



มหาวิทยาลัยราชภัฏนครปฐม



บทที่ 4 อาร์เรย์

ผู้บรรยาย : ผศ.ดร.ณัฐชามณูห์ ศรีจำเริญรัตน์
สาขาวิชาคอมพิวเตอร์ธุรกิจ คณะวิทยาการจัดการ
มหาวิทยาลัยราชภัฏนครปฐม



มหาวิทยาลัยราชภัฏนครปฐม
Nakhon Pathom Rajabhat University



อาร์เรย์

- โครงสร้างของอาร์เรย์ 1 มิติ
- โครงสร้างของอาร์เรย์ 2 มิติ
- โครงสร้างของอาร์เรย์ 3 มิติ
- การค้นหาข้อมูลในอาร์เรย์
- การแก้ไขหรือการลบข้อมูลในอาร์เรย์
- การนับจำนวนสมาชิกในอาร์เรย์

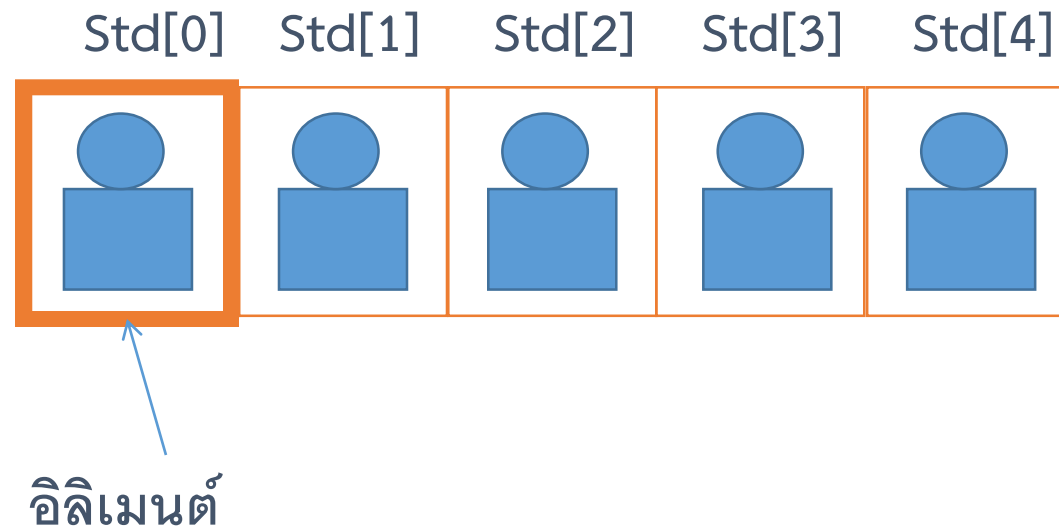


อาร์เรย์

- อาร์เรย์ (array) คือ กลุ่มข้อมูลชนิดเดียวกัน, มีขนาดแน่นอน
- ข้อมูลในอาร์เรย์แต่ละตัว เรียกว่า อิลิเมนต์ (Element)
- การเข้าถึงข้อมูลจะใช้ตัวชี้ตำแหน่ง เรียกว่า ดัชนี (index)
- ตำแหน่งของอาร์เรย์ที่เรียงต่อกัน เรียกว่า มิติของอาร์เรย์
- อาร์เรย์แถวเดียวที่เรียงต่อกัน เรียกว่า อาร์เรย์หนึ่งมิติ
- อาร์เรย์มีมากกว่าอาร์เรย์หนึ่งมิติ เรียกว่า อาร์เรย์ตามจำนวนมิติที่สร้างขึ้น เช่น อาร์เรย์สองมิติ อาร์เรย์สามมิติและอื่น ๆ เป็นต้น

อาร์เรย์

รูปแบบอาร์เรย์ 1 มิติ





4.1 โครงสร้างของอาร์เรย์ 1 มิติ

- อาร์เรย์ 1 มิติ
- อาร์เรย์จะประกอบด้วยข้อมูล 2 ส่วนคือ
 - “ข้อมูล (data)” โดยข้อมูลที่อยู่ในอาร์เรย์เดียวกันจะมีข้อมูลชนิดเดียวกัน
 - “ดัชนี (index)” ใช้อ้างอิงตำแหน่งของข้อมูลภายในอาร์เรย์ โดยใช้ตัวเลขจำนวนเต็มซึ่งเป็นตำแหน่งสมาชิกที่ต้องการอ้างอิง และตำแหน่งอ้างอิงเริ่มต้นจะกำหนดที่ 0 และไม่เกินขนาดของจำนวนสมาชิกในอาร์เรย์



4.1 โครงสร้างของอาร์เรย์ 1 มิติ(ต่อ)

- การประกาศอาร์เรย์ 1 มิติ มีวิธีการกำหนดการประกาศอยู่ 2 รูปแบบ ดังนี้
 - 1) การประกาศอาร์เรย์ 1 มิติ โดยระบุข้อมูลตามจำนวนที่ต้องการจัดเก็บการประกาศลักษณะนี้เป็นการประกาศขนาดของอาร์เรย์เท่ากับจำนวนข้อมูลที่กำหนดไว้ หรือรู้ข้อมูลที่แน่ชัดอยู่ก่อน



4.1 โครงสร้างของอาร์เรย์ 1 มิติ(ต่อ)

การประกาศตัวแปรอาร์เรย์ 1 มิติ

มีการระบุข้อมูลตามจำนวนที่ต้องการจัดเก็บ มีรูปแบบการประกาศ ดังนี้

| | | |
|--------|------------|--|
| | ชนิดข้อมูล | ชื่อตัวแปร[]={ ข้อมูล_1, ข้อมูล_2, ข้อมูล_3, ..., ข้อมูล_n }; |
| โดยที่ | ชนิดข้อมูล | เป็นชนิดข้อมูลที่ต้องการ กำหนดให้กับตัวแปรชนิดอาร์เรย์ (integer, float, double) |
| | ชื่อตัวแปร | เป็นชื่อของอาร์เรย์ |
| | ข้อมูล | เป็นข้อมูลที่ต้องการจัดเก็บในอาร์เรย์ |



4.1 โครงสร้างของอาร์เรย์ 1 มิติ(ต่อ)

ตัวอย่างที่ 4.1 การประกาศอาร์เรย์ 1 มิติ โดยให้มีการระบุข้อมูลตามจำนวนที่ต้องการจัดเก็บ และทำการบวกข้อมูลทั้งหมด

```
1 #include <iostream>
2 using namespace std;
3 int main()
4 {
5     int sizeofarray=5, sum=0;
6     int data[]= {10,20,30,40,50};
7     for (int i=0;i< sizeofarray;i++)
8     {
9         cout<<"Data No. ["<<i+1<<"]: "<<data[i]<<"\n" ;
10        sum=sum+data[i];
11    }
12    cout<<"sum total : "<<sum;
13    system("pause");
14    return 0;
15 }
```



4.1 โครงสร้างของอาร์เรย์ 1 มิติ(ต่อ)

2) การประกาศอาร์เรย์ 1 มิติ (อีกแบบ) โดยการระบุขนาดของจำนวนข้อมูล

รูปแบบการประกาศตัวแปรอาร์เรย์ 1 มิติ โดยให้มีการระบุขนาดตามจำนวนที่ต้องการจัดเก็บ มีรูปแบบการประกาศ ดังภาพด้านล่างต่อไปนี้

| | ชนิดข้อมูล | ชื่อตัวแปร [จำนวนข้อมูล]; |
|--------|-------------|---|
| โดยที่ | ชนิดข้อมูล | เป็นชนิดข้อมูลที่ต้องการกำหนดให้กับตัวแปรชนิดอาร์เรย์ (integer, float, double) |
| | ชื่อตัวแปร | เป็นชื่อของอาร์เรย์ |
| | จำนวนข้อมูล | เป็นจำนวนอิลิเมนต์ที่ต้องการกำหนด |



4.1 โครงสร้างของอาร์เรย์ 1 มิติ(ต่อ)

ตัวอย่างการประกาศตัวแปรอาร์เรย์โดยให้มีการระบุขนาดตามจำนวนที่ต้องการจัดเก็บ เช่น

```
int sample[5]
```

ตัวแปร sample จะสามารถแทนค่าการประกาศตัวแปรได้ ดังนี้

| | | | | | |
|--------|------------|------------|------------|------------|------------|
| ดัชนี | sample [0] | sample [1] | sample [2] | sample [3] | sample [4] |
| ข้อมูล | 1 | 3 | 5 | 7 | 9 |



4.1 โครงสร้างของอาร์เรย์ 1 มิติ(ต่อ)

ตัวอย่างที่ 4.2 การอ้างอิงข้อมูลแต่ละตัวในอาร์เรย์

sample [0] = 1;

sample [1] = 3;

sample [2] = 5;

sample [3] = 7;

sample [4] = 9;



4.1 โครงสร้างของอาร์เรย์ 1 มิติ(ต่อ)

- การกำหนดค่าเริ่มต้นให้กับอาร์เรย์ 1 มิติ
 - กำหนดค่าเริ่มต้นให้กับข้อมูลแต่ละอิลิเมนต์
 - ข้อมูลแต่ละตัวจะถูกแบ่งด้วยเครื่องหมายคอมมา (,)
 - ค่าที่เก็บในอาร์เรย์แต่ละอิลิเมนต์จะเรียงลำดับข้อมูลที่เก็บ โดย
name[0] จะเก็บค่า a
name[1] จะเก็บค่า e ตามลำดับ ดังนี้

```
char name[5]={'a','e','l','o','u'};
```

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



4.1 โครงสร้างของอาร์เรย์ 1 มิติ(ต่อ)

ตัวอย่างที่ 4.3 การประกาศอาร์เรย์ 1 มิติ โดยให้มีการระบุขนาดตามจำนวนที่ต้องการจัดเก็บ พร้อมทั้งแสดงข้อมูลในอาร์เรย์

```
1  #include <iostream>
2  using namespace std;
3  int main()
4  {
5      int data[5]= {10, 20, 30, 40, 50};
6      for (int i=0; i< 5; i++)
7      {
8          cout<<"Data No. ["<<i+1<<"]: "<<data[i]<<"\n" ;
9      }
10     return 0;
11 }
```



4.1 โครงสร้างของอาร์เรย์ 1 มิติ(ต่อ)

ตัวอย่างที่ 4.4 การรับข้อมูลให้กับตัวแปรชนิดอาร์เรย์ 1 มิติ พร้อมทั้งหาผลรวมของข้อมูลที่ได้รับเข้ามา

```
1  #include <iostream>
2  using namespace std;
3  int main()
4  {
5      int data[5];
6      int sum = 0;
7      for (int i=0;i< 5;i++)
8      {
9          cout<<"\n Please enter array No. ["<<i+1<<"]: " ;
10         cin>>data[i];
11         sum=sum+data[i];
12     }
13     cout<<"\n sum total :"<<sum;
14     return 0;
15 }
```



4.2 โครงสร้างของอาร์เรย์ 2 มิติ

โครงสร้างอาร์เรย์ 2 มิติจะมีลักษณะคล้ายกับอาร์เรย์ 1 มิติ

โดยอาร์เรย์ 2 มิติจะประกอบด้วย

- แถว (row) และ คอลัมน์ (column)
- ข้อมูลแต่ละชุดจะมีเครื่องหมายก้ามปูเปิด (“[“) และก้ามปูปิด (“]”)
- ข้อมูลแต่ละแถวจะถูกแบ่งด้วยเครื่องหมายคอมมา (,)



4.2 โครงสร้างของอาร์เรย์ 2 มิติ(ต่อ)

4.2.1 การประกาศอาร์เรย์ 2 มิติ

รูปแบบการประกาศตัวแปรอาร์เรย์ 2 มิติ สามารถจัดการข้อมูลภายในอาร์เรย์สองมิติได้โดยใช้ค่าอินเด็กซ์อ้างอิงค่าในอาร์เรย์ข้อมูลสองมิติ โดย ค่าแรกที่อ้างอิงจะหมายถึงแถวและค่าที่สองหมายถึงคอลัมน์ และให้มีการระบุข้อมูลตามจำนวนที่ต้องการจัดเก็บ



4.2 โครงสร้างของอาร์เรย์ 2 มิติ(ต่อ)

ตัวอย่างที่ 4.5 การสร้างอาร์เรย์ 2 มิติ

```
int num[2][3] = {{11,12,13},{21,22,23}};
```

```
int sample[3][4];
```

คำอธิบาย

ตัวแปรอาร์เรย์ 2 มิติชื่อ sample มีชนิดข้อมูลเป็นตัวเลขจำนวนเต็ม

ขนาด 3x4 (หมายถึง 3 แถวและ 4 คอลัมน์) ประกอบด้วย

ดัชนีสองตัว คือ “ดัชนีแถว” และ “ดัชนีคอลัมน์”

จากตัวอย่างการสร้างอาร์เรย์ 2 มิติ ตัวแปร sample จะสามารถแทนค่าการประกาศตัวแปรปกติ 12 ตัวแปร



4.2 โครงสร้างของอาร์เรย์ 2 มิติ(ต่อ)

การอ้างอิงข้อมูลในอาร์เรย์ 2 มิติ

ตัวแปรอาร์เรย์ 2 มิติ ขนาด 3x4 จากตัวอย่างที่ 4.5

| | | Column (คอลัมน์) | | | |
|-----------|------------|--------------------|--------------------|--------------------|--------------------|
| | | [0] | [1] | [2] | [3] |
| Row (แถว) | sample [0] | 12 sample [0,0] | 34 sample [0,1] | 6 sample [0,2] | 1 sample [0,3] |
| | sample [1] | 44 sample [1,0] | 5 sample [1,1] | 3 sample [1,2] | 25 sample [1,3] |
| | sample [2] | 89 sample [2,0] | 7 sample [2,1] | 65 sample [2,2] | 87 sample [2,3] |



4.2 โครงสร้างของอาร์เรย์ 2 มิติ(ต่อ)

ตัวอย่างที่ 4.6 การอ้างอิงข้อมูลในอาร์เรย์ 2 มิติ โดยใช้ข้อมูลอ้างอิงจากตัวแปรอาร์เรย์ 2 มิติ
ขนาด 3x4 ในตัวอย่างที่ 4.5

sample [2][0] = 89;

sample [0][2] = 6;



4.2 โครงสร้างของอาร์เรย์ 2 มิติ(ต่อ)

ตัวอย่างที่ 4.7 การกำหนดค่าเริ่มต้นให้กับอาร์เรย์ 2 มิติ

- รูปแบบอาร์เรย์

```
char name[2][3] = {  
    {'a', 'e', 'l'},  
    {'x', 'y', 'z'}  
};
```

```
int number[3][5] = {  
    {5, 4, 13, 2, 1},  
    {15, 24, 3, 32, 1},  
    {52, 4, 93, 2, 11}  
};
```

- รูปแบบผลลัพธ์ข้อมูล

| | 0 | 1 | 2 |
|---|---|---|---|
| 0 | a | e | l |
| 1 | x | y | z |

| | 0 | 1 | 2 | 3 | 4 |
|---|----|----|----|----|----|
| 0 | 5 | 4 | 13 | 2 | 1 |
| 1 | 15 | 24 | 3 | 32 | 1 |
| 2 | 52 | 4 | 93 | 2 | 11 |



4.2 โครงสร้างของอาร์เรย์ 2 มิติ(ต่อ)

ตัวอย่างที่ 4.8 จงเขียนโปรแกรมเพื่อหาจำนวนนักศึกษาชั้นปีที่ 1-4 ของแต่ละสาขาวิชา โดยให้ทำการประกาศและการกำหนดข้อมูลให้กับตัวแปรชนิดอาร์เรย์ 2 มิติ ดังรายละเอียดต่อไปนี้

| ลำดับที่ | ชั้นปีที่ 1 | ชั้นปีที่ 2 | ชั้นปีที่ 3 | ชั้นปีที่ 4 |
|----------|-------------|-------------|-------------|-------------|
| 1 | 250 | 170 | 151 | 222 |
| 2 | 297 | 246 | 158 | 287 |
| 3 | 178 | 255 | 291 | 196 |
| 4 | 230 | 170 | 157 | 206 |

หมายเหตุ ลำดับที่ 1 แทนสาขาวิชาคอมพิวเตอร์ธุรกิจ (Business Computer)

ลำดับที่ 2 แทนสาขาวิชาบัญชี (Accountance)

ลำดับที่ 3 แทนสาขาวิชาการตลาด (Marketing)

ลำดับที่ 4 แทนสาขาวิชาการจัดการโลจิสติกส์ (Logistics Management)

ตัวอย่างที่ 4.8 โค้ดโปรแกรม



```
1 #include <iostream>
2 using namespace std;
3 int main()
4 {
5     int student[4][4];
6     int row, col, sum, inputstudent;
7     for ( row = 0; row <=3; row++ )
8     {
9         if (row==0){ cout<<"Business Computer"; }
10        else if (row==1){ cout<<"Accountance"; }
11        else if (row==2){ cout<<"Marketing"; }
12        else { cout<<"Logistics Management"; }
13        cout<<endl;
14        for ( col =0; col <=3; col++ )
15        {
16            cout<<"year "<<col+1<<"=";
17            cin>>inputstudent;
18            student[row][col]=inputstudent;
19        }
20 }
```

```
21 cout<<"=====";
22 for ( row =0; row <=3; row++ )
23 {
24     cout<<endl;
25     if (row==0){ cout<<"Business Computer"; }
26     else if (row==1){ cout<<"Accountance"; }
27     else if (row==2){ cout<<"Marketing"; }
28     else { cout<<"Logistics Management"; }
29     sum=0;
30     for ( col =0; col <=3; col++ )
31     {
32         sum=sum+student[row][col];
33     }
34     cout<<" total ="<<sum ;
35 }
36 return 0;
37 }
```



4.2 โครงสร้างของอาร์เรย์ 2 มิติ(ต่อ)

4.2.2 การประกาศตัวแปรอาร์เรย์ชนิดข้อความ

- ข้อมูลที่เป็นอักขรสามารถกำหนดในรูปแบบตัวแปรประเภท char
- ข้อมูลอักขรที่กำหนดเป็นข้อความจะกำหนดก็ตัวก็ได้
- แต่ต้องน้อยกว่าจำนวนช่องที่ประกาศไว้ เช่น ประกาศไว้ 5 ช่อง จะใส่ค่าได้ 4 อักขรเท่านั้น เพราะช่องสุดท้ายจะเก็บไว้เพื่อเก็บรหัสจบข้อความ หรือ "\0" นั่นเอง



4.2 โครงสร้างของอาร์เรย์ 2 มิติ(ต่อ)

ตัวอย่างที่ 4.9 โปรแกรมแสดงข้อมูลอักษรในรูปแบบอาร์เรย์ 2 มิติ

```
1  #include <iostream>
2  using namespace std;
3  int main()
4  {
5      int row, col;
6      char data[2][3] = {{'a','e','i'},{'x','y','z'}};
7      for ( row=0;row<2;row++)
8      {
9          for ( col=0; col<3; col++)
10         {
11             cout<< data [row][col] << " ";
12         }
13         cout<<endl;
14     }
15 }
```

ผลการทำงานของโปรแกรม

```
a e i
x y z
```



4.3 โครงสร้างของอาร์เรย์ 3 มิติ

- อาร์เรย์ 3 มิติ จะมีลักษณะคล้ายอาร์เรย์ 2 มิติที่นำมาวางซ้อนกัน
- อาร์เรย์ 3 มิติจะประกอบด้วยแถว (row) คอลัมน์ (column) และมีส่วนที่เพิ่มเข้ามา คือ ความลึก
- การใช้งานจะมีลักษณะคล้ายกับการนำอาร์เรย์ 2 มิติมาเรียงซ้อนกัน

□ การประกาศอาร์เรย์ 3 มิติ

รูปแบบการประกาศตัวแปรอาร์เรย์ 3 มิติ โดยให้มีการระบุข้อมูลตามจำนวนที่ต้องการจัดเก็บ
มีรูปแบบการประกาศ

ชนิดข้อมูล ชื่อตัวแปร[ชั้น][แถว][คอลัมน์];



4.3 โครงสร้างของอาร์เรย์ 3 มิติ(ต่อ)

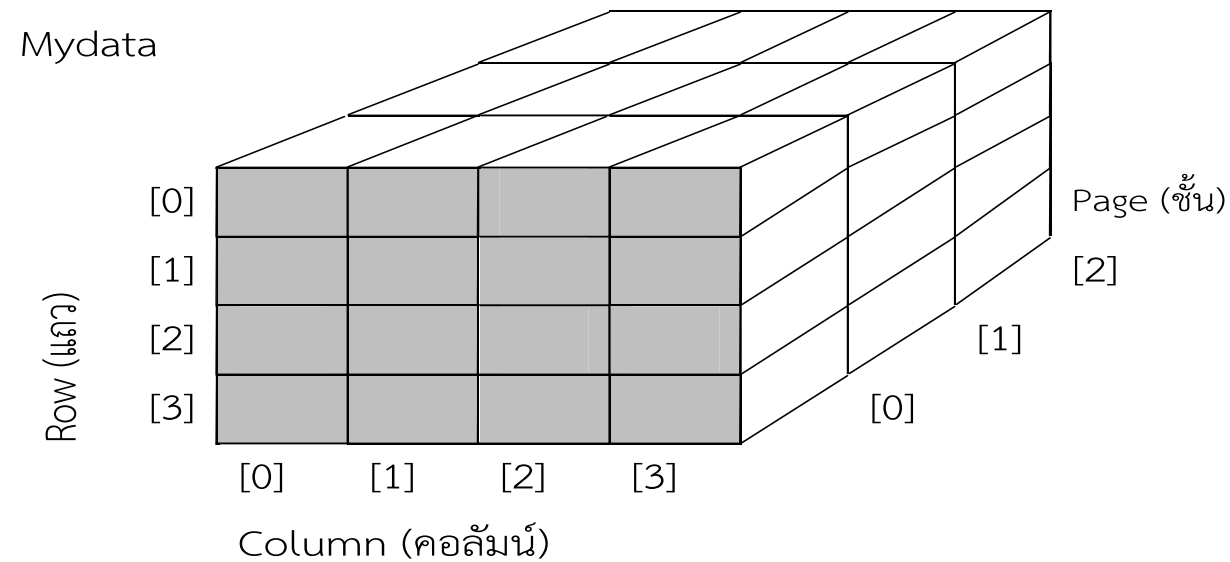
ตัวอย่างที่ 4.10 การสร้างอาร์เรย์ 3 มิติ

```
Int Mydata[3][4][4];
```

4.3 โครงสร้างของอาร์เรย์ 3 มิติ(ต่อ)

คำอธิบาย ตัวอย่างที่ 4.10

ประกาศอาร์เรย์ Mydata เก็บจำนวนเต็ม มีความลึก 3 ชั้น 4 แถว แถวละ 4 คอลัมน์



- รูปแบบการนำเสนอข้อมูลอาร์เรย์ 3 มิติ



4.3 โครงสร้างของอาร์เรย์ 3 มิติ(ต่อ)

ตัวอย่างที่ 4.11 การรับข้อมูลจากคีย์บอร์ดและแสดงข้อมูลจากอาร์เรย์ 3 มิติขนาด 2 x 3 x 3

```
1 #include <iostream>
2 using namespace std;
3 int main()
4 {
5     int x,y,z,data[2][3][3];
6     for(x=0;x<=1;x++)
7     {
8         for(y=0;y<=2;y++)
9         {
10            for(z=0;z<=2;z++)
11            {
12                cout<<"data["<<x<<"]["<<y<<"]["<<z<<"]: ";
13                cin>>data[x][y][z];
14            }
15        }
16    }
17    cout<<"\n===display data===\n";
18    for(x=0;x<=1;x++)
19    {
20        for(y=0;y<=2;y++)
21        {
22            for(z=0;z<=2;z++)
23            {
24                cout<<data[x][y][z]<<" ";
25            }
26            cout<<endl;
27        }
28        cout<<endl;
29    }
30    return 0;
31 }
```



4.4 การค้นหาข้อมูลในอาร์เรย์

การดำเนินงานเพื่อค้นหาข้อมูลในอาร์เรย์ สามารถทำได้หลากหลายวิธี เช่น

- การค้นหาแบบแบ่งทีละครึ่ง
- การค้นหาแบบเรียงลำดับ



4.4 การค้นหาข้อมูลในอาร์เรย์(ต่อ)

ตัวอย่างที่ 4.12 โปรแกรมการค้นหาข้อมูลในอาร์เรย์

```
1 #include <iostream>
2 using namespace std;
3 int main()
4 {
5     int i,key;
6     int data[5]={1,3,5,7,9};
7     cout<<"Please enter number : ";
8     cin>>key;
9     for ( i=0;i<5;i++)
10    {
11        if (key==data[i])
12        {
13            cout<<"Found target at position : "<<i<<endl;
14        }
15    }
16    return 0;
17 }
```

ผลการทำงานของโปรแกรม

Please enter number : 1

Found target at position : 0



4.5 การแก้ไขหรือการลบข้อมูลในอาร์เรย์

การแก้ไขข้อมูลหรือการลบข้อมูลในอาร์เรย์มีการทำงานเหมือนกับการรับข้อมูลเข้าอาร์เรย์ ดังนั้นการรับข้อมูลเพื่อแก้ไขจำเป็นต้องทราบตำแหน่งที่ต้องการแก้ไขหรือตำแหน่งที่ต้องการลบเสียก่อน ดังแสดงในตัวอย่างต่อไปนี้



4.5 การแก้ไขหรือการลบข้อมูลในอาร์เรย์(ต่อ)

ตัวอย่างที่ 4.13 จงเขียนโปรแกรมเพื่อกำหนดอาร์เรย์ชื่อ mydata และทำการลบข้อมูลในอาร์เรย์ตำแหน่ง mydata[1][1] และ แก้ไขข้อมูลในอาร์เรย์ในตำแหน่ง mydata [2][2]

```
1  #include <iostream>
2  using namespace std;
3  int main()
4  {
5      int mydata[3][3] = {
6          { 1,2,3 } , { 4,5,6 } , { 7,8,9 }
7      };
8      for (int j = 0 ; j <=2 ; j++ )
9      {
10         for (int i = 0 ; i <= 2 ; i++ )
11             cout<<mydata[j][i]<<"  ";
12         cout<<endl;
```

```
13     }
14     mydata[1][1]=NULL; //delete
15     mydata[2][2]=55; //update
16     cout<<"====display====";
17     cout<<endl;
18     for (int j = 0 ; j <=2 ; j++ )
19     {
20         for (int i = 0 ; i <= 2 ; i++ )
21             cout<<mydata[j][i]<<"  ";
22         cout<<endl;
23     }
24     return 0;
```



4.5 การแก้ไขหรือการลบข้อมูลในอาร์เรย์ (ต่อ)

ผลการทำงานของโปรแกรม

```
1  2  3
4  5  6
7  8  9
=====display=====
1  2  3
4  0  6
7  8  55
```

ลบข้อมูลในอาร์เรย์ตำแหน่ง mydata[1][1]
แก้ไขข้อมูลในอาร์เรย์ในตำแหน่ง mydata [2][2]



4.6 การนับจำนวนสมาชิกในอาร์เรย์

4.6.1 การนับจำนวนสมาชิกในอาร์เรย์ 1 มิติ

- จำนวนสมาชิกสามารถคำนวณได้จากตำแหน่งดัชนีตำแหน่งเริ่มต้นและตำแหน่งสิ้นสุด
- ปกติจะเริ่มจาก 1, 2, 3,...,20 หรือเริ่มจาก -2, -1, 0, 1, 2 เป็นต้น
- โดยมากมักนิยามกำหนดค่าดัชนีเป็นตัวเลข



4.6 การนับจำนวนสมาชิกในอาร์เรย์(ต่อ)

สมการการนับจำนวนสมาชิกในอาร์เรย์ 1 มิติ

$$\text{จำนวนสมาชิก} = (L1 - B1) + 1$$

โดย $L1$ = ค่าขอบเขตดัชนีบนสุดของแถว

$B1$ = ค่าขอบเขตดัชนีล่างสุดของแถว

การนับจำนวนสมาชิกในอาร์เรย์ 1 มิติเราจำเป็นต้องมีข้อมูลขอบเขตดัชนีบนสุดสุดและค่าขอบเขตดัชนีต่ำสุดเพื่อใช้ในสมการข้างต้น ในการนับจำนวนข้อมูลสมาชิก



4.6 การนับจำนวนสมาชิกในอาร์เรย์(ต่อ)

ตัวอย่างที่ 4.14 จงคำนวณหาจำนวนสมาชิกของอาร์เรย์ที่ใช้เก็บคะแนนสอบของ

ชมรมดนตรีคนที่ 123 – 137

ค่าขอบเขตดัชนีสูงสุด = 137

ค่าขอบเขตดัชนีต่ำสุด = 123

แทนค่าในสมการนับจำนวนสมาชิกในอาร์เรย์ $= (137 - 123) + 1$

$$= 14 + 1$$

$$= 15$$

จำนวนสมาชิกของอาร์เรย์เท่ากับ 15



4.6 การนับจำนวนสมาชิกในอาร์เรย์(ต่อ)

ตัวอย่างที่ 4.15 ร้านขายหนังสือต้องการสำรวจจำนวนหนังสือที่ซื้อตั้งแต่ปี 2549 ถึงปี 2558

จงคำนวณหาจำนวนสมาชิกของอาร์เรย์ที่ใช้เก็บข้อมูลชุดนี้

ค่าขอบเขตดัชนีสูงสุด = 2558

ค่าขอบเขตดัชนีต่ำสุด = 2549

$$\begin{aligned}\text{แทนค่าในสมการการนับจำนวนสมาชิกในอาร์เรย์} &= (2558 - 2549) + 1 \\ &= 9 + 1 \\ &= 10\end{aligned}$$

จำนวนสมาชิกของอาร์เรย์เท่ากับ 10



4.6 การนับจำนวนสมาชิกในอาร์เรย์(ต่อ)

4.6.2 การนับจำนวนสมาชิกในอาร์เรย์ 2 มิติ

อาร์เรย์ 2 มิติสามารถหาจำนวนสมาชิกได้โดยการอ้างอิงจากการหาสมาชิกของอาร์เรย์ 1 มิติ จากการนับจำนวนสมาชิกในอาร์เรย์ได้จากสมการต่อไปนี้

$$\text{จำนวนสมาชิก} = ((L1-B1)+1) \times ((L2-B2)+1)$$

- โดย
- L1 = ค่าขอบเขตดัชนีบนสุดของแถว
 - B1 = ค่าขอบเขตดัชนีล่างสุดของแถว
 - L2 = ค่าขอบเขตดัชนีบนสุดของคอลัมน์
 - B2 = ค่าขอบเขตดัชนีล่างสุดของคอลัมน์



4.6 การนับจำนวนสมาชิกในอาร์เรย์(ต่อ)

ตัวอย่างที่ 4.16 จงคำนวณหาจำนวนสมาชิกของอาร์เรย์ชื่อ mydata[5][4]

ค่าขอบเขตดัชนีสูงสุดของแถว = 4

ค่าขอบเขตดัชนีต่ำสุดของแถว = 0

ค่าขอบเขตดัชนีสูงสุดของคอลัมน์ = 3

ค่าขอบเขตดัชนีต่ำสุดของคอลัมน์ = 1

$$\begin{aligned}\text{แทนค่าในสมการการนับจำนวนสมาชิกในอาร์เรย์} &= ((4-0)+1) \times ((3-0)+1) \\ &= (4+1) \times (3+1) \\ &= 5 \times 4 \\ &= 20\end{aligned}$$

จำนวนสมาชิกของอาร์เรย์เท่ากับ 20



4.6 การนับจำนวนสมาชิกในอาร์เรย์(ต่อ)

ภาพประกอบอาร์เรย์ 2 มิติ

mydata[5][4]

| | | COLUMN | | | |
|-----|---|-------------|-------------|-------------|-------------|
| | | 0 | 1 | 2 | 3 |
| ROW | 0 | mydata[0,0] | mydata[0,1] | mydata[0,2] | mydata[0,3] |
| | 1 | mydata[1,0] | mydata[1,1] | mydata[1,2] | mydata[1,3] |
| | 2 | mydata[2,0] | mydata[2,1] | mydata[2,2] | mydata[2,3] |
| | 3 | mydata[3,0] | mydata[3,1] | mydata[3,2] | mydata[3,3] |
| | 4 | mydata[4,0] | mydata[4,1] | mydata[4,2] | mydata[4,3] |



4.6 การนับจำนวนสมาชิกในอาร์เรย์(ต่อ)

4.6.3 การนับจำนวนสมาชิกในอาร์เรย์ 3 มิติ

อาร์เรย์ 3 มิติ สามารถคำนวณหาจำนวนสมาชิกได้โดยการอ้างอิงจากจำนวนชั้นของอาร์เรย์ แถวและคอลัมน์ ซึ่งคล้ายกับการหาสมาชิกของอาร์เรย์ 2 มิติ มาคำนวณแต่เพิ่มจำนวนชั้นของอาร์เรย์เข้ามาคำนวณด้วยนั่นเอง โดยเราสามารถนับจำนวนสมาชิกในอาร์เรย์ได้จากสมการต่อไปนี้

$$\text{จำนวนสมาชิก} = ((L1-B1)+1) \times ((L2-B2)+1) \times ((L3-B3)+1)$$

โดย

- L1 = ค่าขอบเขตดัชนีล่างสุดของชั้น
- B1 = ค่าขอบเขตดัชนีบนสุดของชั้น
- L2 = ค่าขอบเขตดัชนีล่างสุดของแถว
- B2 = ค่าขอบเขตดัชนีล่างสุดของแถว
- L3 = ค่าขอบเขตดัชนีล่างสุดของคอลัมน์
- B3 = ค่าขอบเขตดัชนีล่างสุดของคอลัมน์



4.6 การนับจำนวนสมาชิกในอาร์เรย์(ต่อ)

ตัวอย่างที่ 4.17 จงคำนวณหาจำนวนสมาชิกของอาร์เรย์ชื่อ mydata [3][4][5]

ประกาศให้อาร์เรย์ชื่อ mydata มีสมาชิก 3 ชั้น 4 แถว 5 คอลัมน์

สามารถคำนวณได้จากสมการการคำนวณหาจำนวนสมาชิกในอาร์เรย์

$$\begin{aligned}\text{จำนวนสมาชิก} &= ((L1-B1)+1) \times ((L2-B2)+1) \times ((L3-B3)+1) \\ &= ((2-0)+1) \times ((3-0)+1) \times ((4-0)+1) \\ &= 3 \times 4 \times 5 \\ &= 60\end{aligned}$$

จำนวนสมาชิกของอาร์เรย์เท่ากับ 60



มหาวิทยาลัยราชภัฏนครปฐม