

ครั้งที่ 7

โครงสร้างข้อมูลชนิด **struct**

Struct

- เป็นโครงสร้าง ระเบียบ ข้อมูลที่ผู้เขียน โปรแกรมกำหนดขึ้น เพื่อใช้ในการทำงาน
- เป็นโครงสร้างที่เก็บรายการข้อมูล(**field**) ที่เกี่ยวข้องกันไว้ด้วยกัน

Struct ของนักศึกษา

- กำหนด ระเบียนนักศึกษาประกอบด้วย รายการข้อมูลดังนี้
 - รหัสนักศึกษา
 - ชื่อนักศึกษา
 - คะแนนสอบกลางภาค
 - คะแนนสอบปลายภาค
 - เกรด

struct Type Declaration

รูปแบบ: struct *struct-type*

```
{  
    Type1 id-list1;  
    Type2 id-list2;  
    .  
    typen id-listn;  
};
```

Struct ของนักศึกษา

- กำหนด ระเบียบนักศึกษาประกอบด้วย รายการข้อมูลดังนี้
 - รหัสนักศึกษา
 - ชื่อนักศึกษา
 - คะแนนสอบกลางภาค
 - คะแนนสอบปลายภาค
 - เกรด

```
struct student
{
    string id;
    string name;
    int mid;
    int final;
    char grade;
};
```

การประกาศตัวแปร

```
struct student
{
    string id;
    string name;
    int mid;
    int final;
    char grade;
};
```

```
student a;
student b,c;
student d[5];
student e[2][3];
```

การปฏิบัติงานกับตัวแปรชนิด student

```
struct student
{
    string id;
    string name;
    int mid, final;
    char grade;
};
```

student a;

a.id	a.name	a.mid	a.fianl	a.grade
------	--------	-------	---------	---------

--	--	--	--	--

```
a.id = "01";
a.name = "urai";
a.mid = 34;
a.final= 45;
a.grade='P';
```

a.id	a.name	a.mid	a.fianl	a.grade
------	--------	-------	---------	---------

01	"urai"	34	45	'P'
----	--------	----	----	-----

การปฏิบัติงานกับตัวแปรชนิด student

```
struct student
{
    string id;
    string name;
    int mid, final;
    char grade;
};
```

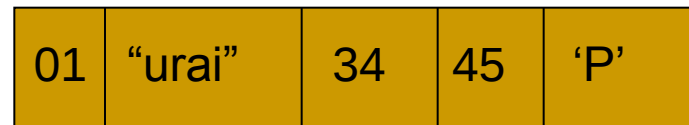
```
student d[5];
```

d[0] d[1] d[2] d[3] d[4]



```
d[4].id = "01";
d[4].name = "urai";
d[4].mid = 34;
d[4].final = 45;
d[4].grade = 'P';
```

d[4].id d[4].name d[4].mid d[4].final d[4].grade



ตัวอย่าง

■ สมมุติต้องการเก็บข้อมูลของ
นักศึกษาประกอบด้วย

□ คะแนนสอบ 3 ครั้ง

□ คะแนนเฉลี่ย

□ เกรด

สามารถกำหนดโครงสร้าง Struct
ที่ชื่อ examStats ได้ดังนี้

```
struct examStats
{
    string stuName;
    int scores[3];
    float average;
    char grade;
};
examStats aStudent;
```

ตัวอย่าง

```
struct examStats
{
    string stuName;
    int scores[3];
    float average;
    char grade;
};
examStats aStudent;
```

<u>StuName</u>	Judy		
Scores	55	90	87
Average	77.3333		
Grade	C		

กรณีศึกษา

- จงเขียนโปรแกรมรับข้อมูลลูกค้าจำนวนหนึ่งโดยข้อมูลลูกค้าแต่ละคนประกอบด้วย
 - ชื่อลูกค้า(name)
 - จำนวนเงินสะสมที่ลูกค้าซื้อสินค้า(money)
 - ชนิดของสมาชิกกรณีที่เป็น 1 แสดงว่าเป็นลูกค้าชั้นดี ถ้าเป็น 0 แสดงว่าเป็นลูกค้าปกติ (type)
- โดยเก็บไว้ในตัวแปรอาร์เรย์ชื่อ **customer**

กรณีศึกษา

INPUT:

urai	50000	1
chanin	10000	0
prayat	20000	1
somsak	15000	0

← **Sentinal** บอกว่าข้อมูลหมดแล้ว

กรณีศึกษา

จงสรุปว่ามีลูกค้าทั้งหมดกี่คน ใครเป็นลูกค้าปกติบ้างและใครเป็นลูกค้าชั้นดี อีกทั้งยังมีการสรุปยอดเงินของลูกค้าแต่ละกลุ่ม และยอดรวมของเงินสะสมที่ลูกค้าซื้อสินค้าทั้งหมดอีกด้วย

กรณีศึกษา

Number of Customer = 4

Group 0

Chanin	10000
--------	-------

Somsak	15000
--------	-------

total	25000
-------	-------

Group 1

Urai	50000
------	-------

Prayat	20000
--------	-------

total	70000
-------	-------

Total all 95000

กรณีศึกษา: ร้านขายสินค้า

ปัญหา ต้องการเก็บข้อมูลสินค้า ในร้านขายสินค้าแห่งหนึ่ง โดยข้อมูลที่ต้องการเก็บ ประกอบด้วยสินค้าที่แตกต่างกันทั้งหมด **100** ชนิด แต่ละชนิด ต้องการเก็บข้อมูลดังนี้

- ❑ รหัสสินค้า ใช้ตัวแปร (id)
- ❑ ชื่อสินค้า ใช้ตัวแปร (name)
- ❑ ราคาขาย ใช้ตัวแปร (cost)
- ❑ จำนวนสินค้าในสต็อก (instock)

กรณีศึกษา : ร้านขายสินค้า

จุดประสงค์ในการออกแบบโปรแกรม

- นำสินค้า 100 ชนิดไปเก็บในตัวแปรอาร์เรย์ของ struct ชื่อ g
- สร้างเมนูเลือกการทำงาน
 - ซื้อสินค้าเพิ่มในสต็อก
 - ขายสินค้าให้ลูกค้า
 - พิมพ์สินค้าคงคลังทั้งหมด
 - หยุดการทำงาน

โดยเลือก 1/2/3/4

โดยให้มีการออกแบบหน้าจอในลักษณะดังนี้

please choose

1. buy
2. sell
3. print
4. end

select :1/2/3/4 ?

กรณีศึกษา : ร้านขายสินค้า

การทำงานเมื่อผู้ใช้

เลือก 1 โปรแกรมจะให้ผู้ใช้งานป้อนรหัสสินค้าที่ต้องการเพิ่ม ระบบจะตรวจสอบรหัสว่าถูกต้องหรือไม่

กรณีที่ถูกต้อง โปรแกรมจะให้ผู้ใช้งานจำนวนสินค้าที่ต้องการเพิ่ม และจะทำการปรับปรุงจำนวนสินค้าให้เพิ่มขึ้น

กรณีไม่ถูกต้อง จะมี **Message** บอกถึงความผิดพลาด

กรณีศึกษา : ร้านขายสินค้า

- เลือก 2 โปรแกรมจะให้ผู้ซื้อป้อนรหัสสินค้าที่ลูกค้าซื้อ ทำนองเดียวกันระบบจะตรวจสอบความถูกต้องของรหัสสินค้าในกรณีที่ถูกต้องจะให้ผู้ซื้อป้อนจำนวนสินค้าที่ต้องการซื้อ ระบบจะตรวจสอบจำนวนสินค้าที่ลูกค้าต้องการซื้อ กับจำนวนสินค้าที่มีอยู่ในสต็อกถ้ามีสินค้าเพียงพอจะนำราคาขายมาคูณกับจำนวนเงินเพื่อคิดราคาสินค้าและตัดสต็อกสินค้า เพื่อให้จำนวนสินค้าที่มีอยู่ในสต็อกถูกต้อง ในกรณีที่เกิดความผิดพลาด เช่น ป้อนรหัสสินค้าผิด หรือ จำนวนสินค้าใน สต็อกมีไม่เพียงพอกับความต้องการของลูกค้า จะมี **Message** แจ้งถึงความผิดพลาด

กรณีศึกษา : ร้านขายสินค้า

- เลือก 3 โปรแกรมจะพิมพ์รายละเอียดของสินค้าทั้งหมดออกทางจอภาพ
- เลือก 4 จบการทำงาน

กรณีศึกษา : ร้านขายสินค้า

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
    struct good
    {
        int id;
        string name;
        double cost ;
        int instock;
    };
}
```

```
good g[3];
char ch;
int key , n ;
int i ;

for (i=0;i<3 ; i++)
{
    cout<<"id=";cin>>g[i].id;
    cout<<"name";cin>>g[i].name;
    cout<<"cost=";cin>>g[i].cost;
    cout<<"instock=";cin>>g[i].instock;
}
```

กรณีศึกษา : ร้านขายสินค้า

```
do {  
    cout << " please choose" << endl;  
    cout << "1. buy " << endl;  
    cout << "2. sell " << endl;  
    cout << "3. print " << endl;  
    cout << "4.end" << endl ;  
    cout << "select : ? " ;  
    cin>> ch;
```

```
if (ch=='1')  
{  
    cout<<"Key=" ; cin >> key;  
    for (i=0;i<3;i++)  
        if (key==g[i].id)    break;  
    if (i<3)  
    {  
        cout << " number of item to add = " ;  
        cin >> n ;  
        g[i].instock += n ;  
    }  
    else  
        cout << "not found" << endl;  
}
```

```
else if (ch == '2')
{
    cout<<"id=" ; cin >> key;
    for (i=0;i<3;i++)
        if (key==g[i].id)    break;
    if (i<3)
    {
        cout << " number of item to sell = " ;
        cin >> n ;
        if (n<g[i].instock)
        {
            g[i].instock -= n ;
            cout << "total = " << n * g[i].cost ;
        }
        else
            cout << "in stock = " << g[i].instock    << endl;
    }
    else
        cout <<"not found" << endl;
}
```

```
else if (ch == '3')
{
    for (i=0;i<3;i++)
    {
        cout << "id:" <<g[i].id <<endl;
        cout << "name :" <<g[i].name <<endl;
        cout << "cost:" << g[i].cost<<endl; ;
        cout << "instock:"<<g[i].instock<<endl ;
    }
}
}while (ch!='4');
cout << "bye bye " << endl;
return 0 ;
}
```

```
#include <iostream>
#include <string>
using namespace std;
struct good
{
    int id;
    string name;
    double cost ;
    int instock;
};

void input1(good []);
int search(const good g[] , int key);
void menu() ;
void process(good []);
void print(const good []);
void buy(good []);
void sell(good []);
```