

บทที่หก

ตัวแปรชุด (Array)

ตัวแปรชุด หมายถึง การตั้งชื่อตัวแปรในหน่วยความจำที่ใช้ชื่อเพียงชื่อเดียว แต่มีตัวเลขแสดงตำแหน่งกำกับไว้เพื่อเป็นการบอกว่าเป็นตัวแปรตัวที่เท่าไรในชุดนั้น ในภาษา C ตัวแปรชุดสามารถมีหลายมิติที่นิยมใช้มี

6.1 ตัวแปรชุด 1 มิติ (One Dimension)

หมายถึงตัวแปรชุดที่มีตัวเลขแสดงตำแหน่งเพียงตัวเดียว

รูปแบบ

```
type array – name [n];
```

type หมายถึง ชนิดของตัวแปร

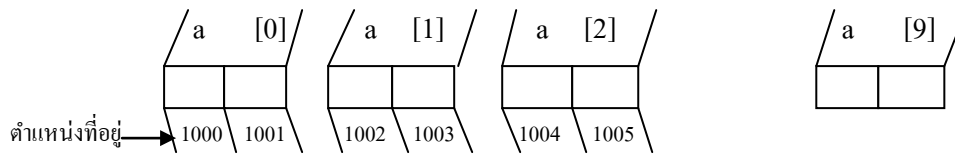
array – name หมายถึง ชื่อตัวแปรที่ต้องการประกาศเป็นตัวแปรชุด

n หมายถึง ตัวเลขที่แสดงตำแหน่งของข้อมูลในชุดนั้น ซึ่งจะต้องเป็นเลขจำนวนเต็มเสมอ

ตัวอย่างที่ 1

```
int a[10];
```

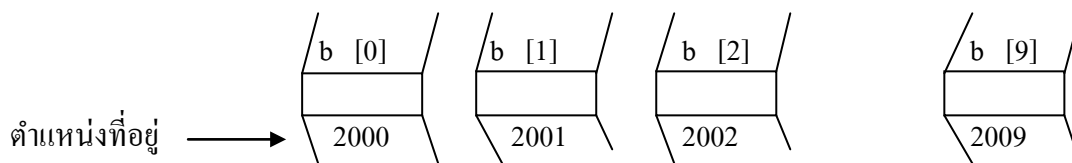
หมายถึง คอมไพเลอร์จะจองเนื้อที่ในหน่วยความจำให้ สำหรับตัวแปร a เป็นตัวแปรชุดจำนวนเต็มชนิด 1 มิติ ขนาด 10 สมาชิก (element) โดยในหน่วยความจำจะเตรียมที่ไว้ให้ 2 ไบต์ สำหรับ 1 ชื่อตัวแปร ดังนี้



ตัวอย่างที่ 2

```
char b[10];
```

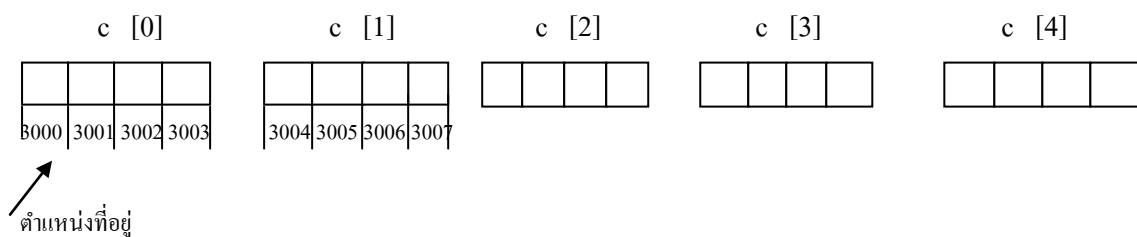
หมายความว่า คอมพิวเตอร์จองพื้นที่สำหรับตัวแปร b ให้เป็นตัวแปรชนิด character 1 มิติ ขนาด 10 สมาชิก โดยหน่วยความจำจะเตรียมที่ให้ 1 ไบต์ สำหรับ 1 ชื่อตัวแปรดังนี้



ตัวอย่างที่ 3

```
float c[5];
```

หมายความว่า คอมพิวเตอร์จองพื้นที่สำหรับตัวแปร C ให้เป็นสมาชิกตัวแปรทุกชนิดทศนิยม 1 มิติ ขนาด 5 สมาชิก โดยหน่วยความจำจะเตรียมที่ให้ 4 ไบต์ สำหรับ 1 ชื่อตัวแปรดังนี้



6.2 ตัวแปรชนิด 2 มิติ (Two Dimension)

หมายถึง ตัวแปรชุดที่มีการจัดข้อมูลเป็นแถวเป็นคอลัมน์ เช่น ข้อมูลแบบ matrix ก็จะมีตัวเลขที่แสดงตำแหน่ง 2 ตัว โดยตัวแรกจะเป็นตัวเลขที่แสดงตำแหน่งแถว ส่วนตัวที่สองจะเป็นตัวเลขที่แสดงตำแหน่งคอลัมน์

รูปแบบ

```
type array – name [n] [m];
```

type หมายถึง ชนิดของตัวแปรที่ต้องการประกาศเป็นตัวแปรชุด

array – name หมายถึง ชื่อตัวแปรที่ต้องการประกาศเป็นตัวแปรชุด

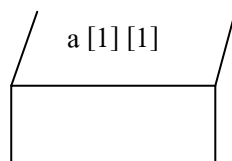
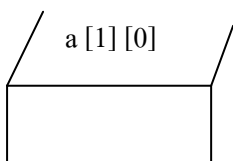
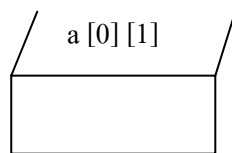
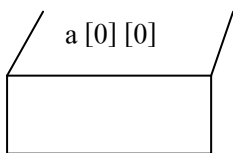
n หมายถึง ตัวเลขที่แสดงตำแหน่งของแถว

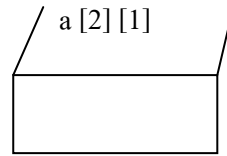
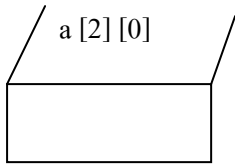
m หมายถึง ตัวเลขที่แสดงตำแหน่งของคอลัมน์

ตัวอย่างที่ 4

```
int a[3] [2];
```

หมายความว่า คอมพิวเตอร์จะเตรียมที่ในหน่วยความจำ 6 ที่ สำหรับตัวแปร a ซึ่งเป็นตัวแปรชุด ชุดจำนวนเต็ม 2 มิติโดยมีลักษณะในหน่วยความจำ ดังนี้





6.3 การกำหนดค่าในตัวแปรชุด (Array Initialization)

ภาษา C จะสามารถกำหนดค่าให้กับตัวแปรชุดได้โดยใช้รูปแบบที่กำหนดไว้สำหรับตัวแปรแต่ละชนิด ดังนี้

6.3.1 การกำหนดค่าให้กับตัวแปรชุดที่มีค่าเป็นตัวเลข

รูปแบบ

```
type array - name [n] = {value - list};
```

value - list หมายถึง ค่าที่ต้องการกำหนดให้ตัวแปรชุด ถ้ามีหลาย ๆ ค่าแต่ละค่าจะคั่นด้วยเครื่องหมายคอมม่าและจะต้องเขียนอยู่ภายในเครื่องหมาย { }

ตัวอย่างที่ 5

```
int i [10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
```

หมายความว่า มีการกำหนดค่าให้ตัวแปร i[0] มีค่าเป็น 1 i[1] มีค่าเป็น 2 ไปเรื่อยๆจนถึง i[9] มีค่า 10

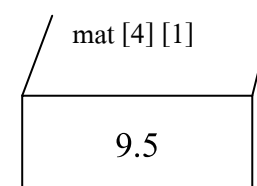
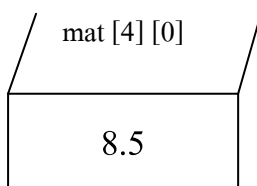
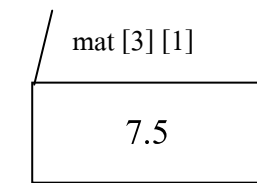
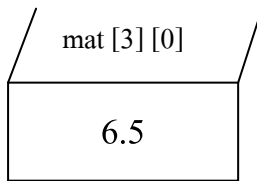
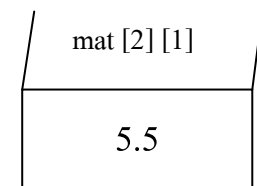
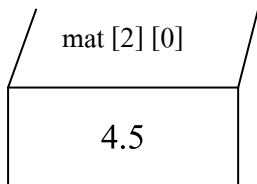
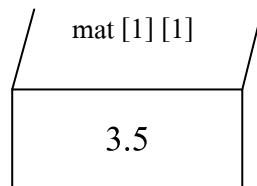
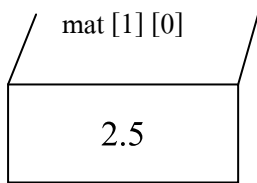
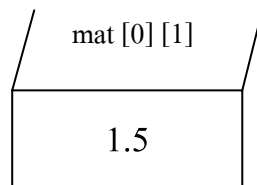
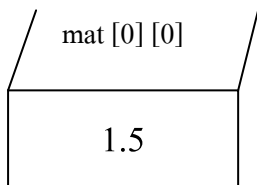
ตัวอย่างที่ 6

```
float mat [5] [2] = {
```

```
1.5,      1.5,
```

```
2.5,    3.5,  
  
4.5,    5.5,  
  
6.5     7.5,  
  
8.5,    9.5,  
  
};
```

หมายความว่า มีการกำหนดค่าให้ตัวแปร mat ไว้ในหน่วยความจำ ดังนี้



6.3.2 การกำหนดค่าให้กับตัวแปรชุด ชนิด Character

รูปแบบ

```
char array - name [n] = "string";
```

string หมายถึง ข้อความที่ต้องการกำหนดให้กับตัวแปรชุดชนิด character ซึ่งต้องเขียน

ภายในเครื่องหมายคำพูด "....."

ตัวอย่างที่ 7

```
char ch [9] = "SEWESDEE"
```

หมายความว่า มีการกำหนดค่าให้กับตัวแปรชุด ch[0] เก็บค่า S ch[1] เก็บค่า A จนถึง ch [7] เก็บค่า E ส่วนตัวแปร ch[8] เครื่องจะเก็บค่า \0 ให้เองโดยอัตโนมัติ เพื่อเป็นการแสดงการสิ้นสุดของข้อความ (string) หรือสามารถกำหนดค่าให้กับตัวแปรชุดที่ละตัวได้ดังนี้

```
char ch [9] = {'S', 'A', 'W', 'A', 'S', 'D', 'E', 'E', '\0'};
```

ในการกำหนดค่าให้กับตัวแปรชุดสามารถที่กำหนดค่าให้โดยไม่ต้องบอกขนาดของตัวแปรชุดได้ คอมไพเลอร์จะเตรียมที่ของตัวแปร ชุดให้เท่ากับจำนวนของข้อมูลจริง

ตัวอย่างที่ 8

```
char str [ ] = "ENTER - DATA";
```

หมายความว่า คอมไพเลอร์จะเตรียมที่ให้ตัวแปรชุด str ทั้งหมด 11 ที่โดย str[0] จะเก็บ E str[1] จะเก็บ N.....str[9] จะเก็บ A ส่วนstr[10] จะเก็บ \0

ในกรณีที่เป็นตัวแปรชนิด 2 มิติ สามารถที่จะไม่กำหนดขนาดของแถวได้ แต่จะไม่กำหนดขนาดของคอลัมน์ หรือไม่กำหนดของขนาดแถวและคอลัมน์เลยไม่ได้

ตัวอย่างที่ 9

```
int ary [ ] [2] = {  
  
    1, 2  
  
    3, 4  
  
    5, 6  
  
};
```

6.4 การนำข้อมูลเข้าไปเก็บและแสดงผลของข้อมูลที่อยู่ในตัวแปรชุด

การนำข้อมูลเข้ามาเก็บและแสดงผลข้อมูลที่อยู่ในตัวแปรชุดจะใช้คำสั่ง for มาช่วยในการทำงาน เพื่อให้เห็นการทำงานชัดเจนจะแบ่งเป็น

6.4.1 การนำข้อมูลเข้าไปเก็บและแสดงผลของข้อมูลในตัวแปรชุด 1 มิติ

ตัวอย่างที่ 10 แสดงการรับค่ามาจากแป้นพิมพ์ 5 ค่าเพื่อเก็บไว้ในตัวแปรชุด in แล้วนำมาคำนวณหาค่าเฉลี่ย

```
main ()  
  
{  
  
    int in [5], i, avg;
```

```
for (i = 0; i < 5; i++) {  
  
    printf ("enter number %d :", i);  
  
    scanf ("%d", &in[i]);  
  
}  
  
avg = 0;  
  
/*now, add up the number */  
  
printf ("The average is %d \n", avg / 5);  
  
}
```

ผลลัพธ์

enter number 0 : 2

enter number 1 : 3

enter number 2 : 9

enter number 3 : 5

enter number 4 : 6

The average is 5

ตัวอย่างที่ 11 ให้เขียนโปรแกรมรับข้อความ CHULALONGKHORN เข้ามาแล้วให้นับว่ามีกี่ตัวอักษร

พร้อมให้พิมพ์ข้อความนี้ย้อนกลับจากตัวสุดท้ายขึ้นมา


```
*/Program print backward by reducing number of array */
```

```
main ( )
```

```
{
```

```
char c, word [20];
```

```
int len = 0, i;
```

```
printf ("Enter a word:");
```

```
gets (word);
```

```
for (;;) {
```

```
    if (word [len] != '\0');
```

```
        len = ++;
```

```
    else
```

```
        break;
```

```
}
```

```
printf ("The length of %s = %d \n", &word, len);
```

```
for (i = len; i >= 0; -- i)
```

```
{
```

```
c = word [i];

printf ("%c", c);

}

printf ("\n");

}
```

Enter a word : CHULALONGKORN

The length of CHULALONGKORN = 13

NROKGNOLALUHC

ตัวอย่างที่ 12 จงเขียนโปรแกรม เพื่อเรียงลำดับข้อมูลที่เป็นตัวเลขจำนวนเต็ม 5 ค่า จากน้อยไปหา
มาก

```
main ()

{

int i, n, s ;

float x [100], temp ;

printf ("Enter number of sort :");

scanf ("%d", &i);

for (n = 0; n < i ; n ++ ) {
```

```
printf("Enter a value %d:", n + 1);

scanf("%f", &x [n]);

}

printf("THE ORIGINAL LIST \n");

for ( n= 0; n < i ; n ++ )

printf("%10.2 \n", x [n]);

/* sort value */

do {

s = 0;

for (n = 0; n < i - 1; n ++ ) {

if (x [n] <= x [n + 1])

;

else {

temp = x [n];

x [n] = x [n+1];

x [n+1] = temp;
```

```
        s = 1; }  
  
    }  
  
}  
  
while (s!=0);  
  
printf("SORT LIST \n");  
  
for (n = 0 < i ; n ++)  
  
    printf("%10.2f\n", x [n]);  
  
}
```

Enter number of sort : 5

Enter a value 1 : 12

Enter a value 2 : 5

Enter a value 3 : 35

Enter a value 4 : 8

Enter a value 5 : 9

THE ORIGINAL LIST

12.0

5.00

35.00

8.00

9.00

SORT LIST

5.00

8.00

9.00

12.00

35.00

6.4.2 การนำข้อมูลเข้ามาเก็บและแสดงผลของข้อมูลในตัวแปรชนิด 2 มิติ

ในกรณีที่เป็นตัวแปรชนิด 2 มิติ ก็จะใช้ for ซ้อนกันเป็น 2 วงจร เพื่อช่วยในการ

รับข้อมูลและแสดงผลข้อมูล

ตัวอย่างที่ 13 จงเขียนโปรแกรมนำข้อมูลของ Matrix ขนาด 3 x 4 ดังนี้

2 4 6 1

8 9 10 7

12 14 16 3

มาเก็บไว้ในตัวแปรที่ชื่อว่า m แล้วพิมพ์ matrix m นี้ออกมาทางจอภาพ

```
main ()  
  
{  
  
int a[3][4]  
  
    for (r = 0; r <= 2; r++)  
  
        for (c = 0; c <= 3; c++)  
  
            scanf ("%d", &a [r][c]);  
  
for (r = 0; r <= 2; r++)  
  
    for (c = 0; c <= 3; r++)  
  
{  
  
printf ("\n");  
  
printf ("%4d", a [r][c]);  
  
    }  
  
}
```

ตัวอย่างที่ 14 จงเขียนโปรแกรมหาค่าผลบวกในแนวแถวและแนวคอลัมน์ของ matrix ขนาด 4 x 5

```
/* Program summation of row and column of matrix */
```

```

main ()

{

int a [4][5], r, c, sum – r[4], sum – c[5], total = 0;

printf ("Enter 4 rows of number each row has 5 values \n");

for ( r = 0; r <= 3; r++)

{

sum – r[r] = 0;

printf ("Row %d =", r);

for (c = 0; c <= 4; c ++)

{

scanf ("%d", &a[r] [c]);

sum – r[r] = sum – r[r] + a[r][c];

}

}

printf ("          sum row \n");

for (r = 0; r <= 3; r++)

```

```
{  
  
printf ("      ")  
  
for (c = 0; c <= 4; c ++)  
  
    printf ("%4d", a[r][c]);  
  
    printf ("%9d", sum - r[r]);  
  
    printf ("\n");  
  
}  
  
printf ("\n");  
  
printf ("sum column");  
  
for (c = 0; c <= 4; c ++)  
  
{  
  
sum - c[c] = 0;  
  
for (r = 0; r <= 3; r ++);  
  
    {  
  
        sum - c[c] = sum - c[c] + a[r][c];  
  
    }  
  
}
```



```

printf ("%4d", sum - c[c]);

total = total + sum - c[c];

}

printf ("%9d \n", total);

}

```

Enter 4 rows of number each row has 5 values

Row 0 = 0 1 2 3 4

Row 1 = 1 2 3 4 5

Row 2 = 2 3 4 5 6

Row 3 = 3 4 5 6 7

		sum	row
	0 1 2 3 4		10
	1 2 3 4 5		15
	2 3 4 5 6		20
	3 4 5 6 7		25
sum	column	6 10 14 18 22	70

แบบฝึกหัด

1. จงหาที่ผิดของโปรแกรมต่อไปนี้

```
char array [100];

main ()

{

    int n;

    for (n = 0; n <= 100; n ++ )

        array[n] = 'a';

}
```

2. จงเขียนโปรแกรมรับข้อมูลที่เป็นคำตอบของข้อสอบของนักเรียน ประมาณ 50 ข้อ แล้วให้ตรวจสอบว่าในข้อใดแต่ละข้อนั้นมีนักเรียนตอบคำตอบเป็นข้อ 1 กี่คน ข้อ 2 กี่คน ข้อ 3 กี่คน โดยให้แสดงผลเป็นเลขข้อ และจำนวนคนที่ตอบในแต่ละข้อออกทางจอภาพ
3. จงเขียนโปรแกรมรับข้อมูลที่เป็น Matrix ขนาด 10x10 แล้วให้คำนวณหาค่า
 1. ผลบวกของสมาชิกทั้งหมด
 2. ผลบวกในแนวแถว
 3. ผลบวกในแนวคอลัมน์
 4. ผลบวกในแนวเส้นทแยงมุมทั้ง 2 เส้น