงผล

01006012 Computer Programming

```
printf ("%m?", data);
```

- เป็นการจองพื้นที่หน้าจอจำนวน m ตัวอักษร แล้วแสดงผล data ชิดด้านขวาของพื้นที่ส่วนที่จองไว้ โดยชนิดข้อมูลตาม ? (หากความยาวเกินส่วนที่จองไว้ก็จะเลื่อนออกไป)

```
printf ("% -m?", data);
```

- เป็นการจองพื้นที่หน้าจอขนาด m ตัวอักษร แล้วแสดงผล data ชิดด้านซ้ายของพื้นที่ส่วนที่จองไว้ โดยชนิดข้อมูลตาม ?

ตัวอย่างการใช้งานคำสั่ง printf โดยมีการกำหนดรูปแบบ-1



01006012 Computer Programming

```
#include<stdio.h>
int main()
{
    printf ("123456789012345678901234567890");
    printf ("\n%20d*",46);
    printf ("\n%-20d*",46);
    printf ("\n%3d*",46);
    printf ("\n%3d*",2550);
    return 0;
}
```

```
123456789012345678901234567890
                                46*
                                *
46
46*
2550*
```

ตัวอย่างการใช้งานคำสั่ง printf โดยมีการกำหนดรูปแบบ-2



01006012 Computer Programming

```
#include <stdio.h>
int main()
{
    printf ("123456789012345678901234567890");
    printf ("\n%20c*", 'c');
    printf ("\n%-20c*", 'c');
    printf ("\n%10s*", "Pro");
    printf ("\n%10s*", "Programming");
    return 0;
}
```

```
123456789012345678901234567890
                                c*
                                *
        Pro*
Programming*
```

2.4 ส่วนแสดงชนิดข้อมูลที่กำหนดรูปแบบการแสดงผล

01006012 Computer Programming

```
printf ("% .n?", data);
```

- เป็นการกำหนดให้แสดงจำนวนทศนิยม n ตำแหน่ง สำหรับ %f หรือแสดงอักขระจำนวน n ตัว สำหรับ %s

```
printf ("%m .n?", data);
```

- เป็นการจองพื้นที่หน้าจอขนาด m ตัวอักษรแล้วแสดงผล data จำนวนทศนิยม n ตำแหน่ง สำหรับ %f หรือแสดงอักขระจำนวน n ตัว สำหรับ %s

ตัวอย่างการใช้งานคำสั่ง printf โดยมีการกำหนดรูปแบบ-3



01006012 Computer Programming

```
printf ("123456789012345678901234567890") ;  
printf ("\n%20s*", "programming") ;  
printf ("\n%-20s*", "programming") ;  
printf ("\n%.3s*", "programming") ;  
printf ("\n%20.3s*", "programming") ;  
printf ("\n%-20.3s*", "programming") ;
```

123456789012345678901234567890

programming*

programming *

pro*

pro*

pro * _

ตัวอย่างการใช้งานคำสั่ง printf โดยมีการกำหนดรูปแบบ-4



01006012 Computer Programming

```
printf ("123456789012345678901234567890") ;  
printf ("\n%20f*", 1234.56789) ;  
printf ("\n%-20f*", 1234.56789) ;  
printf ("\n%.3f*", 1234.56789) ;  
printf ("\n%20.3f*", 1234.56789) ;  
printf ("\n%-20.3f*", 1234.56789) ;
```

123456789012345678901234567890

1234.567890*

1234.567890 *

1234.568*

1234.568*

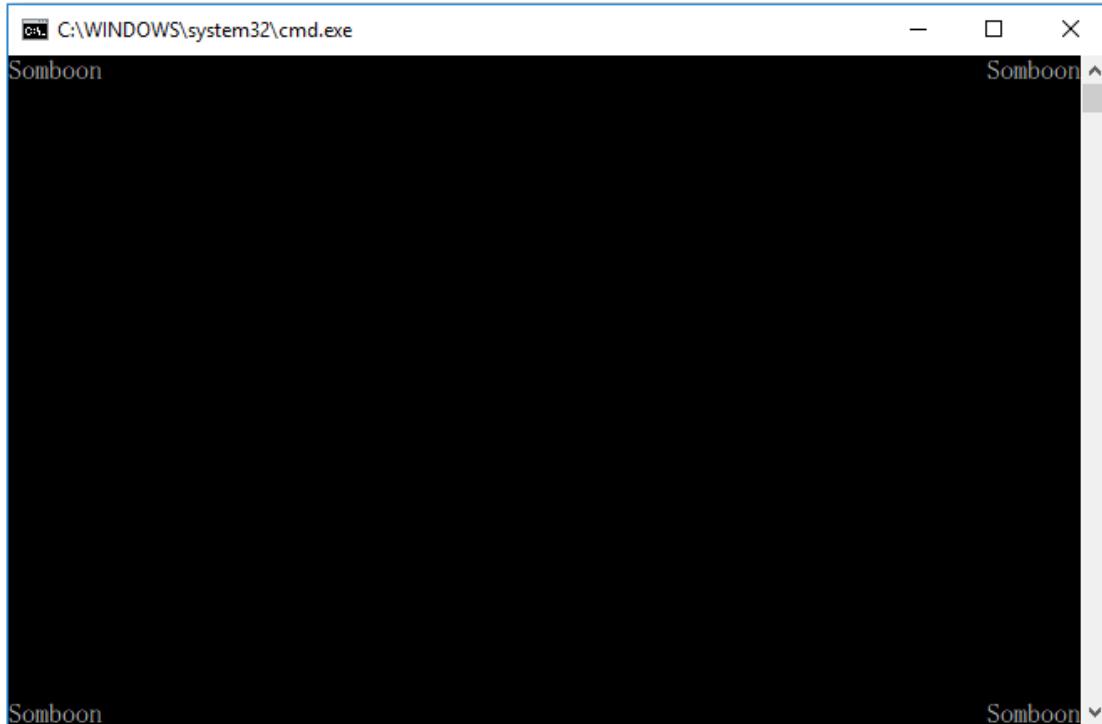
1234.568 * _

2.5 คำอธิบายโปรแกรม (Comment)

01006012 Computer Programming

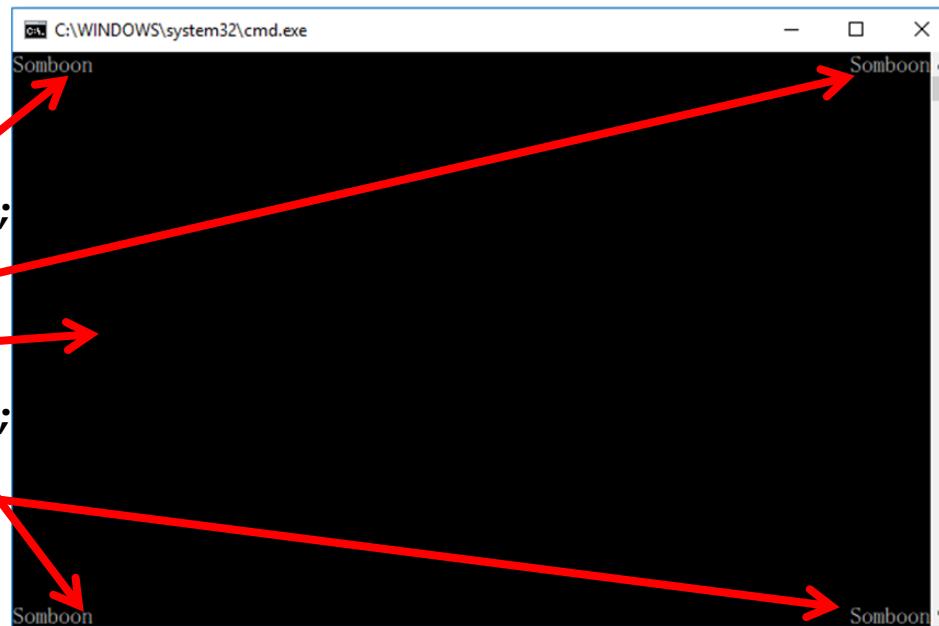
- คำอธิบายโปรแกรมเป็นส่วนที่เพิ่มในโปรแกรมเพื่อช่วยให้ผู้ที่เขียน หรือผู้ที่อ่านโปรแกรมสามารถเข้าใจกับตัวโปรแกรมได้ง่ายขึ้น
- โดยที่คำอธิบายโปรแกรมนี้จะไม่มีผลต่อการทำงานของโปรแกรม คือเวลาที่ทำการคอมไพล์ จะข้ามข้อความที่อยู่ในส่วนอธิบายโปรแกรมไป

จงเขียนโปรแกรมแสดงผลชื่อของตัวเองปรากฏที่มุมทั้ง 4 ของจอภาพ (จอภาพ 80 คอลัมน์ 25 บรรทัด)



```
#include <stdio.h>
int main()
{
    printf("%-40s","Somboon") ;
    printf("%40s","Somboon") ;
    printf("%1840s","");
    printf("%-40s","Somboon") ;
    printf("%40s","Somboon") ;

    return 0 ;
}
```



รูปแบบคำอธิบายโปรแกรม

01006012 Computer Programming

```
/* Comment Sentence 1  
Comment Sentence 2  
...  
Comment Sentence n */
```

```
/* Comment Sentence */
```

```
// Comment Sentence
```

โจทย์ : โปรแกรมแสดงผลข้อมูลส่วนตัว

01006012 Computer Programming

- จงเขียนโปรแกรมแสดงผล ชื่อ-นามสกุล อายุ
 - ตัวอย่างการแสดงผล

Name : Kmitl

Surname : Engineering

Age : 46

```
printf ("Name\t: KMITL");
printf ("\nSurname\t: Engineering");
printf ("\nAge\t: 46");
printf ("\n");
```

โจทย์ : โปรแกรมรายได้

01006012 Computer Programming

- จงเขียนโปรแกรมแสดงรายได้ต่อเดือน แล้วแสดงว่าต่อสัปดาห์ คิดเป็นเท่าไหร (ทศนิยม 2 ตำแหน่ง) ต่อวันคิดเป็นเท่าไหร (ทศนิยม 3 ตำแหน่ง)
 - ตัวอย่างการแสดงผล

Salary : 6500

Money/Week : 1625.00

Money/Day : 232.143

โจทย์ : โปรแกรมแสดงเกรด

01006012 Computer Programming

- จงเขียนโปรแกรมแสดงผล เกรดวิชา Programming, Drawing, Mechanics, Math1 พร้อมแสดงเกรดเฉลี่ย (ทศนิยม 2 ตำแหน่ง)
 - ตัวอย่างการแสดงผล

Programming : A

Drawing : B+

Mechanics : B+

Math : B

GPS : 3.50

คำสั่งที่อยู่บก

01006012 Computer Programming

1. แสดงผลของโปรแกรมต่อไปนี้

```
#include<stdio.h>
int main()
{
    /* float radius, pi, area;
    pi = 22/7; // pi = 3.14;
    printf ("Enter Radius of Circular : ");
    scanf ("%f",&radius);
    area = pi * radius * radius; */
    printf ("Programming Language",2008);
    return 0;
}
```



คำถ้ามทัยบท (ต่อ)

01006012 Computer Programming

2. จงเขียนส่วนของโปรแกรมให้แสดงผลดังต่อไปนี้

12\06\2008

King Mongkut's Institute of Technology Ladkrabang
Bangkok, Thailand.

3. จงเขียนส่วนของโปรแกรมให้แสดงผลตามเงื่อนไขต่อไปนี้

บรรทัดที่ 1 แสดง รหัสประจำตัว นศ.

บรรทัดที่ 2 แสดง ชื่อ นามสกุล

บรรทัดที่ 3 แสดง วัน/เดือน/ปี เกิด

บรรทัดที่ 8 แสดงข้อความ <http://www.ce.kmitl.ac.th>

ขิดชอบข่าวของบรรทัด

บรรทัดที่ 10 แสดงข้อความ bye bye ตรงกลางบรรทัด