

小複習

記憶體位址 與 資料型態

資料型態	資料大小	記憶體位址	value
int	4bytes	0x0000 0000 ~ 0x0000 0003	15
float	4bytes	0x0000 0004 ~ 0x0000 0007	15.03
char	1bytes	0x0000 0008	C

記憶體位址

0x0000 0000

0x0000 0004

0x0000 0008

記憶體

15

15.03

C

⋮

儲存空間
儲存單位
資料

```
int main(void){  
    char str[5] = "1234";  
    printf("%p\n", &str[0]);  
    printf("%p\n", &str[1]);  
    printf("%p\n", &str[2]);  
    printf("%p\n", &str[3]);  
    return 0;  
}
```

```
0x7ffc7a60690  
0x7ffc7a60691  
0x7ffc7a60692  
0x7ffc7a60693
```

陣列

☐ 陣列

☐ `int months[12];`

☐ `int days[]={31,29,.....};`

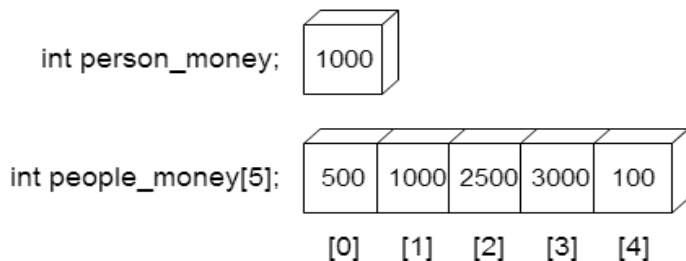
array 所佔用的是一段連續的記憶體區塊。

以 `int months[12]` 為例，其所佔用的即是 $12 * 4 \text{ bytes} = 48 \text{ bytes}$ 的連續空間

☐ 變數可以幫我們儲存一筆資料，非常方便；但是要儲存多筆資料的時候就顯得麻煩。

☐ 試想要記錄五個人在銀行的存款，若用只用變數宣告會寫成

☐ `money` 重複出現好幾次！別忘了程式就是要協助人類解決 **大量重複** 的事情，當然有更好的方法可以用。



```
int money1, money2, money3, money4, money5;
```

```
int person_money = 1000;
```

```
int people_money[5] = {500, 1000, 2500, 3000, 100};
```

字元陣列 與 字串

字串就是一串文字，在 C++ 談到字串的話，一個意義是指字元組成的陣列，最後加上一個空 (null) 字元 '\0'，例如底下是個 "hello" 字串：

```
char text[] = {'h', 'e', 'l', 'l', 'o', '\0'};
```

也可以使用 "" 來包含文字等價於上面，例如：

```
char text[] = "hello";
```

變數與記憶體空間 - char

char 字元 (1 byte = 8bits)

ASCII碼

□ 以256種變化存取對應的字元 = 2^8

Ctrl	Dec	Hex	Char	Code	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
^@	0	00		NUL	32	20		64	40	@	96	60	`
^A	1	01		SOH	33	21	!	65	41	A	97	61	a
^B	2	02		STX	34	22	"	66	42	B	98	62	b
^C	3	03		ETX	35	23	#	67	43	C	99	63	c
^D	4	04		EOT	36	24	\$	68	44	D	100	64	d
^E	5	05		ENQ	37	25	%	69	45	E	101	65	e
^F	6	06		ACK	38	26	&	70	46	F	102	66	f
^G	7	07		BEL	39	27	'	71	47	G	103	67	g
^H	8	08		BS	40	28	(72	48	H	104	68	h
^I	9	09		HT	41	29)	73	49	I	105	69	i
^J	10	0A		LF	42	2A	*	74	4A	J	106	6A	j

變數與記憶體空間 – char vs string

存取字串可使用兩種表達方式

1. char array

```
char str[5] = {1,2,3,4,\0}; // with {..., \0};
```

```
char str[5] = "1234" ;           // with "..."
```

address	0x00	0x01	0x02	0x03	0x05
value	1	2	3	4	\0

&str 會取 str 的第1個address => **&str = 0x00**

2. string

```
string str = "1234" ;
```

string 使用的是C++的**類別**的概念，所宣告的 str 為一個該類別的實例，又稱為**物件**。

所宣告出的物件可以使用已定義於該類別的資料與函式。後面上課會提到！！！！

```
ex:    int len = str.length(); // str call the function length() which is defined in class
        string
        cout << len << endl; // 4
```

變數與記憶體空間 – int

- int 整數

int 以 4 byte(32bit) 儲存

$$2^{32} = 4294967296$$

分為正負 -2147483648 ~ 2147483647

- unsigned int 無符號整數

unsigned int 也是以4byte(32bit)儲存

但沒有負值所以範圍為:0 ~ 4294967295

- 若要使用更大數字運算

long long int 為 8 byte(64bit)儲存

Another practice

Ex: 寫一個大小寫互換的程式

A sub-function (大寫轉小寫)

A sub-function (小寫轉大寫)

A sub-function (自動大小寫互換)

傳入一個字元陣列，印出大小寫互換結果

Homework

在密碼學裡面有一種很簡單的加密方式，就是把明碼的每個字元加上某一個整數 K 而得到密碼的字元（明碼及密碼字元一定都在ASCII碼中可列印的範圍內）。例如若 $K=2$ ，那麼apple經過加密後就變成crrng了。解密則是反過來做。這個問題是給你一個密碼字串，請你依照上述的解密方式輸出明碼。

hint: strlen

輸入說明:

輸入共一行，每行含有1個字串，就是需要解密的明碼

輸入後移數字 K

輸出說明

對每一測試資料，請輸出解密後的密碼。

範例輸入

abcdABC 3

Xc`p_ 2

範例輸出

defgDEF

Zebra

