一個美妙的不等式

1 邁向新的千禧年

這是個怎麼樣的世代呢？除了傳統的數學領域及其研究方法之外，現在似乎又多了一個活動的空間，可以讓我們那自由且豐富的創造力與想像力在一個更寬更廣的世界中遨翔。在此，我們計劃藉著數學套裝軟體 MATHEMATICA[1] 計算(包括數值計算及符號演算)、繪圖(包括平面繪圖、立體繪圖及動態畫)與程式的功能，來擺脫一般傳統所習慣的「定義、定理、證明」之介紹模式，試圖以問話、啓發的方式，用更生動、更活潑的面貌來迎向那嶄新的未來，一起邁入另一個新的千禧年(Millennium)。

我們將挑出數學中一些經常出現且較有名的常數，運用實驗的技巧，一起來探討相關的數學公式及其基本觀念。這是一個新的嘗試！若你在閱讀數學當中，發現到好的、有趣的題材，也可依樣畫葫蘆的進入 MATHEMATICA 的世界中暢遊一番。現在就讓我們從一個非常熟悉而又有點陌生的歐拉[2]數 e 談起吧！

2 歐拉數爲何物？

在 MATHEMATICA 中，我們用大寫的 E 來表示歐拉數 e，而且可任由你來選擇其精確度。譬如，你要看到歐拉數精確到五十五位數，則可輸入指令`N[E,55]`，得到結果如下：

```
In[1]:=N[E,55]
Out[1]=2.718281828459045235360287471352662497757247093699959575
```

若你是 MATHEMATICA 的初學者或對 MATHEMATICA 不熟，有需要的話可先閱讀本章後面的 MATHEMATICA 簡介，也可拜訪東海大學理學院首頁的教學諮詢中心欄，在此你可找到 MATHEMATICA 的簡易使用手冊。

[1] MATHEMATICA 是由美國 Wolfram Research 公司所研發出來的一套透過電腦來演算數學的系統 (a system for performing mathematics by computer)。自從 1988 年問世以來，由於其多才多藝，MATHEMATICA 已建立起自己的形象而成爲眾多使用者所選擇的電腦代數系統 (Computer Algebra System, 簡稱爲 CAS)。其實在科學的各個領域上，它也是一套強而有力的研究工具。

[2] 歐拉 LEONHARD EULER (1707–1783) 瑞士數學家。他的著作等身，撇開身爲 13 個小孩的父親且後來甚至變成全盲不講，仍然有辦法寫超過 800 篇的論文；而且範圍遍及他那時期數學的每一分支，並都有實質的貢獻。有人說他計算如同呼吸一樣的簡單。在眾多新記號中，歐拉引進三大常數的符號 π, e, i, 和的符號 Σ 及函數的符號 $f(x)$。他的《*Introductio In Analysin Infinitirum*》是十八世紀末最重要的數學教科書。在眾多貢獻當中，且讓我們介紹一個歐拉本人也引以爲傲的公式：

$$1+\frac{1}{2^2}+\frac{1}{3^2}+\cdots+\frac{1}{n^2}+\cdots=\frac{\pi^2}{6}。$$

歐拉數e是個很漂亮的無理數。究竟漂亮到什麼程度呢？首先，以此爲底的指數函數e^x是數學中非常重要，應用也非常廣泛的一個函數；而其導數就是它自己，這又使得此函數e^x變成一個極其單純簡便的函數，你還能要求比這更簡單的公式嗎？其次，我們至少可以從四個方面來理解這個數：

(a) 從微分的觀點：觀察指數函數族$\{ a^x \mid a > 0 \}$的圖形。每一函數圖形都會經過點$(0,1)$，而經過這一點的切線斜率與實數之間有一對一的對應關係；其中那使得過$(0,1)$之切線斜率爲1的唯一正數a，我們用e來表示。Douglas N. Arnold 對此有一 MATHEMATICA 的動畫程式，請拜訪他的網頁。其網址爲：`http://www.math.psu.edu/dna/graphics.html#exponential`

(b) 從積分的觀點：令$a > 1$。在區間$[1, a]$，考慮函數$y = 1/x$圖形下方所包圍的區域。顯而易見，此區域的面積隨著a的增大而增大。當這個面積剛好是一個單位時的那個唯一的正數a，我們就定義此數爲e。換句話說，e就是那個唯一的正數使得定積分$\int_1^e \frac{1}{t}\, dt$之值爲1者。

(c) 從極限的觀點：我們可以定義e爲底下數列的極限值

$$\left\{ (1 + \frac{1}{n})^n \right\}_{n=1}^{\infty} 。$$

(d) 從級數的觀點：我們也可以定義e爲下述無窮級數的和

$$1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \cdots = \sum_{n=0}^{\infty} \frac{1}{n!} 。$$

爲了方便起見，我們稱第四個定義爲級數表示法，而第三個定義爲極限表示法。首先我們來比較一下這四個定義。第一個定義的方法不是太好，因爲連你要去估計一下它的大小，都困難重重！第二個定義的方法好一些，至少有其幾何意義，使我們對歐拉數更有感覺;而且可透過黎曼和或其他的法則來計算其近似值。級數表示法遠比極限表示法好，當然這裏所謂的好，是比較它趨近e的速度。前者遠比後者快，而且快的很多。請看下面的計算：

$$\sum_{k=0}^{10} \frac{1}{k!} \approx 2.71828, 18011, 46384, 47971, 78130, 51146, 4$$

$$\sum_{k=0}^{22} \frac{1}{k!} \approx 2.71828, 18284, 59045, 23536, 02471, 10869, 1$$

$$(1 + \frac{1}{10})^{10} = 2.59374, 24601$$

$$(1 + \frac{1}{10^8})^{10^8} \approx 2.71828, 17864$$

在 MATHEMATICA 中，和的指令爲：`Sum[f[k],{k,1,n}]` 意表

$$\sum_{k=1}^{n} f(k) = f(1) + f(2) + f(3) + \cdots + f(n)。$$

所以上面的計算可由 MATHEMATICA 執行如下：

當 $n = 10$ 時，級數表示法精確到小數點之後第 7 位；而 $n = 22$ 時，則可精確到小數點之後第 22 位。但極限表示法，即使 n 大到 $100000000 = 10^8$ 時，才精確到小數點之後第 6 位。

實際上，要估計 $S_n = \sum_{k=0}^{n} \frac{1}{k!}$ 與 e 之間的誤差，是非常容易的。且看下面的不等式：

$$e - S_n = \sum_{k=n+1}^{\infty} \frac{1}{k!} \leq \frac{1}{(n+1)!} \left[1 + \frac{1}{n+1} + \frac{1}{(n+1)^2} + \cdots \right] = \frac{1}{n!n}$$

所以我們有

$$e - S_{10} < \frac{1}{(10! \times 10)} < 10^{-7}$$

以及

$$e - S_{22} < \frac{1}{(22! \times 22)} < 10^{-22}$$

如上面計算所顯示出來的。

雖然如此，極限表示法並非一無是處！且看下面的數據：

$$\left(1 + \frac{1}{10}\right)^{11} = 2.85311670611$$

```
In[3]:= N[(1+1/10)^11,32]
Out[3]= 2.8531167061100000000000000000000
```

這讓我們看到

$$\left(1 + \frac{1}{10}\right)^{10} < e < \left(1 + \frac{1}{10}\right)^{11}$$

那麼這樣的不等式是否只有 $n = 10$ 的時候才對呢？這當然需要更多的觀察才能下結論。且看下面的實驗：

3　實驗：一個美妙的不等式

要建立一個與自然數n有關的不等式之第一步就是先觀察前面幾個自然數的情況，從而猜測是否對所有的自然數仍然會成立。我們所面對的三個數是

$$\left(1+\frac{1}{n}\right)^n，\ e \quad 以及 \quad \left(1+\frac{1}{n}\right)^{n+1}。$$

(a) 用 Mathematica 中 Table 的功能，列出$\left\{\left(1+\frac{1}{n}\right)^n, e, \left(1+\frac{1}{n}\right)^{n+1}\right\}$, n從1到100，觀察三者之關係如何？

(b) 是否對所有的n，上述所觀察到的關係恆成立？我們可將n看成一連續變數，則圖表變成函數圖形。如此一來，當n很大的時候，就更容易推測會有何結果。請用 Mathematica 中 Plot 的功能，分別畫出函數$y=\left(1+\frac{1}{x}\right)^x$, $y=e$及$y=\left(1+\frac{1}{x}\right)^{x+1}$在區間$x\in[1,1000]$的圖形。

(c) 再將這三個圖形放在同一座標平面上，其關係就一目了然。

(d) 經過前述三個步驟，你的結論如何？

4　分析：化繁爲簡與無言的證明

由上面的實驗可知，圖形遠比列表來得簡潔有力而且更具說服力，讓人們一眼即可讀出其中所蘊含的重要信息。且看實驗 (b) 與實驗 (c) 之圖形。所以我們有很好的理由，作如下的猜測：

猜測：對所有的正整數n，下面不等式恆成立

$$\left(1+\frac{1}{n}\right)^n < e < \left(1+\frac{1}{n}\right)^{n+1}。$$

讓我們來分析一下這個不等式。我們所面臨的這三個數，乍看之下，挺抽象的而且不太友善，因爲都是指數函數的值。但是指數函數的反函數就是對數函數，所以很自然的，我們就引進自然對數。這下子，全部都煥然一新了。請看：

$$\begin{aligned}
& \left(1+\frac{1}{n}\right)^n < e < \left(1+\frac{1}{n}\right)^{n+1} \\
\Longleftrightarrow\quad & n\log\left(1+\frac{1}{n}\right) < 1 < (n+1)\log\left(1+\frac{1}{n}\right) \\
\Longleftrightarrow\quad & \frac{1}{n+1} < \log\left(1+\frac{1}{n}\right) < \frac{1}{n}\ 。
\end{aligned}$$

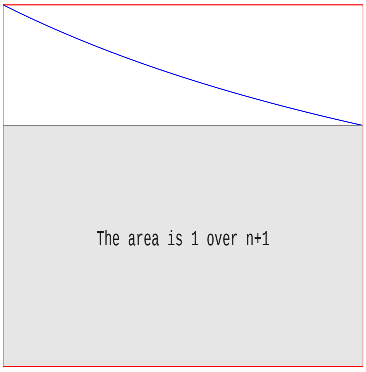

圖 1: 在區間 $[1, 1+\frac{1}{n}]$ 上，R(曲線至 x-軸所包圍的區域)，R_1 (灰色區)，及 R_2(上方水平線至 x-軸的長方形區域)之圖形。

現在這三個數

$$\frac{1}{n+1}, \quad \log(1+\frac{1}{n}) \quad 及 \quad \frac{1}{n},$$

比前三個數友善多了，怎麼說呢？首先，$\log(1+\frac{1}{n})$ 這個數代表函數 $f(x)=\frac{1}{x}$ 圖形底下從 $x=1$ 到 $x=1+\frac{1}{n}$ 所包圍區域 R 的面積，亦即下面這個定積分的值：

$$\log(1+\frac{1}{n}) = \int_1^{1+\frac{1}{n}} \frac{1}{x}\,dx。$$

因爲函數 $f(x)=\frac{1}{x}$ 在區間 $[1, 1+\frac{1}{n}]$ 是遞減的，所以可得到不等式：

$$\frac{n}{n+1} = \frac{1}{1+\frac{1}{n}} < \frac{1}{x} < \frac{1}{1} = 1。$$

在區間 $[1, 1+\frac{1}{n}]$ 上取定積分，得不等式：

$$\int_1^{1+\frac{1}{n}} \frac{n}{n+1}\,dx < \int_1^{1+\frac{1}{n}} \frac{1}{x}\,dx < \int_1^{1+\frac{1}{n}} 1\,dx,$$

亦即：

$$\frac{1}{n+1} < \log(1+\frac{1}{n}) < \frac{1}{n}。$$

這証明了我們的猜測是對的，而且証明非常簡單。如果從圖?? (MATHEMATICA 的指令如下) 來看這個不等式，那更是不言而喻：

```
In[8]:= p4=Graphics[{{GrayLevel[0.9],
                    Rectangle[{1,0},{1.5,1/1.5}]}}];
        p5=Plot[1/x,{x,1,1.5},PlotStyle->RGBColor[0,0,1]];
        p6=ListPlot[{{1,0},{1,1},{1.5,1},{1.5,0},{1,0}},
        PlotJoined->True,PlotStyle->RGBColor[1,0,0]]
        p7=Graphics[{
           {GrayLevel[0.5],Line[{{1,1/1.5},{1.5,1/1.5}}]},
           {RGBColor[1,0,0],Line[{{1,0},{1.5,0}}]}     }];
        p8=Graphics[{{GrayLevel[0.1],
             Text["The area is 1 over n+1",{1.25,.35}]}}];
        Show[{p4,p5,p6,p7,p8}]
```

令R_1與R_2分別爲上述區域R的內接(灰色區)與外接(上面那條水平線到x-軸)長方形區域，則其共同邊長爲$\frac{1}{n}$，而另一邊長分別爲$\frac{1}{1+\frac{1}{n}}$與1。所以其對應的面積分別就是$\frac{1}{n}\cdot\frac{1}{1+\frac{1}{n}} = \frac{1}{n+1}$與$\frac{1}{n}\cdot 1 = \frac{1}{n}$，當然$R$的面積$\log(1+\frac{1}{n})$就介於這兩個數之間。這就是所謂的「無言的證明」，請參閱蔡聰明所寫的《數學的發現趣談》，當中可看到一些其他的例子。若你想要找到更多的例子，可查閱「美國數學協會(The Mathematical Association of America)」所出版的數學雜誌(Mathematics Magazine)中不定期的專欄 “Proof Without Words”。另外還有一本由Roger B. Nelsen所寫的，書名就叫Proofs Without Words，也是由「美國數學協會」所出版。

5 Mathematica 簡介

Mathematica是一個強而有力的數學程式環境。它提供了數值的、符號的及圖形的工具，來協助我們解決數學方面的問題。已有相當多的人使用Mathematica來觀察並分析工程、數學、物理學、經濟學及其他科學領域上的問題。Mathematica也可以當成高階的程式語言來使用。其工作的平台相當廣泛，從Cray的超級電腦到桌上型及膝上型輕便電腦皆可。在超過一百多萬個使用者當中，約有28%爲工程師，21%爲電腦科學家，20%爲物理方面的專家，12%爲數學方面的專家，12%爲管理科學、社會科學及生命科學家。約有三分之二的使用者是在工業界及政府部門工作，而僅僅8%爲學生使用者。

Mathematica是由兩部分所組成的，就是所謂的前端(Front End)及核心(Kernel)。核心乃是其計算的引擎，負責所有的計算工作。前端則透過記事本介面(notebook interface)或指令行介面(command line interface)，來作爲使用者與核心之間的溝通橋樑。

當你啓動Mathematica之後，它會自動開啓一個工作視窗[3]和一個常用的基本輸入面板(Basic Input Palette)。我們稱此工作視窗爲記事本(Notebook)，而

[3]這是一個未命名的空白檔案，名稱爲untitled-1，亦即未命名的第一個檔案。

MATHEMATICA 用一種特殊的檔案格式來儲存工作視窗的內容，其附加檔名爲 .nb。這些記事本有如一般的文字處理機 (word processor) 一樣，你可在上面加註解[4]、做結論還可匯入或匯出多種不同格式的圖形檔。要發表的論文資料或上課講義可在此預備，也可在此下達指令透過印表機將這些文件列印出來。其中的資料可在記事本之內或之間互相剪貼，目的當然就是希望能再使用或是經過修改其文字、圖形或計算式子後成爲我們所要的文件。

前端記事本檔案將裡面的資料分類安排放在所謂的 "Cells" 當中，所以這些 Cells 就是構成此記事本檔案最基本的單元。在這些文字、圖形或計算式的 Cells 之最右端都有「右中括弧」，而這些「右中括弧」就代表整個文字、圖形或計算式單元。輸入 Cells 含有 MATHEMATICA 的指令者可按下 `SHIFT-RETURN` 鍵 (先按住 `SHIFT` 鍵然後再按 `RETURN` 鍵) 來執行這些指令。文字 Cells 僅包含有文字信息，所以不需經由 Kernel 來計算。圖 Cells 則包含有描繪圖及曲線圖。

我們可以把 Cells 格式化成具備各式各樣的屬性，如所展示的字體其大小與顏色等。這些 Cells 也可用摘要的形式來組織一文件，其方法如下：先點取此 Cells 最右端的「右中括弧」，再從工作視窗上點取 Format 選單中的 Style 然後選取 Title, Subtitle, Section, Subsection, $\cdots$ 即可。

接著我們介紹一下 MATHEMATICA 所用的慣用語法，你可透過這些規則來了解 MATHEMATICA 的指令是怎麼下達的。

- 基本四則運算及指數所用的語法跟其他的程式語言完全一樣。
- 變數通常用小寫字母表示，但也可以是一個字串如 `yvalue=`$\cdots$。
- 函數的變數 `x,y,...` 以中括弧 `[x,y,...]` 括起來，而小括弧則用來達到分組的效果。
- 串列 (List) 是 MATHEMATICA 最原始的資料結構，以大括弧`{}` 括起來，其中的元素則用逗點分開。如一維串列 `{1,2,3}` 爲一向量，而二維串列 `{{1,2,3},{4,5,6}}` 則表矩陣其第一列就是第一個元素 `{1,2,3}`。
- 內建函數與內建常數都是以大寫字母起頭，如 `Sin[Pi/2]`。
- 乘號是用 `*` 或是空格來代表，如 `a*b` 或 `a b`。變數與整個括號相乘可不用留空格，如 `a(b-1)`。數字與變數應注意前後次序，如 `7x` 表示 `7*x`，但 `x7` 表示一變數名爲 `x7`，而 `x 7` 則表示乘積 `x*7`。
- 符號 `=` 意指代換，如 `t=1`；而相等則以符號 `==` 表示之，如 `Equal[x,t]` 或 `x==t` 只有當 `x` 與 `t` 有相同的值才會得真。

[4]若程式過於龐大，使用者得發很多時間從頭看起；所以一旦加上註解，使用者便可以清楚的知道程式的內容。一般的註解可直接寫在畫面上，但如此一來電腦在執行計算時便容易產生混亂。因此我們可先點取最右端的「右中括弧」，再從工作視窗上點取 Format 選單中的 Style 然後選取 Text 即可。

- 否定指令Not可用 !表示之，如 x!=t 爲真若 x與 t有不同的値時。
- 上一個輸入以 % 表示之，而 %n 指的是第 n 個輸入即 In[n]。所以 %% 意指上上一個輸入$\cdots$等等。
- 每一個Cells都是以 In[n]:=開始的，但你絕不可鍵入這些，而只需鍵入你要的文字或指令，因爲MATHEMATICA會自動在你執行(即按下SHIFT-RETURN鍵)之後將之冠在前頭的位置。

6 MATHEMATICA 指令簡介

最後我們將此章用過的指令按字母順序介紹一下。你若想做進一步的探討，可透過Stephen Wolfram的書[?]或其他入門書[?, ?, ?, ?]來達成你的美夢。或者也可以從MATHEMATICA「Help視窗」當中輸入你所要了解的指令，詳細的介紹就會出現在你的眼前。注意MATHEMATICA對大小寫是靈敏的(case sensitive)，字母是大寫就必須大寫，否則會出現錯誤信息。

- AxesLabel->{"x","y"}] 座標軸的名稱，分別爲x-軸與y-軸。
- E 自然對數的底e，其値約爲$2.71828182845904523\cdots$。
- Factorial[n](或n!) n階乘，即$n(n-1)(n-2)\cdots 3\cdot 2\cdot 1$。
- Graphics[*list*] 爲一二維圖形，*list*包含三種類型的基本元件，一是RGBColor[ ]，二是Point[$\{x,y\}$]，三式選項，而Graphics的結構則爲

  ```
  Graphics[{RGBColor[ ],{Points[ ]}},{Options}]
  ```

  RGBColor[ ]、Points[ ]與Options代表繪圖三要素中的圖形控制指令(Graphics Directives如點的大小、線條的粗細和顏色等)、基本圖元(Graphics Primitives如點、線、圓、圓盤和多邊形等)與繪圖選項(Graphics Options如PlotLabel, AspectRatio等)，也就是說所有的Graphics Objects都是由此三要素組成的。
- GrayLevel[*level*] 爲一圖形指令，專管黑白兩種顏色的調配，其濃度由0(黑)到1(白)變動。
- Hue[h]

  爲一圖形style的指令，指的是從0到1之間的系列色彩。

  Hue[h,s,b] 指定系列色彩(Hue)、色飽度(Saturation)與亮度(Brightness)。其範圍爲0到1。
- Line[$\{\{x_1,y_1\},\{x_2,y_2\},\cdots\}$] 經過點$(x_1,y_1),(x_2,y_2),\cdots$的直線。
- ListPlot[$\{y_1,y_2,\cdots\}$ 畫出點$(1,y_1),(2,y_2),\cdots$。

  ListPlot[$\{(x_1,y_1),(x_2,y_2),\cdots\}$ 畫出點$(x_1,y_1),(x_2,y_2),\cdots$。

- `MatrixForm[`*list*`]`　將串列*list*的元素用矩陣的形式列印出來。
- `N[`*expr*`]`　算式*expr*的近似值，其精確度爲預設值6位。

  `N[`*expr*, *n*`]`　算式*expr*的近似值，其精確度爲*n*位。
- `Plot[`f, $\{x, xmin, xmax\}$`]`　函數f在區間$[xmin, xmax]$的圖形。

  `Plot[`$\{f_1, f_2, \cdots\}$, $\{x, xmin, xmax\}$`]`　將函數$f_1(x), f_2(x), \cdots$在區間$[xmin, xmax]$的圖形放在同一個座標平面上。
- `PlotJoined->True`　在`ListPlot`所描的點中，依序將相鄰兩點用直線連結。
- `PlotStyle->`$\{\}$　爲指令Plot與ListPlot中用來指明所要畫之線或點的style的一種選擇。
- `Rectangle[`$\{xmin, ymin\}$, $\{xmax, ymax\}$`]`　填滿顏色的長方形。
- `RGBColor[`*red*, *green*, *blue*`]`　爲一圖形style的指令，專管三基本色紅(*red*)、綠(*green*)、藍(*blue*)的調配，其濃度由0到1變動。
- `Show[`$\{plot1, \cdots\}$, *option*`]`

  將數張圖片*plot*1, ⋯拼成一張成爲一個圖形顯現出來。

  `Show[GraphicsArray[`$\{plot1, plot2, \cdots\}$`]]`

  將圖形$\{plot1, plot2, \cdots\}$橫向排列。

  `Show[GraphicsArray[`$\{\{plot1\}, \{plot2\}, \cdots\}$`]]`

  將圖形$\{plot1, plot2, \cdots\}$縱向排列。

  `Show[GraphicsArray[`$\{\{plot1, plot2, \cdots\}, \cdots\}$`]]`

  將圖形以二維矩陣的形式排列。
- `Sum[`f, $\{i, imax\}$`]`　這些$f(i)$的和，i從1到*imax*。

  `Sum[`f, $\{i, imin, imax\}$`]`　這些$f(i)$的和，i從*imin*到*imax*。

  `Sum[`f, $\{i, imin, imax, di\}$`]`

  這些$f(i)$的和，i從*imin*到*imax*，其間距爲*di*。
- `Table[`*expr*, $\{imax\}$`]`　產生一包含*imax*個*expr*的串列。

  `Table[`*expr*, $\{i, imax\}$`]`

  產生一*expr*之値的串列，i從1到*imax*。

  `Table[`*expr*, $\{i, imin, imax\}$`]`

  產生一*expr*之値的串列，i從*imin*到*imax*。

  `Table[`*expr*, $\{i, imin, imax, di\}$`]`

產生一$expr$之値的串列，i從$imin$到$imax$，其間距爲di。

`Table[`$expr$, $\{i, imin, imax\}$, $\{j, jmin, jmax\}$`]`　產生一$expr$之値的二維串列，i從$imin$到$imax$，而j從$jmin$到$jmax$。

- `Text[`$expr$, $\{x, y\}$`]`　以(x, y)爲中心點的文字$expr$。
- `Ticks->None`　指令Show中的選項之一，指明座標軸之刻度，其預設値爲`Automatic`，而設定爲`None`時就沒有刻度記號出現。

參考文獻

[1] Abell, Martha L. and Braselton, James P. :*Mathematica by Example*, 2nd Edition, Academic Press, San Diego, 1997.

[2] Clapham, Christopher: *A Concise Oxford Dictionary of Mathematics*, Oxford University Press, Oxford/New York, 1990.

[3] Daintith, John/Nelson, R.D.: *The Penguin Dictionary of Mathematics*, Penguin Books Ltd., 1989.

[4] Nelsen, Roger B.: *Proofs Without Words* , MAA, Wahington D.C., 2000.

[5] 沈淵源，從尤拉數e到Stirling常數，數學傳播第二十卷第一期(77)，85年3月，第34–45頁。http://episte.math.ntu.edu.tw/articles/mm/mm_20_1_03/index.html

[6] 蔡聰明，《數學的發現趣談》，三民書局出版，90年2月。