

## Chapter 3 資料表示法

- 研讀完本章後，你應該可以：
  - 區分類比與數位資訊。
  - 解釋資料壓縮與計算壓縮比。
  - 解釋負數與浮點數的二進制格式。
  - 描述 ASCII 與 Unicode 字元集的特徵。
  - 完成多種文件類型的壓縮。
  - 解釋聲音的本質以及它的表示法。
  - 解釋 RGB 數值如何定義一種色彩。
  - 區分光柵圖片與向量圖片。
  - 解釋時間性與空間性視訊的壓縮。

1

Ch03 資料表示法

## 資料與電腦

- 資料 (data)
  - 是事實的基本數值，可以是無特定結構以及缺乏脈絡的。
  - 電腦是處理大量各類型資料的多媒體 (multimedia) 裝置。電腦儲存、呈現以及協助我們修改下列資訊類型：
    - 數字 (numbers)、文字 (text)、音訊 (audio)、影像與圖片 (images and graphics)、視訊 (video)
- 資訊 (Information)
  - 則是經過組織與/或以一種能有效解決某種問題的方法處理過的資料。
  - 可以幫助我們回答問題 (它「告知」(informs))。

2

Ch03 資料表示法

## 資料與電腦

- 資料壓縮 (data compression)：減小儲存一筆資料所需的空間
  - 無漏失型 (lossless)
    - 它指的是資料重新取得時沒有漏失任何原始資訊。
  - 漏失型 (lossy)
    - 它指的則是資料於壓縮過程會漏失某些資訊。
- 壓縮率 (compression ratio)
  - 是壓縮後資料大小除以原始資料大小，資料大小數值可以用位元或字元 (或者任何適合的單位) 來表示，只要是兩個數值的度量單位相同即可，**壓縮率**用來表示有多少壓縮發生。

3

Ch03 資料表示法

## 類比與數位資訊

- 大部分的自然世界是連續與無限的，數字線也是連續的，而且其數值是無限往大與小的方向成長。
  - 也就是說，你永遠可以找到比給定數字更大或更小的數字。兩個整數之間的數字空間也是無限的。
- 相反地，電腦是有限的。
  - 電腦只能利用其有限的記憶體與其他硬體裝置的空間來儲存與操作某一定數量範圍內的資料。
  - 我們無法用有限的機器來表示無限的世界。設計電腦的各種儲存資訊格式的目的是要利用電腦來充分描繪世界，讓我們可以感覺視覺與聲音的世界，來做為資料的表述即可。
  - 在算數運算上，只要能夠滿足我們計算需求(指精確到某一個數字以下，如  $10^{-9}$ )。

4

Ch03 資料表示法

## 類比與數位資訊

- 類比資料 (analog data)
  - 是一種連續式的表示法，類似於它所表示的實際資訊。
- 數位資料 (digital data)
  - 則是一種不連續的表示法，它將資訊拆散成分開的元素。

圖 3.1 水銀溫度計以正比於溫度的方式連續上升



5

Ch03 資料表示法

## 電子訊號

- 類比訊號是電壓連續地高低變動，但數位訊號則只有高或低兩種狀態，恰好可以對應於兩個二進制數元。
- 當電子訊號沿線往下移動時，所有的電子訊號(類比與數位兩種都一樣)都會衰減，也就是說訊號的電壓值會由於環境的影響而變動。

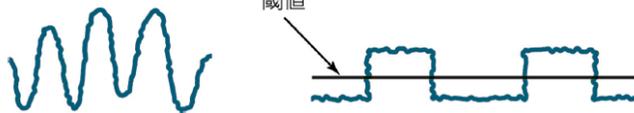
6

Ch03 資料表示法

## 類比與數位資訊



圖 3.2 類比與數位訊號



需要重新計時 (reclocked)

圖 3.3 類比與數位訊號的降級

7

Ch03 資料表示法

## 類比與數位資訊

- 電腦處理類比資訊時無法運作得很好，因此我們藉由將類比資訊切成幾個部分、並對各個部份分別予以表示的方式來將類比資訊 **數位化 (digitize)**。
- 為什麼我們要使用二進制呢？現代的電腦被設計成使用與管理二進制數值，因為如果用來儲存與管理資料的裝置只需要表示兩種可能數值的其中一種，則它們會大大比較便宜、而且比較可靠。

8

Ch03 資料表示法

## 二進制表示法

- 1個位元可以是1或0，只能代表兩件事物。
- 2個位元可以表示成00、01、10以及11，可以代表四件事物。
- 3個位元表示成000、001、010、011、100、101、110、111，可以代表八件事物。
- 4個位元可以表示16件事物。
- 5個位元可以表示32件事物.....等。
- 在2進制，n位元可以表示 $2^n$ 件事物，因為可以由n位元形成 $2^n$ 種0與1的組合。
  - 注意每次增加1個位元，我們可以表示的事物的個數便會乘以2。

9

Ch03 資料表示法

## 二進制表示法

1位元	2位元	3位元	4位元	5位元
0	00	000	0000	00000
1	01	001	0001	00001
	10	010	0010	00010
	11	011	0011	00011
		100	0100	00100
		101	0101	00101
		110	0110	00110
		111	0111	00111
			1000	01000
			1001	01001
			1010	01010
			1011	01011
			1100	01100
			1101	01101
			1110	01110
			1111	01111

圖 3.4 位元組合

10

Ch03 資料表示法

## 二進制表示法(續)

1位元	2位元	3位元	4位元	5位元
				10000
				10001
				10010
				10011
				10100
				10101
				10110
				10111
				11000
				11001
				11010
				11011
				11100
				11101
				11110
				11111

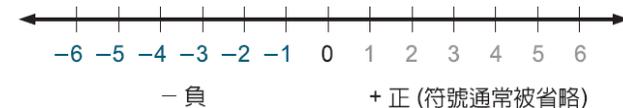
圖 3.4 位元組合

11

Ch03 資料表示法

## 數字的表示法：負數的表示 帶號—數量表示法

- 典型的數線看起來像下圖，負號意謂數字在0的左邊，正號則是數字在0的右邊：

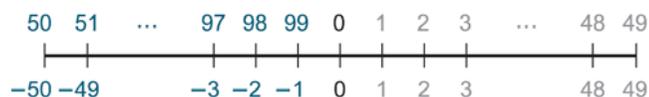


- 用帶號—數量表示法有個問題存在：**有兩個0**，亦即正零(+0)與負零(-0)。
- 電腦裡零有兩種表示法會引起不必要的複雜性，例如當需要判斷某個運算結果是否為零時，需要檢查兩次(+0與-0)。所以要使用其他的負數表示法。
- 設計如何於電腦儲存數字時，如果我們只容許一個固定總數的數字，則我們可以只用整數來代表數字，讓其中一半用來表示負數。

12

Ch03 資料表示法

## 固定大小數字表示法



- 例如，如果我們可以表示的十進位數共 100 個，則我們可以令數元 1 至 49 為正數 1 至 49，數元 50 至 99 表示負數 -50 至 -1。
- 雖然人類傾向於以符號及數量來表示數字，但當數字涉及電子式計算時，**補數**的策略真的是比其他方式來得容易多了。

## 負數的表示(續)

- 在上頁的數字表示方案中要進行兩數相加，只要將兩個數字的代表數元加在一起，並捨棄其進位即可。

符號-數量	新方案
$\begin{array}{r} 5 \\ + -6 \\ \hline -1 \end{array}$	$\begin{array}{r} 5 \\ + 94 \\ \hline 99 \end{array}$
$\begin{array}{r} -4 \\ + 6 \\ \hline 2 \end{array}$	$\begin{array}{r} 96 \\ + 6 \\ \hline 2 \end{array}$
$\begin{array}{r} -2 \\ + -4 \\ \hline -6 \end{array}$	$\begin{array}{r} 98 \\ + 96 \\ \hline 94 \end{array}$

## 負數的表示(續)

- 在上頁的數字表示方案中要進行兩數相減，利用  $A-B=A+(-B)$ ，我們可以藉由加上第二個數字的負值至第一個數字作減法運算。

符號-數量	新方案	加負值
$\begin{array}{r} -5 \\ + 3 \\ \hline -8 \end{array}$	$\begin{array}{r} 95 \\ - 3 \\ \hline 92 \end{array}$	$\begin{array}{r} 95 \\ + 97 \\ \hline 92 \end{array}$

## 負數的表示(續)

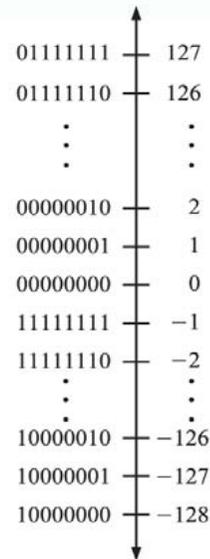
- 我們可以用以下公式來計算負值表示法

$$\text{Negative}(I) = 10^k - I, \text{ 此處 } k \text{ 是數元的數目。}$$

- 例如在 3 個十位數元中：
  - $\text{Negative}(1) = 10^3 - 1 = 999$
- 這種負值表示法稱為**十的補數 (ten's complement)**

## 二的補數

- 我們可以將十的補數這個負數表示概念擴充至其他基底，最常見的就是二的補數，如右圖
- 讓我們假定某個數字必須以8個位元來表示。
- 以這種表示法表示，負數的最左邊位元始終為1。



Ch03 資料表示法

## 負數的表示(續)

- 完成加法與減法的方式與十的補數算術相同
  - $-127 \quad 10000001$
  - $+ \quad 1 \quad \underline{00000001}$
  - $-126 \quad 10000010$
- 練習：請問 8 個位元的 2 進位制以 2 的補數表示負整數時， $-113$  的表示法為何？

## 數字溢位

- 溢位 (overflow)
  - 是發生在計算所得的數值無法以我們所分配的位元數目來代表，也就是超過我們所分配的位元數目的時候。例如，如果每個數值(含正負數)都是以八位元來儲存，3 加上 127 將會得到 -126 而不是 130，發生溢位：

$$\begin{array}{r} 01111111 \\ + 00000011 \\ \hline 10000010 \end{array}$$

- 溢位是我們將無限世界映射至有限機器時所遭遇的問題中的典型例子。

## 實數的表示

- 自然界的實數含有整數部分與小數部分。例如 104.32、0.999999、357.0 以及 3.14159 等數元所表示的數值是根據它的位置，且這些位置數值與它的基底相關。
- 小數點右邊的位置數值是  $1/10$  位置 ( $10^{-1}$ )、 $1/100$  位置 ( $10^{-2}$ ) 等等。
- 在二進制中也應用相同的規則，只不過基底數值為 2。因為我們並非以十進制來計算，小數點便以**基數點 (radix point)** 來表示。
- 二進制中基數點右邊的位置是  $1/2$  位置 ( $2^{-1}$ )、 $1/4$  位置 ( $2^{-2}$ ) 等等。

## 實數的表示

- 在電腦的領域中，我們稱所有可以被表示的非整數為**實數 (real value)**，其實電腦是無法表示所有實數的，有大小限制，有小數點後的位數限制，還有分數限制(如  $1/7$ )，所有無理數也無法在電腦正確表示，只能逼近。
- 以十為基底的實數可以用下列公式定義它：  
正負符號 \* 假數 \*  $10^{\text{指數}}$
- 假數的英文為 mantissa
- 這種表示法稱為**浮點 (floating point)** 表示法，因為數元數目固定但基數點浮動。

21

Ch03 資料表示法

## 實數的表示

- 二進制浮點數也可以用下列公式來定義：

$$\text{符號} * \text{假數} * 2^{\text{指數}}$$

- 讓我們將.75轉換成二進制。

$$.75 * 2 = 1.50$$

$$.50 * 2 = 1.00$$

因此，十進制的.75以二進制表示是.11。

22

Ch03 資料表示法

## 實數的表示

- 我們藉由轉換十進制的 20.25 為二進制來討論整個轉換過程。首先我們轉換20。

$\frac{10}{2}$	$\frac{5}{2}$	$\frac{2}{2}$	$\frac{1}{2}$	$\frac{0}{2}$
20	10	4	2	0
0	0	1	0	1

- 20以二進制表示是10100。現在我們轉換分數部分：  
 $.25 * 2 = 0.50$   
 $.50 * 2 = 1.00$

因此，以十進制表示的 20.25 以二進制表示是 10100.01。

23

Ch03 資料表示法

## 實數的表示

表 3.1 以小數表示法與以浮點表示法 (5 個數元) 表示的數值

實數	浮點數
12001.00	$12001 * 10^0$
-120.01	$-12001 * 10^{-2}$
0.12000	$12000 * 10^{-5}$
-123.10	$-12310 * 10^{-2}$
155555000.00	$15555 * 10^3$

24

Ch03 資料表示法

## 實數的表示(續)

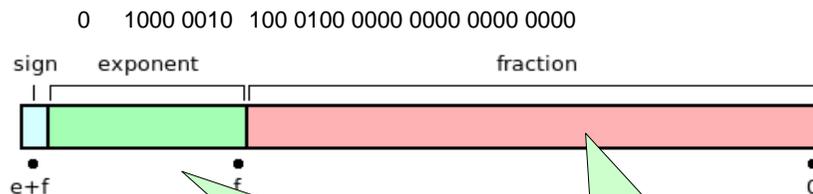
- 科學表示法 (scientific notation) 是一個你們可能已經極為熟悉的名詞。科學表示法是浮點表示法的其中一種形式，它的小數點始終保持在最左邊非零數元的右邊。
- 例如，12001.32708以科學表示法表示將寫成  $1.200132708 \text{ E} +4$ 。

## 二進位浮點數表示法

- 二進位正規化
  - 將數字轉換成  $1.\text{ddd} \times 2^{\text{指數}}$  的形式
  - 例如  $(12.25)_{10}$  的二進位為  $(1100.01)_2$ ，正規化表示為  $1.10001 \times 2^3$
- 二進位浮點數
  - IEEE 標準(請參考：[http://en.wikipedia.org/wiki/IEEE\\_floating-point\\_standard](http://en.wikipedia.org/wiki/IEEE_floating-point_standard))，依位元數可分
    - 32 位元的單精確度 (Single Precision)：1+8+23
    - 64 位元的雙精確度 (Double Precision)：1+11+52

## 二進位浮點數表示法

- $(12.25)_{10}$  的單精確度浮點數表示法



1 bit, 表示正或負  
0 表示正, 1 表示為負

8 bits, 偏差指數  
以 127 為指數偏差值, 也就是將該值正規化後的指數部份加上 127 所得之值 [只能為 0 或正整數] 的 2 進制表示  
本例為  $3+127=130$ , 其 2 進制為 1000 0010

23 bits, 小數部份  
指該值正規化後的小數部份, 以 2 進制表示

## 練習

- 請問  $(3.125)_{10}$  的單精確度浮點數表示法為何?
- 請問以下單精確度浮點數表示法  
0 01111110 1010000 00000000 00000000  
代表的值以 10 進位表示為何?