

C++基礎語法4

鄭詠堯，這是最後一次上課惹QAQ

今天課程

指標、基礎記憶體配置原理

總複習!

指標

指標可以視為是一種變數型態，但用法有點不太一樣，最大的區別在於，指標儲存的不是一個值，而是儲存一個記憶體位置aka指向某一塊記憶體。

記憶體配置

每當你宣告一個變數後，編譯器在編譯時便會分配一塊記憶體空間給該變數。而這塊記憶體自然就會有一個“位址”。而透過位址與值的搭配，C++就會開始讓人學不會(X 有些奇妙的寫法可以用ㄌO。

宣告指標變數

宣告一個指標變數的話，需要在宣告變數的名稱前面，加上一個 `*` 號就會形成該型態的指標。大家習慣會寫成 `int* x;` 以表示整數指標`x`，但要注意 `*` 字號接在 `int` 後面只是習慣，那個 `*` 字號只會影響到 `x`。

哪些是指標變數

```
int* x, y, *z;
```

哪些是指標，哪些是普通變數？

怎麼對指標賦值

或者也可以說是，如何取得一個變數的位址(因為指標只能儲存位址)。

這就要利用一個新的運算子：取址運算子`&`，用法是加在一個變數前面，就會得到該變數的位址。

像是`int* pointer = &value;`

就會使`pointer`指標儲存的址為`value`的位址

取址之後的使用

取址之後，若要使用這個位址儲存的值，用到的是 取值運算子:*

以前一頁為例 `*pointer` 跟 `value`是完全一樣的。

指標的一些運用

函式

串列

指標在函式的運用

這裡就是上禮拜有提過的 pass-by-address 傳址 做法是在宣告函式的傳入值時，使用指標變數，就會自動傳入變數的位址了，像下面那樣

```
int passByAddressFunction(int *pointer) {  
    //你的code們  
}
```

串列

串列是個很複雜的概念，基本上用途在於可以動態配置記憶體，以及降低增減元素的複雜度。

實務上來說，串列的做法就是透過struct儲存多種資料，並且除了儲存本來就要存的值以外，還另外儲存一塊指標，指向下一個元素。

實作的話這堂課上不完：)，知道概念就好惹

補充

我看你們ㄉ狀況決定講不講

陣列的記憶體配置

陣列其實指的是一塊連續的記憶體，而做[i]其實指的是陣列開始後的第i格，有點抽象，我用小畫家畫給大家看XD

位址的加減法運算

當你對一個位址做加減法運算，那電腦會根據該位址的型態，去移動記憶體格位，比如說int型態佔4個位元組，那`pointer+1`就會等同於4格後的記憶體，因此等價於`pointer[1]`，但這東西有點不重要，看看就好XD。

複習!

接下來要複習ㄟㄟ

重點整理

I/O流

變數型態

運算子

陣列/多維陣列

更多重點整理

選擇結構

重複結構

巢狀重複結構

判斷式

Struct結構

函式寫法

函式怎麼用

遞迴

傳值/傳址/傳參考

可愛又有趣ㄉ題目們

Zj d074

Zj a034

Ckeisc oj 015

你們寫完的話跟我說，我再找你題目給你們XD