

Chapter 2.

基礎 C++ 程式設計導論 II

Introduction to Basic C++ Program Design II

嘉義女中數資班資訊組課程

National Chia-Yi Girls' Senior High School Information Course

講師：陳俊安 Colten

字元與字串 char and string

- char & string
- stringstream
- ASCII

string 字串

- 字串就是一個被很多字元串接起來的東西
 - 因此在 C 中，字串的表示法為一個字元陣列
- 既然字串是一個字元陣列，那我們應該如何存取字串中某一個字元？
- 我們可以直接將字串當一般陣列使用，用索引指定字元位置

```
string s = "Hello";  
  
cout << s[0] << " " << s[2] << " " << s.size() << "\n"; // H l 5
```

ASCII Code

- 這個會輸出什麼東西？
- 為什麼字元可以被轉型成數字？依據是什麼？

```
19  
20     cout << (int)'c' << "\n"; // 99  
21
```

ASCII Code

16進制表示法：0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F、10...

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
^K	11	0B		VT	43	2B	+	75	4B	K	107	6B	k
^L	12	0C		FF	44	2C	,	76	4C	L	108	6C	l
^M	13	0D		CR	45	2D	-	77	4D	M	109	6D	m
^N	14	0E		SO	46	2E	.	78	4E	N	110	6E	n
^O	15	0F		SI	47	2F	/	79	4F	O	111	6F	o
^P	16	10		DLE	48	30	0	80	50	P	112	70	p
^Q	17	11		DC1	49	31	1	81	51	Q	113	71	q
^R	18	12		DC2	50	32	2	82	52	R	114	72	r
^S	19	13		DC3	51	33	3	83	53	S	115	73	s
^T	20	14		DC4	52	34	4	84	54	T	116	74	t
^U	21	15		NAK	53	35	5	85	55	U	117	75	u
^V	22	16		SYN	54	36	6	86	56	V	118	76	v
^W	23	17		ETB	55	37	7	87	57	W	119	77	w
^X	24	18		CAN	56	38	8	88	58	X	120	78	x
^Y	25	19		EM	57	39	9	89	59	Y	121	79	y
^Z	26	1A		SUB	58	3A	:	90	5A	Z	122	7A	z
^[27	1B		ESC	59	3B	;	91	5B	[123	7B	{
^\	28	1C		FS	60	3C	<	92	5C	\	124	7C	
^]	29	1D		GS	61	3D	=	93	5D]	125	7D	}
^^	30	1E	▲	RS	62	3E	>	94	5E	^	126	7E	~
^-	31	1F	▼	US	63	3F	?	95	5F	_	127	7F	␣*

* ASCII 碼 127 具有代碼 DEL。在 MS-DOS 下，這個代碼與 ASCII 8 (BS) 的效果相同。DEL 代碼可以由 CTRL + BKSP 鍵產生。

ASCII Code

- 所以當你拿字元做加減乘除的運算的時候
- 程式會自動幫你把字元轉成相對應的 ASCII Code 整數才做運算

ASCII Code

- 如果我現在想判斷一個數字有沒有某一個位數出現 2 或 3
 - 出現的話要把這一個位數 + x ($1 \leq x \leq 5$)
- 但我這個數字大到 10^{100} , 使用 C++ 該怎麼做？
 - Example : 12345 , x = 5
 - Answer : 17845
- Hint: 可以掃過整個字串

ASCII Code

- 把整個字串掃過一次，碰到 2 or 3 的時候就直接 +x
 - 字元 + 整數 x = 字元的 ASCII + x
 - 然後由於我們是存到字元裡面，所以新的字元的 ASCII Code 會等於 ASCII + x ，程式會自動幫你轉回字元型態

```
20
21     int x;
22     string s;
23     cin >> s >> x;
24
25     for(int i=0;i<s.size();i++)
26     {
27         if( s[i] == '2' || s[i] == '3' ) s[i] += x;
28     }
29
30     cout << s << "\n";
```


stringstream

- `#include <sstream>`
- 一種處理字元的好工具
- 兩大功能
 - 切割字串
 - 型態轉型

stringstream 切割字串

- 一般陣列的題目都會先跟你說他會輸入幾個
- 但如果你遇到的題目一開始不會告訴你他要輸入幾個怎麼辦？
- 這個時候我們就只能先把整段字串先讀取進來了！

stringstream 切割字串

- stringstream 是一個工具，使用前必須先宣告

```
stringstream name;
```

stringstream 切割字串

- 接下來有兩種操作，假設我的 stringstream 的名字是 ss
 - `ss << 變數; // 將變數放入 ss 裡面`
 - `ss >> 變數; // 將 ss 裡面的東西放入到變數`

stringstream 切割字串

- 現在有一個字串 100 200 300，我想把他切成三份
- 每一份都是一個整數，依序存入 `a[0],a[1],a[2]`
- 如何用 `stringstream` 達成？
- 我們來解析一下吧！

```
21     string s;  
22     getline(cin,s);  
23  
24     stringstream ss;  
25  
26     ss << s;  
27  
28     int number,a[3],idx = 0;  
29  
30     while( ss >> number )  
31     {  
32         a[idx] = number;  
33         idx++;  
34     }
```

getline(cin,s)

- 由於我們必須使用字串讀入，字串含有空白字元
- 所以我們必須使用 getline

```
21     string s;  
22     getline(cin,s);  
23  
24     stringstream ss;  
25  
26     ss << s;  
27  
28     int number,a[3],idx = 0;  
29  
30     while( ss >> number )  
31     {  
32         a[idx] = number;  
33         idx++;  
34     }
```

stringstream ss; & ss << s;

- 宣告 stringstream
- 然後把我們要處理的字串丟入到這一個 stringstream 裡面

```
20
21     string s;
22     getline(cin,s);
23
24     stringstream ss;
25
26     ss << s;
27
28     int number,a[3],idx = 0;
29
30     while( ss >> number )
31     {
32         a[idx] = number;
33         idx++;
34     }
```

int number , a[3] , idx = 0;

- 宣告一個整數的 number，用來存 stringstream 切下來的東西
 - 因為我們切下來的東西是整數，所以 number 要是整數
- 宣告 a[3] 與 idx 來存取資料

```
21     string s;  
22     getline(cin,s);  
23  
24     stringstream ss;  
25  
26     ss << s;  
27  
28     int number,a[3],idx = 0;  
29  
30     while( ss >> number )  
31     {  
32         a[idx] = number;  
33         idx++;  
34     }
```


while(ss >> number)

- stringstream 會依序把東西切下來然後放入 number
- 使用 while 迴圈的目的是：當沒有切到東西的時候才會停止

```
20
21     string s;
22     getline(cin,s);
23
24     stringstream ss;
25
26     ss << s;
27
28     int number,a[3],idx = 0;
29
30     while( ss >> number )
31     {
32         a[idx] = number;
33         idx++;
34     }
```

大功告成！

- 這麼一來我們的 $a[0] = 100$, $a[1] = 200$, $a[2] = 300$ 了！
- 都是整數型態的哦！

```
21     string s;  
22     getline(cin,s);  
23  
24     stringstream ss;  
25  
26     ss << s;  
27  
28     int number,a[3],idx = 0;  
29  
30     while( ss >> number )  
31     {  
32         a[idx] = number;  
33         idx++;  
34     }
```