

String 文字處理

string

1

String文字/字符串

1.宣告: char s1[10];
char *s2;
char s3[] = "Chan Tai Man";
 char s4[20] = "Chan Siu Ming";
char s5[] = {'H','e','l','l','o','\0'};

	0	1	2	3	4	5	6	7	8	9	10	11	12
s3	C	h	a	n		T	a	i	\0	M	a	n	\0

結束
字符

改變字元

```
printf ("%s\n", s3);  
s3[8]='\0';  
printf ("%s\n", s3);
```



string

2

2.輸出文字

```
char s[]="Game Over\n";  
printf (s);
```

```
 printf ("Game Over\n");  
printf ("Game"  
      " "  
      "Over"  
      "\n");    // 不用逗號
```

```
 printf ("%s\n", s);
```

```
_____ (stdout, "%s", s);  
_____ (s, stdout);
```

string

3

3.輸入文字

沒有
&

• scanf ("%s", s); 見空白即停

```
 _____ (s); // 會否混亂??
```

• scanf ("%s%s", firstname, lastname);

```
 _____ (s, 20, stdin);  
_____ (stdin, "%s", s);
```

keyboard

string

4

4.比較文字

↓ 'c' - 'C' = 99 - 67 ↓

```
char s[10]="chan tm", t[10]="Chan TM";
```

- if (strcmp(s,t) ___)
 printf ("%s=%s\n", s, t);
- if (strcmp(s,t) ___)
 printf ("%s<%s\n", s, t);
- if (strcmp(s,t) ___)
 printf ("%s>%s\n", s, t);
- n = strcmp(s,t);
0 // ignore cases ('A' = 'a')

32	' '
48	'0'
49	'1'
64	'@'
65	'A'
66	'B'
67	'C'
97	'a'
98	'b'
99	'c'

5

string

4.比較文字

```
int stringCompare (const char *s1, const char *s2){  
    while(*s1==*s2){  
        if(*s1=='\0' && *s2=='\0') return 0;  
        s1++;  
        s2++;  
    }  
    if(*s1>*s2) return 1;  
    return -1;  
}
```

```
main(){  
    char s[10]="chan tm", t[10]="Chan TM";  
    n = stringCompare (s, t);  
}
```

string

6

5.文字檔

```
_____ *fp, *fp2;           // 宣告 file pointers
// 開啓檔案
fp = _____("abc.txt", "r"); // read
fp2 = fopen ("xyz.txt", _____ // write

while (! _____ ){           // 只要未到檔案末
    _____(s, 20, fp);       // 讀入一行
    printf ("%s", s);
    _____ (fp2, "%s", s); // 寫入
}
// 關閉檔案
_____ (fp);
fclose (fp2);
```

string

7

5.文字檔

輸出/寫入

```
_____ (fp, "%s", s);
fputs (s, fp);
```

輸入/讀取

```
_____ (s, 10, fp);
// 讀取一行(或最多10char)
fscanf (fp, "%s", s);
```

```
Chan TM
Chan Tai Man
...
```

```
_____ (fp, "%i", _____
```

```
50 70 90 60
77 88 66 55
...
```

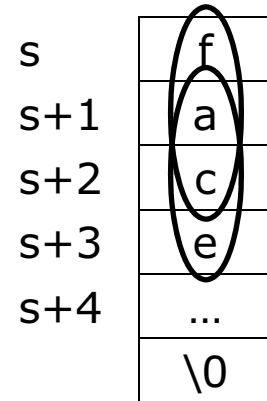
string

8

6.常用函數

```
char s[20]="face...", t[20];
```

```
n=_____ (s);
// string length 字串字長
```



s = "face";
 _____ (s, "face");

// illegal 不合法
 // string copy 複製字串

部分複製

```
_____ (t, s, 3);
strncpy (t, s+1, 3);
```

"fac"
"ace"

```
t = s[0]+s[1]+s[2]
```

```
t = s[1]+s[2]+s[3]
```

string

9

6.常用函數(續)

```
char s[20], t[20];
```

if (s=="face").. // illegal 不合法
 if (_____ (s, "face") ..) // 比較字串

```
if (strstr (s, ".txt")==NULL) // 字串s不包含".txt"
    strcat (s, ".txt"); // string concatenate 連接
```

```
char s[]="A123456(7)";
```

```
printf("%s", ...);
s
s+1
s+2
...
```

```
printf("%c", ...);
s[0]
s[1]
...
```

```
printf("%i", ...);
s[0]
s[1]
...
```

string

10

其他: `char *p, key;`

找尋
對象

```
p = str_____ ("abc.txt html jpg", ' ');  
printf ("%s", p);
```

```
p = strrchr ("abc.txt html jpg", ' ');  
printf ("%s", p);
```

```
if (str_____ ("abc.txt html jpg", "txt"))  
    printf ("txt找到了!\n");
```

```
printf ("Q: [Y]es [N]o [C]ancel ");
```

```
do{
```

```
    key = toupper (getch());
```

只可輸入YNC

```
}while (_____ ("YNC", key) == NULL);
```

string

11

```
char name[40][20];  
strcpy(name[0],"chan tai man"); // string copy
```

```
gets(name[2]); scanf ("%s",name[2]);
```

```
for (i=0;i<40;i++)  
    _____(name[i]); // 輸入40個名字
```

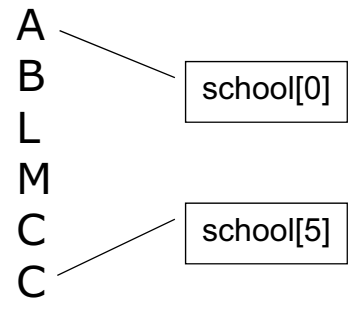
```
for (i=0;i<40;i++)  
    _____(name[i]); // 輸出40個名字
```

```
for (i=0;i<40;i++)  
    puts (name_____); // 輸出40個名字
```

string

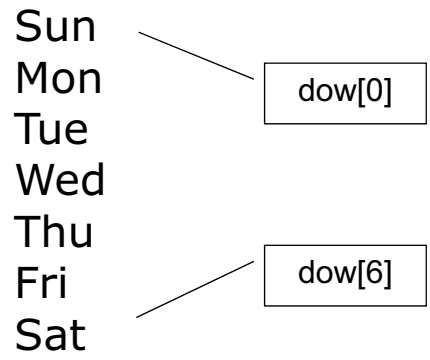
12

```
char school[7] = "ABLMCC";
for (i=0; i<____(school); i++)
    printf ("%__\n", school____);
```



```
char dow____[4]
={"Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat"}
```

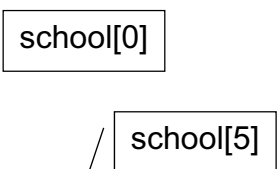
```
for (i=0; i<7; i++)
    printf ("%__\n", dow____);
```



string

```
char school[7] = "ABLMCC";
```

文字"..."

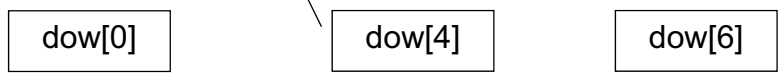


包含6個字元
'A', ..., 'C'

```
char dow[7][4]
={"Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat"};
```

文字陣列

字元
dow[6][0]
dow[6][1]
dow[6][2]



文字 文字 文字

string

```
char s[13] = "chan tai man";
```

```
s = toupper (s);          s[0] = toupper (s[0]);  
_____ = toupper (_____);
```

```
...  
s[10] = toupper (s[10]);  
s[11] = toupper (s[11]);
```

(字元)

```
for (i=0; i<12; i++)  
    _____ = toupper (_____);
```

```
printf ("%s\n", s);
```

CHAN TAI MAN

```
printf ("%s\n", _____(s) ); // upper大寫  
printf ("%s\n", _____(s) ); // lower小寫  
printf ("%s\n", _____(s) ); // reverse倒轉
```

string

15

```
char cno[5] = "1A01", t[3] = "99";  
gets (cno);
```

```
ok=0;  
if (_____ (cno) == 4){  
    ok=1;  
    if (    cno[0] < '1'  
        cno[1]  
        cno[2]  
        ) ok=0;  
  
    if (ok == 1){  
        strncpy(t, _____);  
        n = atoi(t); // 文字轉數字  
        if (_____ ) ok=0;  
    }  
}
```

格式
檢查

```
if (ok == 0) printf("Error\n");
```

string

16


```
char idno[11] = "A123456(7)";  
gets (idno);
```

格式
檢查

```
ok=0;  
if ( _____ ==10  
    && idno[0]>= _____ && idno[0]<= _____ ){  
    ok=1;  
    for (i=1; _____; i++)  
        if ( _____ ) ok=0;  
  
    if ( ! ( _____ || _____ ) )  
        ok=0;  
}
```

```
if (ok==0) printf("Error\n");
```

string

17

```
char idno[11] = "A123456(7)";  
gets (idno);
```

檢查數位總和

```
sum=58*9 + ( _____ ) * 8 ;  
    sum += (idno[1] _____) * 7;  
    sum += (idno[2] _____) * 6;  
    ...  
    sum += (idno[5] _____) * 3;  
    sum += (idno[6] _____) * 2;  
for (i=1; i<7; i++)  
    sum +=  
  
if (idno[8]=='A')  
    else  
  
if (sum _____) printf ("Valid\n");  
    else printf ("Invalid\n");
```

```
printf("%s", ...);  
idno  
idno+1  
idno+2  
...
```

```
printf("%c", ...);  
idno[0]  
idno[1]  
...
```

```
printf("%i", ...);  
idno[0]  
idno[1]  
...
```

string

18

7. 程式編寫練習: (a)
 Q: Enter text: Chan Tai Man
 Q: n=? 2
 A: s[2] = 'a', "Ch"
 :

(b)
 Q: Enter text: Chan Tai Man
 A: 12 char, 4 vowels, 2 spaces
 :

s[2]='\\0';

Q: Enter text: ↵
 A: Bye

Chan Tai Man
 han Tai Man
 an Tai Man
 n Tai Man
 ...

C
 H
 A
 N
 ...

C
 H
 A
 N
 ...

(c)
 Q: Enter text: Chan Tai Man
 A: ChanTaiMan
 :
 Q: Enter text: ↵
 A: Bye

string

19

8. 程式編寫練習:

(a)
 Q: Enter Name: Chan
 A: You are Chan
 :
 Q: Enter Name: QUIT
 A: Bye

(c)
 Q: Enter alphabet: D
 No.of letters: 5
 A: DEFGH

(b)
 Q: Enter password: 1314
 A: Incorrect!
 :
 Q: Enter password: axe
 A: correct!

(d)
 n = posn ("banana", "ana");
 isSym = symmetric ("ana");
 isSym = symmetric ("bana");
 palindrome ("abc", s);
 palindrome ("cba", s);
 n = anagram ("act", "cat");

string

20

(1) 文字 String 測驗 <http://www.ablmc.edu.hk/~scy/home/javascript/strncpy.swf>

(0)	char name[30]="Chan_Tai_Man";	// 來源 source
	strcpy (name, "Chan Tai Man"); name = "Chan_Tai_Man";	// 文字複製 // 錯誤!
(1)	int p = 5 , n = 3 ;	// 起點(p)、字元數目(n)
(2)	char ch = name [p];	_____ // 抽取第p個字元(字符)
(3)	char dst[30]; 1 strcpy (dst, name); 2 strncpy (dst, name, n); 3 strncpy (dst, name+p, n);	// 目的地 destination // 文字複製 string copy // 部分複製 dst[] ← _____ // dst[] ← _____
(4)	char str[]="520"; strcat (str,dst); x = strlen (str);	// 文字合併 concatenate (str ← str+dst) // str[] ← _____ // x ← _____

(2) 複製部分字串 strncpy(target目標, source來源, length長度);

```
char t[20]; // target
char s[20] = "a1b3c5d7e90p2q4r6s8t";
先選取 上面 ↑ 部分字串，再按 確定
```

(3) 複製部分字串 strncpy(target目標, source來源, length長度);

```
char t[20] = "xyzopqr";
char s[11] = "a1b3c5d7e9";
strncpy (t+1 , s+4 , 4); t ← _____
```

(4) Relational Operators & String Comparison 測驗 strcmp.swf

```
string 文字 compare 比較: '0'=48, 'A'=65, 'B'=66, 'a'=97, 'b'=98
n = strcmp ("ab","bc"); → _____
n = strcmp ("ab","ab"); → _____
n = strcmp ("ba","ab"); → _____
```

(5) input: "abs", "Abs", "abst", "ab", "cab"

		char input[10]; //string文字	char grade; //單字/字符
等於	==	if(strcmp(input, "abs") ==0) ...	if(grade == 'A') ...
不等於	!=	if(strcmp(input, "abs") !=0) ...	if(grade != 'U') ...
大於	>	if(strcmp(input, "abs") >0) ...	if(grade > 'C') ...
小於	<	if(strcmp(input, "abs") <0) ...	if(grade < 'F') ...

請填上 (==, !=, <, >)

"abs"	"Abs"	"abst"	"ab"	"cab"

A. 一般變數

類別	變數名稱	宣告句子	輸入句子
1 文字	name	char name[10]="A123";	gets (_____);
2 字符	grade	char grade = 'A';	scanf ("____", &grade);
3 整數	mark	int mark =100;	scanf ("____", &mark);
4 小數	average	float average;	scanf ("____", &average);

B. 陣列

類別	變數名稱	宣告陣列	for (i=0; i<100; i++)
1 文字	name	char names[100][10];	gets(_____);
2 字符	grade	char grades[100];	scanf("%c", &grades__);
3 整數	mark	int marks[100];	scanf("%i ", &marks__);
4 小數	average	float avgs[100];	scanf("%f", &avgs__);

C. 文字 : char name[20] = "Chan Tai Man", t[20];

1	字長 strlen	n = strlen(name);
2	複製 strcpy	strcpy (t, name);
3	複製 strncpy	strncpy (t, name+__, 4); // "an T"
4	比較 strcmp	n = strcmp (t, _____);
5	單字/字符	name[0], name[1], ..., name[____]
6	轉大寫	for (i=0; i<strlen(name); i++) name_____ = toupper(name_____);
7	輸出	printf ("%s", name); for (i=0; i<strlen(name); i++) printf ("____", name_____);

D. 文字陣列 : char names[100][10] = {"Chan KK", "Chan YY", "Chan 7"};

		for (i=0; i<100; i++){
1	字長 strlen	n = strlen(names_____);
2	複製 strcpy	strcpy (t, names_____);
3	複製 strncpy	strncpy (t, names_____, 4);
4	比較 strcmp	n = strcmp (t, names_____);
5	單字/字符	names_____[2]='e'; names_____[4]=',';
		}
6	轉大寫 names[0]	for (i=0; i<strlen(names[0]); i++) names_____ = toupper(names_____);
7	輸出文字	printf ("____", names[0]);
8	輸出	for (i=0; i<strlen(names[0]); i++) printf ("____", names[0]__);