

แถวลำดับ (Array)

นายสายัณห์ อินทมาตย์

ครู คศ.2

โรงเรียนปทุมเทพวิทยาคาร

วิทยาการ : โครงการโอลิมปิกวิชาการ

ค่าย 1 วิชาคอมพิวเตอร์

แถวลำดับ (Array)

1. อาร์เรย์ 1 มิติ
2. การเก็บตัวอักษรหรือสายอักขระ 1 มิติ
3. อาร์เรย์ 2 มิติ
4. การเก็บตัวอักษรหรือสายอักขระ 2 มิติ
5. ฟังก์ชัน การทำงานกับสายอักขระ
6. การค้นหาตัวอักษรหรือค่าใน Array 1 มิติและ 2 มิติ
(String matching)

ข้อมูลชนิดอาร์เรย์

-12	23	45	65	12	27	86
-----	----	----	----	----	----	----



ตัวแปรอาร์เรย์เก็บจำนวนเต็ม 7 จำนวน

12.8	85.21	32.1	23.9	43.5
------	-------	------	------	------



ตัวแปรอาร์เรย์เก็บทศนิยม 5 จำนวน

แถวลำดับ (Array)

ตัวแปรประเภทอาร์เรย์ เป็นตัวแปรที่สามารถเก็บข้อมูลหลาย ๆ ค่าไว้ในตัวแปรชื่อเดียวกันได้ โดยระบบจะใช้พื้นที่หน่วยความจำต่อเนื่องกัน เพื่อเก็บข้อมูลชนิดเดียวกันหลายจำนวน

การประกาศตัวแปรอาร์เรย์

จำนวนสมาชิกของตัวแปรอาร์เรย์

ชนิดข้อมูล ชื่อตัวแปรอาร์เรย์[n];

```
int A[10];  
float B[5];
```

ประเภทข้อมูลในตัวแปรอาร์เรย์

ตัวอย่าง

- `int n[10];`

ประกาศตัวแปรอาร์เรย์ชื่อ `n` มีขนาด 10 หน่วย แต่ละหน่วยเก็บเลขจำนวนเต็ม

- `char a[20];`

ประกาศตัวแปรอาร์เรย์ชื่อ `a` มีขนาด 20 หน่วย แต่ละหน่วยเก็บตัวอักษร

- `float g[5];`

ประกาศตัวแปรอาร์เรย์ชื่อ `g` มีขนาด 5 หน่วย แต่ละหน่วยเก็บเลขทศนิยม

ถ้าหากประกาศตัวแปรเป็นสตริง ตัวแปรนั้นก็คืออาร์เรย์ของ `char` นั่นเองครับ

การประกาศตัวแปรแถวลำดับ

```
int num[10]
```

ประกาศตัวแปรแถวลำดับชื่อ num ประกอบด้วยสมาชิก 10 หน่วย แต่ละหน่วยเก็บเลขจำนวนเต็ม โดยลักษณะของตัวแปรแถวลำดับในหน่วยความจำจะเรียงต่อกันดังนี้

หน่วยของแถวลำดับ

หน่วยความจำ

num[0]

num[1]

num[2]

num[3]

num[4]

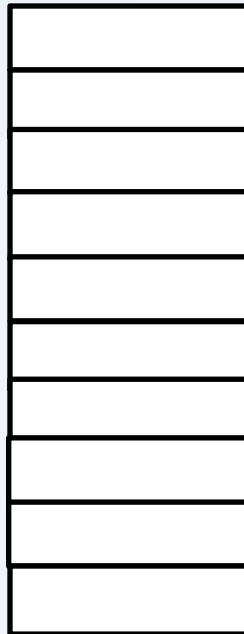
num[5]

num[6]

num[7]

num[8]

num[9]



การอ้างอิงสมาชิกในอาร์เรย์

สามารถอ้างอิงสมาชิกแต่ละหน่วยของอาร์เรย์ดังนี้

ชื่อตัวแปรอาร์เรย์[ดัชนีกำกับ]

เลขจำนวนเต็มตั้งแต่ 0 ถึง $n-1$

ตัวอย่าง

```
int n[5];
```

```
n[4] = 25;
```

สามารถอ้างอิงสมาชิกแต่ละหน่วยของอาร์เรย์ `n[]` โดยใช้ `n[0], n[1], n[2], n[3]`

และ `n[4]`

ตัวอย่าง

ถ้าหากมีข้อมูลกลุ่มหนึ่งเป็นคะแนนของนักศึกษา 8 คน สามารถเก็บได้ดังนี้

หมายเลข	X[0]	X[1]	X[2]	X[3]	X[4]	X[5]	X[6]	X[7]
คะแนน	18	20	35	84	21	45	65	74

ถ้าหากมีการอ้างอิงอาเรย์อาจเป็นดังต่อไปนี้

X[2]	อ้างอิงเซลล์ที่ 2 มีค่าเท่ากับ 35
X[2] + X[3]	นำเซลล์ที่ 2 บวกกับเซลล์ที่ 3 จะได้ $35 + 84$ เท่ากับ 119
X[1+3]	อ้างอิงเซลล์ที่ 4 มีค่าเท่ากับ 21
X[5] + 1	นำเซลล์ที่ 5 มาบวกด้วย 1 จะได้เท่ากับ 46

ขนาดหน่วยความจำของอาร์เรย์

ขึ้นอยู่กับประเภทของข้อมูลและจำนวนสมาชิกที่จองไว้

ตัวอย่าง

```
int x[20];
```

x[0] x[1] x[2] x[3] x[17] x[18] x[19]



← 40 Byte →

จองเนื้อที่ช่องละ 2 ไบต์ จำนวน 20 ช่อง รวมเป็น 40 ไบต์

```
char name[20];
```



ใช้พื้นที่ 20 ไบต์

ตัวอย่างที่ 1

โปรแกรมรับข้อมูล 10 ค่า แล้วหาผลรวมของข้อมูลเหล่านั้น

```
#include <stdio.h>

main()
{
    int num[10],sum,i;

    for(i = 0; i < 10; i++)
        scanf("%d",&num[i]);

    sum = 0;

    for(i=0;i<10;i++)
        sum = sum + num[i];

    printf("sum is %d\n",sum);
}
```

ตัวอย่างที่ 2

โปรแกรมเพื่ออ่านข้อมูลชนิดจำนวนจริง 100 จำนวน หาค่าเฉลี่ย
และจำนวนข้อมูลที่มีค่าสูงกว่าค่าเฉลี่ย

```
#include <stdio.h>
```

```
void main()
```

```
{
```

```
    int n, count,k;
```

```
    float avg, d, sum =0;
```

```
    float large;
```

```
    float list[100];
```

```
    scanf("%d",&n);
```

```
    for(count = 0; count<n; count++)
```

```
{
```

```
scanf("%f",&list[count]);  
    sum =sum +list[count];  
}  
avg =sum /n;  
printf("The average is %f\n",avg);  
k = 0;  
for(count = 0; count<n; count++)  
if(list[count] > avg)  
    k= k+1;
```

```
printf("number above average is %d\n",k);
```

```
large = list[0];
```

```
for(count = 1; count<n; count++);
```

```
    if(list[count]>large)
```

```
        large = list[count];
```

```
    printf ("The largest number is %f\n",large);
```

```
}
```

ตัวอย่างที่ 3

โปรแกรมเพื่ออ่านข้อมูลชนิดจำนวนจริง 25 จำนวน แล้วพิมพ์ตัวเลขเหล่านั้นในลำดับ
ตรงกันข้ามกับที่อ่านเข้ามา

```
#include <stdio.h>

void main()
{
    int i;

    float a[25];

    for(i = 0; i<25; i++)
        scanf("%f", &a[i]);

    for(i = 24; i>= 0; i--)
        printf("%f\n",a[i]);
}
```

ตัวอย่างที่ 4

โปรแกรมเพื่ออ่านข้อมูลเข้าเป็นคะแนนสอบ ซึ่งเป็นจำนวนจริงแล้วเก็บในตัวแปรแถวลำดับ score[] อ่านจนกว่าข้อมูลเข้าจะมีค่าน้อยกว่าศูนย์ แล้วพิมพ์จำนวนนักเรียนที่เข้าสอบทั้งหมด ซึ่งมีไม่เกิน 500 คน

```
#include <stdio.h>
```

```
void main()
```

```
{
```

```
    int i=0;
```

```
    float score[500], temp;
```

```
    scanf("%f", &temp);
```

```
    while (temp>=0)
```

```
    {
```

```
        score[i] = temp;
```

```
        i = i + 1;
```

```
        scanf("%f",&temp);
```

```
    }
```

```
    printf("Number of student taking the exam is %d",i);
```

```
}
```

ตัวแปรอาร์เรย์เพื่อเก็บข้อมูลชนิดอักขระ

- ในการเก็บข้อมูลที่เป็นสายอักขระ (string) ต้องเก็บแต่ละอักขระไว้ในตัวแปรอาร์เรย์ โดยที่อักขระหนึ่งตัว คือ สมาชิกหน่วยหนึ่งของอาร์เรย์ซึ่งใช้เนื้อที่ 1 ไบต์ และจบสายอักขระด้วย '\0'

ตัวอย่าง

การเก็บสายอักขระในหน่วยความจำ

สายอักขระ “string” จะเก็บในตัวแปรอาร์เรย์ดังนี้

หน่วยของแวลลำดับ

0
1
2
3
4
5
6

หน่วยความจำ

s
t
r
i
n
g
\0

การประกาศตัวแปร

char ชื่อตัวแปรลำดับ[n];

n คือ จำนวนเต็มบวก ที่มีค่ามากกว่าหรือเท่ากับ จำนวนอักขระบวกด้วยหนึ่ง (สำหรับ'\0') การอ่านหรือแสดงสายอักขระให้ใช้ %s และใช้ชื่อตัวแปรลำดับที่เก็บสายอักขระนี้

ตัวอย่าง

```
#include <stdio.h>
```

```
void main()
```

```
{
```

```
    char message[14];
```

```
    message[0] = 'I';
```

```
    message[1] = ' ';
```

```
    message[2] = 'a';
```

```
    message[3] = 'm';
```

```
    message[4] = ' ';
```

```
    message[5] = 'f';
```

```
    message[6] = 'i';
```

```
    message[7] = 'n';
```

```
    message[8] = 'e';
```

```
    message[9] = '.';
```

```
    message[10] = '\0';
```

```
    print("%s",message);
```

```
}
```

ผลลัพธ์

I am fine.

ตัวอย่าง

```
#include <stdio.h>
```

```
void main()
```

```
{
```

```
    char string[6];
```

```
    scanf("%7s",string);
```

```
    printf("%s",string);
```

```
}
```

ถ้าข้อมูลเข้าคือ Computer Science

ผลลัพธ์ที่ได้คือ

Compute

แบบฝึกหัด

1. เติมโปรแกรมด้านล่างเพื่อให้พิมพ์ผลลัพธ์ดังนี้

1

4

3

5

```
#include <stdio>
main()
{
    int a[5] = {1,2,3,4,5};
    printf("%d\n",a[...]);
    printf("%d\n",a[...]);
    printf("%d\n",a[...]);
    printf("%d\n",a[...]);
}
```

แบบฝึกหัด

2. เติม Source code โปรแกรมเพื่อให้พิมพ์ผลลัพธ์ดังนี้

1 2 3 4 5

2 3 4 5 6

3 4 5 6 7

4 5 6 7 8

5 6 7 8 9

แบบฝึกหัด

```
#include <stdio>
```

```
main()
```

```
{
```

```
    int a[5][5];
```

```
    for(int i=0; i<5; i++) {
```

```
        for(int j=0; j<5; j++) {
```

```
            a[i][j] = ..... ;
```

```
        }
```

```
    }
```

```
for(int i=0; i<5; i++) {
```

```
    for(int j=0; j<5; j++) {
```

```
        printf("%d ",a[i][j]);
```

```
    }
```

```
    printf("\n");
```

```
}
```

```
}
```

แบบฝึกหัดท้ายบท

1. ถ้าหากมีข้อมูลอยู่ 10 ค่า จงเขียนโปรแกรมเรียงข้อมูลจากค่าน้อยที่สุดไปหาค่ามากที่สุด
2. จำนวนเฉพาะคือจำนวนเต็มบวก ซึ่งหารด้วยจำนวนอื่นไม่ลงตัว ยกเว้น 1 กับตัวมันเอง ตัวอย่างเช่น 7 เป็นจำนวนเฉพาะ แต่ 6 ไม่ใช่
จงสร้างตารางจำนวนเฉพาะ n จำนวนแรก
3. จงเขียนโปรแกรมแปลงเลขฐานสิบเป็นเลขฐานสอง