

運算子、運算式與敘述

運算式與運算子

- 運算式由運算元(operand)與運算子(operator)所組成
 - 運算元可以是常數、變數甚至是函數
 - 運算子是數學上的運算符號，如「+」「-」「*」「/」等

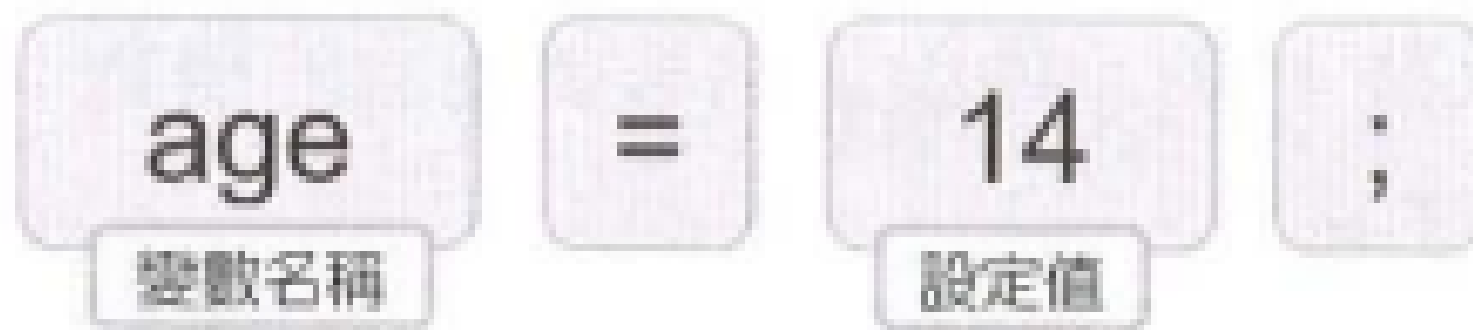


圖 5.1.1

運算式是由運算元
與運算子所組成

設定運算子

- 「=」是設定(指派)的意思，絕對不能解讀成等於，運算應該由右到左。



將整數14設定給變數age

- 記得是「設定」「設定」「設定」

練習-設定運算子

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    int age = 14;
```

```
    printf("age = %d \n",age);
```

```
    age = age+1; //將age+1後，在設定給age存放
```

```
    printf("將age加1後,age=%d\n",age);
```

```
    system("pause");
```

```
    return 0;
```

```
}
```

一元運算子

一元運算子(unary operator)

表 5.1.2 一元運算子的說明

一元運算子	意義	範例	說明
+	正號	a=+5	同 a=5，相當於設定 a 等於正 5
-	負號	a=-3	設定 a 等於-3
!	NOT，否	a=!b	把 b 的值取 NOT，再設給 a 存放

練習-「!」運算的用法

- 在C語言裡，1與0可以分別表示邏輯運算中的true與false，因此「!」運算子可以視為邏輯運算裡的「否定」運算

```
#include <stdio.h>

int main(void)
{
    int a = 0;
    int b = 6;
    printf("a=%d, !a=%d\n", a, !a);
    printf("b=%d, !b=%d\n", b, !b);
    system("pause");
    return 0;
}
```

算術運算子

➤ 算術運算子(mathematical operator)

表 5.1.3 算數運算子的說明

算數運算子	意義	範例	說明
+	加法	2+4	計算 2+4
-	減法	3-6	計算 3-6
*	乘法	7*9	計算 7*9
/	除法	6.4/3	計算 6.4/3
%	取餘數	21%9	計算 21 除以 9 的餘數

練習-餘數運算子

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    printf("12%%4=%d\n", 12%4);
```

```
    printf("12%%5=%d\n", 12%5);
```

```
    printf("12%%16=%d\n", 12%16);
```

```
    system("pause");
```

```
    return 0;
```

```
}
```


關係運算子與if敘述

► 常搭配if判斷條件使用

► If語法

```
if(條件判斷)
{
    條件成立
}
```

表 5.1.4 關係運算子的說明

關係運算子	意義	範例	說明
>	大於	$a > b$	判別 a 是否大於 b
<	小於	$a < b$	判別 a 是否小於 b
>=	大於等於	$a \geq b$	判別 a 是否大於等於 b
<=	小於等於	$a \leq b$	判別 a 是否小於等於 b
==	等於	$a == b$	判別 a 是否等於 b
!=	不等於	$a != b$	判別 a 是否不等於 b

練習-關係運算子

```
#include <stdio.h>

int main(void)
{
    if(5>2)
        printf("5>2成立\n");
    if(1)
        printf("此行一定被執行\n");
    if(3==8)
        printf("3==8成立\n");
    int num=9;
    if(num=8)
        printf("num=8成立\n");
    system("pause");
    return 0;
}
```

遞增與遞減運算子

表 5.1.5 遞增與遞減運算子

遞增與遞減運算子	意義	範例	說明
++	遞增，變數值加 1	a++	a 加 1 後再設定給 a 存放
--	遞減，變數值減 1	a--	a 減 1 後再設定給 a 存放

練習-遞增運算子

```
#include <stdio.h>

int main(void)
{
    int a=3,b=3;
    printf("a=%d",a);
    printf(",a++的傳回值為%d",a++);
    printf(",a=%d\n",a);

    printf("b=%d",b);
    printf(",++b的傳回值為%d",++b);
    printf(",b=%d\n",b);
    system("pause");
    return 0;
}
```

邏輯運算子

表 5.1.6 邏輯運算子的說明

邏輯運算子	意義	範例	說明
&&	AND，且	a&&b	計算 a AND b 的結果
	OR，或	a b	計算 a OR b 的結果

表 5.1.7 (a) AND 的真值表

AND	T	F
T	T	F
F	F	F

表 5.1.7 (b) OR 的真值表

OR	T	F
T	T	T
F	T	F

練習-邏輯運算子範例

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    int score;
```

```
    printf("請輸入成績:");
```

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

```
    if(score<0 || score>100)
```

```
    ...    printf("成績輸入錯誤\n");
```

```
    if(score<60 && score>49)
```

```
    ...    printf("需要補考!!\n");
```

```
    system("pause");
```

```
    return 0;
```

```
}
```

括號運算子

表 5.1.8 括號運算子的說明

括號運算子	意義
()	提高括號中運算式的優先順序

運算子的優先順序

表 5.2.1 運算子的優先順序

優先順序	運算子	類別	結合性
1	()	括號運算子	由左至右
1	[]	方括號運算子	由左至右
2	!、+ (正號)、- (負號)	一元運算子	由右至左
2	~	位元邏輯運算子	由右至左
2	++、--	遞增與遞減運算子	由右至左
3	*、/、%	算數運算子	由左至右
4	+、-	算數運算子	由左至右
5	<<、>>	位元左移、右移運算子	由左至右
6	>、>=、<、<=	關係運算子	由左至右
7	==、!=	關係運算子	由左至右
8	& (位元運算的 AND)	位元邏輯運算子	由左至右
9	^ (位元運算的 XOR)	位元邏輯運算子	由左至右
10	(位元運算的 OR)	位元邏輯運算子	由左至右
11	&&	邏輯運算子	由左至右
12		邏輯運算子	由左至右
13	?:	條件運算子	由右至左
14	=	設定運算子	由右至左

運算式

表 5.3.1 簡潔的運算子與使用說明

運算子	範例用法	說明	意義
+=	$a+=b$	$a+b$ 的值存放到 a 中	$a=a+b$
-=	$a-=b$	$a-b$ 的值存放到 a 中	$a=a-b$
=	$a=b$	$a*b$ 的值存放到 a 中	$a=a*b$
/=	$a/=b$	a/b 的值存放到 a 中	$a=a/b$
%=	$a\%=b$	$a\%b$ 的值存放到 a 中	$a=a\%b$

練習-簡潔運算式

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    int a=3,b=5;
```

```
    printf("計算前: a=%d,b=%d\n",a,b);
```

```
    a+=b;
```

```
    printf("計算後: a=%d,b=%d\n",a,b);
```

```
    system("pause");
```

```
    return 0;
```

```
}
```

運算式的型態轉換

- C語言處理型態轉換時，是以不流失資料為前提來進行資料間的型態轉換，將表示範圍小的型態轉換成表示範圍大的型態。當編譯器發現運算式中運算元的型態不行時，便會先進行自動型態轉換，再執行運算。

■ EX

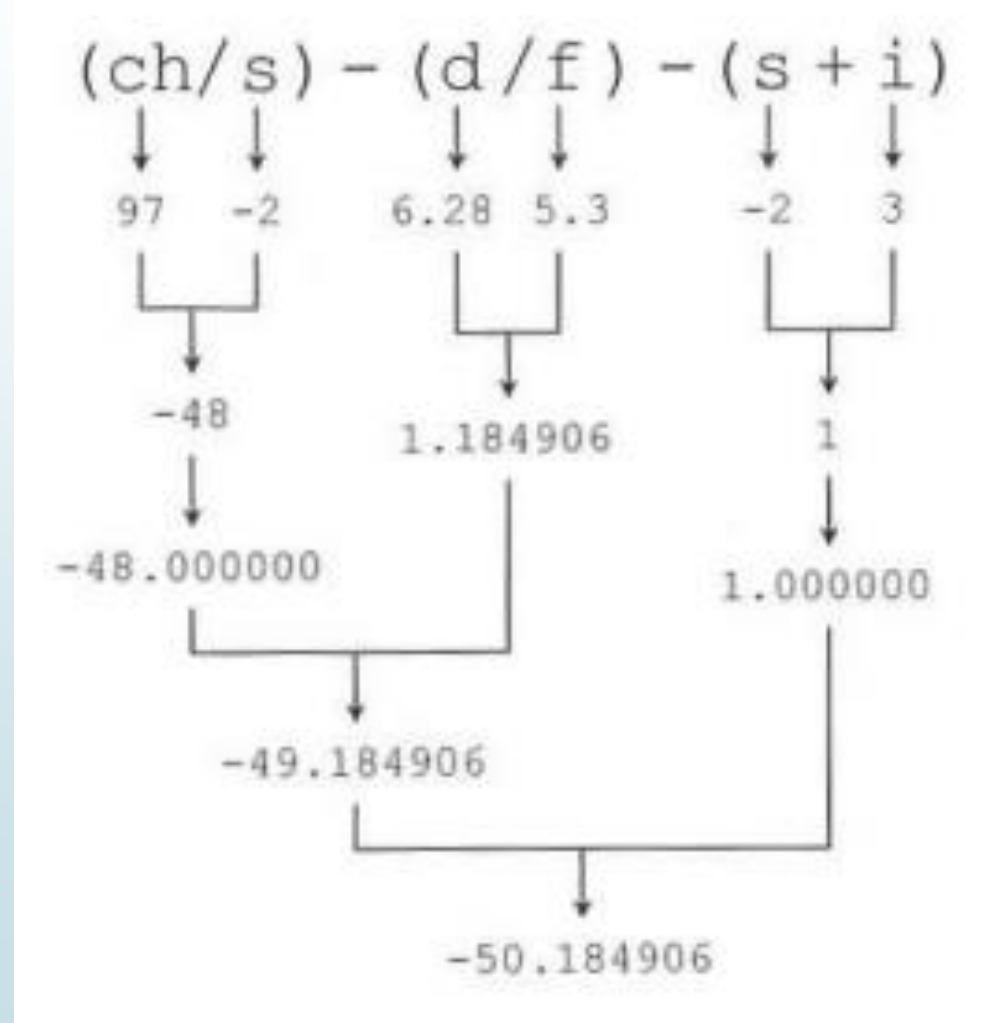
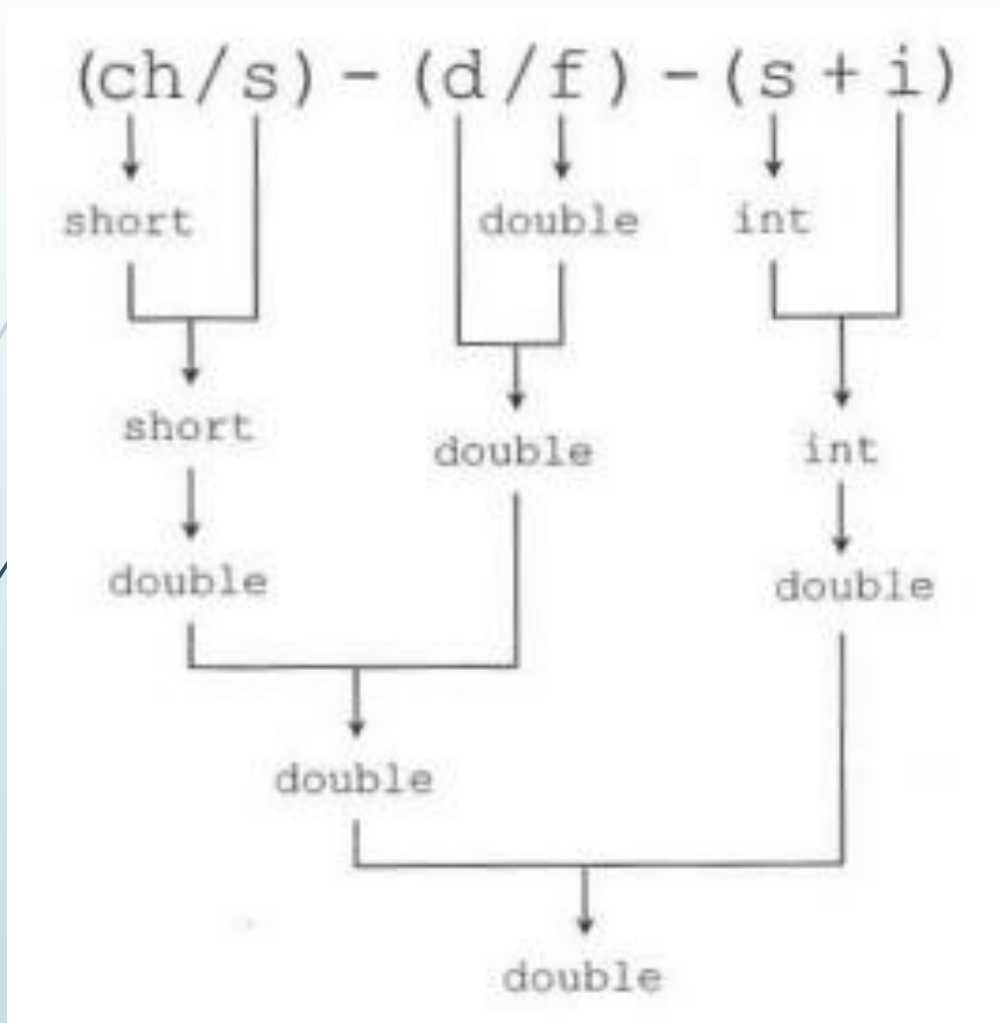
- int型態的變數i與float型態的變數f相加時，因為float的表示範圍較int大，因此變數i會被掀轉換成float型態，再與變數f做相加運算。
- char型態的變數c和int型態的變數i相乘，因為char的表示範圍(0~255)比int來得小，所以變數c會被先轉換成int型態，再與變數i相乘。
- 在C語言所提供的型態中，表示範圍從大到小依序
 - double → float → long → int → short → char

練習-運算式的型態轉換

```
#include <stdio.h>

int main(void)
{
    char ch = 'a';
    short s=-2;
    int i=3;
    float f=5.3f;
    double d=6.28;
    printf("(ch/s)-(d/f)-(s+i)=%f\n", (ch/s)-(d/f)-(s+i) );
    printf("size=%d\n", sizeof((ch/s)-(d/f)-(s+i)));
    system("pause");
    return 0;
}
```

資料運算過程



習題

21

■ 試寫一程式，可由鍵盤輸入攝氏溫度，程式輸出華氏溫度。

■ 轉換公式

■ 華氏溫度 = $(9/5) \times \text{攝氏溫度} + 32$

 Z:\C2講義\C語言\C語言教學手冊\課程範例\05_習題_溫度轉換.exe

```
請輸入攝氏溫度:30  
華氏溫度為:86.000000  
請按任意鍵繼續 . . .
```

習題

■ 試寫一程式，可由鍵盤輸入公里，程式輸出英哩。

■ 轉換公式

■ $1 \text{ 英哩} = 1.6 \text{ 公里}$

 Z:\C2講義\C語言\C語言教學手冊\課程範例\05_習題_公里轉英哩.exe

```
輸入公里數: 30  
英里數為: 18.750000  
請按任意鍵繼續 . . .
```