

2 Dimension Array

อะเรย์ 2 มิติ

Natsuda Kaothanthong

4 October, 2014

1D Array

- อะเรย์ 1 มิติ ใช้เก็บชุดข้อมูล

```
int x[3] = {3, 4, 5};
```



แต่ละข้อมูลเป็น **primitive type** (ชนิดทั่วไป) เช่น int, float, double, char

- ข้อมูลทั้ง 3 ตัว ถูกเก็บภายใต้ชื่อ **x**

ข้อจำกัดของ 1D Array

- ตัวอย่าง เก็บค่าสตริงตั้งแต่แม่ 1 ถึง แม่ 3

| | | | | | | | | | | | | |
|---------------|---|---|---|----|----|----|----|----|----|----|----|----|
| int mult1[12] | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| int mult2[12] | 2 | 4 | 6 | 8 | 10 | 12 | 14 | 16 | 18 | 20 | 22 | 24 |
| int mult3[12] | 3 | 6 | 9 | 12 | 15 | 18 | 21 | 24 | 27 | 30 | 33 | 36 |

| | | | | | | | | | | | | |
|---------------|---|---|---|----|----|----|----|----|----|----|----|----|
| int mult1[12] | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| int mult2[12] | 2 | 4 | 6 | 8 | 10 | 12 | 14 | 16 | 18 | 20 | 22 | 24 |
| int mult3[12] | 3 | 6 | 9 | 12 | 15 | 18 | 21 | 24 | 27 | 30 | 33 | 36 |

- หากต้องการแสดงค่าแต่ละค่า ต้องใช้ 3 ลูป เพื่อแสดงค่าในแต่ละอะเรย์

```
for(i = 0; i < 12; i++)  
{ printf("1*%d = %d\n",i+1, mult1[i]); }
```

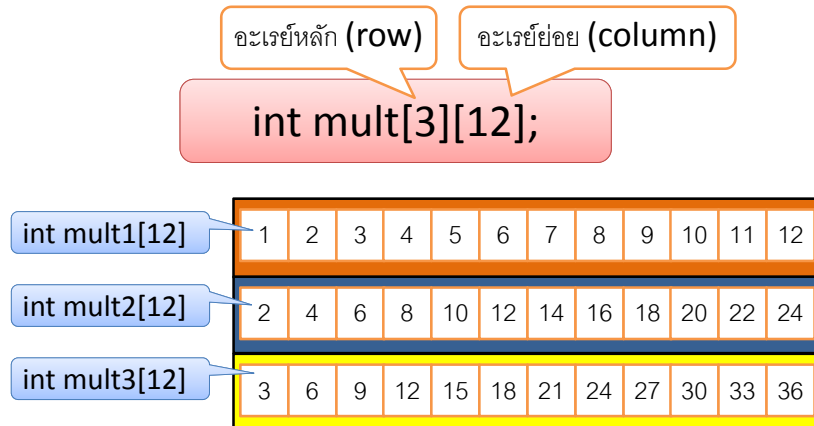
```
for(i = 0; i < 12; i++)  
{ printf("2*%d = %d\n",i+1, mult1[i]); }
```

```
for(i = 0; i < 12; i++)  
{ printf("3*%d = %d\n",i+1, mult1[i]); }
```

จำนวนลูป
เท่ากับจำนวน
ตัวแปรอะเรย์

2D Array

- อะเรย์ 2 มิติ หรือ อะเรย์ของอะเรย์ 1 มิติ



mult

column

row

| | | | | | | | | | | | |
|---|---|---|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| 2 | 4 | 6 | 8 | 10 | 12 | 14 | 16 | 18 | 20 | 22 | 24 |
| 3 | 6 | 9 | 12 | 15 | 18 | 21 | 24 | 27 | 30 | 33 | 36 |

การเรียกใช้ข้อมูลแต่ละตัวในอะเรย์

อะเรย์หลัก (row) อะเรย์ย่อย (column)

mult[row_num][col_num]

การประกาศ 2D Array

ชนิดข้อมูล ชื่อตัวแปร[row][column];

1. `int x[3][5] = {1,2,3,4,5,2,4,6,8,10,3,6,9,12,15};`
2. `int x[3][5] = {{1,2,3,4,5},
 {2,4,6,8,10},
 {3,6,9,12,15}};`
3. `int x[][5] = {{1,2,3,4,5},
 {2,4,6,8,10},
 {3,6,9,12,15}};`
4. `int x[3][5] = {0};`

การกำหนดค่าให้แต่ละข้อมูลใน 2D อะเรย์

1. `x[2][1] = 2;` `x[3][3] = 9;`
2. `scanf("%d",&x[3][5]);`
3. `printf("2 * 3 = %d", x[1][2]);`
4. `x[0][4] = 1*5;`

การใช้ 2D array และ nested loop

- ส่วนใหญ่ 2 มิติจะทำงานกับ loop ซ้อน
- ตัวอย่างโค้ดสำหรับกำหนดค่าของ `int x[3][4];`

```
for(row = 0; row < 3; row++)
{
    for(col = 0; col < 4; col++)
    {
        x[row][col] = (row)*(col);
    }
}
```

โปรแกรมแสดงค่าสูตรคูณแม่ 1 - 9

```
int main()
{
    int mult[9][12];
    int row, col;

    for(row = 0; row < 9; row++)
    {
        for(col = 0; col < 12; col++)
        {
            mult[row][col] = (row+1)*(col+1);
        }
    }

    // Display
    printf("**\t| %5d%5d%5d%5d%5d%5d%5d%5d%5d%5d%5d\n", 1,2,3,4,5,6,7,8,9,10,11,12);
    printf("-----\n");
    for(row = 0; row < 9; row++)
    {
        printf("%d\t|", row+1);
        for(col = 0; col < 12; col++)
        {
            printf("%5d", mult[row][col]);
        }
        printf("\n");
    }
    return 0;
}
```

multiplication1_9.c

```
int main()
{
    int num[10][10] = {0};
    int i, j, count = 0;
    for(i = 0; i < 10; i++)
    {
        for(j = 0; j <= i; j++)
        {
            num[i][j] = count;
            printf("%5d", num[i][j]);
            count++;
        }
        printf("\n");
    }
}
```

triangle_num.c

Array of String

- String เป็นอาร์เรย์ของตัวอักษร

```
char name1[] = "Chalee";
char name2[] = "Manee";
char name3[] = "Wittaya";
```

- Array of string คืออาร์เรย์ที่แต่ละข้อมูลมีชนิด String

```
char name[3][50] = {"Chalee", "Manee", "Wittaya"};
```

หรือ

```
char name[][50] = {"Chalee", "Manee", "Wittaya"};
```

จำนวนข้อมูลถูกกำหนดจาก
จำนวนชื่อ เมื่อประกาศตัวแปร

ต้องกำหนดจำนวน
ตัวอักษรมากที่สุด

โปรแกรมสำหรับค้นหาชื่อ

```
1  #include <stdio.h>
2
3  int main()
4  {
5      char name[3][50]={"Chalee", "Manee", "Wittaya"};
6      char search[50];
7      printf("Enter a name to search: ");
8      scanf("%s", search);
9
10     int i;
11     for(i = 0; i < 3; i++)
12     {
13         if(strcmp(search, name[i]) == 0)
14         {
15             printf("Name: %s is found in the database.\n", search);
16             return 0;
17         }
18     }
19     printf("Name: %s is not in the database.\n", search);
20
21     return 0;
22 }
```