

# 第七講 12/1 數、計算與文明

Moore 定律、電腦每兩年，容量就大一倍，速度快一倍。

★ [了解創造的歷程，有助了解創造本質。  
能提升個人創造力、完備個人人文素養。]

## 語言內建基本運算(計算)

電子計算機的發明並非靈光一現，而是經年累月的進化，數量觀念伴隨語言而生。

- 智人必定有語言，但不一定有文字。
- 文字是創造文明的關鍵，沒有就很難處理複雜問題。  
↳ 就沒有數字的計算。(較大數值就不行了)

## 計算工具 第一代：算籌/算盤

「計算機」概念在自動。

圓周率 記作  $\pi$ 、 $P = 2\pi r$  為定義。

17 世紀以前，只有割圓術能測量。→ 內接圓。

各種割圓術表示，巴比倫、60 進位 / 中國  $\sqrt{2}$  萬高。

阿基米得

劉徽

祖沖之

切了 100 多邊。

200 多邊，設半徑長一百萬

一萬多邊。

$$\frac{22}{7} < \pi < \frac{223}{71}$$

$$\pi \approx 314,1866$$

半徑一千萬

維持近一千年。 ← 密率  $\frac{355}{113}$

需求才是推動我們的真正動力。

自動機械計算機。〈第二代工〉

航海帶動相關計算興起。(還有零件)

當時已有指數對數、(對數律)、必須查表確認  
(17th). 但仍有誤差, 進化之後的計算方式即「微積分」

Babbage、數學家 → 計算機製作

差分機、⇒ 分析機、(從甲卡織機概念延伸)

(用差分機的井偷偷研究)

※ 差別在於「可變程式」

分析機

分成三塊

設計

倉儲: 儲存參數與輸入數值

工廠: 計算

傳輸: 讓數值從 store 送到 mill

但可惜沒製造成中。不過催生了「程式設計」的概念、

科學運算。

先利用計算算出星球的軌跡表, 才找到星球、

(還是使用大量人力去做計算) 《關鍵少數》

電機型計算機。⇒ 可儲程式的電子計算機

(但沒有倉儲)。

馮諾以曼

1. 十進制 → 二進制。  
2. 指令 + 資料, 存在機器內 ⇒ 催生「記憶體」

1952、電晶體取代真空管

高階程式誕生、文字進入電腦、  
(188)

計算、方法和工具、如人兩腿、前後交替、而致進步。

## 感想

聽完有聲書的第一個感想 = 這就是「文化脈絡中的數學」吧！前幾堂從數學(郵票)出發,這次則是從「計算」開始,數學的出現的確與計算脫不了關係。

由於電腦在現代科技實在太理所當然,從來沒想過在電子計算機出現以前,所有複雜的計算都必須通過「人」來完成。

我覺得數學就像是施魔法一樣,必須先有魔力、元素(數)的概念,透過正確的詠唱(計算)咒語,才能施放各種魔法。魔法越難,所需時間及成本也越大,也越少人能理解、能詠唱。

光是從計算方式演變去看待計算的發展,完全無法想像用機械、用電流、用各項工具,能夠算出天體運行,能夠解出各種方程式。

果然數學家就是魔法師吧!

# 第七講、數、計算與文明

林欽提

一開始提到的 Moore's Law 應該是叫做摩爾定律。雖然只是小口誤，但會以為是化學上的莫爾。這半導體也是我現在在學的（有夠難！），相信這個

semiconductor 產業會越做越小越精密，今天幸好亂...

老師第二部份提到的“沒人知道語言是怎麼來的”其實我有聽到一個非常合理的解釋，我有聽說人是從“叫聲”來的，我覺得超有道理，當做固定的事會有固定的叫聲，久而久之演化成來的，但我其實最想知道，非常好奇這一切是如何來的，我現在寫的手足如何來的。

再來我算是長眼了，原來 V 有 55 的意思，而且羅馬數字有 5 進位！

接着原來 Calculate 是從 Calculi 來的，這也超合理的，從石頭計數。

(這邊開始我換了網路上說的正確握筆方式，手會比較穩)

不當時只要死住就是可以刻在臺上的現在打太高或本書，人類真的很強！

微積分真的是陪我後半人生的東西，高之前就只知道是非常非常難，然後0概念，結果學了之後，其實真的難，但說真的，超好玩。我現在整個 mind 裡一堆微積分，工數，程式設計的核心技術在重複，我這邊有一個完美範例。

```
S0951033FinalVersion.cpp X
S0951033FinalVersion.cpp > main()
1 #include <iostream>
2 using namespace std;
3 /*思路：已知西元1年1月1號為星期一，非閏年有365天，
4 365/7=52...1，則西元2年1月1號為星期二，倘若遇到閏年，
5 則會多一天*/
6 int main(){
7     int year;
8     cout << "Enter a Year:";
9     cin >> year;
10    int day[12] = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};
11    //逢四年一閏，逢百年不閏，逢四百年再閏，若為閏年，二月會變成29天
12    if(year%4==0){
13        if(year%100==0){
14            if(year%400==0)
15                day[1] = 29;
16            else
17                day[1] = 28;
18        }
19        else
20            day[1] = 29;
21    }
22    else{
23        day[1] = 28;
24    }
25    /*這邊開始宣告從一月到十二月需要的變數，而jan, feb這些代表的是，該年該月的禮拜一
26    前面會有幾個空格，一月的話，思路是這樣，我們已知兩件事，第一，西元元年一月一號是禮拜一，
27    第二，一年365天，而365/7會整1，那就代表說，西元二年一月一號是禮拜二，再來重點，若遇到閏年，
28    會加一天，所以我以幾年再加上總共過了幾個閏年，再除以七，即可知道該年一月是禮拜幾，而再來二月算法，
29    已一月第一天加上一月天數除以七，就知道二月第一天是禮拜幾，以此類推*/
30    int a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x;
31    int jan, feb, mar, apr, may, jun, jul, aug, sep, oct, nov, dec;
32    jan=(year-1)/4-(year-1)/100+(year-1)/400; //利用過了幾年知道我輸入那一年的一月一號是禮拜幾
33    feb=(jan+day[0])%7; //Feb變數表該年二月一號禮拜幾，以此類推
34    mar=(feb+day[1])%7;
35    apr=(mar+day[2])%7;
36    may=(apr+day[3])%7;
37    jun=(may+day[4])%7;
38    jul=(jun+day[5])%7;
```

這邊反正就是先跟電腦說，閏年的規則是什麼，因為電腦不知道

```
... S0951033FinalVersion.cpp X
WORKSPACE
  .vscode
  1.cpp
  1.exe
  c.c
  c.exe
  Calendar.cpp
  chan.cpp
  chan.exe
  check.cpp
  check.exe
  DuanKao.cpp
  DuanKao.exe
  for.cpp
  forwhechain.cpp
  forwhechain.exe
  HowMuch.cpp
  HowMuch.exe
  S0951033.cpp
  S0951033.exe
  S0951033FinalVersion.cpp
  S0951033FinishedVersion.exe
  S095103395.cpp
  tempCodeRunnerFile.exe
  Test.cpp
  Test.exe
  workspace.code-workspace
  Yeha.cpp
  Yeha.exe
  大綱
S0951033FinalVersion.cpp X
main()
{
    /*這邊開始宣告從一月到十二月需要的變數，而jan,feb這些代表的是，該年該月的禮拜一
    前面會有幾個空格，一月的話，思路是這樣，我們已知兩件事，第一，西元元年一月一號是禮拜一，
    第二，一年365天，而365/7會餘1，那就代表說，西元二年一月一號是禮拜二，再來重點，若遇到閏年，
    會加一天，所以我以幾年再加上總共過了幾個閏年，再除以七，即可知道該年一月是禮拜幾，而再來二月算法，
    已一月第一天加上一月天數除以七，就知道二月第一天是禮拜幾，以此類推*/
    int a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x;
    int jan, feb, mar, apr, may, jun, jul, aug, sep, oct, nov, dec;
    jan=(year+(year-1)/4-(year-1)/100+(year-1)/400)%7;//利用過了幾年知道我輸入那一年的一月一號是禮拜幾
    feb=(jan+day[0])%7;//feb變數表該年二月一號禮拜幾，以此類推
    mar=(feb+day[1])%7;
    apr=(mar+day[2])%7;
    may=(apr+day[3])%7;
    jun=(may+day[4])%7;
    jul=(jun+day[5])%7;
    aug=(jul+day[6])%7;
    sep=(aug+day[7])%7;
    oct=(sep+day[8])%7;
    nov=(oct+day[9])%7;
    dec=(nov+day[10])%7;
    /*這邊就是連續輸出十二個月，
    而我換行的依據是，空格加上跑到的天數，
    跑到那天如果加上空格除以七餘數是0，換行
    以下這個來解釋，接著以此類推12次
    cout<<"某月"<<endl;
    cout<<" SUN MON TUE WED THU FRI SAT"<<endl;
    for(變數1=1;變數1<=某月的變數;變數1++) {
        cout<<" "; //這邊有四個空格，才能跟一個空格然後三個英文字對齊
    }
    for(變數2=1;變數2<=day[某月在陣列裡的位置];變數2++) { //變數2指的其實就是天數，從一號開始跑
        if(變數2>=10)
        {
            cout<<" "<<變數2;
        }
        else
        {
            cout<<" "<<變數2;
        }
    }
}
```

```
... S0951033FinalVersion.cpp X
main()
{
    cout<<"\n"; //換行到下個月*/
    cout<<"January"<<endl;
    cout<<" SUN MON TUE WED THU FRI SAT"<<endl;
    for(a=1;a<=jan;a++) {
        cout<<" ";
    }
    for(b=1;b<=day[0];b++) {
        if(b>=10)
        {
            cout<<" "<<b;
        }
        else
        {
            cout<<" "<<b;
        }
        if((jan+b)%7==0) {
            cout<<"\n";
        }
    }
    cout<<"\n";
    cout<<"February"<<endl;
    cout<<" SUN MON TUE WED THU FRI SAT"<<endl;
    for(c=1;c<=feb;c++) {
        cout<<" ";
    }
    for(d=1;d<=day[1];d++) {
        if(d>=10)
        {
            cout<<" "<<d;
        }
        else
        {
            cout<<" "<<d;
        }
        if((feb+d)%7==0) {
            cout<<"\n";
        }
    }
}
```

這邊就是我所謂的重複  
動作是：先算1號是禮拜幾  
印空格(比如說  
1號只有1個格子，要加  
1格才可配合1位  
數以上。  
印天數

```
... S0951033FinalVersion.cpp X
main()
{
    cout<<"\n";
    for(v=1;v<=day[10];v++) {
        if(v>=10)
        {
            cout<<" "<<v;
        }
        else
        {
            cout<<" "<<v;
        }
        if((nov+v)%7==0) {
            cout<<"\n";
        }
    }
    cout<<"\n";
    cout<<"December"<<endl;
    cout<<" SUN MON TUE WED THU FRI SAT"<<endl;
    for(w=1;w<=dec;w++) {
        cout<<" ";
    }
    for(x=1;x<=day[11];x++) {
        if(x>=10)
        {
            cout<<" "<<x;
        }
        else
        {
            cout<<" "<<x;
        }
        if((dec+x)%7==0) {
            cout<<"\n";
        }
    }
    system("pause");
    return 0;
}
```

做"重複"的事到  
December

這是一下修的最喜歡的程序設計課，這第一  
個小專題 Perpetual Calendar 就萬年曆。  
我的程序很笨重，又有更簡單的，如 for while  
但那程序有點大 我不太會。  
至少是從頭到尾自己研卒慢慢摸的。

```
D:\Workspace\S0951033FinishedVersion.exe
Enter a Year: 2021
January
SUN MON TUE WED THU FRI SAT
    1  2  3  4  5  6  7
 10 11 12 13 14 15 16
 17 18 19 20 21 22 23
 24 25 26 27 28 29 30
 31
February
SUN MON TUE WED THU FRI SAT
    1  2  3  4  5  6
    7  8  9 10 11 12 13
   14 15 16 17 18 19 20
   21 22 23 24 25 26 27
   28
March
SUN MON TUE WED THU FRI SAT
    1  2  3  4  5  6
    7  8  9 10 11 12 13
   14 15 16 17 18 19 20
   21 22 23 24 25 26 27
   28 29 30 31
April
SUN MON TUE WED THU FRI SAT
    1  2  3  4  5  6  7
    8  9 10 11 12 13 14
   15 16 17 18 19 20 21
   22 23 24 25 26 27 28
   29 30
May
SUN MON TUE WED THU FRI SAT
    1  2  3  4  5  6  7
    8  9 10 11 12 13 14
   15 16 17 18 19 20 21
   22 23 24 25 26 27 28
   29 30 31
June
SUN MON TUE WED THU FRI SAT
    1  2  3  4  5
    6  7  8  9 10 11 12
   13 14 15 16 17 18 19
   20 21 22 23 24 25 26
   27 28 29 30
July
SUN MON TUE WED THU FRI SAT
    1  2  3
    4  5  6  7  8  9 10
   11 12 13 14 15 16 17
```

2021 就大概這樣。

最後想提下，Konrad Zuse 那畫真好看...

這期不錯看，因為跟自己比較有共鳴，而且  
算有興趣，但我最喜歡的還是艾雪那期