

Chapter12：結構與自訂資料型態

一、結構 (structure)

宣告格式：

```
struct 結構名稱
{
    資料型態 欄位名稱 1;
    資料型態 欄位名稱 2;
    ...
};
```

範例：

1. 先設置結構變數，使用時再宣告並設定初始值

```
struct data /* 結構名稱 */
{
    char name[15]; /* 欄位 */
    int score; /* 欄位 */
};

struct data t = {"Vivian",100}; /* 宣告結構變數並設定初值 */
```

2. 在設置結構變數時就宣告結構變數並設定初值

```
struct data /* 結構名稱 */
{
    char name[15]; /* 欄位 */
    int score; /* 欄位 */
} t = {"Vivian",100};
/* 宣告結構變數並設定初值 */
```

注：公用的結構通常都設置成外部變數，也就是在 main 函式的外面

二、巢狀結構

範例：

```
struct date /* 結構名稱 */
{
    int month;
    int day;
};
struct newdata /* 結構名稱 */
{
    char name[15];
    struct date birthday; /* 結構變數 */
    int score;
};
int main(void)
{
    struct newdata student = {"Vivian", {7, 10}, 100};
    printf("%s's birthday is %d/%d \n",
        student.name, student.birthday.month,
        student.birthday.day);
    printf("She got %d points! \n", student.score);
    return 0;
}
```

三、結構的陣列

範例：

```
struct mydata student[10];
for(i=0; i<10; i++)
    printf("%s got %d points! \n", student[i].name,
        student[i].score);
```

四、結構的指標

範例：

```
struct data student[10]; /* 結構陣列 */
struct data *ptr = student; /* 結構指標，初值為 student 陣列的起
始位址 */
for (i = 0; i < 10; i++)
{
    printf("Name, Score:");
    scanf("%s, %d", (student+i)-> name, &(student+i)->score);
    printf("%s got %d points!\n", ptr->name, ptr->score);
    ptr++;
}
```

Q&A：

點運算子(.)和箭頭運算子(->)，這兩者之間的使用有什麼區別麼？

相同的點：兩者都是二元操作符，而且右邊的運算元都是成員的名稱。

不同的點：

點運算子(.)的左邊的運算元是一個結果為結構的表示式；

箭頭運算子(->)的左邊的運算元是一個指向結構體的指標。

五、結構與函數

範例：

```
void get_data(struct mydata *p); /* 函式原型，參數為結構指標 */
void print_data(struct mydata a); /* 函式原型，參數為結構變數 */
int main(void)
{
    struct mydata student;

    get_data(&student);
    print_data(student);
    return 0;
}
void get_data(struct mydata *p)
{
    printf("Name:");
    scanf("%s", p->name);
    printf("Score:");
    scanf("%d", &p->score);
}
void print_data(struct mydata a)
{
    printf("%s got %d !\n", a.name, a.score);
    return;
}
```

六、自訂資料型態 (typedef)

宣告格式：

typedef 資料型態 識別字;

typedef	#define
由編譯器執行	由前置處理器主導
範例： typedef int clock; clock hour, second;	範例： #define CLOCK int CLOCK hour, second;

使用 typedef 來自訂新的資料型態：

```
typedef struct
{
    int minite;
    float second;
} time;
time record = {3, 27.25};
```