Lab 08 - 結構與聯集

授課：ANT 實驗室

簡報製作：Vincent Chi < song374561@chivincent.net >

實驗一：struct

實驗目的

瞭解 struct 的意義與使用方式

實驗內容

試實作電話簿工具，需具備以下函式以完成操作

- print()：列出現在電話簿的內容
- attach()：新增資料
- remove()：移除指定資料
- update()：更新指定資料

實驗一：struct - 第一步，定義結構

```
struct 結構型態
{
    欄項資料型態 欄項變數名稱；
    欄項資料型態 欄項變數名稱；
    欄項資料型態 欄項變數名稱；
} 變數Ⅰ,變數Ⅱ..... ;
```

```
struct  Car    //定義「車」型態
{
    char  brand [10]；  //品牌
    char   id [8]；        //車牌號碼
    int   mileage；      //里程數
}；  //別忘了加分號ㄛ！
```

實驗一：struct - 第二步，實例與初始化

```
struct Car  car1 = {“Jaguar”, “ABC-1234”, 87654} ;

struct Car  car2 = { .mileage = 1324 , .brand = “Pagani” , .id = “QQQ-7777” } ;

struct Car  car3 ;

strcpy(car3.brand, "Porsche");

strcpy(car3.id, "REG-9487");

car3.mileage=9527;
```

```
struct  Car  {

   char  brand[10] ;  //品牌

   char   id[8] ;        //車牌號碼

   int  mileage ;      //里程數

} car4 =  {“Spyker”, “AAA-7894”, 0} ,  car5  ;
```

實驗一：struct - 第二步，實例與初始化

```
struct Car cars[ ] = {
    {“Jaguar”, “ABC-1234”, 87654},
    {“Pagani”, “QQQ-7777”, 1324},
    {“Porsche”, “REG-9527”, 6666}
} ;
```

```
for(int i = 0; i < 3; i++) {
   printf(“car : %s,\t%.2f\n", cars[i].brand, cars[i].id);
}
```

實驗二：搭配 typedef 使用 struct

實驗目的

學習 typedef，並且搭配 struct 使用

實驗內容

將實驗一的電話簿定義為 ContactBook 並實現以下操作

ContactBook 需支援實驗一的所有操作

ContactBook clone(ContactBook cb)：複製一份一模一樣的電話簿

bool equal(ContactBook a, ContactBook b)：比較兩個電話簿是否相同

實驗二：搭配 typedef 使用 struct

```
struct  Car {
   char  brand [10]；  //品牌
   char   id [8]；        //車牌號碼
} ;
typedef  struct Car  Auto;
```

```
typedef   struct  {
   char  brand [10]；  //品牌
   char   id [8]；        //車牌號碼
} Auto ;
```

Auto car1, car2 ;

實驗三：聯集 (Union)

實驗目的

瞭解聯集的目的與意義

瞭解聯集與 struct 的差別

實驗三：聯集 (Union)

union 聯集名稱標籤 {

資料型態 資料變數元素1 ;

資料型態 資料變數元素2 ;

…… } ;

```
union unionType
{
    char a[4];
    short b;
};
```

Union 是將不同data type 儲存在同一個記憶體空間中的自定型別
一次只會儲存一比變數資料。
上面的unionType 只會有 4Bytes 大小。

實驗三：聯集 (Union)

```
union unionType {   char a[4];    short b;};  //定義一個聯集型別 unionType

int main(void){

    union unionType u ;

    u.b=20;

    printf("sizeof=%d u.b=%d, u.a[1]=%d, u.a[0]=%d\n", sizeof(u), u.b, u.a[1], u.a[0]);

}
```

sizeof=4 u.b=20, u.a[1]=0, u.a[0]=20

作業一：PHP Shell

作業內容

試實現簡易 PHP Shell，需包含以下功能

- 變數宣告（支援整數及字串）
- echo 運算子
- == 跟 === 比較運算子實作
- exit 結束程式

參考資料

https://goo.gl/GrcLPJ - PHP Source Code