



# บทที่ 3 ตัวแปรในภาษาซี คำสั่งรับข้อมูล และการคำนวณ

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

# วัตถุประสงค์



01001012 Principle of Computer Programming

- สามารถสร้างตัวแปรและใช้งานตัวแปรได้อย่างถูกต้อง
- สามารถแสดงข้อมูลความหลากหลายทางอาชีพและรับข้อมูลจากคีย์บอร์ดได้
- สามารถใช้ Operator ทางคณิตศาสตร์ได้
- เขียนโปรแกรมประมวลผลตัวอักษรได้

### 3.0 ตัวแปรในภาษาซี

01001012 Principle of Computer Programming

- การประการศตัวแปรมี 2 ลักษณะ ได้แก่ **ตัวแปรโภคอล** และ **ตัวแปรโอล寇ล** ซึ่งมีรูปแบบการประการศตัวแปรที่เหมือนกัน แต่จะมีคุณสมบัติต่างกัน โดยจะเห็นได้ชัดเจนเมื่อมีการสร้างฟังก์ชันมาใช้งาน
- เนื้อหาส่วนต้น จะใช้งานเฉพาะตัวแปรชนิดโอล寇ล โดยจะประการตัวแปรในส่วนเริ่มต้นของฟังก์ชัน
- ตัวแปรมีหน้าที่เก็บค่าต่างๆ เพื่อใช้ในการประมวลผล

### 3.1 ตัวแปรกับชนิดของข้อมูลในภาษาซี

01001012 Principle of Computer Programming

- ตัวแปรใช้เก็บข้อมูลเพื่อใช้ในการประมวลผล
- ต้องกำหนดชนิดข้อมูลที่จะเก็บให้กับตัวแปร
  - กำหนดชนิดได – เก็บไดเฉพาะชนิดนั้น
- ชนิดข้อมูลหลักที่ภาษาซีรูจัก (และใช้กำหนดชนิดตัวแปร)

**int**                    ข้อมูลชนิดจำนวนเต็ม

**float**                ข้อมูลชนิดจำนวนทศนิยม

**char**                ข้อมูลชนิดอักขระ

## 3.2 รูปแบบการประกาศตัวแปรในภาษาซี

01001012 Principle of Computer Programming

### 3.2.1 การประกาศตัวแปรแบบไม่กำหนดค่าเริ่มต้น

int            A, B, C;  
type          var1;

type          var1, var2, ... , varN;

type          กือ ชนิดของข้อมูลที่จะกำหนดให้กับตัวแปร

varX         กือ ชื่อของตัวแปรที่จะตั้ง

# ตัวอย่างการประกาศตัวแปร



01001012 Principle of Computer Programming

int number;

int a, b, c;

float real;

float point1, point2;

char choice;

char ch1,ch2;

### 3.2.2 รูปแบบการประกาศและกำหนดค่าตัวแปรในภาษาซี

01001012 Principle of Computer Programming

int A ;

type var1 = value1;

int A = 1, B = 2 ;

type var1 = value1, varN = valueN;

**type** กือ ชนิดของข้อมูลที่จะกำหนดให้กับตัวแปร

**varX** กือ ชื่อของตัวแปรที่จะตั้ง

**valueX** กือ ค่าของตัวแปรที่ต้องการกำหนดให้

# ตัวอย่างการประกาศและกำหนดค่าตัวแปร



01001012 Principle of Computer Programming

```
int number = 25;
```

```
int a = 1, b = 2, c = 3;
```

```
float real = 99.99;
```

```
float point1 = 45.2, point2 = 30;
```

```
char choice = 'a';
```

```
char ch1 = 'o', ch2 = 'z';
```

# หลักการตั้งชื่อตัวแปรในภาษาซี

01001012 Principle of Computer Programming

- ขึ้นต้นด้วยอักษร **A-Z, a-z** หรือ เครื่องหมาย \_ เท่านั้น
- ภายในตัวแปร ห้ามมีช่องว่าง
- ภายในตัวแปรประกอบด้วยอักษร **A-Z, a-z**, ตัวเลข **0-9** หรือ เครื่องหมาย \_ เท่านั้น
- การใช้อักษรตัวใหญ่และตัวเล็ก มีความแตกต่างกัน
- ห้ามใช้คำสlang เป็นชื่อตัวแปร
- ควรตั้งชื่อตัวแปรให้สัมพันธ์กับค่าที่ใช้เก็บ

int a, A ไม่ควร



# คำสংวน (Reserved word) ในภาษาซี

01001012 Principle of Computer Programming

auto	break	case	char	const
continue	default	do	double	else
enum	extern	float	for	goto
if	int	long	register	return
short	singned	sizeof	static	struct
switch	typedef	union	unsigned	void
volatile	while			

Hint : ในโปรแกรม Turbo C++ Reserved Word จะเป็นตัวหนาสีดำ

# ตัวอย่างการตั้งชื่อตัวแปร



01001012 Principle of Computer Programming

//ตั้งชื่อตัวแปรถูกต้อง

int \_money;

float salary\_ot;

char M11223\_223;

double GOTO\_data;

↑  
ดำเนินงาน แบบ  
ห้ามใช้

//ตั้งชื่อตัวแปรผิด

int \$money;

float static;

char 1M1223\_223;

double GOTO-data;

### 3.3 ตัวแปรชนิดข้อความในภาษาซี

01001012 Principle of Computer Programming

- ในภาษาซีจะ ไม่มีข้อมูลชนิดข้อความโดยเฉพาะ ซึ่งในการเขียนโปรแกรมส่วนใหญ่จำเป็นต้องมีการรับข้อมูลที่เป็นข้อความ เราสามารถใช้ตัวแปรข้อมูลชนิดอักขระหลายๆ ตัวมาใช้งานได้ในระดับหนึ่ง แต่ยังไม่สะดวกเมื่อข้อความมีความยาวมาก  
ตัวอย่าง เช่น ต้องการใช้ข้อความว่า Hello สามารถใช้ตัวแปรชนิดอักขระ 5 ตัวแทน

```
char ch1='H' ,ch2='e' ,ch3='l' ,  
      ch4='l' ,ch5='o' ;
```

# ตัวแปรชนิดข้อความในภาษาซี

01001012 Principle of Computer Programming

- ตัวแปรชนิดข้อความในภาษาซี คือ การนำอักษรมาเรียนต่อ กัน ดังนั้นสามารถสร้างตัวแปรชนิดอักขระเรียงต่อกันหลาย ๆ ตัว ให้เป็นตัวแปรชนิดແவล์ดับ ทำให้สามารถใช้เก็บข้อมูลชนิด ข้อความได้
- โดยตัวแปรชนิดข้อความในภาษาซีจะอยู่ในเครื่องหมาย Double quote ""

char kuy [20] = " Fuck You "

### 3.3.1 รูปแบบการประกาศตัวแปรชนิดข้อความ

01001012 Principle of Computer Programming

**char**      **var1 [M1]** ;

**char**      **var1 [M1] , var2 [M2]** ;

**varX**      ก็คือ ชื่อตัวแปร

**MX**      ก็คือ จำนวนของอักขระที่จะใช้เก็บบวกด้วย 1

การใช้ตัวแปรแล้วลำดับชนิดอักขระเป็นตัวแปรข้อความ ในภาษาซีกำหนดไว้ว่าตัวสุดท้ายของตัวแปรแล้วลำดับคือ \0

\0 → Null character

### 3.3.2 รูปแบบการประกาศตัวแปรและกำหนดค่า

01001012 Principle of Computer Programming

**char**      **var [M]** = "??...?";

**char**      **var [M]** = { '?' , '?' , ... , '?' } ;

**char**      **var []** = "??...? ";

**var**      กือ ชื่อตัวแปร

**M**      กือ จำนวนของอักขระที่จะใช้เก็บบวกด้วย 1

**?**      กือ อักขระที่จะกำหนดค่าให้ข้อมูล จำนวน m-1 ตัว

# ตัวอย่างการประกาศและกำหนดค่าตัวแปร

01001012 Principle of Computer Programming

```
char subject[12] = "Programming";
```

```
char nick[4] = "Com";
```

```
char nick_1[4] = { 'C', 'o', 'm', '\0' };
```

```
char name[] = "Somsak";
```

subject[12]



nick[4]



name[]



nick\_1[4]



### 3.3.3 รูปแบบการอ้างอิงอักขระในตัวแปรข้อความ

01001012 Principle of Computer Programming

#### variable [N]

**variable**      ก cioèชื่อตัวแปร

**N**                  ก cioèลำดับอักขระที่จะอ้างอิงในตัวแปรข้อความ

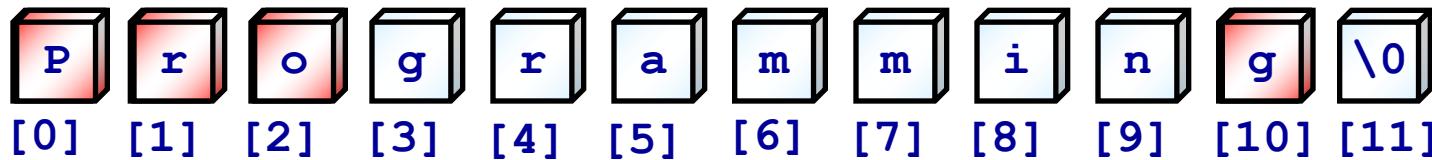
เริ่มนับอักขรตัวแรกเป็นตำแหน่งที่ 0

# ตัวอย่างการอ้างอิงอักขระในตัวเปลี่ยนความ

01001012 Principle of Computer Programming

```
char subject[12] = "Programming";
```

**subject[12]**



**subject[0]** → 'P'

**subject[1]** → 'r'

**subject[2]** → 'o'

**subject[10]** → 'g'

## 3.4 การใช้งานตัวแปรร่วมกับคำสั่ง printf

01001012 Principle of Computer Programming

- ในคำสั่ง printf มีส่วนแสดงชนิดข้อมูล ซึ่งตัวแปรที่ใช้งานก็เป็นส่วนอย่างของข้อมูลชนิดต่าง ๆ

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

- จึงสามารถใช้ตัวแปรแทนในส่วนของ data-list ได้ เช่น

```
#include<stdio.h>
int main()
{
    printf ("GPA : %.2f", 3.5);
    return 0;
}
```

```
#include<stdio.h>
int main()
{
    float g = 3.5;
    printf ("GPA : %.2f", g);
    return 0;
}
```

# ตัวอย่างการใช้งานคำสั่ง printf โดยไม่ใช้ตัวแปร



01001012 Principle of Computer Programming

ต้องการเขียนโปรแกรมแสดงคำว่า **Programming** จำนวน 5 บรรทัดที่จอกคอมพิวเตอร์

```
#include <stdio.h>
int main()
{
    printf ("%s\n","Programming");
    printf ("%s\n","Programming");
    printf ("%s\n","Programming");
    printf ("%s\n","Programming");
    printf ("%s\n","Programming");
    return 0;
}
```

หากต้องการ  
เปลี่ยนจากคำว่า  
**Programming**  
เป็น **Kmitl**  
จะทำอย่างไร



# ตัวอย่างการใช้งานคำสั่ง printf โดยใช้ตัวแปร

01001012 Principle of Computer Programming

ต้องการเขียนโปรแกรมแสดงคำว่า **Programming** จำนวน 5 บรรทัดที่จอกคอมพิวเตอร์ โดยใช้ตัวแปรข้อความ

```
#include <stdio.h>
int main()
{
    char name[20] = "Kmitl";
    printf ("%s\n", name);
    return 0;
}
```

หากต้องการ  
แสดงข้อความ  
ตามผู้ใช้กำหนด  
จะทำอย่างไร?



# Example

01001012 Principle of Computer Programming

เมื่อสั่งให้โปรแกรมด้านล่างทำงาน ผลลัพธ์ที่ได้เป็นอย่างไร

```
#include <stdio.h>
int main()
{
    int A,B,C ;
    A = 20 ;
    B = 30 ;
    C = A+B ;
    printf("%d + %d = %d",A,B,C) ;
    return 0 ;
}
```

20 + 30 = 50

A      B      C  
20 + 30 = 50 Press

```
#include <stdio.h>
#include <string.h>
int main()
{
    char Name[31] ;
    strcpy(Name,"Somchai Jaidee") ;
    printf("\nI Love %s",Name) ;
    printf("\nThe character number 0 is %c",Name[0]) ;
    printf("\nThe character number 9 is %c",Name[9]) ;
    return 0 ;
}
```

I love Som J  
The char number 0 is S  
9 is a

### Output

I Love Somchai Jaidee  
The character number 0 is S  
The character number 9 is a

### 3.5 การรับค่าด้วยคำสั่ง scanf

01001012 Principle of Computer Programming

- โปรแกรมโดยทั่วไปต้องมีการรับค่า ข้อมูลจากผู้ใช้โปรแกรม เพื่อนำมาผลลัพธ์ตามกระบวนการการทำงานของโปรแกรม หรือ ตามความต้องการของผู้ใช้งาน
- คำสั่งที่ใช้สำหรับการรับค่าในภาษาซึ่งมีหลักคำสั่ง แต่ที่สามารถใช้งานได้ครอบคลุมและนิยมใช้กัน คือ คำสั่ง scanf

### 3.5.1 รูปแบบของคำสั่ง scanf

01001012 Principle of Computer Programming

**scanf (format-string, address-list);**

format-string

คล้ายในคำสั่ง printf แต่จะมีเฉพาะส่วนแสดง  
ชนิดข้อมูล และอยู่ในเครื่องหมาย " "

address-list

คือ ตำแหน่งของตัวแปรที่ต้องการเก็บข้อมูลไว้  
(การใช้งานตำแหน่งของตัวแปรจะใช้  
เครื่องหมาย & นำหน้าชื่อตัวแปร ยกเว้นตัว  
แปรชนิดข้อความ)

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

# ตัวอย่างการใช้คำสั่ง scanf

01001012 Principle of Computer Programming

```
#include<stdio.h>
int main()
{
    float point;
    char name[20];
    printf ("Enter your name : ");
    scanf ("%s",name);
    printf ("Enter your point : ");
    scanf ("%f",&point);
    return 0;
}
```

Enter your name : Chanya  
Enter your point : 98.5  
Press

ตัวแปรข้อมูล  
ไม่ต้องมี & หน้าตัวแปร

ตัวแปรทศนิยม  
ต้องมี & หน้าตัวแปร

⇒ printf("\n");

# ตัวอย่างการใช้คำสั่ง printf และ scanf

01001012 Principle of Computer Programming

```
#include<stdio.h>
int main()
{
    char first[20],last[20];

    printf ("Enter your name and surname : ");
    scanf ("%s %s",first,last);
    printf ("Hi %s %s\nHow are you?",first,last);
    return 0;
}
```

การรับตัวแปรมากกว่า 1 ตัวในคำสั่งเดียว  
ต้องป้อนข้อมูลให้มีรูปแบบเหมือนกัน

```
Enter your name and surname : Somsak Jaidee
Hi Somsak Jaidee
How are you?
```

## 3.5.2 การใช้คำสั่ง scanf รับข้อความที่มีการเว้น (spacebar)

01001012 Principle of Computer Programming

- คำสั่ง scanf ไม่สามารถใช้รับข้อความที่มีการเว้นเพื่อเก็บในตัวแปรข้อความตัวเดียวได้ (เมื่อใช้ %s)
- เราสามารถใช้คำสั่ง scanf เพื่อให้รับข้อความที่มีการเว้นไปเก็บในตัวแปรชนิดข้อความได้ดังนี้

```
scanf ("%[^\\n]", string);
```

printf ("%./.s", "love yan");

# ตัวอย่างการใช้คำสั่ง printf และ scanf

01001012 Principle of Computer Programming

```
#include<stdio.h>
int main()
{
    char name[40];
    float gpa;
    printf ("Enter your name : ");
    scanf ("%[^\\n]",name);
    printf ("Enter your GPA : ");
    scanf ("%f", &gpa);
    printf ("Name : %s\\n",name);
    printf ("Gpa : %f",gpa);
    return 0;
}
```

(Inactive C:\TCWIN45\BIN\NONAME00.EXE)

```
Enter your name : Com eng
Enter your GPA : 3.7
Name : Com eng
Gpa : 3.700000
```



### 3.6 เครื่องหมายคำนวณทางคณิตศาสตร์



01001012 Principle of Computer Programming

เครื่องหมาย	การทำงาน	ตัวอย่าง
+	บวก	<b>ans = a + b;</b>
-	ลบ	<b>ans = a - b;</b>
*	คูณ	<b>ans = a * b;</b>
/	หาร	<b>ans = a / b;</b>
%	โมดูลัส (modulo)	<b>ans = a % b;</b>

# การกำหนดค่าให้ตัวแปร

01001012 Principle of Computer Programming

- การกำหนดค่าให้กับตัวแปรในภาษาซึ่งใช้เครื่องหมาย = โดยการทำนั่นค่าที่อยู่ทางขวามือ (จำนวน อักขระ ข้อความ ค่าจากตัวแปร หรือผลลัพธ์จากฟังก์ชัน) ให้กับตัวแปรที่อยู่ทางซ้ายมือ

```
variable = value;
```

```
num1 = 99;
```

```
n = num % 10;
```

```
point = mid + final;    ch = '9';
```

```
ans = pow(x, y);
```

### 3.6.1 การเพิ่มลดค่าตัวแปร

01001012 Principle of Computer Programming

เครื่องหมาย	การทำงาน	ตัวอย่าง	ขั้นตอนการทำงาน	
++	เพิ่มค่าทีละ 1 (Increment)	$x++ ; \quad ++x ;$	เพิ่มค่า x ขึ้น 1	
		$y = ++x ;$	เพิ่มค่า x ขึ้น 1	กำหนดค่าให้ y
		$y = x++ ;$	กำหนดค่าให้ y	เพิ่มค่า x ขึ้น 1
--	ลดค่าทีละ 1 (Decrement)	$x-- ; \quad --x ;$	ลดค่า x ลง 1	
		$y = --x ;$	ลดค่า x ลง 1	กำหนดค่าให้ y
		$y = x-- ;$	กำหนดค่าให้ y	ลดค่า x ลง 1

## 3.6.2 เครื่องหมายแบบลดรูป

01001012 Principle of Computer Programming

เครื่องหมาย	ตัวอย่างการใช้งาน	ตัวอย่างรูปแบบเต็ม
$+=$	$y += x;$	$y = y + x;$
$-=$	$y -= x;$	$y = y - x;$
$*=$	$y *= x;$	$y = y * x;$
$/=$	$y /= x;$	$y = y / x;$
$\%=$	$y \%= x;$	$y = y \% x;$

### 3.6.3 ตัวดำเนินการ & ตัวถูกดำเนินการ

01001012 Principle of Computer Programming

Type 1	Operator	Type 2	Result	Exam	
<b>int</b>	<b>+,-,*,/</b>	<b>int</b>	<b>int</b>	$3*3$	9
				$19/2$	9
<b>int</b>	<b>+,-,*,/</b>	<b>float</b>	<b>float</b>	$3*3.0$	9.000000
				$19/2.0$	9.500000
<b>float</b>	<b>+,-,*,/</b>	<b>int</b>	<b>float</b>	$3.0*3$	9.000000
				$19.0/2$	9.500000
<b>float</b>	<b>+,-,*,/</b>	<b>float</b>	<b>float</b>	$3.0*3.0$	9.000000
				$19.0/2.0$	9.500000
<b>int</b>	<b>%</b>	<b>int</b>	<b>int</b>	$7\%4$	3
				$8\%4$	0

### 3.6.4 นิพจน์ในภาษาซี

01001012 Principle of Computer Programming

ลำดับความสำคัญ	เครื่องหมาย
1	()
2	!, ++, --, (type)
3	* , / , %
4	+ , -
5	< , <= , > , >=
6	== , !=
7	&&
8	
9	*= , /= , %= , += , -=

# ตัวอย่าง แสดงลำดับความสำคัญสัญลักษณ์ทางคณิตศาสตร์



01001012 Principle of Computer Programming

```
#include <stdio.h>
int main() //ผลลัพธ์
{
    int a = 10;           60 - 10/7      59
    printf("%d", 10*2*3-10/7);
    printf("%d", 10*2*(3-10)/7); -19
    printf("%d", 10*2>2+3);   1
    printf("%d", 10*2<2+3);   0
    return 0;
}
```

### 3.7 คำสั่งแสดงผล และรับข้อมูลอื่นๆ

01001012 Principle of Computer Programming

**putchar(ch) ;**

พิมพ์ 1 ตัว

**puts(str) ;**

พิมพ์หลายตัว

**ch = getchar() ;**

**ch = getch() ;**

**gets(str) ;**

# รูปแบบและการใช้คำสั่ง getchar

01001012 Principle of Computer Programming

- เป็นคำสั่งที่ใช้รับข้อมูลชนิดอักขระจากผู้ใช้งานเพียงตัวเดียว โดยเมื่อป้อนข้อมูลแล้วต้องกด Enter
- คำสั่ง getchar มีรูปแบบการใช้งานคำสั่งดังนี้

```
ch = getchar();
```

**ch** ก็อตัวแปรชนิดข้อความที่ต้องการเก็บข้อมูลไว้

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

# รูปแบบและการใช้คำสั่ง getch

01001012 Principle of Computer Programming

- เป็นคำสั่งที่ใช้รับข้อมูลชนิดอักขระจากผู้ใช้งานเพียงตัวเดียว โดยเมื่อป้อนข้อมูลแล้ว โปรแกรมจะทำงานคำสั่งต่อไปทันที และจะ ไม่แสดงอักขระที่พิมพ์ไป
- คำสั่ง getch มีรูปแบบการใช้งานคำสั่งดังนี้

```
ch = getch();
```

**ch** กือตัวแปรชนิดข้อความที่ต้องการเก็บข้อมูลไว้

**หมายเหตุ** คำสั่ง getch ต้องเรียกใช้ Preprocessor Directive `#include<conio.h>`

# รูปแบบและการใช้คำสั่ง gets

01001012 Principle of Computer Programming

- เป็นคำสั่งที่ใช้รับข้อมูลชนิดข้อความจากผู้ใช้งาน โดยสามารถใส่ข้อมูลที่มีการเว้นช่องว่างภายในได้
- คำสั่ง gets มีรูปแบบการใช้งานคำสั่งดังนี้

**gets (str) ;**

**str**      กือตัวแปรชนิดข้อความที่ต้องการแสดงผล

หมายเหตุ คำสั่ง gets ต้องเรียกใช้ Preprocessor Directive **#include<stdio.h>**

# ตัวอย่างคำสั่งแสดงผล และรับข้อมูล

01001012 Principle of Computer Programming

```
char ch1,ch2;  
printf ("Enter Character 1 : ");  
ch1 = getchar();  
printf ("Enter Character 2 : ");  
ch2 = getch();  
puts ("\n**** Output ****");  
printf ("Char 1 = %c\nChar 2 = %c", ch1,ch2);
```

```
Enter Character 1 : J  
Enter Character 2 :  
**** Output ****  
Char 1 = J  
Char 2 = O
```

พิมพ์ O แต่ไม่แสดงออกมานะ



# Variable

01001012 Principle of Computer Programming

ป้อนชื่อ(ข้อความ)และ เงินเดือน(ทศนิยม) แสดง 5 ตัวอักษรแรกของชื่อและ 25 เปอร์เซ็นต์ของเงินเดือน  
ทศนิยม 2 ตำแหน่ง

```
#include <stdio.h>
int main()
{
    char Name[31];
    float Salary;

    scanf("%s %f", Name, &Salary);
    printf("\nFirst 5 chars of %s = %.5s", Name, Name);
    printf("\n25 percent of %.2f = %.2f", Salary, Salary/4.0);

    return 0;
}
```

Jakrapong 10000  
 First 5 char of Jakrapong = Jakra  
 25 percent of 10000.00 =

A diagram illustrating variable binding. It shows the variables `Name` and `Salary` from the `scanf` statement being passed to the `printf` statements. The `Name` variable is used twice in the first `printf` statement, and the `Salary` variable is used once in the second `printf` statement. The `Salary` variable is also annotated with a crossed-out `f`.

### 3.8 คำถ้ามทัยบท



01001012 Principle of Computer Programming

1. ชนิดข้อมูลแบบข้อความเหมือนหรือแตกต่างจากชนิดข้อมูล

แบบอักขระอย่างไร

Name: Pao

2. จากตัวอย่างโปรแกรมต่อไปนี้

Age : 80

Your name is Pao

Your age is 80

```
#include <stdio.h>
int main() {
    char name[20];
    int age;
    printf("Enter your name: "); scanf("%s", name);
    printf("Enter your age "); scanf("%d", age);
    printf("Your name is : %s\n Your age is : %d", name, age);
    return 0; }
```

เมื่อ Run แล้วจะได้ผลอย่างไร

# คำถ้ามท้ายบท (ต่อ)

01001012 Principle of Computer Programming

## 3. จากโปรแกรมต่อไปนี้

```
#include <stdio.h>
int main()
{
    printf("%d", (40/4*3+4*5/2));
    return 0;
}
```

40

30

10

เมื่อ Run แล้วจะได้ผลอย่างไร