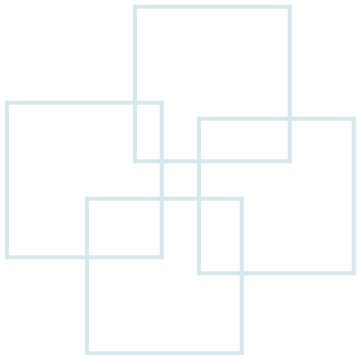


Chapter 4

Formatted Input and

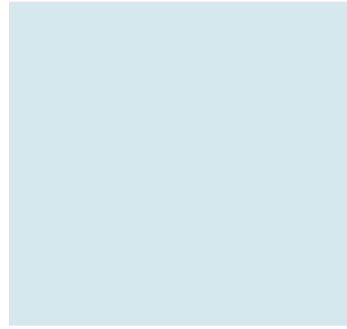
Output



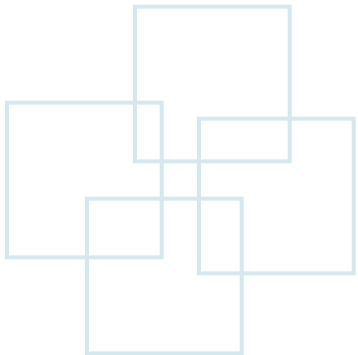


Outline

- Input and output functions
 - printf(), scanf()
- Character input and output functions
 - getchar(), putchar(), getch(), getche()



Input and Output Function printf()





Input Function printf()

- *printf* is composed of *print* and *format*.

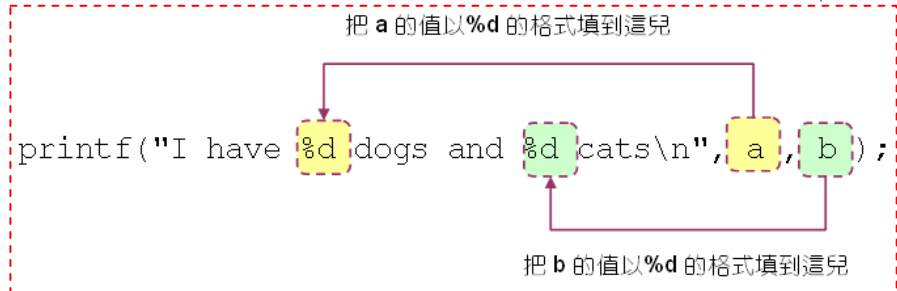
- **Format:**

Format of printf()

```
printf("FormattedString", item1, item2, ...);
```

- **Example:**

```
01
02 #include <stdio.h>
03 #include <stdlib.h>
04 int main(void)
05 {
06     int a=2;
07     int b=4;
08     printf("I have %d dogs and %d cats\n", a, b); /* 呼叫 printf() 函數 */
09
10     system("pause");
11     return 0;
12 }
```



/* OUTPUT -----

I have 2 dogs and 4 cats

-----*/



Input Function printf() (Cont.)

```
01  /* prog4_2, 印出字串 */
02  #include <stdio.h>
03  #include <stdlib.h>
04  int main(void)
05  {
06      printf("Have a nice day!!\n");      /* 印出字串內容 */
07
08      system("pause");
09      return 0;
10  }
```

/* OUTPUT--

Have a nice day!!

-----*/



Format of printf()

Format	Description	Format	Description
%c	Character	%%	Print %
%d	Decimal integer	%o	Unsigned octal integer
%ld	Long integer	%s	String
%e	Floating-point with e format	%u	Unsigned decimal integer
%f	Floating-point with decimal-point format	%x	Unsigned hexadecimal integer
%lf	Double with decimal-point format		



Escape Sequence

跳脫序列	功能	跳脫序列	功能
\a	警告音	\"	印出雙引號
\b	倒退	\\	印出反斜線
\n	換行	\/	印出斜線
\r	歸位		
\t	跳格		
\'	印出單引號		



Application of Escape Sequence and Formation

```

01  /* prog4_3, 使用 printf() 函數 */
02  #include <stdio.h>
03  #include <stdlib.h>
04  int main(void)
05  {
06      int num=25;
07      printf("\'%d%%的學生來自小康家庭'\n", num); /* 印出字串 */
08
09      system("pause");
10      return 0;
11  }

```

/* OUTPUT---

"25%的學生來自小康家庭"

-----*/

格式碼，用來印出整數值 跳脫序列，用來印出雙引號

printf("\' %d %%% 的學生來自小康家庭 \' \' \n ", num);

格式碼，用來印出百分比符號 跳脫序列，用來進行換行

跳脫序列，用來印出雙引號



Output Width

```

01  /* prog4_4, 印出特定格式 */
02  #include <stdio.h>
03  #include <stdlib.h>
04  int main(void)
05  {
06      int num1=32, num2=1024;
07      float num3=12.3478f;
08
09      printf("num1=%6d 公里\n", num1); /* 以「%6d」格式印出 num1 */
10      printf("num2=%-6d 公里\n", num2); /* 以「%-6d」格式印出 num2 */
11      printf("num3=%6.2f 英哩\n", num3); /* 以「%6.2f」格式印出 num3 */
12
13      system("pause");
14      return 0;
15  }

```

num1 = 32 公里

%6d, 佔 6 格, 靠右對齊

num2 = 1024 公里

%-6d, 佔 6 格, 靠左對齊

num3 = 12.35 英哩

%6.2f, 佔 6 格, 靠右對齊

/* OUTPUT---

```

num1=      32 公里
num2=1024  公里
num3= 12.35 英哩

```

---*/



Modifiers

修飾子	功能	舉例
-	靠左對齊	%-3d
+	將數值的正負號顯示出來	%+5d
空白	數值為正值時，留一格空白；為負值時，顯示負號	% 6f
0	將固定欄位長度的數值前空白處填上 0（與負號「-」同時使用時，此功能無效）	%07.2f

資料內容 格式 執行結果

12345 %10d

					1	2	3	4	5
--	--	--	--	--	---	---	---	---	---

12345 %+d

+	1	2	3	4	5
---	---	---	---	---	---

12345 %09d

0	0	0	0	1	2	3	4	5
---	---	---	---	---	---	---	---	---

12345 %-10d

1	2	3	4	5					
---	---	---	---	---	--	--	--	--	--

資料內容 格式 執行結果

12345 % d

	1	2	3	4	5
--	---	---	---	---	---

123.456 %7.2f

	1	2	3	.	4	6
--	---	---	---	---	---	---

123.456 %010.3f

0	0	0	1	2	3	.	4	5	6
---	---	---	---	---	---	---	---	---	---

123.456 %+10.4f

	+	1	2	3	.	4	5	6	0
--	---	---	---	---	---	---	---	---	---



Modifiers (Cont.)

```
01
02  #include <stdio.h>
03  #include <stdlib.h>
04  int main(void)
05  {
06      int i=1234;
07      printf("i=%+08d\n",i);      /* 呼叫 printf() 函數 */
08
09      system("pause");
10      return 0;
11  }
```

```
/* OUTPUT--
```

```
i=+0001234
```

```
-----*/
```



Octal and Hexadecimal Formats

```
01
02  #include <stdio.h>
03  #include <stdlib.h>
04  int main(void)
05  {
06      printf("42 的八進位是 %o\n", 42);          /* 印出 42 的八進位 */
07      printf("42 的十六進位是 %x\n", 42);       /* 印出 42 的十六進位 */
08
09      system("pause");
10      return 0;
11  }
```

/* OUTPUT--

42 的八進位是 52

42 的十六進位是 2a

-----*/



Data Type vs. Output Format

```

01
02  #include <stdio.h>
03  #include <stdlib.h>
04  int main(void)
05  {
06      int a=15;
07
08      printf("a=%d\n", a);
09      printf("以浮點數型態印出: %f\n", a);
10      printf("以指數型態印出 : %e\n", a);
11
12      system("pause");
13      return 0;
14  }

```

Inconsistent
format

/* 宣告整數變數 a，並設值為 15 */

/* 印出 a 的值 */

/* 以%f 格式碼印出 a 的值 */

/* 以%e 格式碼印出 a 的值 */

/* **OUTPUT**-----

a=15

以浮點數型態印出: 0.000000

以指數型態印出 : 1.910519e-297

-----*/

Incorrect
value



Data Type vs. Output Format (Cont.)

```

01
02  #include <stdio.h>
03  #include <stdlib.h>
04  int main(void)
05  {
06      int a=15;                /* 宣告整數變數 a，並設值為 15 */
07
08      printf("a=%d\n", a);    /* 印出 a 的值 */
09      printf("以浮點數型態印出: %f\n", (float) a); /* 以浮點數型態印出 a */
10      printf("以指數型態印出: %e\n", (double) a); /* 以指數型態印出 a */
11
12      system("pause");
13      return 0;
14  }

```

Type
conversion

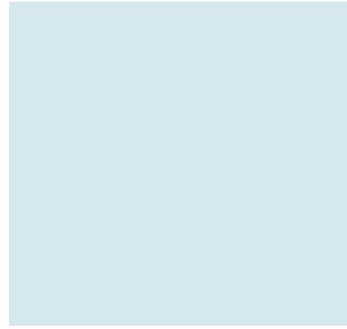
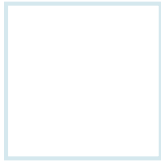
/* OUTPUT-----

a=15

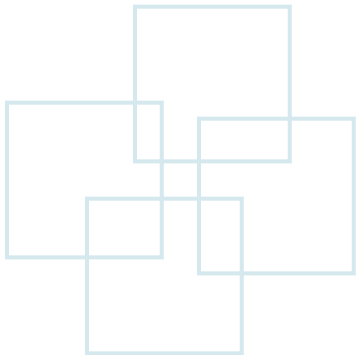
以浮點數型態印出: 15.000000

以指數型態印出: 1.500000e+001

-----*/



Input and Output Functions `scanf()`





Output Function scanf()

- scanf() is used for the input of characters, numbers, or strings.

Format of scanf()

```
scanf("FormattedString", &Var1, &Var2, ...);
```

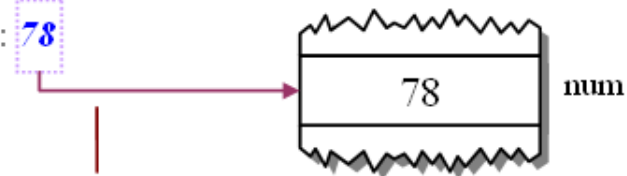
```

01
02 #include <stdio.h>
03 #include <stdlib.h>
04 int main(void)
05 {
06     int num;
07
08     printf("請輸入一個整數:");
09     scanf("%d",&num);
10     printf("num=%d\n",num);
11
12     system("pause");
13     return 0;
14 }
```

```
scanf("%d",&num);
```

請輸入一個整數: 78

Address of the variable



將數值 78 寫到變數 num 裡

/* 由鍵盤輸入整數，並指定給 num 存放 */

/* 印出 num 的內容 */

/* OUTPUT ---

請輸入一個整數: 78

num=78



Output Function scanf() (Cont.)

```

01
02 #include <stdio.h>
03 #include <stdlib.h>
04 int main(void)
05 {
06     int a,b;
07
08     printf("請輸入兩個整數: ");
09     scanf("%d %d",&a,&b);
10     printf("%d+%d=%d\n",a,b,a+b);
11
12     system("pause");
13     return 0;
14 }

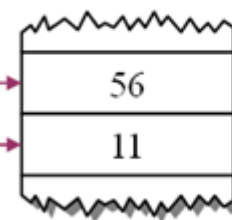
```

```
scanf("%d %d",&a,&b);
```

請輸入兩個整數:

56

11



a

b

將數值 56 與 11 分別寫到變數 a 與 b 裡

```
/* OUTPUT--
```

請輸入兩個整數: 56 11

56+11=67

```
-----*/
```



Output Function scanf() (Cont.)

```
01  /* prog4_11, 使用逗號區隔輸入格式 */
02  #include <stdio.h>
03  #include <stdlib.h>
04  int main(void)
05  {
06      int a,b;
07
08      printf("請輸入兩個整數，請用逗號隔開數值： ");
09      scanf("%d,%d",&a,&b);          /* 以「,」隔開兩個輸入格式碼 */
10      printf("%d+%d=%d\n",a,b,a+b); /* 計算總和並印出內容 */
11
12      system("pause");
13      return 0;
14  }
```

/* OUTPUT-----

請輸入兩個整數，請用逗號隔開數值： **14,36**

14+36=50

-----*



Format of scanf()

輸入格式	輸入敘述	輸入格式	輸入敘述
%c	字元	%s	字串
%d	十進位整數	%o	八進位整數
%f	浮點數	%x	十六進位整數
%lf	倍精度浮點數 (注意%lf 裡的 l 是英文小寫字母 l)		



Format of scanf() (Cont.)

```
01
02 #include <stdio.h>
03 #include <stdlib.h>
04 int main(void)
05 {
06     int num;
07
08     printf("請輸入十六進位的整數：");
09     scanf("%x",&num);          /* 輸入十六進位數值，並指定給變數 num */
10     printf("%x 的十進位為%d\n",num,num); /* 將十六進位數值以十進位印出 */
11
12     system("pause");
13     return 0;
14 }
```

/* OUTPUT----

請輸入十六進位的整數：**12ab**

12ab 的十進位為 4779

-----*/



Character Handling of scanf()

```
01
02  #include <stdio.h>
03  #include <stdlib.h>
04  int main(void)
05  {
06      char ch;
07
08      printf("Input a character:");
09      scanf("%c",&ch);      /* 由鍵盤輸入字元並指定給變數 ch */
10      printf("ch=%c, ascii code is %d\n",ch,ch);
11      system("pause");
12      return 0;
13  }
```

/* OUTPUT----

Input a character: **R** → 先輸入一個空白鍵再輸入 R
ch= , ascii code is 32

-----*/



Character Handling of scanf() (Cont.)

```
01
02  #include <stdio.h>
03  #include <stdlib.h>
04  int main(void)
05  {
06      char ch;
07      printf("Input a character:");
08      scanf("%c", &ch); /* 這裡輸入一個空格 (The leading zeros could be ignored)
09                          /* 由鍵盤輸入字元並指定給變數 ch */
10      printf("ch=%c, ascii code is %d\n", ch, ch);
11
12      system("pause");
13      return 0;
14  }
```

/* OUTPUT---

Input a character: **R** → 先輸入一個空白鍵再輸入 R
ch=R, ascii code is 82

-----*/

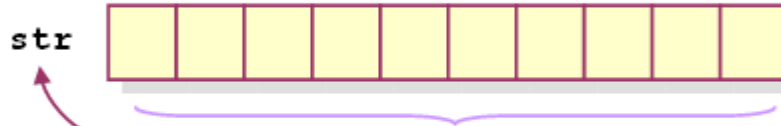


String Handling of scanf()

Declaration of String

```
char StrVar[StrLength];
```

```
char str[10];
```



可容納10個字元的陣列

```
scanf("%s", str);
```

& is not needed

Read string from keyboard and store the string into the "str" array



String Handling of scanf() (Cont.)

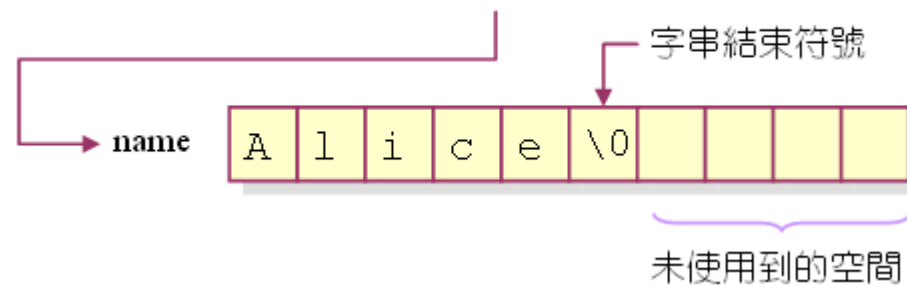
- 字串輸入的範例：

```

01
02 #include <stdio.h>
03 #include <stdlib.h>
04 int main(void)
05 {
06     char name[10];          /* 宣告字元陣列 */
07
08     printf("What's your name: ");
09     scanf("%s", name);     /* 輸入字串，並由字元陣列 name 所接收 */
10     printf("Hi, %s, How are you?\n", name); /* 印出字串的內容 */
11     system("pause");
12     return 0;
13 }

```

What's your name: *Alice*



/* OUTPUT----

What's your name: *Alice*

Hi, Alice, How are you?

-----*/

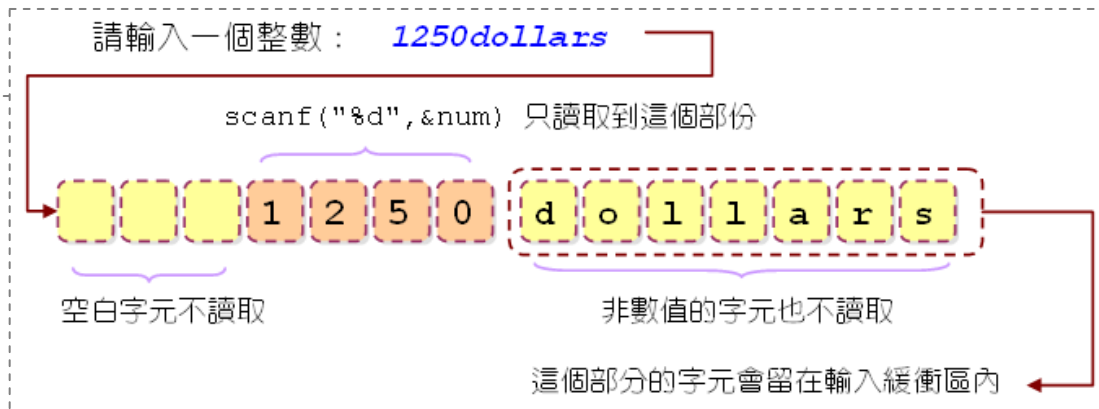


Common Problems of Using scanf()

```

01
02 #include <stdio.h>
03 #include <stdlib.h>
04 int main(void)
05 {
06     int num;
07
08     printf("請輸入一個整數：");
09     scanf("%d",&num);          /* 輸入整數並設定給變數 num 存放 */
10     printf("num=%d\n",num);    /* 印出 num 的值 */
11
12     system("pause");
13     return 0;
14 }

```



/* OUTPUT-----

請輸入一個整數： 1250dollars → 先輸入三個空白，再輸入 1250dollars
num=1250

-----*/



Common Problems of Using scanf() (Cont.)

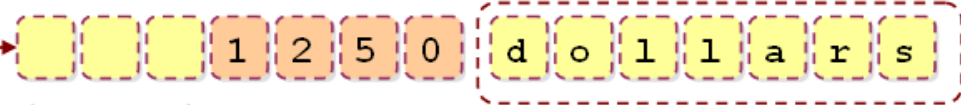
```

01
02 #include <stdio.h>
03 #include <stdlib.h>
04 int main(void)
05 {
06     int num;
07     char str[10];
08     printf("請輸入一個整數：");
09     scanf("%d",&num);
10     printf("num=%d\n", num);
11     printf("請輸入一個字串：");
12     scanf("%s",str);          /* 輸入字串 */
13     printf("str=%s\n",str);  /* 印出字串的內容 */
14     system("pause");
15     return 0;
16 }

```

請輸入一個整數： 1250dollars

scanf("%d",&num) 只讀取到這個部份



空白字元不讀取

scanf("%s",str) 會從緩衝區內讀取到這個部分

/* OUTPUT

請輸入一個整數： 1250dollars

num=1250

請輸入一個字串：str=dollars

先輸入三個空白，再輸入 1250dollars

*/

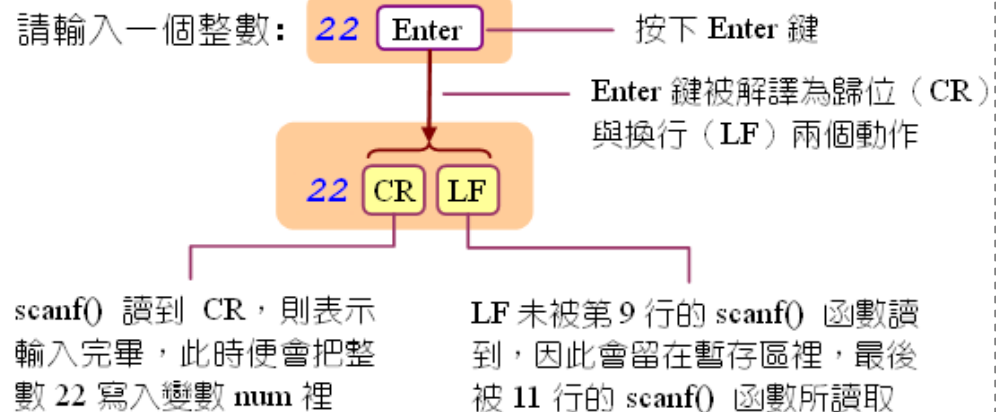


Common Errors of Using scanf()

```

01
02 #include <stdio.h>
03 #include <stdlib.h>
04 int main(void)
05 {
06     int num;
07     char ch;
08     printf("請輸入一個整數: ");
09     scanf("%d",&num); /* 由鍵盤輸入整數，並指定給變數 num */
10     printf("請輸入一個字元: ");
11     scanf("%c",&ch); /* 由鍵盤輸入字元，並指定給變數 ch */
12     printf("num=%d, ascii of ch=%d\n", num, ch); /* 印出 num 與 ch 的 ascii 碼 */
13     system("pause");
14     return 0;
15 }

```



/* OUTPUT -----

請輸入一個整數: 22
 請輸入一個字元: num=22, ascii of ch=10

***/**



Common Errors of Using scanf() (Cont.)

```
01
02 #include <stdio.h>
03 #include <stdlib.h>
04 int main(void)
05 {
06     int num;
07     char ch;
08
09     printf("請輸入一個整數: ");
10     scanf("%d",&num); /* 由鍵盤輸入整數，並指定給變數 num */
11     printf("請輸入一個字元: ");
12     scanf("%c",&ch); /* 由鍵盤輸入字元，並指定給變數 ch */
13     printf("num=%d, ascii of ch=%d\n",num,ch); /* 印出 num 與 ch 的 ascii 碼 */
14     system("pause");
15     return 0;
16 }
```

/* OUTPUT----

請輸入一個整數: 22
請輸入一個字元: k
num=22, ascii of ch=107
-----*/



Flushing of Input Buffer

- `fflush()` could clear data in the input buffer

Format of `fflush()`

```
fflush(stdin);    /* Clear the buffer */
```



Flushing of Input Buffer (Cont.)

```
01
02 #include <stdio.h>
03 #include <stdlib.h>
04 int main(void)
05 {
06     int num;
07     char ch;
08
09     printf("請輸入一個整數: ");
10     scanf("%d", &num);
11     fflush(stdin);          /* 清空緩衝區內的資料 */
12     printf("請輸入一個字元: ");
13     scanf("%c", &ch);
14     printf("num=%d, ascii of ch=%d\n", num, ch);
15     system("pause");
16     return 0;
17 }
```

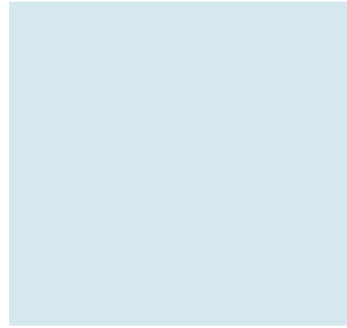
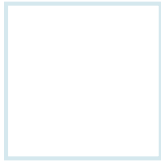
/* OUTPUT-----

請輸入一個整數: 2332

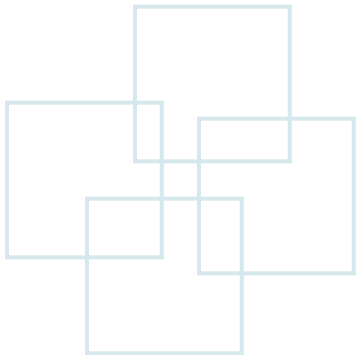
請輸入一個字元: k

num=2332, ascii of ch=107

-----*/



Character Input and Output Functions





Getting and Putting of a Character

- Read a character from keyboard after “Enter” is pressed.

Format of getchar()

```
CharVar=getchar();    /* Read one character and  
                        store it in "CharVar" */
```

- Print a character on screen

Format of putchar()

```
putchar(CharVar);    /* Print one character from  
                        "CharVar" to screen */
```




Getting and Putting of a Character (Cont.)

```
01
02 #include <stdio.h>
03 #include <stdlib.h>
04 int main(void)
05 {
06     char ch;
07     printf("請輸入一個字元: ");
08     ch=getchar();          /* 輸入一個字元，並指定給變數 ch */
09     printf("您輸入的字元是: ");
10     putchar(ch);          /* 將字元 ch 輸出到螢幕上 */
11     putchar('\n');        /* 將換行字元 ch 輸出到螢幕上 */
12
13     system("pause");
14     return 0;
15 }
```

/* OUTPUT---

請輸入一個字元: **h**

您輸入的字元是: h

-----*/



Getting of a Character without Enter

- Whether the input character shown on screen?
 - `getche()` — yes (echo)
 - `getch()` — no (no echo)

Format of `getche()` and `getch()`

```
CharVar=getche( );    /* Read a character with  
                        showing it on screen */
```

```
CharVar=getch( );    /* Read a character without  
                       showing it on screen */
```



Getting of a Character without Enter (Cont.)

```
01
02 #include <stdio.h>
03 #include <conio.h>          /* 載入 conio.h 標頭檔 */
04 #include <stdlib.h>
05 int main(void)
06 {
07     char ch;
08     printf("請輸入一個字元: ");
09     ch=getche();            /* 利用 getche() 輸入字元 */
10     printf(" 您輸入的字元是: %c\n",ch);
11
12     printf("請輸入一個字元: ");
13     ch=getch();            /* 利用 getch() 輸入一個字元 */
14     printf(" 您輸入的字元是: %c\n",ch);
15
16     system("pause");
17     return 0;
18 }
```

/* OUTPUT -----

請輸入一個字元: 8 您輸入的字元是: 8
請輸入一個字元: 您輸入的字元是: h

***/**



Lab 04

- Use printf() to show the following string on the screen:

```
"100/4=25"
```

- Declare a variable "int num=28.47f;" and use printf() to print it as follows:

```
num=0028.47  
num=+0028.47
```

- Write a program that you can input an integer that is output on screen in octal and hexadecimal format.
 - E.g., Type in "31" then output "37" and "1F".