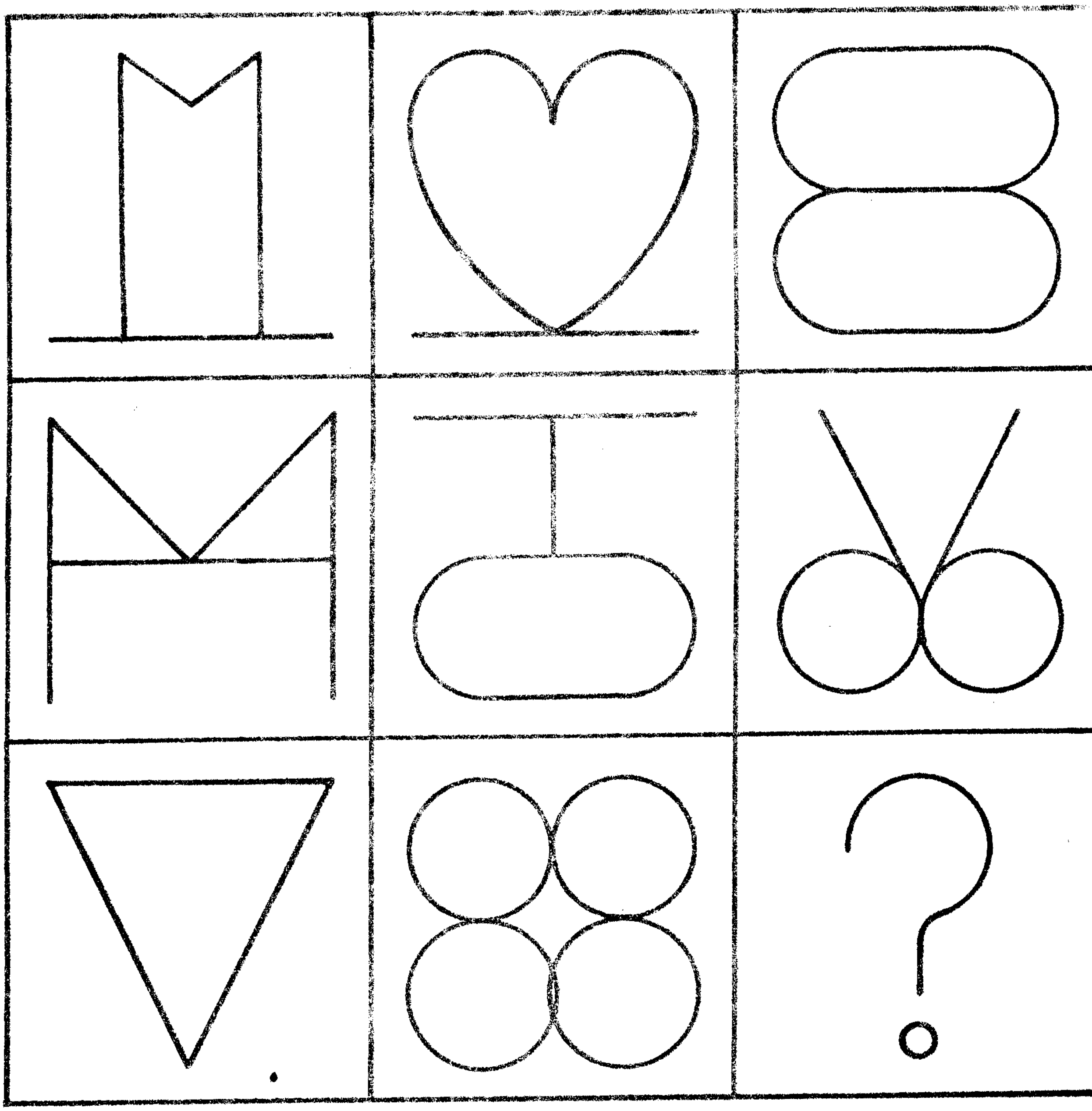


SCHOOL MATHEMATICS NEWSLETTER

June 1983

Mathematics Section, Advisory Inspectorate

Education Department, Hong Kong

5

Please ensure that every member of your mathematics staff has an opportunity to read this Newsletter.

The views expressed in the articles in this Newsletter are not necessarily those of the Education Department, Hong Kong.

Cover : The figures in the grid on the cover are designed in the same manner. What's special about them? What should the figure represented by the question-mark look like? Here are the answers : The figures are all symmetric, the right sides being the integers 1, 2, 3, etc. and the left sides their mirror images. The question-mark represents P9.

IN THIS ISSUE

FOREWORD

This fifth issue of the School Mathematics Newsletter (SMN) brings you a variety of ideas, articles and news on Mathematics and Computer Studies. These include features on special topics of the two subjects, student homework assignment, teaching experience, in-service teacher training, current trends in mathematics teaching, views on computer literacy, criticism on BASIC, etc., together with puzzles, games, cartoons and news on new publications. I hope you find this issue informative besides enjoyable.

The new cover design and new layout are intended to make the SMN more inviting and eye-catching. I would like to hear your comments and suggestions on both the layout and the content.

May I take this opportunity to thank all those who have contributed to this issue of the SMN.

S. B. Teng
Principal Inspector (Mathematics)

漫談小學生的算術家課

陳衍輝

	算草
一　農場裏有鷄12隻，有鴨19隻。問共有動物多少隻？ 12隻＋19隻＝31隻 答：共有動物31隻	12隻 ＋19隻 31隻
二　一隻鷄可賣得30元，一隻鴨可賣得20元。有鷄12隻，鴨19隻，問共可賣得若干元？ 30元×12＋20元×19＝740元 答：共可賣得740元	30元 ×12 30 60 360元 20元 ×19 20 180 380元 360元 ＋380元 740元

上列是兩題從小學生練習簿抄出來的家課。教師吩咐他不須抄題，文字題是筆者特地寫下來，以方便以下的討論。

這個小學生依從老師所教導的格式，來計算文字題。這些格式是：

㈠為文字題列出橫式。
㈡在算草範圍寫出直式。
㈢用〞答：〞開始，寫出解答字句。
㈣在所有答案下畫二直線，以清楚顯出答案。
㈤在每一步驟，書寫單位。

有些教師確實很注重這些格式，學生每做一題算術，必須用這些格式進行。因為他們認為這些格式，除能幫助學生解決

數學問題外，還收到下列額外的效果：

㈠使學生從數學練習中，得到書寫整齊的訓練和習慣。

㈡給予學生用文字表達算術結果的機會，使算術和語文均有俾益。

㈢作爲向校長及家長的交代，證明曾經督促學生做練習。

本來，要求學生以這種詳盡的格式來做算術練習是一件好事。但是近年來小學生的家課實在太多了。一週內，一個小學生要做不少國語作業，健教作業，要寫不少英文生字，社會生字等等。上述的〃額外效果〃，已能從其他科目達到，所以不少家長見自己的子女至深夜還未做完家課，便索性自已親自動手，替子女做了，免致子女受教師的責罰。

有些算術問題，數字很顯淺，學生在未列出橫式之前，就像解答電視節目的校際問答一般，已憑心算，計出答案。爲什麽還強要他們列出橫式，列出算草，寫好單位，寫好答案，一字不漏來滿足老師的要求呢？

上列兩題，如照下列方法計算及列式，應該更爲簡潔實際。

一、農場裏有雞 12 隻，有鴨 19 隻。問共有動物多少隻？ 答：共有動物（ 12 + 19 ）隻 = 31 隻	$\begin{array}{r} 12 \\ +19 \\ \hline 31 \end{array}$
二、一隻雞可賣得 30 元，一隻鴨可賣得 20 元。有雞 12 隻，鴨 19 隻，問共可賣得若干元？ 答：共可賣得（ 30 × 12 + 20 × 19 ）元 = 740 元	$\begin{array}{r} 12 \\ \times 30 \\ \hline 360 \end{array}$ $\begin{array}{r} 19 \\ \times 20 \\ \hline 380 \end{array}$ $\begin{array}{r} 360 \\ +380 \\ \hline 740 \end{array}$

這種解答方式的特點有三：

㈠把文字和數式聯成一答案句子。

㈡把運算部份用括號括起來，單位寫在括號外。如有若干演算步驟，全部用等號串聯起來。

㈢算草裏不再要求書寫單位。（當然寫亦不拘。）

有些現行的課本都逐漸推廣這種解答方式，下列為它的優點：

㈠訓練學生用文字表達數學意念，使數式作為文字的一部份。

㈡從數學觀點來看，這種解答方式比較合理。因為當我們把數字運算的時候，單位是不在運算之列的。當運算完畢後，單位才與數字合拼，構成答案。這種解答方式，便正正寫出構成答案的程序。

從函數的角度來看，加、減、乘、除都是二元函數。此等函數的定義域為純數構成的序偶，以加法來說：

$$+ : (12, 19) \xrightarrow{\text{加 成}} 31$$

$$+ : (12\text{隻}, 19\text{隻}) \xrightarrow{\text{不合"+"的定義}}$$

惟有將（12隻，19隻）看成（12，19），使函數"加"得以進行，然後再將單位附在運算的結果後面。那麼從數學觀點來看，為何不把

12隻 + 19隻 = 31隻

寫成

(12 + 19) 隻 = 31隻，

那豈非更合理！

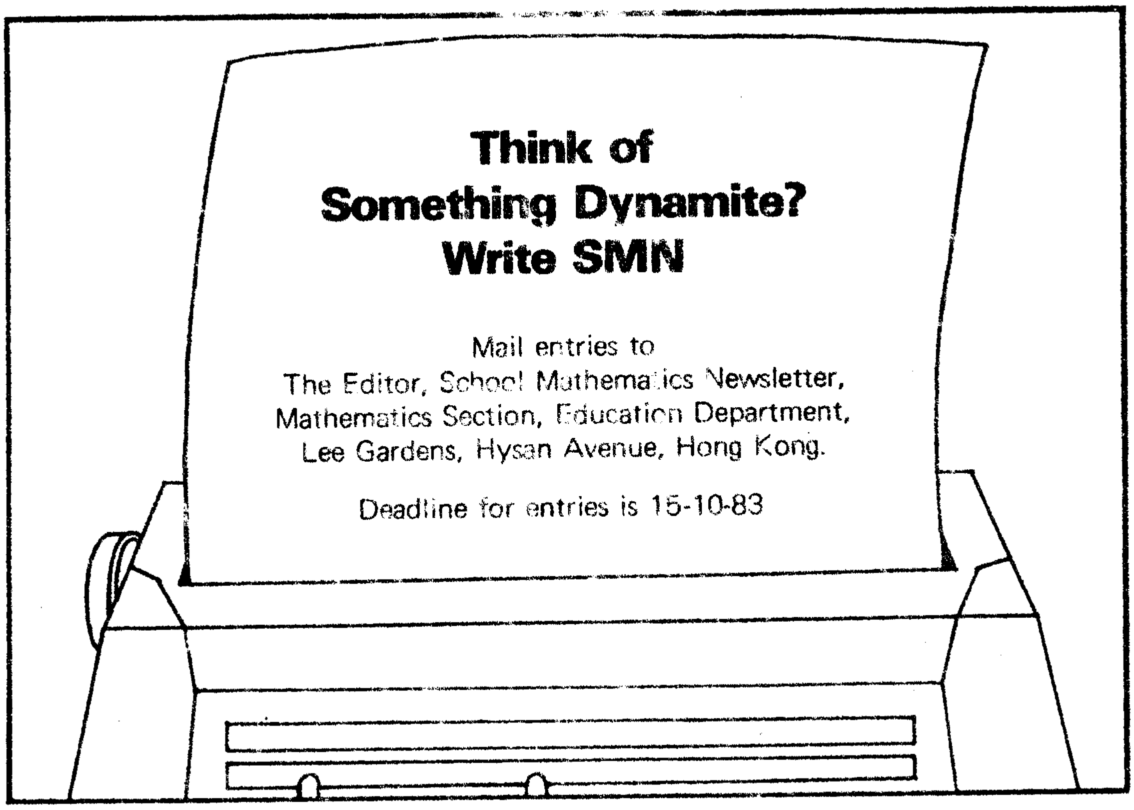
有些學生做家課時，把答案做對了，算草亦做對了，但是在算草沒有寫單位，就如

12隻 + 19隻 = 31隻	$\begin{array}{r} 12 \\ +19 \\ \hline 31 \end{array}$

教師可能以不合格式（算草裏沒有寫單位）爲理由予以扣分，並命令學生在改正欄裏將題目重做。但從函數的觀點來看，函數"加"是不採納單位的，寫單位與不寫單位是隨學生之便。寫了單位亦只把它當作注釋般看待，不寫單位亦不妨。

單位的書寫過多，可能是一種負累，應可免則免。但在處理複名數問題時，單位的書寫則無法省略。

以上的討論，提供與教師和家長參考，希望對小學生的學習有所幫助。SMN



淺談空間想像力之訓練

黃毅英
黃棣珊紀念中學

隨着教育之全面化與普及，數學不再局限於知識技巧的授受。中學數學教育之目標應在於培養學生的計算能力、邏輯思維能力與空間想像力（見〔1〕、〔2〕、〔3〕）。但作爲一個普通教師，怎樣才可達到上述之目標呢？

計算能力之訓練在中學數學課程中已算不少，邏輯思維也可從定理之推演中得到啓示，但空間想像力便較爲空泛和難於測度的了。我們希望藉本文與大家討論一下這方面的一點點經驗。

現時，學生在中四便可開始從立體三角學之計算接觸到三維空間，預科班時更會涉及直線與平面的方程。兩者似乎較難拉上關係，故似不易貫通，亦不能顯突空間想像力之訓練。其實在此之前，中三學生已可從體積計算中接觸柱體、正立方體、錐體等模型，而我們認爲製作立體模型不失爲一種引起空間想像力之具體有效方法。

在數學課外活動裡〔4〕，我們曾嘗試指導中三學生自製立體模型，讓他們對空間形像更爲親切，我們不只局限於正多面體 (Platonic Solids) 之正規製作（用平面圖則摺成），我們任由學生貼貼糊糊，又或將正立方體割成三個錐體、用幾個立體拼成非正規凹多面體 (irregular concave polyhedra) 等…務求學生親手將模型製成，並對其每一結構都熟識清楚。

其實單從立體模型所引出的範圍也十分多樣化和廣泛，所涉及的深度亦富彈性，比如我們又可探討圓錐體而引出圓錐曲線 (conic sections)，而以一套「錐體與球體」教育電視節目作壓軸。我們相信，這些都要比課堂講解來得生動有趣。

此外，中四學生每每覺得解立體三角學問題傷透腦筋。當然除了在黑板講解外，我們必得利用立體模型，甚至窗門

、晉朱、柏何等。但學生仍感到陌生和害怕，因爲他們和模型之間仍有一度隔膜。比方以下一題便會經花費不少唇舌也未能把圖像顯示給學生的了：

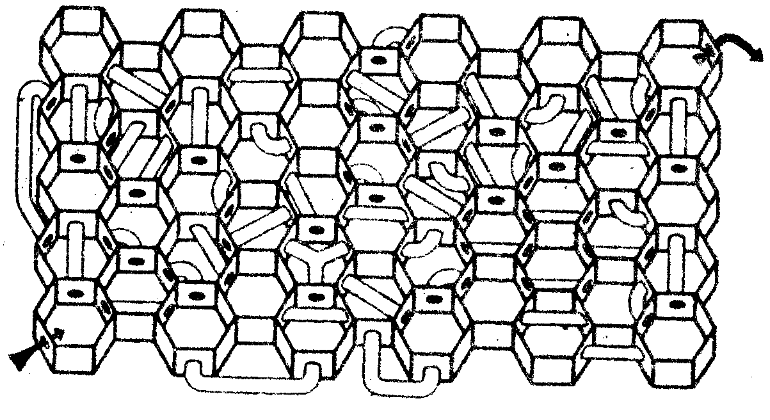
「今有一旗竿高二十米，向東傾斜十度。太陽之仰視角爲旗竿南面之七十度，求影長。」

結果我們用另一種方法，學生便於十數分鐘的時間洞悉圖的結構與解答方案了！我們叫學生自己按題目製作模型，學生不只即能理解立體結構，最重要的是他們自己能將模型一步一步的製作出來，這更使他們留下深刻印象。

在預科的情況，由於在立體幾何的教授中，一開始便涉及抽象的代數算式，所以使學生難以想像。假使學生在低班素有訓練，消除了恐懼心理，再從具體模型出發，問題必不致太大。從初中所接觸過的模型出發，更能貫徹以往空間想像力之訓練。比方圓錐體及其切面便是一例。

在圓錐切面的例子中，我們更可從幾個角度探討問題：用模型看出幾個切面之形狀研究其方程；用幾何方法證明切面是橢圓（用 Pendelin 球體）；用代數計出切面方程；甚或帶到斜面噴水之濕潤地帶爲一橢圓等〔5〕；務求一題多解，使學生貫通來龍去脈。

但最重要的是要學生不生畏懼，這便得從低年班做起了，而數學遊戲是一值得參考之途徑。比如立體打井〔6〕及一些特別的迷宮（見附圖）都會是相當富吸引力的。



〔參考書目〕

〔1〕 中學數學教材教法，十三院校協編，1980。

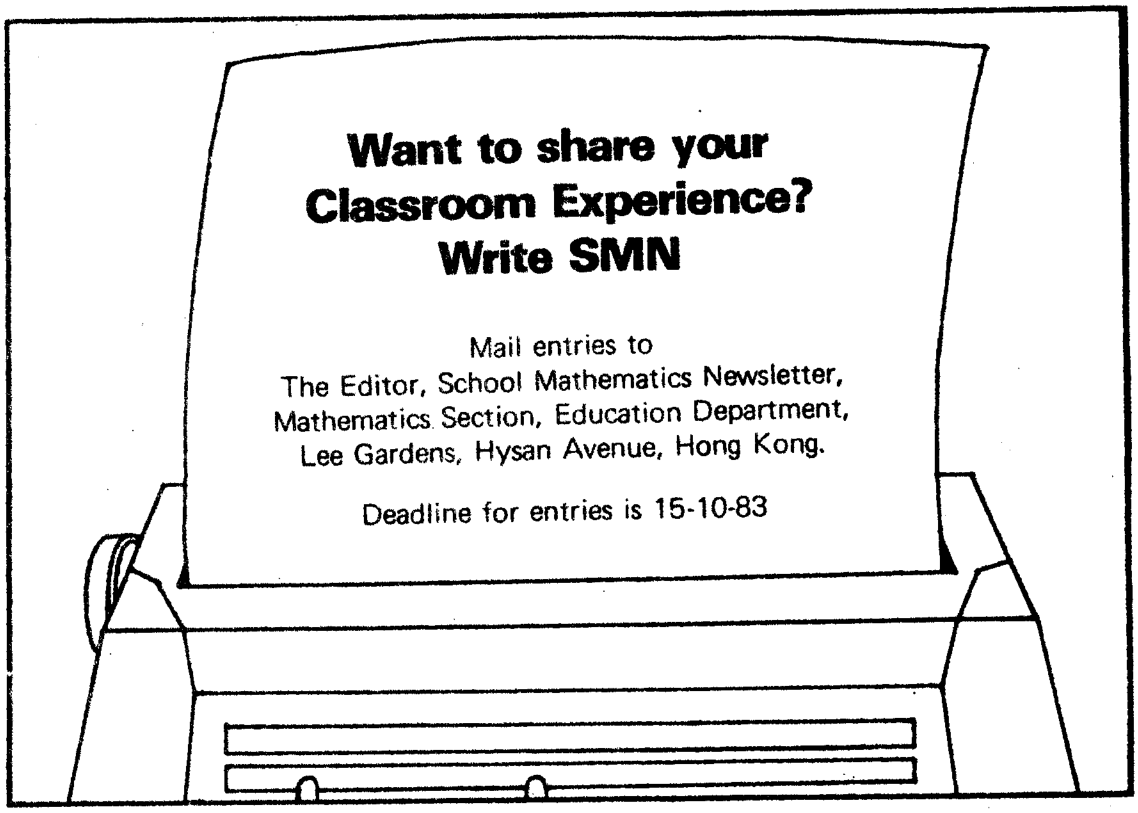
〔2〕 爲甚麽要學習數學？蕭文強，學生時代出版社。

〔3〕 對國內中學數學教學大綱的一些意見——兼談數學教育之目的，陳鳳潔、周錫昌、蕭文強，數學教學論叢，1981。

〔4〕 辦數學學會的一些經驗，黃毅英，數學教學季刊(4)，1982。

〔5〕 The Cross Section of Surfaces, N.Y. Wong, Math. Bulletin (2), 1981.

〔6〕 匈牙利幻立方，唐伯螢，數學通報(2)，1981。



遼闊的概率天地

林建
香港大學

時下一般的中學教科書，在討論概率論時所用的例子，不外都是抽紙牌，抽球或是擲骰，很容易給人一個印象，就是概率的計算，只能應用於一些有關碰機會的遊戲。其實概率論是一門有廣泛應用的學科。日常生活中，概率的應用也是很多的。本文試圖漫遊一下遼闊的概率世界，漫遊的起點，就選在香港大學的紐魯斯樓吧！

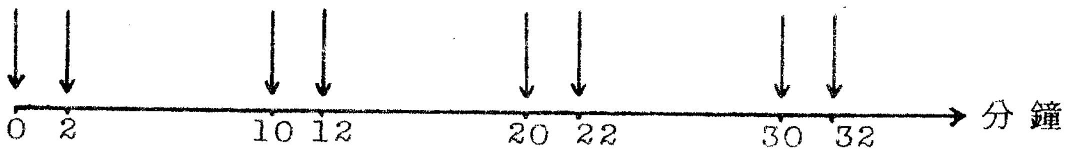
等電梯的概率

香港大學的紐魯斯樓，樓高十層，我的辦公室在七樓，而我上課的班房則在二樓，所以我常常要坐電梯來往二樓、七樓之間，日久之後，就讓我發現了一個有趣的現象。每當我在七樓等電梯去上課時，第一部到達七樓的電梯，多是往上跑的。於是那些和我一同等電梯上十樓開會的同事，都比我先上電梯。但是當我上完課在二樓等電梯回辦公室時，那些要到地下去的學生，又通常會快我一步，先等到向下的電梯。爲什麼和我一起等電梯的人，總能較快地等到電梯呢？是幸運之神在玩弄我，還是別有原因呢？

想深一層，這個現象是有道理的。爲了清楚地說明這個道理，讓我假設紐魯斯樓只有一部電梯。當我在七樓等電梯時，如果電梯在七至十樓之間，那麼它到七樓時必然向下。反之，如果電梯在地下和七樓之間，那麼它到達七樓時必然向上。但電梯在七至十樓的機會較它在地下至七樓的機會爲小（比例爲３：７），所以我只有３／１０的機會能夠先上電梯。但是當我在二樓等電梯時，情況便不同了。除非電梯在二樓之下，否則它到達二樓時必然向下。我能夠先上電梯的機會只有２／１０。你看，這樣一個簡單的概率計算，就能解決了日常生活中的一個疑惑了。

某路巴士，平均每五分鐘開出一班。我是這路巴士的常客，每天等巴士時，閒來無事，便計算一下要等多久才有巴士到站。有一天心血來潮，把歷來等巴士的時間平均一下，發覺平均3.4分鐘才有一部巴士到站。但這個數字好像很不合理。既然巴士每五分鐘一班，最長要等五分鐘，最短要等零分鐘，那麼平均不是要等2.5分鐘嗎？爲什麼我偏要等三分多鐘呢？

其實，問題的癥結，就在巴士平均五分鐘一輛這個假設上面。通常巴士並不絕對準時，有時巴士開出的時間，相隔比五分鐘多，但有時又比五分鐘少。上面的假設，只可解釋爲相隔時間平均爲五分鐘，而不眞的是每五分鐘開出一輛巴士。可以想像以下的情況，這路巴士每隔兩分鐘及八分鐘開出一輛，這時巴士到站的時間，會如下圖箭嘴所表示：



平均來說，這路巴士還是五分鐘一輛。但從乘客的角度來說，在0（分鐘）－2（分鐘）（或10－12，20－22，30－32，···）抵達的乘客，平均只要等一分鐘；而在2（分鐘）－10（分鐘）（或12－20，22－30，···）抵達的乘客，平均要等四分鐘。由於0－2這個區間，只爲2－10這個區間的1/4，所以前者的人數，爲後者之1/4，因而對整體乘客而言，平均要等

$$\frac{1}{5}(1)+\frac{4}{5}(4)=3.4\text{（分鐘）}$$

當然，上面例子也是不符合實際的，巴士不會準時五分鐘一班，也不會準時兩分鐘及八分鐘一班。我們只是借用上面的例子來說明乘客平均等候時間（W），並不是巴士平均相隔時間（M）的一半。事實上，「應用概率論」中的「排隊論」（queueing theory），就給出了W及M的關係。

用一點概率理論，可以證明在一定的假設底下，W與M的關係爲：

$$W = \frac{M}{2}\left[1 + \frac{S^2}{M^2} \right]$$

其中 S^2 爲巴士相隔時間的方差（**variance**）。如果巴士眞的是準時M分鐘一班，那時 $S^2 = 0$，我們便有 $W = \frac{M}{2}$。但由於巴士不能絕對準時，所以通常會有 $W > \frac{M}{2}$。其實，撇開概率計算，我們等巴士的時間，除了和巴士的多少有關以外，還應該和巴士的準時與否有關，這不是合情合理嗎！概率計算只是把這合情合理的現象，加以數量化而已！

<u>觀點與角度</u>

上面的一個例子，乘客的而且確地平均等了3.4分鐘，但從巴士公司的立場來說，也是的而且確，平均每五分鐘有一輛巴士，這是一個「觀點與角度」的問題。這類「觀點與角度」的分歧，在日常生活中，是層出不窮的。例如我們統計系共開21科，其中20科班上有50人。餘下一科有學生200人。從系的立場來看，平均每科有學生

$$\frac{(1)(200) + (20)(50)}{21} = 57.1(\text{人})$$

爲了要知道每一個學生上課時的人數，學生會進行了一項調查，把調查問卷派給所有唸統計的學生（如果一名學生修多於一門統計課，他便會收到多於一份問卷。），問卷要他們填上班上的人數。這時分發給前20科同學的問卷（共 20 × 50 = 1000(份)），答案是50。而發給最後那科同學的問卷（共200份），答案則爲200。所以就整體學生來說，平均每班人數爲：

$$\frac{(200)(200) + (1000)(50)}{1200} = 75$$

由於觀點與角度有所不同，計算得到的平均數便有出入了。

類似的觀點與角度問題，例子多得很。有一名餐廳東主，認爲他的餐廳，生意並不理想，平均每小時只有三、四名顧客。但從顧客角度來看，餐廳的生意十分繁忙。此無它，因爲大部份顧客，都在繁忙時候抵達，因而只見到繁忙時候的擠迫，只有小部份顧客，能夠看到門堪羅雀的情況，但因爲前者佔了絶大多數，所以對整體顧客而言，餐廳生意興隆，但餐廳東主以每小時爲計算單位，平均人數必然少得多了。

<u>幼稚園的生日會</u>

我家的小頑皮去年九月剛入幼稚園，他很喜歡上學，而最令他開心的，就是每月一次的生日會了。他的班有 30 人，學校在每個月的最後一個星期六，都爲該月份生日的小朋友們慶祝生日。九月底他們開了第一個生日會，小頑皮興高彩烈地回家，問我是不是每月都有一個生日會。我隨口答說是的。後來細心想了一下，會不會整個月都沒有小朋友生日呢？那麽小頑皮不是會很失望嗎？於是我便坐下來算一算這個機會。我知他們九月份有三個人生日，小頑皮的生日在六月，餘下 26 人，每人有 1/11 機會在十月生日（因爲他們不在九月生日，故機會爲 1/11 而不是 1/12)，所以十月份沒有生日會的機會爲：

$$\left(\frac{10}{11}\right)^{26} = 0.084$$

還好，這機會不大，他應該不會失望吧！

上面，我只計算了十月份沒有人生日這一個概率。但全年最少有一個月沒有人生日的概率又是若干呢？計算這個概率並不容易，概率論中的「古典佔位模型」（**Classical occupancy model**）便爲這問題提出了解答。這個模型是這樣的：把 n 個球，隨意丟在 m 個球罐裏。如果用 X_r 來代表內有 r 個球的罐數，那麽便有如下的公式：

$$P(X_r = k) \approx \frac{e^{-\lambda}\lambda^k}{k!}$$

其中 $\lambda = me^{-n/m}(\frac{n}{m})^r / r!$

不錯，上面又是一個抽球模型，但這個模型應用很大。例如我們可以把球看成是人。而球罐則看成是他們的生日月份（故共有球罐 12 個）。如果班中有 30 人（即有球 30 個），要計算最少有一個月沒有人生日的概率，我們便要計算 $P(X_0 \geqslant 1)$。由於 $m = 12$，$n = 30$，$r = 0$，故 $\lambda = 0.985$。而

$$P(X_0 \geqslant 1) = 1 - P(X_0 = 0)$$

$$= 1 - \frac{e^{-\lambda}\lambda^0}{0!}$$

$$= 0.627$$

由生日會到兇殺案

上面的球罐模型，有很多應用。如果我們假設兇殺案的發生是獨立事件，我們還可以計算一下同日發生多宗兇殺案的概率。翻查 1981 年的統計數字，該年有兇殺案 106 宗。假設 1983 年兇殺案數字仍為 106 宗，那麼有一日有三宗兇殺案發生的機會為多少？

只要應用「古典佔位模型」，便可以解決上面的問題了。我們可以把每宗兇殺案當做一個球（因而有球 106 個），把他們丟在 365 個球罐內（球罐就是兇殺案的發生日期），是則 X_3 代表有多少日發生三宗謀殺案，而我們要計算 $P(X_3 = 1)$。用上面公式，$n = 106$，$m = 365$，$r = 3$，故 $\lambda = 1.11$。因而

$$P(X_3 = 1) \approx \frac{e^{-1.1}(1.1)}{1} = 0.366$$

有兩日有三宗兇殺案的概率爲：

$$P(X_3 = 2) \approx \frac{e^{-1.1}(1.1)^2}{2} = 0.201$$

法律不離概率

概率不單是紙上談兵的理論，在不少的情況底下，它能夠協助事實的判斷。美國的加州法庭，就曾有過一個利用概率計算來判案的例子。

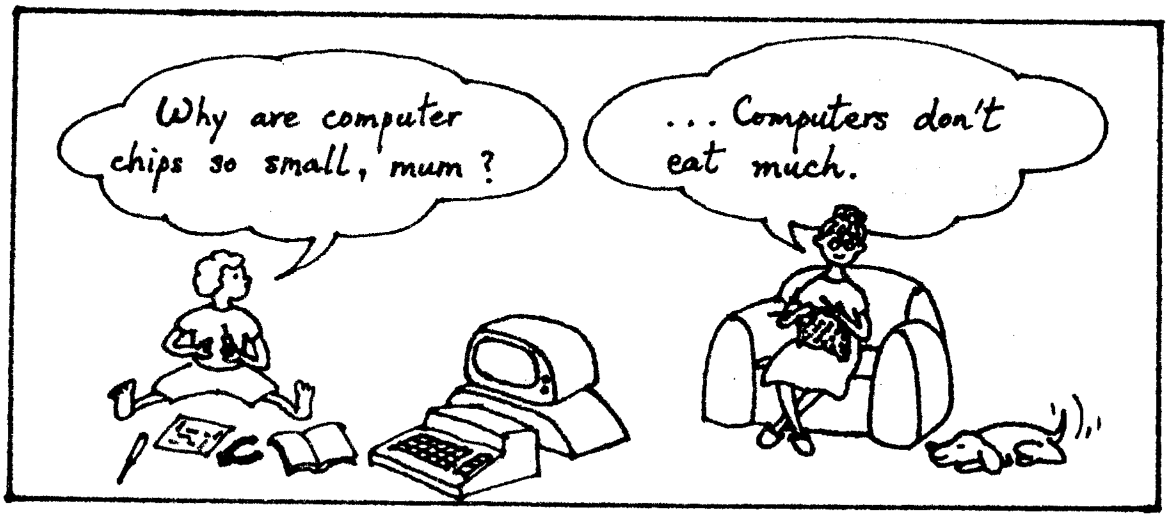
於某兇殺案的現場，有人見到兇手是一名黑人男子，有鬚，並蓄有小鬍子。同行是一名白人女子，馬尾裝，二人行兇後，坐一黃色小跑車逃去。十分鐘後，警察在公路截得一對男女，具有以上特徵，但此外並無其他佐證，指證他們爲兇手。結果勞煩了一輪概率計算，算出這樣組合的機會爲一千二百萬分之一，於是這個出名的 **People vs Collins** （**Collins** 就是案中被告的名字）案件的被告便被判定有罪。

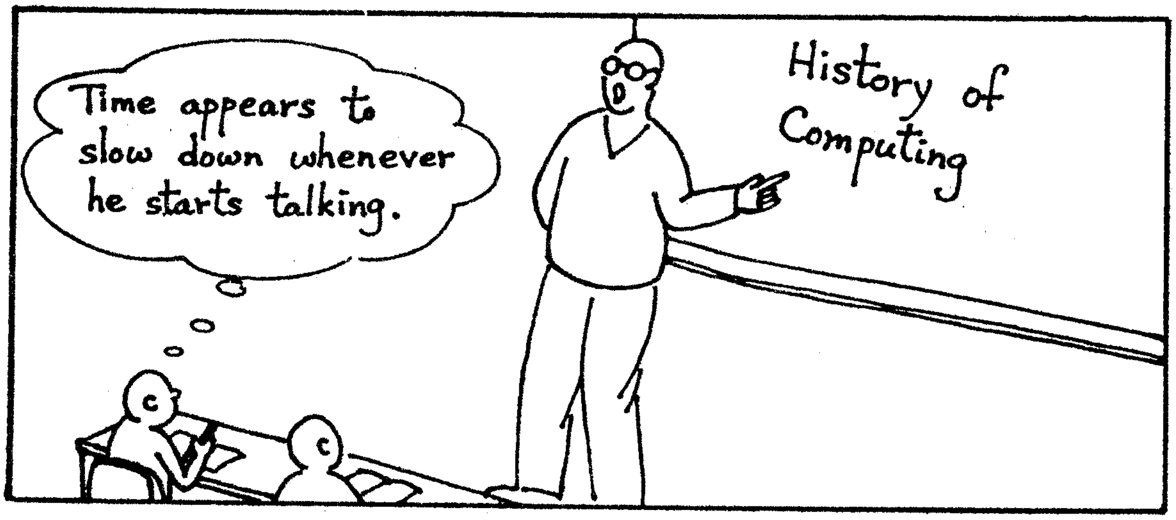
回到球罐模型

不錯，概率論有很廣泛的應用，並不只是局限於抽球及放球入罐那麼簡單。數學家們研究球罐模型（**urn model**），並不眞的是對抽球那麼有興趣，他們只是以抽球來代表一些實際發生的偶然現象而已。例如以上的「古典佔位模型」，便可以有不同的應用。事實上，研究這個模型的馮·米斯（**Von-Mise**），本身就是一個物理學家。他心目中的球是某些氣體分子，而他心目中的罐，則是氣體分子的狀態。他想研究的是氣體分子整體的分佈狀態。在不同假設底下，有關「古典佔位模型」的概率計算又會不同。而這不同的計算方法，正好標誌着古典力學和量子力學的分野。（請讀者參看本文作者和蕭文強博士合著的「概率萬花筒」一書。）

讓我們看看以下的一個球罐模型。球罐內有球 2n 個，每個球都有號碼標記，其中兩個是 1 號，兩個是 2 號，…，兩個是 n 號。今抽出球 r 個，問罐內平均會有多少

對同號碼的球？以上這個球罐模型，是早期概率論發展史上出名的例子。概率論的先驅貝努里（ Bernoulli ）研究這個模型，是爲了實際的需要，他希望估計社會上老年人喪失配偶，不能同偕白首的人數。同號碼的球，代表了一對配偶，把球抽出球罐，則代表死亡，剩下有多少對同號碼的球，就是能夠同偕白首的配偶數目了！我們這個漫遊，從批評用太多抽球例子爲開始，但最後又回到了抽球的例子。也許，漫遊到此，也該到了終點吧！



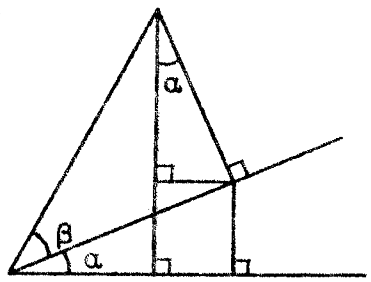


和角公式證明的一點探討

高淄長
信義中學

編者按：本文乃信義中學四年級學生來稿，本刊希望日後有更多學生投稿。

我們在證明兩角和三角函數公式時，通常是用這個圖形來尋求與單元三角函數的關系的。



通過推導而得到

$\sin(\alpha+\beta)=\sin\alpha\cdot\cos\beta+\cos\alpha\cdot\sin\beta$ 和

$\cos(\alpha+\beta)=\cos\alpha\cdot\cos\beta-\sin\alpha\cdot\sin\beta$；

這是大家所熟知的，而且許多教科書也是這樣證明的。

但是應當注意到這樣證明是局限於 $0<\alpha<\frac{\pi}{2}$ 和 $0<\beta<\frac{\pi}{2}$ 此範圍內的。要證明這個公式在任意角的範圍內是正確的，還需步步推導。

那麼怎樣在任意角的範圍對此公式加以證明呢？我們知道，單位圓上的角的大小是具有任意性的。因此，應考慮如何在單位圓上證明此公式。

首先，建立直角坐標系 XOY，以原點 O 爲圓心，作一單位圓。

設 α 角的始邊爲 OX，終邊相交圓於 A；

β 角的始邊亦爲 OX，終邊交圓於 B。

連接 AB。

因此我們得到

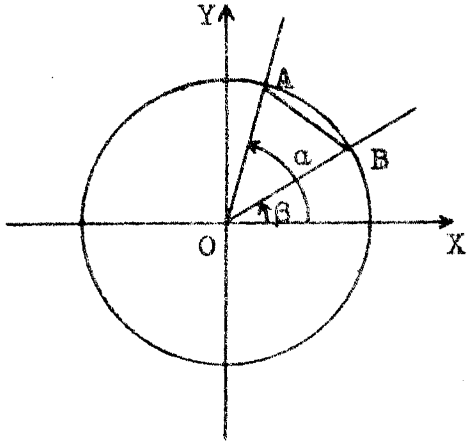
$\angle AOX = \alpha$,

$\angle BOX = \beta$,

$\angle AOB = \alpha - \beta$,

而且 A 的坐標爲 ($\cos\alpha$, $\sin\alpha$),

B 的坐標爲 ($\cos\beta$, $\sin\beta$)。



利用兩點間距離公式得

$$AB^2 = (\cos\beta - \cos\alpha)^2 + (\sin\beta - \sin\alpha)^2$$

$$= \cos^2\beta + \cos^2\alpha - 2\cos\alpha\cdot\cos\beta + \sin^2\beta + \sin^2\alpha - 2\sin\alpha\cdot\sin\beta$$

$$= 2 - 2(\cos\alpha\cdot\cos\beta + \sin\alpha\cdot\sin\beta)$$

在 $\triangle AOB$ 中，利用餘弦定理

$$\cos(\alpha - \beta) = \frac{OA^2 + OB^2 - AB^2}{2\cdot OA\cdot OB}$$

$$= \frac{1 + 1 - 2 + 2(\cos\alpha\cdot\cos\beta + \sin\alpha\cdot\sin\beta)}{2\cdot 1\cdot 1}$$

$$= \cos\alpha\cdot\cos\beta + \sin\alpha\cdot\sin\beta$$

因爲 α 和 β 是任意角，所以

$$\cos(\alpha+\beta) = \cos[\alpha-(-\beta)]$$
$$= \cos\alpha\cdot\cos(-\beta) + \sin\alpha\cdot\sin(-\beta)$$
$$= \cos\alpha\cdot\cos\beta - \sin\alpha\cdot\sin\beta$$

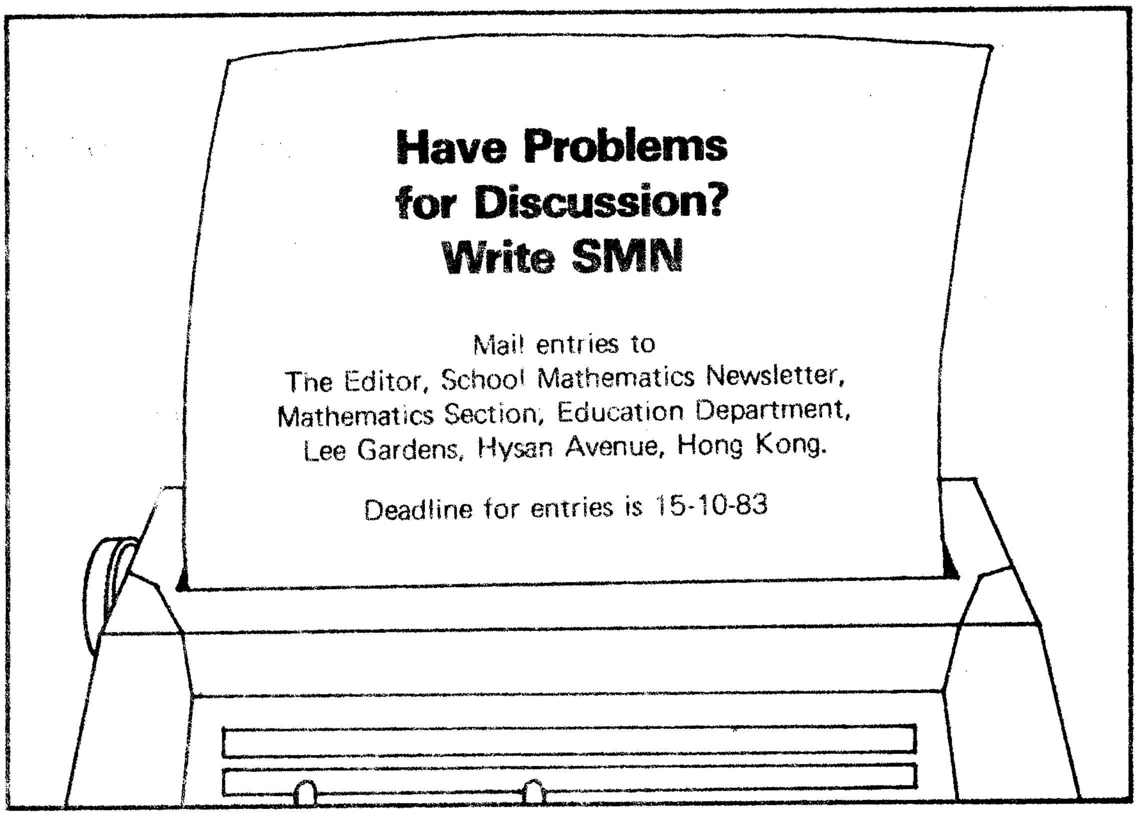
再利用已得公式求 $\sin(\alpha+\beta)$，

$$\sin(\alpha+\beta) = \cos\left[\frac{\pi}{2}-(\alpha+\beta)\right]$$
$$= \cos\left[\left(\frac{\pi}{2}-\alpha\right)-\beta\right]$$
$$= \cos\left(\frac{\pi}{2}-\alpha\right)\cdot\cos\beta + \sin\left(\frac{\pi}{2}-\alpha\right)\cdot\sin\beta$$
$$= \sin\alpha\cdot\cos\beta + \cos\alpha\cdot\sin\beta$$

因此而證明到

$$\sin(\alpha+\beta) = \sin\alpha\cdot\cos\beta + \cos\alpha\cdot\sin\beta$$
$$\cos(\alpha+\beta) = \cos\alpha\cdot\cos\beta - \sin\alpha\cdot\sin\beta$$

在任意範圍內都是正確的。SMN



In-service Training for Mathematics Teachers

F. K. Siu
Northcote College of Education

Mathematics plays an important role in the school curriculum. All schools are responsible for providing students with a sound mathematics education so that they can cope with life in this growing complex society. To meet this demand, we need well qualified and competent mathematics teachers. But mathematics teaching is a demanding job. Mathematics teachers should not regard themselves solely as imparters of mathematical skills. They should also instil into their students the appreciation of and respect for the intellectual richness and innate beauty of mathematics. Modern science and technology develop so rapidly that they greatly influence the school curriculum. We witness the changes of mathematics curriculum in the past two decades and we also envisage more changes in the future. This will make mathematics teaching an even more demanding task. In order to offer better mathematics education to the younger generation, mathematics teachers need to maintain and improve their professional competence.

In-service training (or retraining as some people prefer) is a very general term. A teacher who is aware of the need to up-keep his professional competence will plan his self-devised in-service program. For instance, he reads regularly professional journals and references. He participates in conferences and job-related functions organized by professional associations. (The H.K. Association of Science and Mathematics Education and the H.K. Professional Union of Teachers are at present the only local professional associations that organize such activities for mathematics teachers.) He also frequently discusses with his colleagues teaching methods and new teaching trends. The mathematics panel of a school can also provide some sort of school based in-service training. The panel chairman takes up the responsibility for offering guidance to his junior colleagues. Through frequent meetings mathematics teachers can learn from each other's experience.

However, in-service training frequently refers to the more formal courses organized by institutions such as the School of Education of the two universities, the Mathematics Section, Advisory Inspectorate of the Education Department and the colleges of education. These institutions will be responsible for initiating, planning, implementing and evaluating the in-service programs. It is our intent to include all the afore-mentioned aspects, but only concentrate on the more formal aspects of in-service training in the following discussion.

Aims of In-Service Training Courses for Mathematics Teachers

It is fairly obvious that any in-service program for mathematics teachers should aim at maintaining and improving their professional competence. However it must be made clear that the ultimate beneficiaries of in-service training of teachers must be the students. Thus, although the following aims of in-service training are teacher oriented, they are geared towards the improvement on students' learning.

1. To provide teachers with opportunities, means, facilities and materials for improving their professional competence.
   We trust every mathematics teacher has the urge to improve his competence. However, because of the heavy work load he is shouldering he may not be so self-motivated to read up journals and references regularly. Courses organized by institutions with experienced supervisors are thus welcomed.
2. To assist teachers in developing creative instruction.
   Teachers who already possess some teaching experience would always like to improve their teaching approach. Creative instructions can help students learn the subject in a more meaningful way.
3. To assist teachers in developing new teaching techniques.
   This is particularly important with modern innovations in education technology. The use of such modern technology in the classroom can be learnt in the in-service training program.

To help teachers to develop problem solving ability of their students. Students in Hong Kong, though enjoy a reputation for their mastery of mathematical skills, are in lack of ability in problem solving. This is inevitable because of the examination pressure. Teachers need to learn a way to help their students to get over this deficiency.

5. To provide a means of developing the mathematics curriculum for the individual use in the teachers' own schools.
Teachers need to learn how to response to curriculum changes and design their own curriculum to meet the special needs of their students. Although the Education Department supply schools with teaching syllabuses, individual schools have their own needs and mathematics teachers will take up the responsibility in designing their own mathematics curriculum.

6. To help teachers to let students into appreciation of the cultural aspect of mathematics.
Owing to the tight schedule for completing syllabuses for external examinations, mathematics teachers usually ignore the cultural aspect of the subject. (Or would it be that teachers themselves are not aware of it ?) To make the learning of mathematics more meaningful, teachers need to refresh (or start learning ?) mathematics from a global view and to understand the subject through its cultural development.

7. To help teachers expand their perception of mathematics.
Learning mathematics is an endless task. One can see a simple topic from a variety of angles. A teacher with a broad view of mathematics can, in one way or another, help his students understand the subject better. It is vital that teachers should "live up" with mathematics and expand their perception of it.

Why In-Service Training is Necessary

Pre-service training cannot supply all the expertise that a teacher needs, not even for a beginning teacher. Thus in-service training is necessary right from the start and is vitally important for teachers who have taught for several years.

As a group, teachers are concerned with providing the best possible education for their students. In-service training can play an important role in maintaining and improving their professional competence, thus benefiting their students. A good in-service program also helps to develop self-confidence in the teacher by enabling him to put up better and more effective performance in the classroom. An additional benefit of in-service training courses is that teachers of various schools can come together. They can exchange ideas, information and experience in mathematics and mathematics teaching. This teacher-to-teacher contact through in-service training courses enables teachers to discover better ways to teach.

What Courses can be Offered in In-Service Training ?

The following are some suggested courses for in-service training for mathematics teachers :

1. The use of electronic calculators in teaching mathematical concepts.
   With the development of electronic technology, the price of an electronic calculator has fallen rapidly in recent years. Nowadays almost all secondary students possess a hand-held calculator. However for most of them the calculator is merely a super-convenient substitute for the old four-figure tables, while in fact it can be used as an aid to learn mathematics. There are lots of ways to help students develop mathematical concepts through the use of a calculator. (1)

2. Mathematics laboratory approach.
   Mathematics laboratory in which students are allowed to play with concrete materials to discover mathematical ideas has been a recent development of mathematics teaching. (2)

3. History of mathematics.
   The cultural aspect of mathematics can best be reflected through the development of mathematical ideas. Moreover, teachers may get ideas on teaching strategy if they know something about the past development of the topic they are teaching. (3)

4. Curriculum development.
The need of developing curriculum for schools' individual use become more prominent. Teachers need an overview of curriculum changes in the past decades so that they can design a better curriculum for their own students. (4, 5)

5. Remedial mathematics.
With the introduction of mass education in Hong Kong there is an urgent need for remedial teaching. Teachers have to consider alternatives and solutions for teaching weaker students in mathematics.

6. Teaching in mixed-ability classes.
There is a growing tendency of putting students with different ability in the same class. How to cope with problems arising from this situation becomes an urgent problem for teachers. (6)

7. Develop the ability of problem solving in students.
Problem solving has been laid down as the main theme of mathematics teaching in the 80's (7). What problem solving is and how students can develop such ability should be discussed in depth (8, 9).

There are many other topics that can be included in in-service training courses. Readers are referred to Appendix 3 of (10).

References

(1) National Council of Teachers of Mathematics, Calculators — Readings from the Arithmetic Teacher and the Mathematics Teacher, N.C.T.M., Reston, 1979.

(2) National Council of Teachers of Mathematics, The Mathematics Laboratory — Readings from the Arithmetic Teacher, N.C.T.M., Reston, 1977.

(3) F.K. Siu and M.K. Siu, History of mathematics and its relation to mathematical education, Intern. J. Math. Education in Sci. and Technol., 10 (1979), 561 - 567.

(4) National Council of Teachers of Mathematics, NACOME Report, Overview and Analysis of School Mathematics : Grade K-12, N.C.T.M., Reston, 1975.

(5) 梁鑑添，評論近二十年來中學數學課程改革，抖擻雙月刊，38(1980)，64-75，83頁。

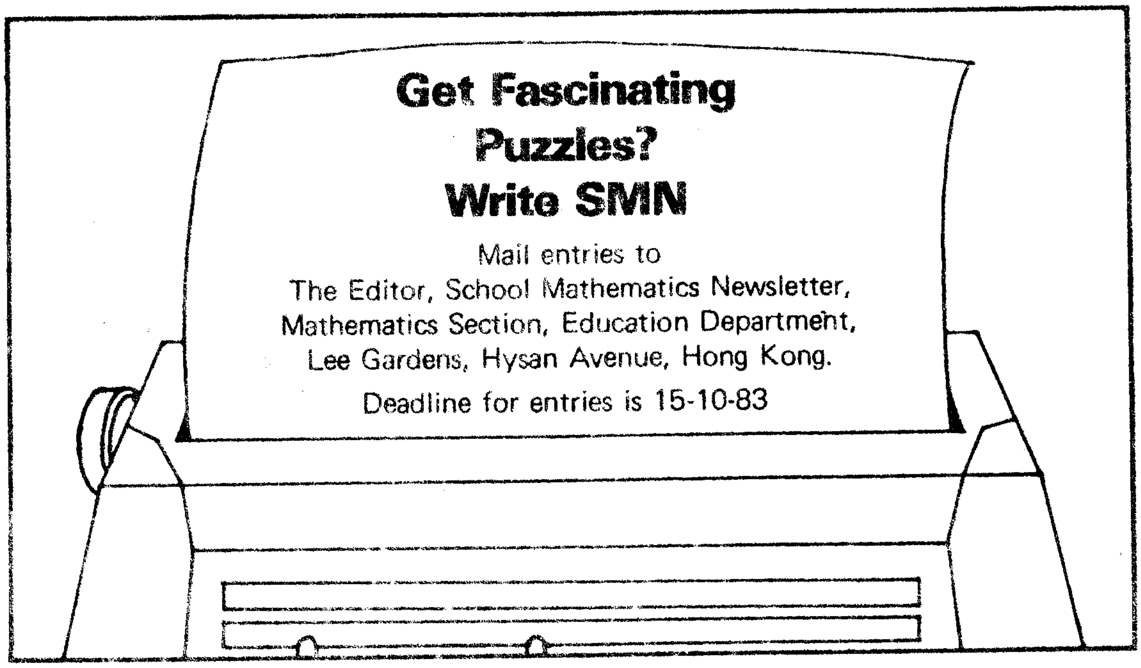
(6) Schools Council Working Group on Mixed-ability Teaching, Mixed-ability Teaching in Mathematics, Schools Council Mathematics Committee, London, 1977.

(7) National Council of Teachers of Mathematics, An Agenda for Action : Recommendations for School Mathematics of the 1980's, N.C.T.M., Reston, 1980.

(8) G. Polya, How To Solve It, Princeton University Press, 1945.

(9) G. Polya, Mathematics and Plausible Reasoning, Princeton University Press, 1954.

(10) National Council of Teachers of Mathematics, An In-Service Handbook for Mathematics Education, N.C.T.M., Reston, 1977.



An Unbased Theorem on Equilibrium

Robert Shin

Kwun Tong Government Secondary Technical School

Let a rigid body be put on a horizontal or an inclined plane (if inclined, it is assumed that friction is large enough to prevent slipping to occur). We want to consider the problem of its equilibrium. Equilibrium requires, of course, as Mr. Newton told us as long ago as his Principia Mathematica of 1686, and Signor Galileo even earlier, that the forces acting on it should nullify the effects of each other. This proclamation, though sound (to the point of being trivial were it not for the illustrious names that go with it), is not of much use in deciding whether a given body, say a cone or a chair, will stand up if left to itself. What is needed is a criterion that relates equilibrium to its shape. Messrs. Humphrey and Topping (A Shorter Intermediate Mechanics) and others, whose names are sufficiently well-known among A Level candidates to justify quoting from their works, have come up with a simple test :

When a body is placed with its base in contact with a plane (rough enough to prevent slipping if inclined), it will be in equilibrium if the vertical line through its centre of gravity meets the plane within the area of the base.

Here 'if' is a loose abbreviation of 'if, and only if', a practice not uncommon among engineers, and 'base' means, according to the Oxford English Dictionary, 'the bottom of any object, considered as its support, or that on which it rests'. This theorem, though seemingly obvious, and widely accepted, is not all that simple. Consider a man standing at ease on level ground, with feet apart. The plumb line through his c.g. passes midway between his feet, and if they are construed to be his support or base the theorem will predict that he cannot maintain his stance, though in actuality his is one of stability and comfort. Similarly, a table rests on four legs, and if the union of these four sets of points of contact is its base, the line through the c.g. will again fall outside it. For this theorem to apply, the base of an object must be something more than 'that on which it rests'. The idea may be sharpened using the notion of a convex set.

1. CONVEX SETS

<u>Definition</u> A set E of points is <u>convex</u> iff for every pair of points A and B of E, the straight-line segment AB joining them lies inside or on the boundary of E; i.e. iff $AB \subseteq E$.

All straight lines (of infinite extent) are convex. For a line is determined by any two of its points, so that if A and B are any two points of a line L, the line L' containing them is no other than L, and by definition the segment $AB \subseteq L' = L$. We shall take as an axiom that all segments of straight lines are convex. All triangles (i.e. the set of points on or inside the three sides) are also convex, though the demonstration will require a little effort.

Theorem The intersection of a family of convex sets is also convex.

Proof Let A and B be any two points of the intersection $\bigcap_{M \in \mathcal{M}} M$ of a family $\mathcal{M}$ of convex sets M. These points are in every $M \in \mathcal{M}$, and therefore by convexity the segment $AB \subseteq M$ for every M, so that $AB \subseteq \bigcap_{M \in \mathcal{M}} M$, q.e.d.

With every set E it is therefore possible to find, by considering the intersection of all the convex sets containing E, a smallest convex set $\hat{E} \supseteq E$. (Convex sets containing E certainly exist; e.g. the entire space $\mathbf{R}^2$ or $\mathbf{R}^3$ are convex.) If $E \neq \emptyset$, $\hat{E} \neq \emptyset$ as well. If E is convex, $\hat{E} = E$.

Examples :

1. The smallest convex set $\{A, B\}^\wedge$ containing two points A and B is the segment AB. For by definition $AB \subseteq \{A, B\}^\wedge$ and the convexity of AB gives $\{A, B\}^\wedge \subseteq AB$, because $\{A, B\}^\wedge$ is minimal.

2. The smallest convex set containing three noncollinear points A, B, C is the set of points on or inside $\triangle ABC$.

Proof A convex set E that contains A, B, C will have to contain the segments AB, BC, CA and hence their union (i.e. the sides of $\triangle ABC$). Let P be any point inside the triangle. Produce AP to meet BC at Q. Because E is convex, $P \in AQ \subseteq E$. This being true of any point P, $\triangle ABC \subseteq E$. This and the convexity of triangles give $\{A, B, C\}^\wedge \subseteq \triangle ABC \subseteq E$. The last inequality holding for any E, it holds in particular for $E = \{A, B, C\}^\wedge$ and this establishes the result.

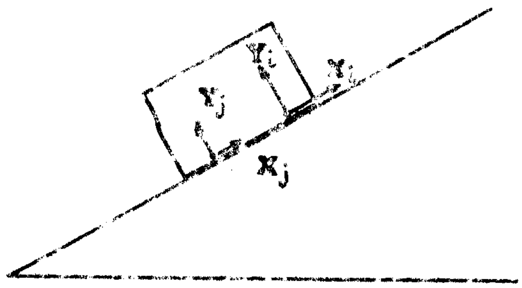
3. For n coplanar points $P_1, P_2, \dots, P_n$, $\{P_1, P_2, \dots, P_n\}^\wedge$ is the set of points on or inside the polygon $P_1 P_2 \dots P_n$, provided that this polygon is convex. If e.g. P_n is inside the polygon $P_1 P_2 \dots P_{n-1}$, $\{P_1, P_2, \dots, P_n\}^\wedge = \{P_1, P_2, \dots, P_{n-1}\}^\wedge$.

Proof Induce on n. A convex set E containing $P_1, P_2, \dots, P_{n-1}, P_n$ contains the polygon $P_1 P_2 \dots P_{n-1}$ and $\triangle P_{n-1} P_n P_1$ (inductive hypothesis) and therefore also their union, which is the polygon $P_1 P_2 \dots P_{n-1} P_n$ (supposing that P_n is outside polygon $P_1 P_2 \dots P_{n-1}$). Etc.

2. A NECESSARY CONDITION FOR EQUILIBRIUM

Theorem Let a rigid body be put on a horizontal or an inclined plane (it being assumed that, if the plane is inclined, the friction between it and the body is large enough to prevent slipping; no glue, however, is applied). Let E be the set of points where the body makes contact with the plane. A necessary condition for the equilibrium of the body is that the vertical line through its c.g. should pass through $\hat{E}$.

Proof Let $E = \{P_1, P_2, \ldots, P_n\}$. Resolve the contact force at P_i in two components, $\mathbf{X}_i$ parallel to or along the plane and $\mathbf{Y}_i$ normal to the plane.



Because the body is not glued to the plane, the forces $\mathbf{Y}_i$ all point upwards (see figure) and the resultant of any two of them, say $\mathbf{Y}_i$ and $\mathbf{Y}_j$, is a like parallel force whose point of application may be taken to be some point P_{ij} of the segment P_iP_j. Similarly, the resultant of $\mathbf{Y}_i + \mathbf{Y}_j$ and $\mathbf{Y}_k$ acts at some point P_{ijk} of $P_{ij}\,P_k$, etc. Proceeding thus, it may be seen that the resultant $\mathbf{Y}_1 + \mathbf{Y}_2 + \ldots + \mathbf{Y}_n$ of the Y-forces must act at some point $P_{12\ldots n}$ of $\hat{E}$.

If we assume that each particle of the body in contact with the plane tends to move in the same direction, the frictional forces $\mathbf{X}_i$ are all like parallel forces and their resultant must, by the same argument, act at some point Q of $\hat{E}$.

There are now three forces acting on the body : its weight $\mathbf{W}$ acting through the c.g., and the resultants $\mathbf{X}_1 + \mathbf{X}_2 + \ldots + \mathbf{X}_n$ and $\mathbf{Y}_1 + \mathbf{Y}_2 + \ldots + \mathbf{Y}_n$. For equilibrium, these forces must be concurrent, which implies that $Q = P_{12\ldots n}$ and that the line of action of $\mathbf{W}$ passes through this point.

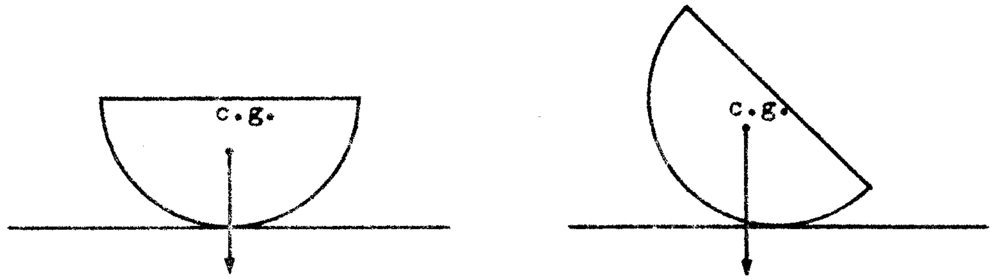
Technical complications arise if E is infinite; these may be left to the reader to ponder over at his leisure.

The condition stated in the theorem is also sufficient, if we assume that the plane is stationary and the body is left to itself (not given any acceleration sideways, for example). For then equilibrium may be broken only by the body's turning about an edge (because of some $\mathbf{Y}_i$ being too large); but as soon as the body is lifted off the plane the normal force $\mathbf{Y}_i$ vanishes and the weight's righting moment will restore it back to position. Thus the forces $\mathbf{Y}_i$ will automatically adjust themselves to bring about equilibrium.

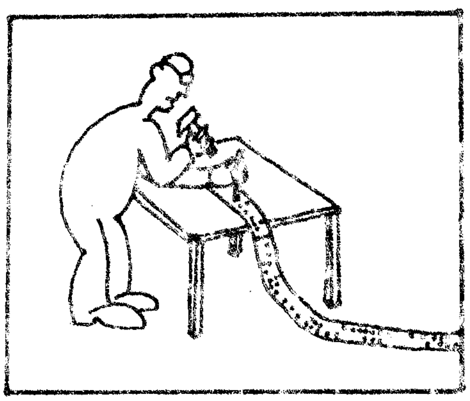
3. STABILITY

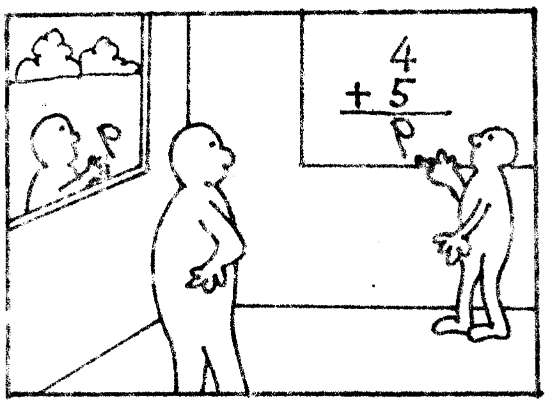
A sequel to the theorem of the last section is to consider the type of equilibrium the body will be in, whether stable, unstable or neutral. We are told, by Nelkon and other authors, that the equilibrium is stable or unstable according as the vertical line through the c.g. still or no longer passes through the base when the body is slightly displaced, and that equilibrium is neutral if the line always goes through the base irrespective of the amplitude of the displacement.

As explained earlier, the base of a body is 'that on which it rests', and that changes with the orientation of the body in relation to those on which it rests. My base when I am sitting on a chair, as I am now, is quite different to my base when I am lying on bed, though in the second case people, following another convention, would still call my base my back. When a chair is displaced slightly, it will make contact with the ground in only two legs (which therefore become the new base). The vertical line through the c.g., however, will still pass through $\hat{E}$, where E is the set of points of contact before displacement (i.e. the bottoms of the four legs of the chair, or the points of the ground they are in contact with — it does not matter which). If $\hat{E}$ is taken to be the 'base' of the chair, the criterion predicts, correctly, stable equilibrium. It need not apply, however, in other situations.



Consider a hollow or solid hemisphere resting on a horizontal plane. Any displacement, however slight, will bring the line through the c.g. outside the base (be this the old or new base), though the hemisphere will return to its undisturbed position if released and is therefore in stable equilibrium. The criterion, despite its being almost 'obvious', is actually fallacious. [redacted]





A Redirection for Mathematics in Schools — The Work of the Spode Group

John Berry, FIMA
The Open University

David Burghes, FIMA
University of Exeter

Ian Huntley, FIMA
Paisley College of Technology

Reproduced from IMA Bulletin, by permission of the Institute of Mathematics and its Applications.

"It is of vital importance that future generations have not only an understanding of basic mathematical operations but also the ability to apply them in a practical context."[1]

1. INTRODUCTION

THESE words are perhaps the driving force behind the formation and work of the Spode Group, which is a group of mathematics teachers from schools, colleges and universities in the U.K. We are all familiar with students who can "turn the handle" and produce the solution to a tightly posed mathematical problem, but who are totally lost when presented with essentially the same problem in a practical context. This has led to a realisation of the need to involve school children in the process of mathematical modelling and the increasing attention to applications over the past few years. It is of no use merely to know the techniques; students must also be able to apply them to solve real problems.

Over the past 4 years one of the authors (D.B.), while at Cranfield Institute of Technology, ran short courses on mathematical modelling for teachers. The aim of these 2 or 3 day courses was to introduce the teachers to the process of modelling. Mathematical modelling is not something that is easily taught, and expertise is gained almost exclusively by actually "doing some modelling". So the teachers were usually thrown in at the deep end with only a statement of the practical situation, and left to come up with a model, find a solution and then validate the model using some data relevant to the problem. Actually by carrying out this process the various activities in mathematical modelling can be experienced and better understood. The courses were very successful on the whole, and the teachers returned to the classroom with both a better appreciation that mathematics is a useful discipline and an insight into how mathematics is used in a practical context. It soon became apparent, however, that there was a tremendous shortage of suitable material in published form, both for use on in-service courses and to introduce mathematical modelling to the wider school audience. Thus the Spode Group was formed.

A group of ten teachers was financed by the Schools Council, and met for a weekend at Spode Conference Centre, Staffordshire, in September, 1980, under the directorship of one of us (D.B.), to write a series of case studies for use in their classrooms and for distribution as widely as possible to interested teachers. The group has met several times since then, and this article describes the work of the group during 1980 and 1981.

2. THE FIRST PROJECT

The first project of the Spode Group was to produce material for use in the upper school - probably the easiest group to which to introduce these ideas. It was decided to write case studies each with a clear problem statement, together with some possible approaches for finding a solution. The format of each case study is essentially the same :

(i) problem statement;
(ii) teaching notes;
(iii) possible solution.

The problem statement is a clear description of a problem of practical non-mathematical origin, and the task is to find an answer to the problem posed. This will usually involve mathematical modelling, and often will not be unique - there will be different ways of solving the problem depending, perhaps, on the assumptions made. The group felt that a teacher may need some background or supporting information when using the material. The "Teaching Notes" provide this back-up, by giving recommendations on what extra information and hints might be needed by the children and advice on how the lesson might be structured. When solving practical problems there are often different approaches that can be made; and it is not usually possible to define solutions as right or wrong - some approaches are better than others, but any work that helps to answer the problem posed is acceptable. Each case study has a possible solution to the problem, but this in no way means to imply that it is the best or only solution - it merely indicates the approach likely to be taken by the pupils, and one which the teacher has available if little progress has been made.

The case studies have been written in such a way that they can be used in two distinctive modes. The first is to encourage the children to experience applying mathematics to practical problems and for them actually to do some mathematical modelling. The idea is that the problem statement should be duplicated and distributed to the class, and the children encouraged to work on the problem in small groups. Each group should be given ample time to make as much progress as possible on its own, with the teacher giving help only when a group is making no headway whatsoever. This method of teaching is probably different from what the class and the teacher are used to.

It proves to be very challenging, but the rewards for the class and the teacher are great. However, it was realised that the case studies could and probably would be used in a rather different way - as a completed mathematical model which would motivate particular mathematical topics.

By using problems set in the real world, the relevance of mathematics can at least be made apparent, although we very much encourage the use of the first mode of presentation. Three books of case studies have been produced.[1-3] The first two volumes are aimed mainly at pupils in the upper half of a secondary school, and use the skills of O- and A-level mathematics. The third volume is aimed at children who are on a CSE course in mathematics. The experience of using these case studies with CSE classes has been very successful. Children with no interest in mathematics have seen the subject come alive when it has been used to solve real problems.

3. AN EXAMPLE OF A CASE STUDY FROM VOLUME I

In this section we reprint a sample case study from the first of the three volumes mentioned.[1] This example illustrates the main features of each case study.

Stock control

(a) Problem statement

Many large organisations need a definite policy for controlling levels of stock. If stock levels are too high, money is being tied up unnecessarily; if levels are kept very low, there is a danger of being out of stock for a period of time. Without any management, a graph of stock level against time would look something like Fig. 1.

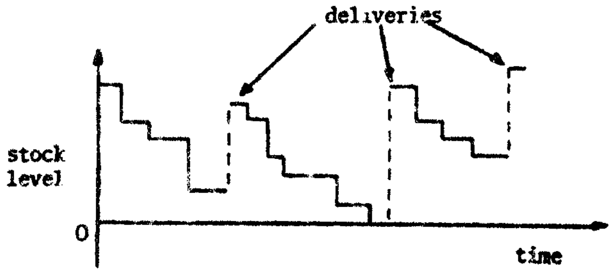


Fig. 1

The following data refer to a particular item used by the National Coal Board :

(i) ordering cost (i.e. the cost of the administrative process) £1.50 per order;
(ii) annual holding cost of an item in store (which refers either to the loss of interest which could be made on the capital used;

or interest charges being paid on borrowed capital) is assessed at 18 per cent of the goods' price;

(iii) the current price per item is £24;

(iv) the average usage level of the item is 26 per month.

Determine an optimal ordering policy; that is, how many items should be ordered and how often in order to minimise the costs?

(b) Teaching notes

The main difficulty in this problem lies in producing a formula, equation (4), for the total annual costs. Bright pupils might be able to get to (4) or an equivalent formula without help, but for the majority guidance will be vital. On the other hand, most pupils should be able to achieve a solution to the problem once equation (4) has been reached.

It is strongly suggested that small group activity should be encouraged on this problem at all stages. In section (d), a number of extensions are proposed. The first deals with the equivalent general problem, using algebraic symbols rather than actual values; whilst the second deals with the problem of discounts.

(c) Possible solution

The stock level situation illustrated in Fig. 1 in section (a) has a certain degree of randomness; but we can start by assuming a constant demand, as illustrated in Fig. 2.

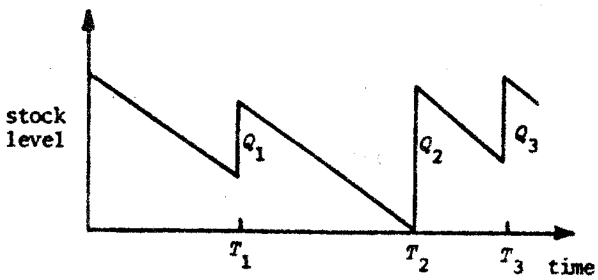


Fig. 2

So we have the situation where several different orders (size Q_i) are placed at times T_i. We can further simplify the analysis by assuming a constant order size Q and reorder time T, and that orders are placed just as stocks run out with instant delivery ! This is illustrated in Fig. 3.

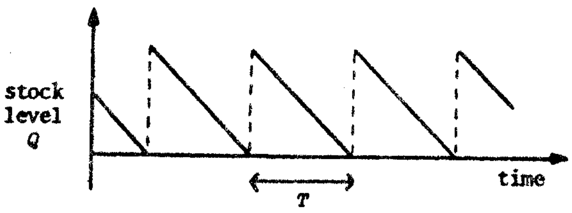


Fig. 3

The next key stage is to introduce the variable for the order quantity, say x, and let

$$y = \text{total annual costs} \\ = \text{annual ordering costs} \\ + \text{annual holding costs} \quad (1)$$

Clearly y is a function of x, and the mathematical problem requires finding the value of x which minimises y. The next stage though is to determine the functional form of y.

(i) Annual ordering costs

Since the order size is x, and the total usage is 26 x 12 = 312 per year, the average number of orders per year is 312/x, and so the annual ordering costs are

$$£\frac{3}{2} \cdot \frac{312}{x} = £\,\frac{468}{x} \quad (2)$$

(ii) Annual holding costs

Looking at the "idealised" stock graph above, we see that the stock level is repeatedly dropping from x to 0 linearly, so that the "average" stock held throughout the year is ½ x. The cost of holding one item in stock for 1 year = 0.18 x £24 = £4.32 and so the annual average holding costs are

$$£4.32 \cdot \frac{1}{2}x = £2.16x \quad (3)$$

Combining (2) and (3) gives the total annual costs in £ or

$$y = \frac{468}{x} + 2.16x \quad (4)$$

We now require to find the value of x which minimises y. There are a number of possible ways to proceed; for example

(i) draw a graph of the function;
(ii) use a numerical procedure on a computer;
(iii) use calculus to find the minimum.

We will take the third method as this can easily be extended for more general problems.

Since y is a continuous function of x, its minimum will occur when $dy/dx = 0$, $d^2y/dx^2 > 0$. Now, from (4)

$$\frac{dy}{dx} = -\frac{468}{x^2} + 2.16 \quad (5)$$

and $dy/dx = 0$ when $x^2 = 468/2.16$, i.e. $x \approx 14.7$. Also $d^2y/dx^2 = 936/x^3 > 0$ for $x = 14.7$, and so we do indeed have a minimum.

So taking the nearest whole number solution, we suggest an order quantity of 15 items. With this order quantity, the number of orders per year is given by

$$\frac{312}{15} \approx 21$$

and so an order must be placed every 52/21 weeks, i.e. about twice every month.

(d) <u>Related problems</u>

(i) Generalisation

Although we have solved a specific problem, clearly the same problem will occur many times for the NCB, and for many other industries as well. Consequently it will be advantageous to have the solution to the general problem where

(i) ordering costs = £a;
(ii) annual holding costs per item = b times the cost of one item;
(iii) cost of item = £c;
(iv) average usage = d items per year.

Again let x denote the number of items in one order, and y the annual total costs. Then

$$y = \text{annual ordering costs} + \text{annual holding costs}$$

$$= a.\frac{d}{x} + b.c.\frac{x}{2},$$

since there will be d/x orders per year, and the average stock held is x/2. Thus the functional form is

$$y = \frac{ad}{x} + \frac{bcx}{2} \qquad (6)$$

Now for a minimum we require $dy/dx = 0$ and $d^2y/dx^2 > 0$. From (6),

$$\frac{dy}{dx} = -\frac{ad}{x^2} + \frac{bc}{2}$$

and $dy/dx = 0$ when $x^2 = 2\,ad/bc$. Also

$$\frac{d^2y}{dx^2} = \frac{2ad}{x^3} > 0 \text{ for positive } x.$$

Hence the optimum order quantity is given by

$$x = (2ad/bc)^{1/2} \qquad (7)$$

(this quantity is known as the "Economic Order Quantity"). Note that we can now use (7) to solve the original problem, as well as others. For the original problem

$$a = 1.50, \quad b = 0.18, \quad c = 24, \quad d = 312$$

and (7) will, of course, again give $x \approx 14.7$.

(ii) Discounts

In many cases it will be possible to obtain discounts for larger order sizes. For example, returning to the original problem, suppose the cost price per item, when the order is x, is given by

$$c = \begin{cases} £24 & \text{if } x < 20, \\ £20 & \text{if } 20 \leqslant x < 50, \\ £16 & \text{if } x \geqslant 50. \end{cases}$$

What now is the economic order quantity ?

Although we can still tackle the problem algebraically, it is probably most constructive to use a graphical approach. The total costs are now given by

$$y = \begin{cases} \dfrac{468}{x} + 2.16x & \text{if } x < 20, \\[2ex] \dfrac{468}{x} + 1.80x & \text{if } 20 \leqslant x < 50, \\[2ex] \dfrac{468}{x} + 1.44x & \text{if } 50 \leqslant x. \end{cases}$$

This function can be illustrated on the graph (Fig. 4).
From Fig. 4 we can deduce that the optimal order quantity is now 20, rather than the previous value 15. An analytic approach would be to evaluate the costs at

(i) the EOQ formula (7);
(ii) all points of discontinuity of the price.

The optimal solution will then occur at the position of global minimum.

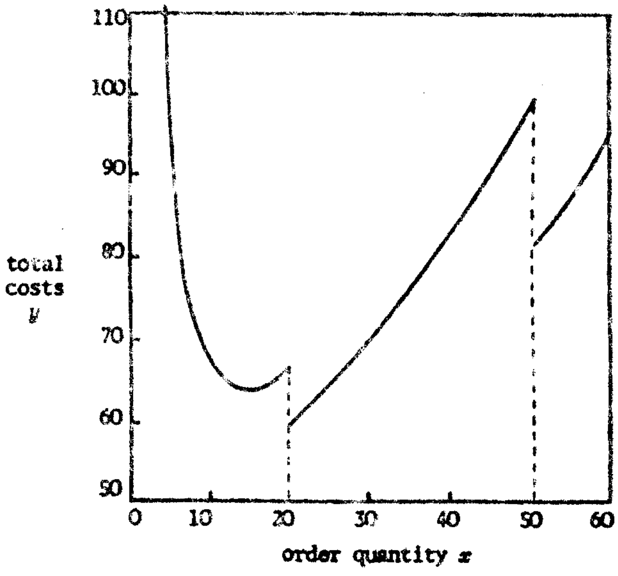


Fig. 4

4. THE SECOND PROJECT

The original aims of the Spode Group were to produce examples which (i) show the relevance of mathematics to practical problems, and (ii) provide applications that can be taught in schools. It was our intention that the case studies would be used to give children the experience of mathematical modelling, but we soon became aware of their wider use for motivating mathematical techniques. Also the case studies which we initially developed only used a small proportion of the present mathematics syllabus; however, because it was diversity of applications we were searching for and not a diversity of techniques.

The second major project undertaken by the Spode Group was to write a source book containing solved problems of practical importance, which illustrate the practical uses of the major topics in A-level mathematics. The solutions to the problems rely on well-known models so that the mathematical modelling process is not emphasised here. The aims were to cover as much of the A-level syllabus as possible, presenting problems from the real world which show the relevance of A-level mathematics. It is of interest to note that despite much time and effort being spent in researching for problems for this book, there were many parts of the present A-level syllabus for which it seemed virtually impossible to find convincing applications. On the other hand, there were many areas where a vast amount of motivational material existed. The authors of this article are the editors of this book and this project was supported jointly by the Schools Council and the Institute of Mathematics and its Applications.

5. THE FUTURE

The teaching of mathematics is a never ending iteration, as teachers seek to motivate themselves and their pupils. The Spode Group is attempting to impress on the educational world that teachers and pupils must see the relevance of the mathematical techniques that they are learning. It is producing material that illustrates the links between real problems and mathematics, showing that mathematics can be used to solve problems of practical importance and that this can be done in the classroom. Further, if the case studies of the first project are used according to the original intentions, then both teachers and pupils will have experienced the excitement of taking a real problem, formulating it as a mathematical problem and thus solving it: in other words the rewards of mathematical modelling. The membership of the group has not been static. Altogether we have now involved nineteen teachers in the writing of these case studies and examples. It is of interest to note that all but one of the original members are still active writers for the Group.

The aims of the Group have also not remained static. Although our next major project is to provide material for the 16–19, again with emphasis on the practical nature of mathematics, we plan to follow this by our most ambitious project to date, which involves writing a full series of books (probably two parallel series in fact) for the new 16+ examination, which is due to replace O-level and CSE sometime in the future. The emphasis will be again on providing a course which shows fully the practical nature and relevance of mathematics but we also realise that many topics cannot be fully justified in the syllabus on relevance alone. So we will be seeking to write a course which will illustrate the (i) relevance, (ii) beauty, (iii) intriguing nature and (iv) creative nature of mathematics, although emphasis will undoubtedly be put on (i).

Enquiries regarding the work of the Spode Group, and offers of help both in writing and testing material, are very much welcomed and should in the first instance be addressed to : Professor David Burghes, FIMA, School of Education, University of Exeter, St. Luke's, Exeter, Devon.

We very much hope that our efforts will eventually provide a positive redirection of school mathematics. Although up to now we have been writing very much for our own benefit, we hope that in the next stage we will provide material that will meet the needs and aspirations of pupils, teachers and employers alike.

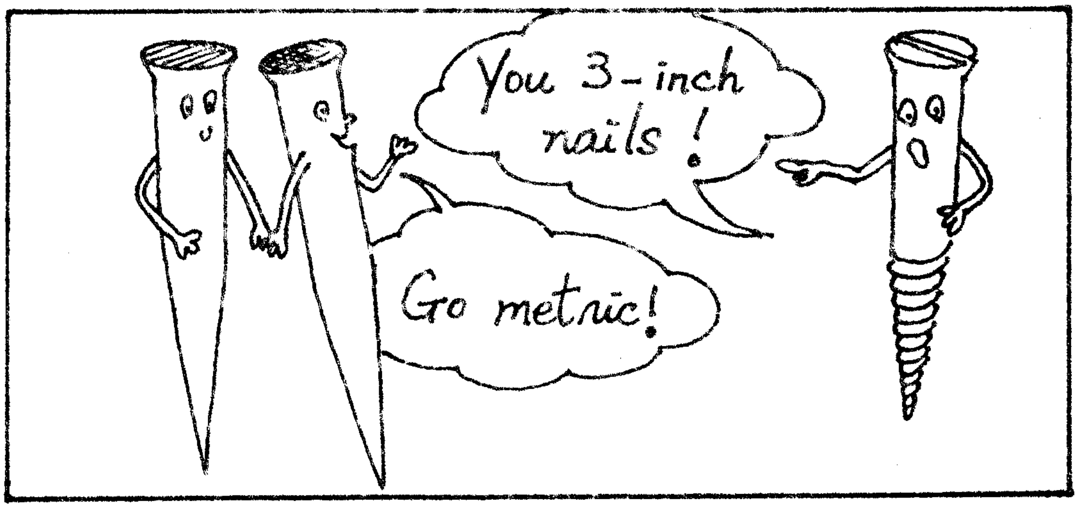
REFERENCES

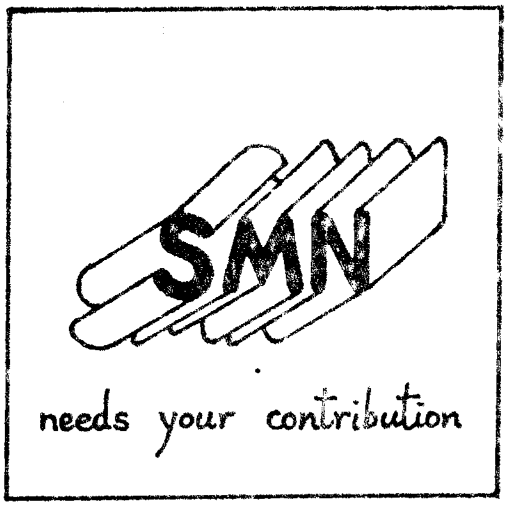
1. "Solving Real Problems with Matnematics", Volume 1, Cranfield Press, 1981.

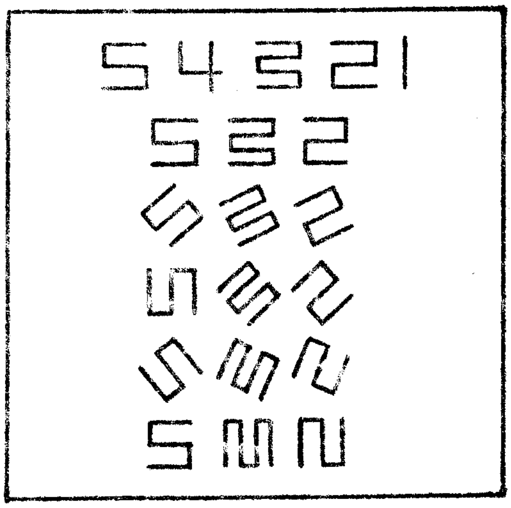
2. "Solving Real Problems with Mathematics", Volume 2, Cranfield Press, Spring 1982.

3. "Solving Real Problems with CSE Mathematics", Cranfield Press, Summer 1982.

4. "Motivational Mathematics for A-level", IMA, in preparation.







Computer Studies in Hong Kong

Mathematics Section
Advisory Inspectorate, E.D.

Background

The use of computers and the number of computer related activities have increased dramatically in recent years. Computer Studies has already been introduced into the school curriculum in many countries. In Hong Kong, the 1978 White Paper on the Development of Senior Secondary and Tertiary Education stated that it is the government's aim to provide a broader curriculum in Forms IV and V so that students, as well as studying languages, science and social subjects, may take at least one practical or technical subject. In line with this proposal, the Education Department decided to include a new subject, Computer Studies, in the secondary school curriculum. The Computer Studies teaching syllabus was approved by the Curriculum Development Committee in 1981. In order to test the practicability of the syllabus, the Education Department has organized a pilot scheme involving 30 schools (5 government and 25 aided schools) in the public sector, starting from September 1982.

Objectives of the Subject

The objectives of the subject Computer Studies are to enable students to understand the functions, uses and limitations of computers, to provide an opportunity for the study of the modern methods of information processing, and to encourage an understanding of the implications of computers in the modern world and to prepare students for further studies in computer science.

Selection of Pilot Schools

When the secondary schools learned of the pilot scheme, a large number of them expressed interest in joining the scheme. The main criteria for the selection of pilot-scheme schools are the enthusiasm and interest shown by the school and the academic qualification of the teaching staff who will eventually teach the subject. Through regular contact with schools, the Education Department is able to identify the schools which are likely to benefit from and to contribute to the scheme.

Provision of Equipment

A total of $2,770,000 was approved for the pilot scheme expenditure. The breakdown of the sum was as follows :

Non-recurrent	Recurrent
$2,384,000	$386,000

Through formal tendering procedures and with detailed evaluation a contract was made with Onflo International Ltd. in 1982.

Each pilot school was provided with the following equipment :

9 A-800 microcomputer with 48K RAM and microsoft BASIC ROM
9 JVC Colour television/monitor
10 A-810 disc drive
1 A-410 cassette recorder
2 A-850 interface
2 Epson MX-80 III printer

Action by Hong Kong Examinations Authority

The Computer Studies Examination Syllabus was prepared by HKEA and was approved in December, 1981. Details of it can be found in Hong Kong Certificate of Education Examination Regulations and Syllabuses 1984, published by HKEA. A sample examination paper was also circulated to all secondary schools for reference purpose. Examination for this subject at Certificate of Education level will be provided by HKEA. It is estimated that about 4,000 students will be sitting for the first examination in May, 1984.

Non Pilot Schools

In principle, permission to start this subject is granted to schools if the applying schools give assurance that adequate equipment will be provided from the schools' own resources. Most of these schools intend to enter candidates for the 1984 examination in Computer Studies.

The model of microcomputer used in these schools varies from school to school. The equipment configuration recommended preferably consists of :

1 system for 10 students (1 system = 1 CPU + 1 Monitor + 1 disc drive)

1 cassette recorder

1 printer

In addition, one system and one printer are required for the teacher.

Implementation

1. Time Allocation

Four periods per week is recommended for this two-year course. Both classroom teaching and practical work are to be scheduled in the lessons.

2. Textbook

Textbooks written for the syllabus are still being prepared by publishers. So far, the book most popularly used by schools is 'Computer Programming in BASIC' by Peter Bishop, published by Thomas Nelson & Sons Ltd.

3. Selection of students

Schools can exercise their own discretion in the selection of students for the course, but are advised to give consideration not only to attainment but also to interest, inclination and aptitude.

4. Reference notes for teachers

Teaching notes are still being prepared by the Curriculum Development Team and distributed monthly to the pilot schools.

Training Courses and Supporting Services

1. In-service courses for teachers

As early as 1977, courses involving the use of computers were organized for mathematics teachers in secondary schools. These courses included six basic courses on topics like general introduction of computer application in education, programming in FORTRAN, BASIC, etc. Four of these courses were repeated during the years. There were also three intermediate courses on programming and one advanced course on machine language. Most of these courses were held at the Mathematics Teaching Centre but a number of them, thanks to the kind co-operation of the Chinese University of Hong Kong and/or the Hong Kong Polytechnic, were held at the Computer Services Centre of the University or at the Polytechnic.

From July to December, 1982, four more courses on basic programming, electronic data processing and machine familiarization were organized to assist teachers of the pilot schools to implement the syllabus and to familiarize them with the equipment.

More intensive in-service courses will continue to be conducted.

2. Reference materials

The Mathematics Section has prepared a list of reference books and a list of audio-visual aids currently available on loan from the Visual Education Section, Education Department for reference by teachers concerned.

3. Professional organization

The Hong Kong Association for Computer Education (HKACE) was formed by a group of enthusiastic teachers with the objective of promoting computer education in Hong Kong. All the teachers teaching this subject in the pilot schools are members of the Association. Regular monthly meetings are held during which teachers can exchange views, experiences and notes on the subject.

4. Advisory services

Mathematics inspectors have regularly been visiting the pilot and non pilot schools so as to obtain feedbacks and give advice where necessary. [redacted]

How to Begin Execution in the Middle of a Multiple Statement

Joseph Shin

An obvious advantage of writing multiple statement is the save of memory space. However, the use of multiple statements suffers from not being able to execute in the middle of a line. This article intends to demonstrate how one can pass execution to the middle of a multiple statement. To make my presentation precise, I shall use TRS-80 Level II BASIC with 16 K RAM as my sample model.

First of all, let us examine the internal coding of a BASIC program such as

```
1ØINPUT"YOUR NAME";A$:PRINTA$:POKE16631,252:POKE16632,66:CONT
```

It is known that the coding starts at memory location 17129. A portion of the contents of memory locations starting from 17129 is listed below :

17129	24	17130	67	17131	10	17132	0
17133	137	17134	34	17135	89	17136	79
17137	85	17138	82	17139	32	17140	78
17141	65	17142	77	17143	69	17144	34
17145	59	17146	65	17147	36	17148	58
17149	178	17150	65	17151	36	17152	58
17153	177	17154	49	17155	54	17156	54
17157	51	17158	49	17159	44	17160	50
17161	53	17162	50	17163	58	17164	177
17165	49	17166	54	17167	54	17168	51
17169	50	17170	44	17171	54	17172	54
17173	58	17174	179	17175	0	17176	0
17177	0						

The memory location pair 16631 and 16632 contains the address to start execution when the computer encounters the command CONT. Since 66*256 + 252 = 17148, we know that after the execution of the last command CONT, control will be passed to address 17148. RUN the program and see what happens. The output is exactly the same as that of the following program :

```
1ØINPUT"YOUR NAME";A$
2ØPRINTA$:GOTO2Ø
```

My first program demonstrates how one can pass control to the middle of a multiple statement. The procedure is as follows :

(1) Determine the address you want to branch to (in this case 17148).

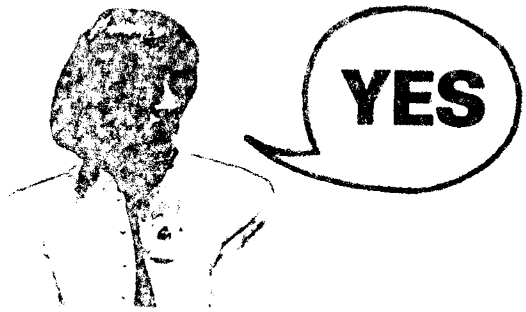
(2) POKE this address to the location pair 16631 and 16632 (in this case 252 in 16631 and 66 in 16632).

(3) Pass control to the desired address by CONT.

Thank you for reading this article.

Must Every Student Become "Computer Literate"?

**The recent introduction of Computer Studies into local secondary schools has received much attention. This article reflects contradictory views of two American educationalists on "Computer Literacy For Every Student". It might be an interesting topic for discussion, and I would be delighted to hear of your opinions.
—— Editor**



Julie McGee is project director of computer curriculum development at Lyons Township High School, La Grange, Illinois. She holds an M.A. in English Literature and has taught English for 17 years. A member of NEA, she also serves on the Standing Committee on Instructional Technology of the National Council of Teachers of English.

A funny thing happened on the way to my English class —— I got a computer and discovered some new ways to teach.

Numerous publications assure us that, like the Russians, "The computers are coming!" And many teachers respond with about the same enthusiasm they might feel if indeed the Russians were landing on our shores.

Some of my colleagues assert, "The computer is just another overhead projector. Everyone said that would change education too." Unfortunately, much of the currently available software uses the computer as a kind of electronic overhead projector, an unnecessarily expensive way to present instructional material.

But if we limit our consideration of the computer to seeing it only as a presenter of traditional instruction, we will fail to see its full potential. Properly programmed, the computer can be a highly effective educational tool.

Teachers need training to understand the various educational functions that computers can perform. The simplest is drill and practice. Students complete exercises much as they would with a workbook, except that they find out immediately if their answers are correct or not. Such a use has its advantages : the computer is patient, willing to repeat, and uncritical. When more quality software becomes available, the computer will be able to gear instruction to meet individual needs.

The computer can, however, do much more. It can stimulate students to think, to make decisions, and to explore a world beyond the classroom.

For example, although our students may never run for President, they can make decisions about campaigning and issues and see how these decisions affect the outcome of a computer-simulated election.

For students to use any already programmed material requires very little "computer literacy". About the only skills one needs are those required to operate a typewriter : the ability to locate keys on a keyboard and to turn a switch on. In fact, one error schools frequently make is to attempt to teach everyone how to program the computer.

Still, a microcomputer has far greater capabilities than the presentation of programmed materials. This article is being written on a microcomputer with a word processing program. I can insert, delete, and rearrange material by pressing a few keys. My well-used bottle of white-out is in the trash can. My students compose on the computer, and because revision does not require recopying an entire paper, they are more willing to revise. The result is better writing.

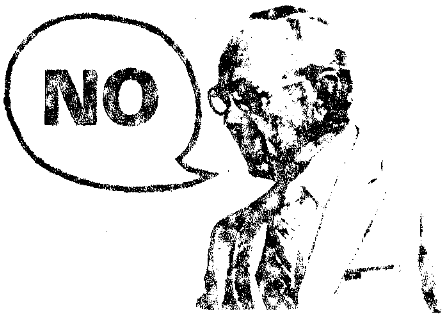
Students who want to learn how to program find that the computer synthesizes their mathematical and language skills. Programming requires the logical use of language -- and may even help us learn how to think. There is little doubt that computer skills will be an asset in tomorrow's job market.

The timorous argue that constantly changing technology will make today's equipment and languages obsolete in a few years. So, they say, let's wait awhile and see what develops. This argument assumes, however, that knowledge gained on one computer or with one language will not be applicable to another. That is rather like arguing that if I learn to drive a Ford, I will have to start all over again when I get a Mercedes.

And while we're waiting for the next generation of computers to arrive, our students are learning nothing about a technology that will be an important part of their futures. Can we afford to wait?

I think not.

The computers ARE coming. No college catalog I have seen offers a course in Overhead Projection 101, but the catalogs overflow with computer offerings. The computer will not go away. What teachers must decide is how this new technology can be used most effectively. These decisions should not be left to commercial interests. As teachers, our challenge is to integrate the microcomputer into a curriculum that meets its objectives while retaining its humanity.



A. Daniel Peck is senior professor of secondary/postsecondary education at San Francisco State University, where he has been deeply involved in teacher education for 30 years. He founded the Educational Technology Center at SFSU in 1960 and ran it until 1975. An NEA life member, he currently supervises student teachers at the middle and high school levels.

As we move into a world that increasingly depends on the new electronic technology of computers, an awesome thing is happening in our schools. It is almost like a new religion — a computer-oriented creed with its own liturgy, clergy, and adherents whose brand of "born again" fervor brooks no heretics.

This year computers will be showing up in record numbers in elementary and secondary classrooms across America. As the new orthodoxy sweeps the land and we hear the cry of "computer literacy for everybody", few teachers seem to be raising questioning voices. Administrators sing the computer's praises and hurry to get on the bandwagon. Parents, too, want their children to grow up computer-smart.

The computer companies are mounting high-powered school sales campaigns, armed with studies showing that their products can be used as educational tools for almost any age group. And schools are buying. Education dollars seem to be available for computers as for nothing else. Some 15 percent of the nation's total instructional materials budget for 1982/83 is reportedly destined for computers and their associated programs (software). Prices are down to the $2,000-$3,000 range for the average microcomputer, and now some of the manufacturers even want to give them away to schools - if Congress will make such donations tax-deductible.

General purpose microcomputers can be used to teach, given adequate time and appropriate software. There's the rub. Researchers at Columbia University's Teachers College recently studied the instructional software currently available - and found most of it boring, of questionable educational value, or both.

Too many of the prepackaged programs concentrate on drill and repetition and fail to encourage students' independent thinking. One software developer charges most publishers with "taking a workbook and giving it to a programmer, so what you get is an electronic workbook" — a waste of the computer's unique capacities as a machine that can think.

Good teachers must spend hours adjusting commercial computer programs to their classroom needs. Some experts suggest that teachers instead create their own software - somewhat akin to expecting teachers to write their own textbooks. Training to help teachers use computers has lagged far behind the purchasing of classroom hardware.

To use this hardware, students must have a high degree of "computer literacy". Computer proselytizers readily admit as much but insist that computer literacy instruction must come first so that the computer can then help children to become literate in basic academic subjects. This is like requiring a course in engine design and development before allowing someone to drive an automobile.

I'm convinced that the financial investment and operational costs involved make computer-assisted instruction the most outlandishly expensive means ever conceived to teach the three R's.

I have yet to be persuaded that our children will be unemployable if they graduate from high school without computer literacy. The technology is changing so fast that computer skills taught in today's K-12 classrooms will be largely obsolete by the time students enter the job market. I expect, for example, development of a computer that speaks our language, which will make the various computer languages passé.

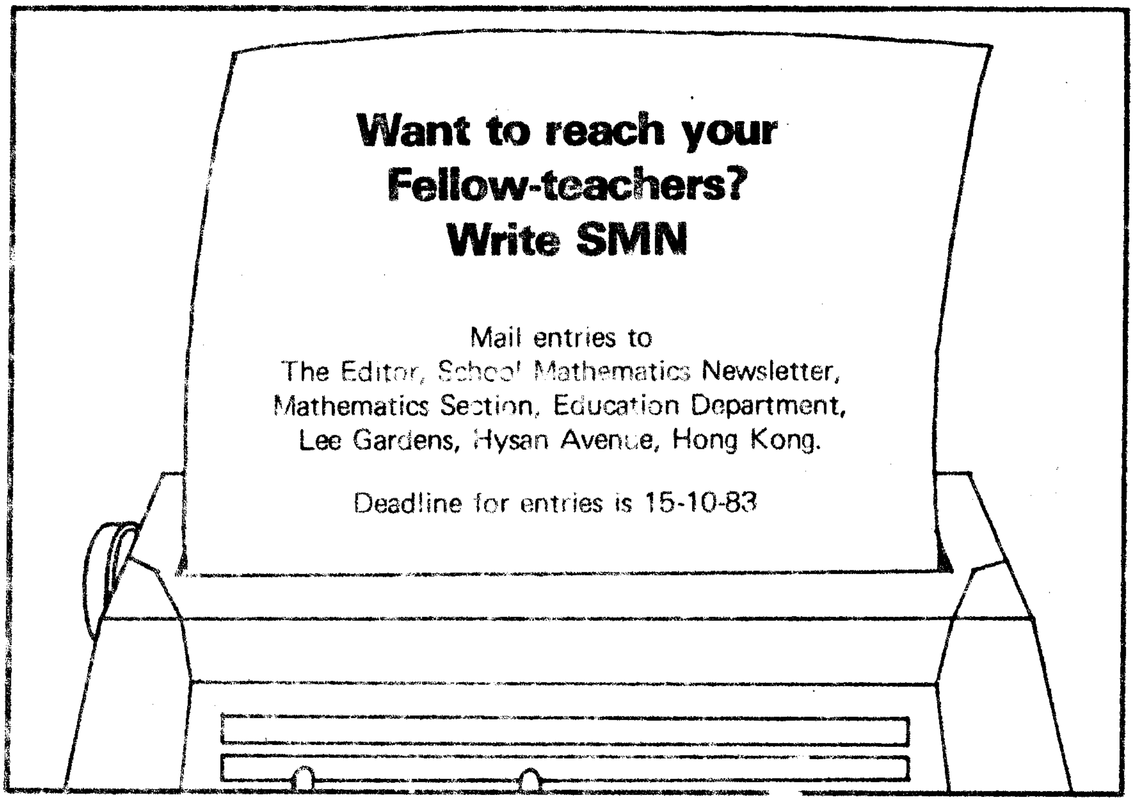
As teachers, I think we have a responsibility to our profession and to our students to ask some pointed questions:

* What happens to students' social skills when they spend hours of classroom time "interfacing" — not with other human beings but with a machine?

* Does the cost of computers pose new equity problems, widening the gap between rich and poor school districts?

* As computer costs go down and teacher salaries go up, are we heading for a time when efficiency-minded school boards will decide it's cheaper to provide each student with a home microcomputer and close the schools?

Knowing the history of educational innovations embraced as panaceas and then abandoned, are we sure we want to hop on this latest bandwagon?



Micros — What to Buy and Why

Gillian Lovegrove
University of Southampton

Reproduced from IMA Bulletin, by permission of the Institute of Mathematics and its Applications.

INTRODUCTION

It will not surprise you to discover that I have no one single processor, no one single printer, disc or cassette that I will advise you to buy, as everyone's needs are different.

There are several stages in making a successful choice of microcomputer:

(i) defining your system requirements and matching them with what is available;

(ii) appreciating what computer systems can offer and how they vary; and

(iii) making a choice from the available equipment.

I will give some ideas on how to define your system requirements; then I shall outline some of the different systems and explain some of the jargon that you will meet in glossy brochures and in conversation with fast-talking salesmen. Finally I will compare some of the more popular computer systems available today.

MATCHING SYSTEM REQUIREMENTS WITH WHAT IS AVAILABLE

This can be split into various questions.

(i) How large a system do you want?

(ii) What kind of input media?

(iii) What kind of output media?

(iv) Does it matter what processor you choose?

(v) How much internal memory is available?

(vi) What external storage media are there?

(vii) What kind of software is provided/easily available?

* standard functions
* games packages
* programming languages
* monitor/operating system
* word-processing

(viii) How good is the documentation?

Let us consider these in detail. It may sound like an endless series of questions, some of which only you can answer, some I can answer, and we hope the salesmen can answer the rest.

(i) <u>How large a system do you want?</u>

Should it be

a small self-teaching system : this is likely to be a single-board computer which is inexpensive but probably cannot be extended; or

a desk-top computer (personal computer); or

more like a mini-computer system : close to the classical computers; or

have you more grandiose ideas : perhaps a network of computers?

There are other questions you need to ask. Do you want to be able to carry it around - does it pack into a briefcase?

I shall refer mainly to personal computers which are likely to be of greater interest.

(ii) <u>What kind of input media?</u>

This is most likely to be a keyboard - do you want both upper and lowercase? You may want to collect data from electronic devices - is there a standard interface? You might want speech recognition - this is not yet readily available, but soon will be.

(iii) <u>What kind of output media?</u>

The most common output is via the Visual Display Unit (VDU). Again there are several questions to ask.

* Will you be happy with a 40-character line of upper-case characters? This is the most common output from micros.

* Will it be sufficient to have a representation of the characters on an ordinary keyboard, or do you want special characters or graphics symbols, or high resolution graphics where the screen is divided into many dots allowing greater flexibility?

* Do you need hard copy (printer)? You may think this is unnecessary, but it is highly desirable for serious programming. Also bear in mind that the quicker the printer moves, the more it costs, and do not buy a faster one than you need. What kind of a printer do you want - one with just keyboard characters, one with the ability to produce characters and graphics shapes, or do you want typewriter-quality characters for word-processing? This affects your choice of printer as you will see later.

* What about output in the form of speech? This does exist, but it is not yet common.

(iv) Does it matter what processor you choose?

There are several different micro-processors and also different versions of the same processor (like the Z80 and Z80A). The Appendix shows some micro-processors and computers that use them. The more recent processors probably have a fast cycle time, which means the time to execute a simple instruction is faster. The cycle time is not too important in comparison with other considerations, since if fast arithmetic is of major importance to you, you would probably need to look beyond micro-computers to mini or mainframe computers.

(v) How much internal memory is available?

Find out by how much it can be extended, and how much of it is actually "free space" for your own programs, and is not taken up by essential system programs.

(vi) What "external" storage media are there?

Are there cassettes or discs? I will discuss their reliability and compare them.

(vii) What software is provided or is easily available?

BASIC is the main language used on micro-computers. You may be greatly concerned that certain arithmetic functions should exist with your BASIC, so check this. Otherwise, these days software for micros is being produced so fast that it is likely that someone (perhaps other than the manufacturer) will already have written some software that you need.

You will need to find how fast the BASIC interpreter is. Are there good error messages? How big a program can you run at any one time - can you find out how much space is left at any time? If you want to run your BASIC program on another computer you will need to check that the BASIC language is exactly the same on both machines (i.e. that your program is compatible). There are many versions of BASIC.

Games packages

Quite apart from the fun we all get from these, they are often the first introduction a child has to a computer and it is important to have good games packages for this reason if for no other.

Languages besides BASIC

(a) Assembler

How easy is it to get assembly language programs in and out of the computer? Sometimes the difficulty incurred in getting a program in and out of the computer far outweighs the difficulty of writing an assembler program.

How good is the assembler itself and its diagnostics?

How good are the debugging facilities?

(b) PASCAL, FORTRAN, COBOL, COMAL et al.

How big a program can you compile at any one time?

Can you compile and link sections?

Do you need extra store/equipment to run these languages?

Monitor/operating system

A good operating system will give you the advantage of direct, easy-to-remember instructions for accessing files and compiling and running programs and it will have a good editor for text files. Is it important to you to use the same operating system as other people? CP/M is a widely used operating system, available on 8080 or Z80 based computer systems.

Word-processing

There are microsystems which specialise in word-processing, but possibly the micro you choose will also have word-processing available. Find out how versatile it is and also how much it costs, as word-processing packages tend to be expensive.

(viii) <u>How good is the documentation</u>?

Make certain that the documentation is good, as the advantages are not to be underestimated. Many hours can be saved by the existence of good manuals - we want to be able to use the computer, not be a slave to it!

WHAT COMPUTER SYSTEMS CAN OFFER AND HOW THEY VARY

To help you to discover about micro-computer systems I will explain some keywords.

(i) Some well known facts

A bit is a binary digit, either a 0 or 1.

A byte is 8 bits (normally).

An integrated circuit (IC) is a semi-conductor circuit in micro form (chip).

A printed circuit is a glass fibre board with a circuit pattern of thin copper conductors to which other electronic components can be added.

A circuit board is a printed circuit with these other components.

(ii) Internal memory

Most micros use 2 bytes, i.e. 16 bits to hold memory addresses. Now 2 ↑ 16 = 65 536 = 64K where K = 1024, so there is an upper limit of 64K different addresses or locations available.

In micro-computers there are usually two kinds of memory, RAM and ROM.

RAMs are Random Access Memories into which you can write as well as read. However, if the power is switched off the contents of the RAM are lost. There are two types of RAM - static and dynamic. The static RAM is more expensive, uses more power and therefore gets hotter so there may be heat dispersal problems. The dynamic RAM however is less expensive but needs extra chips (at extra cost!) to "refresh" the memory as the transistor cells cannot hold the information for very long, so there is actually little to choose between static and dynamic RAMs.

It would be useful to have some instructions or even programs stored permanently in the internal memory. If the machine is switched off and later is switched on again, it will automatically run this permanently-stored program (often known as the Monitor program). It may then wait for you to do something, or it may tell you when the system is entirely loaded.

This memory is called a ROM, i.e. read only memory. We can only read from this memory and cannot write to it, so that the contents cannot accidentally be destroyed. It is said to be a "non-volatile" store, as the contents are permanently stored and will not disappear if the power is switched off.

Another set of similar terms are PROM - this is a programmable ROM, and the EPROM - this is an erasable programmable ROM. It is a relatively easy job to put a new program into a PROM or EPROM if you have the right equipment, but this is not generally so for the personal computer user.

The software in the monitor ROM is called the firmware. (No guesses as to how it got its name!) It may also include a simple BASIC interpreter.

(iii) The BUS system

This is the complex electrical network linking the parts of the computer together. The bus is usually constructed of a number of parallel conductors.

There are single board computers (e.g. Acorn Atom, Sinclair ZX81, Commodore VIC) where the computer system bus is on one printed circuit board. Generally there is an expansion slot to extend the bus to allow for an increase in memory.

There are several board computers which probably have a printed circuit back plane or mother board with these conductors on it. In this case the memory, processor, input and output circuitry can each be on separate boards which link into the back plane.

Here are some examples of different bus interfaces and some computers which use them.

S-100	used by the Cromemco Z80 (the S100 bus has its own international standard)
SS-50	used by the RML380Z, the MSI6800 and SWTPc6809
SS-44	
SS-48	
IEEE-488	

It is advisable to take great care whilst considering fitting boards or systems of the same bus standard. Not all manufacturers follow the same standard. It is best to check with your dealer.

(iv) External storage

An example of this is the domestic cassette recorder. This has the advantage of being cheap. However the rate of transmission is relatively slow - between 100 and 1000 bits per second, which may mean it would take about 10 minutes to load a sizeable program. Not only that, but the quality of the data recording is dubious and the electronics have to be tolerant of speed variation. So at the end of the 10 minutes, the program may not have loaded correctly! There is also no standard recording format. The Kansas City standard is very common, and another is the KIM standard.

Even when you have ensured that the same standard has been used, the tape may not be transferable to another machine, simply because of head alignment problems. However, if you are always going to use the tapes on the same machine this may not be a serious problem.

An alternative is disc storage. A disc drive is at least ten times the cost of an average cassette recorder, and a disc costs about three times tha of a cassette. There are floppy discs and hard discs (which are much more expensive).

Apart from the fact that floppy discs do not like magnetic fields or extremes of temperature, they are quite robust, and are exchangeable.

A common size is the 5¼ inch disc. The amount of information that can be held on a disc varies from 75K bytes (single density) to 180K bytes (double density). Some drives have two heads and thus both sides of the disc can be used (double sided); so there could be up to 360K bytes held on a double sided double density disc.

There is also an 8 inch floppy disc which can hold from 256K bytes to 1.2 Mbytes (one megabyte is 10↑6 bytes).

There are also hard discs which are appearing more frequently on the micro-computer systems. These have a much better rate of transmission of data and are clearly more robust but are more expensive. The 8 inch Winchester disc can hold up to 26 Mbytes and there are some 5¼ inch Winchester drives available now. However these discs are fixed, not exchangeable, so it would be impossible to transfer information from one computer system to another via a Winchester disc.

(v) Input/output devices

VDU

On certain computers there is an integral screen, but some computers have a socket for a connection to a TV aerial on the back.

However Germany, UK, France, USA and Japan all have different standards for TV especially for colour and you may need a UK (PAL) card to do the conversion which might cost extra.

Alternatively a special monitor screen can be used (a colour monitor for those computers offering colour). The definition on a monitor is generally better.

Character generation

If you have 25 lines of 40 characters, which is common on micro-computer VDU's, then you would need 25 x 40 = 1000 memory locations to store information.

The computer then transfers the contents of these memory locations straight on to the screen. With 8 bits of information you could have $2 \uparrow 8 = 256$ different characters possible. The ASCII system of coding characters is the most common and uses 7 bits (128 characters). The other 128 are used to make graphics characters.

There is often a ROM (called the character generator) which will interpret the code into actual characters and graphic symbols. There is, however, no standard and, for example, the PET and the SORCERER have different graphics symbols.

For high resolution graphics, for example Apple II and RML380Z, the screen is split into a series of discrete points. For example for the Apple II there are 280 x 190 points. Since each point requires three bits to represent it for colour and brightness, it needs 18K bytes in total for screen representation.

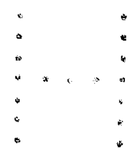
This is very useful if you want good graphics but it does require memory, leaving less for free programming.

Printers

Most printers and VDU's are designed to be machine independent and therefore need a standard interface. This is generally the RS232 (or RS232C) serial interface. The rate of transmission over the interface must be compatible with the computer : for example 110, or 300 or 1200 baud (about bits per second). Another standard interface is the Centronics parallel interface but it is important to remember that the interface must be compatible.

Matrix printers are very common. The printing is not quite as clear as that from an electric typewriter or other printers but it has the advantage of being fast and relatively cheap. The image of the character to be produced is made up of dots from a vertical set of hammer heads moving across the page. Typically there would be seven vertical hammer heads and there would be five dots across the page making up a character (i.e. a 5 x 7 matrix printer).

The character "H" when magnified would look like this :



A matrix of 7 x 9 will give good quality characters.

If your computer has a graphics capability, a dot matrix printer must be used otherwise the special characters cannot be produced since the characters are special characters (i.e. beyond the normal ASCII characters). In fact extra care has to be taken with the PET computer, which must have a PET-compatible printer to reproduce its graphics.

If you want to be able to use different line-widths (i.e. 40, 80, 132 characters etc.), make sure that the printer is wide enough and capable of being programmed or preset to do this.

Another printer is the daisy wheel printer. This produces good quality printing by means of a wheel with characters on it that rotates at high speed, each character which is required being struck by a hammer. This wheel is termed the "daisy wheel", and can be changed very simply to give different type faces or fonts. This is very good for word-processing applications. The disadvantages are that the printer is slower (about 45 characters per second) and also more expensive.

All the printers mentioned so far use normal paper and normal "typewriter" sort of ribbon.

There are also the thermal printers. Think very carefully before buying one of these. They use heat-sensitive paper : a dot-matrix heating element moves across the paper and a blue dot appears when the paper is made hot. The great advantages are that thermal printers are quick and quiet, but this has to be weighed against the cost of the heat-sensitive paper.

Other input/output

So far we have considered various output devices of a standard kind. Input devices are typically keyboards, which may only differ from each other in minor details.

However, it may be necessary to attach to your computer other external input or output devices. If so you will have to take great care in choosing the devices, making sure that the interface is an acceptable one.

The most common standard interfaces are the RS232 (as mentioned above for printers) and IEEE-488. The RS232 gives 8-bit serial transmission and is the most commonly used interface for teletypes, VDU's and modems (devices for sending data over telephone lines, say from a remote terminal). The IEEE-488 (or the GPIB, the general purpose interface bus) gives 16-bit parallel transmission of data and is a standard for controlling and taking information from electronic instruments. Commodore particularly have used it for the PET and PET peripherals like their discs and printers.

Another interface which is used is the S-100 bus which is also 16-bit parallel but is generally incompatible.

MORE POPULAR COMPUTER SYSTEMS

There are many points to be borne in mind when choosing a computer and a computer system. It would be impossible to say that one specific system is best. So, instead I shall mention some of the more popular micro-computer systems that are found in educational establishments today. (The percentages quoted came from an informal survey done by Computer Weekly of micro-computer systems bought by educational establishments.)

The most popular is the PET (50 per cent). It is a desktop system with a 1 MHz 6502 micro-processor and an integral 9 inch video display. There is also a good graphics capability and a wide range of input and output possibilities with its IEEE-488 interface. It has Microsoft BASIC and COMAL and there is also a tremendous range of other software available either from Commodore or other suppliers. It is good value for money.

Then there is the RML380Z (26 per cent). This has an SS-50 bus based system with a faster micro-processor (a 4 MHz Z80A micro-processor). It has been designed for school and educational use, and a school buying this computer as its first computer would have the advantage of a 50 per cent reduction in price! There are plenty of peripherals, good graphics, and the software available is quite considerable, and as with most software for micros, the range of programs is increasing rapidly. However, the RML 380Z is on the expensive side.

The Tandy TRS-80 and its "look-alike" the Video Genie between them have 9 per cent of the educational market. Outside the field of education the TRS-80 has been the world's largest selling computer. The TRS-80I is a smaller cheaper version and the TRS-80II offers more for more money. Both have standard interfaces, floppy discs, cassettes and good software.

Close behind them come the Apple II and its "look-alike" the ITT 2020 with 6 per cent. The Apple is very versatile and has many input/output ports with standard interfaces (IEEE-488 or RS232). The supporting software is generally good and a wide range of languages including Pascal can be used. Also a Microsoft card can be added enabling a Z80A micro-processor to be used with the CP/M operating system. The Apple also has good graphics and colour.

These are all of the larger variety of personal computers, but there are single board computers which are much cheaper, but clearly are more limited in what they can offer. The one in the limelight (deservedly so as good value for money) at the moment is the Sinclair ZX81, but it is worth-while waiting to see what the BBC Micro based on the Acorn Atom will have to offer. At the moment there is not much known about its software capabilities, but it will have far more to offer than the Sinclair ZX81 at a price of around £250.

All these computer systems can be well recommended but there may well be many other systems that would suit your requirements.

It is worth-while getting in touch with your local User Group or Computer Club. There is a list of these in Robin Bradbeer's book referred to in the Bibliography. The User Group can tell you which local dealers can give you good servicing support. This may also influence your choice of computer.

NOTE TO TEACHERS

There are two more points to bear in mind when selecting a computer system : they are robustness and safety. Look out for connections that could become loose in time or wires that could easily be touched. Children will test a computer system to the extreme, so these are important considerations.

CONCLUDING REMARKS

The world of micro-computers has expanded and is continuing to expand rapidly. As time goes on more possibilities occur and there are exciting prospects ahead. Some of today's micro-computers will undoubtedly become obsolete within a few years, but the principles behind making the correct choice of computer at any time will not change so quickly. By following the guidelines suggested in this paper you can look forward to choosing a computer successfully and realising its potential.

I am grateful to Professor D.W. Barron for his helpful comments on this paper.

BIBLIOGRAPHY

There are many books and periodicals on different aspects of personal computing. I shall refer you to only a few, each of which can give you a host of further references to suit your particular needs.

The first is "The Personal Computer Handbook" by Robin Bradbeer, obtainable from Gower Publishing Co. Ltd.

Next are two periodicals, Practical Computing and Personal Computer World, which in addition to being invaluable sources of information also have details of many bargains to be obtained in the second hand computer market.

APPENDIX

Most Common Micro-processors

8-bit	8080	One of the first - now old-fashioned.
	Z80	Based on the 8080 : uses 8080 instruction set and a lot more : Z80A is a fast-speed version with a clock frequency of 4 MHz.
	6800	Also old-fashioned.
	6502	Based on the 6800 but is more flexible : very fast when used with the BASIC interpreter from Microsoft Ltd : a 4 MHz version exists and is very fast.
16-bit	8085 8086	Based on the 8080 and halfway to a true 16-bit processor.
	9900 68000 Z8000	These are 16-bit micros and the computer systems which incorporate them are rather more expensive - in the region of £8000 to £10000.
32-bit?		Micro mainframe! Watch the computer magazines for news of these!

Some micro-processors and the computers which incorporate them

6502

Acorn Atom
(BBC Micro)
ITT 2020
Commodore PET
Apple II

Z80A

Sinclair ZX81
Sharp MZ80K,
MZ80B
TRS-80/II
Horizon (North Star)
RML380Z

Z80

Tandy TRS-80/81
Video Genie
Cromemco System 2

HP

Hewlett Packard's own

6809

SWTP6809 [redacted]

How BASIC Causes Almost Irreversible Brain Damage and How it Can be Made Safe

Roy Atherton, AFIMA
Bulmershe College of Higher Education

Reproduced from IMA Bulletin, by permission of the Institute of Mathematics and its Applications.

1. EXAMPLES OF BAD PROGRAMMING STYLE

The first example came to my notice in about 1975 and contributed to general unease felt about BASIC, although, as a teacher, I had welcomed it in 1970 because it had three overriding virtues:

(i) simple syntax;

(ii) easy operating environment;

(iii) cheap to implement on small computers.

```
20 PRINT "TIME(S)","HEIGHT(M)","VEL(M/S)","FUEL(KG)","BURN(KG/S)"
30 G0=1.62\M=26000\D=10000\F=13000\T=1\T1=0\S=1
33 P=1125\R=3\V=100\U=100
40 PRINT T1,D,V,F,\INPUT B,D2\B=ABS(B)
45 FOR X=S TO D2 STEP S\R2=1.700000E+06
47 G=G0-2*D/R2\IF D<1.000000E+07 THEN 50\PRINT "TOO FAR OUT"\STOP
50 V=U+T*G-(B*P*T)/((M+(M-T*B))/2)\T1=T1+T
60 M=M-(T*B)\D=D-((U+V)/2*T)\U=V\F=F-B*T
65 K=K+(B*.23)-9.87854\IF P*B+G*100>250000 THEN 220
70 IF D<=0 THEN 110\NEXT X\IF K>2000 THEN 200\IF K>1500 THEN 210
75 IF F>=0 THEN 40
90 PRINT "OUT OF FUEL AT";T1\B=0\S=1.000000E-03\GOTO 45
110 PRINT "ON MOON AT ";T1;"SECONDS.LANDING VELOCITY";V
120 IF V<10 THEN 160\IF V<20 THEN 170
130 PRINT "ALL CREW KILLED.BLASTED NEW CRATER ";V*11.78;"KM WIDE"\STOP
160 PRINT "SAFE LANDING"\STOP
170 PRINT "CREW INJURED ";INT(3.2*V/17.46);" BONES BROKEN"\STOP
200 PRINT "POWER TUBE BURN OUT";R-1;"LEFT"\R=R-1\IF R=0 THEN 207
205 P=P/(4-R)\GOTO 75
207 PRINT "ALL TUBES GONE"\S=1.000000E-03\B=0\GOTO 45
210 PRINT "POWER TUBES TOO HOT"\GOTO 75
220 Z=RND(0)*30\PRINT "BLACK-OUT FOR";INT(Z*D2);"SECS"
230 D2=Z\B=25\GOTO 45
```

Fig. 1 Moon landing program

This moon landing simulation (Fig. 1) was written in a local school for a 4K PDP8 minicomputer using EDUSYSTEM 10 — a primitive BASIC. The use of multiple-statement lines may have been necessary to squeeze it into a small memory. However, it has been unravelled and the first stage just cleans up the mis-use of the multiple-statement facility. It was still a mess. Space does not allow the reproduction of all stages but it has been programmed properly without GOTO statements.

Let us move into higher education for a second example, taken from a batch of statistical and other programs written by staff in the Geography Department of the University of Liverpool. The segment shown in Fig. 2 is taken from KENRANK which helps statistical users. It was written for a PET microcomputer and there are 79 more, not all as bad. Generally when the program segments are sequential or use FOR loops they are much better but there is little evidence of any attempt to structure the programs properly.

```
670 IFR(I+M)>=0THEN 690
680 X=R(I+M): I3=I3+1
690 NEXTI:IFI3<=0THEN830
700 Y=X:C3=0
710 FORI=ITON
720 IFR(I+M)<>XTHEN740
730 C3=C3+1
740 NEXTI
750 IFC3=0THEN640
760 IFM=NTHEN800
770 IFK1=ITHEN790
780 T1=T1+C3*(C3-1)/2:GOTO640
790 T1=T1+(C3 3-C3)/12:GOTO640
800 IFK1=ITHEN820
810 T2=T2+C3*(C3-1)/2:GOTO640
820 T2=T2+(C3 3-C3)/12:GOTO640
830 NEXTM
840 IFT1<>0THEN870
850 IFT2<>0THEN870
860 T=S/(.5*S3):GOTO880
870 T=S/(SQR(.5*S3-T1)*SQR(.5*S3-T2))
880 S1=SQR(( 2*(S2+S2+5))/(9*S3))
```

Fig. 2 A segment from the program "KENRANK"
The flowlines show the "spaghetti-like" structure.

Finally let us consider a small program (Fig. 3) taken from Christine Doerr's book "Microcomputers and the 3 r's".[2] It can be criticised on three points:

(i) the short variable names (C, X, Y, Z, T, G, B) are unhelpful and unnecessarily abstract;

(ii) the spaghetti-like nature of the flow of the control is difficult to follow. Consider, for example, an attempt to trace an error in line 90 (Fig. 5);

(iii) even in BASIC the program could be better structured. The tracing of routes for error correction or development is very much a "hit or miss" affair.

```
21 PRINT "ADDITION PROBLEMS."
30 C=C+1
40 IF C>5 THEN 190
50 T=1
60 X=INT(RND(TI)*10)
70 Y=INT (RND(TI)*10)
80 PRINT X"+"Y"=";
90 INPUT Z
100 IF Z<>X+Y THEN 124
110 IF T=1 THEN G=G+1
120 GOTO 30
124 T=T+1
126 IF T>3 THEN 160
130 PRINT "NOPE,TRY THAT ONE AGAIN"
150 GOTO 80
160 PRINT "SORRY, THE ANSWER IS "X+Y
180 GOTO 30
190 PRINT "GOOD, YOU GOT "G"OUT"
191 PRINT "OF 5 CORRECT."
200 INPUT "DO YOU WANT MORE (1=YES 0=NO)";B
210 IF B=1 THEN C=0:T=0:G=0:GOTO 30
```

Fig. 3 Arithmetic test program in BASIC
The lines show "flow of control" during execution.

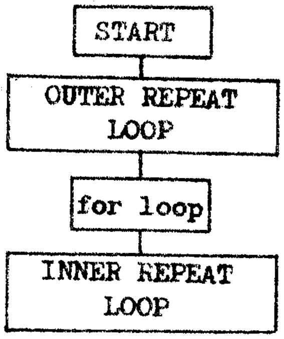


Fig. 4 Route to an error in line 100 of COMAL program

(1) 30, 40, 50, 60, 70, 80

(2) 30, 40, 190, 210, 30 then as (1)

(3) 30, 40, 50, 60, 70, 80, 90, 100, 110, 120 then as (2)

(4) 30, to line 100, 124, 126, 160, 180 then as (1)

etc. ....

Fig. 5 Possible routes to an error in line 90 of the BASIC program

```
10 print "Addition Problems."
20 repeat
30   SCORE=0
40   for PROBLEM=1 TO 5
50     FIRST=int(rnd(1)*10): SECOND=int(rnd(1)*10)
60     SUM=FIRST+SECOND: TRY=0
70     repeat
80       TRY=TRY+1
90       print FIRST; "+"; SECOND; "=";
100      input ANS
110      if ANS<>SUM then print "No, try again."
120    until ANS=SUM or TRY=3
130    if TRY=1 then SCORE=SCORE+1
140    if ANS<>SUM then print "The answer is"; SUM
150  next PROBLEM
160  print "You scored"; SCORE; "out of 5."
170  print "Type 0 to finish or 1 to continue."
180  input CONTINUE
190 until CONTINUE=0
```

Fig. 6 The arithmetic test program in COMAL

<u>Notes</u> 1. The expression int(rnd(1)*10) generates a number randomly.

2. The symbol <> means "is not equal to".

3. The indenting illuminates the structure.

The COMAL program (Fig. 6) performs exactly the same task and the above criticisms disappear.

1. The long variable names (SCORE, PROBLEM, TRY, FIRST, SECOND, ANS, SUM, CONTINUE) help the readability and avoid a level of abstractness.

2. The program reads from top to bottom and each structure (two "repeats" and one "for") is clearly identifiable and properly related to the other two, i.e. nested. System forced indenting and system-forced lower case for keywords also helps.

3. The program is about as well-structured as it can be because the essential concepts (two repeat loops and one for loop) were properly identified at the analysis stage. Debugging or amending would be easy because the flow of control leading to any point is always from the start of the program downwards through each structure to that point (see Fig. 4).

While it is perfectly possible for a highly disciplined experienced programmer to write well-structured programs in any language the knowledge of the essential concepts is vital. In practice only a small minority of programs are well written and the embodiment of the concepts (for, repeat, while, if/then/else, cases, procedures) in a language would guarantee that user would not only write much better programs but also use the ideas at the problem analysis stage.

2. EXAMPLES OF BAD PROBLEM ANALYSIS PROGRAM DESIGN

It is suggested that computers program people. In particular the syntax of a programming language influences users. The concepts embodied in the syntax feed back into the problem analysis style and affect, for better or worse, the way problem-analysis skills develop. This is not to deny the possibility of a good programming style with a non-structured or poorly-structured language. An experienced, well-read or well-taught programmer can discipline himself to do this, but there are now tens of thousands of people, including many teachers, who have "picked up" BASIC from books or short courses which take little account of the software developments since about 1968 when Dijkstra published his paper[3] "GOTO statement considered harmful". The following are examples of the way the GOTO/Flowchart concepts have become part of the analytical equipment of those who produced them.

The first example is taken from a very useful little book[4] about data structures. It is aimed at teachers, A-level or college students or others interested in computing science and its practice. Although the book gathers together some good material and presents it very well, the suggested algorithms leave much to be desired (see Fig. 7).

This one performs a Pre-order Tree Traversal, and might be used to convert a mathematical expression into Polish (Pre-fix) notation.

1. Access the root node.
2. Access the eldest child of the currently accessed node.
3. If the currently accessed node is a parent node then go to Step 7.
4. If the next eldest twin node of the current node exists, access it and go to Step 3.
5. If the stack is empty then finish.
6. Unstack a node pointer and if this node's next eldest twin node exists, access the next eldest twin node and go to Step 3, otherwise go to Step 5.
7. Stack a pointer to the current node and go to Step 2.

```
1.  Access root node.
2.  repeat

      if son exists then
        while there is a next eldest son stack him
        visit the eldest son

      else
        if stack is empty then stop
        unstack node
    until  2 = 1
```

Fig. 7 Unstructured and structured "tree-traversal" algorithms

The second example is a flowchart taken from Computer Education.[5] Fig. 8 shows a solution to a queue simulation problem which seems popular in GCE computer studies courses. I have witnessed the kinds of difficulties this particular problem can cause in a GCE class when a flowchart approach is adopted. The problem relates to the simplest type of queue, for example, television sets arriving in a random manner for repair.

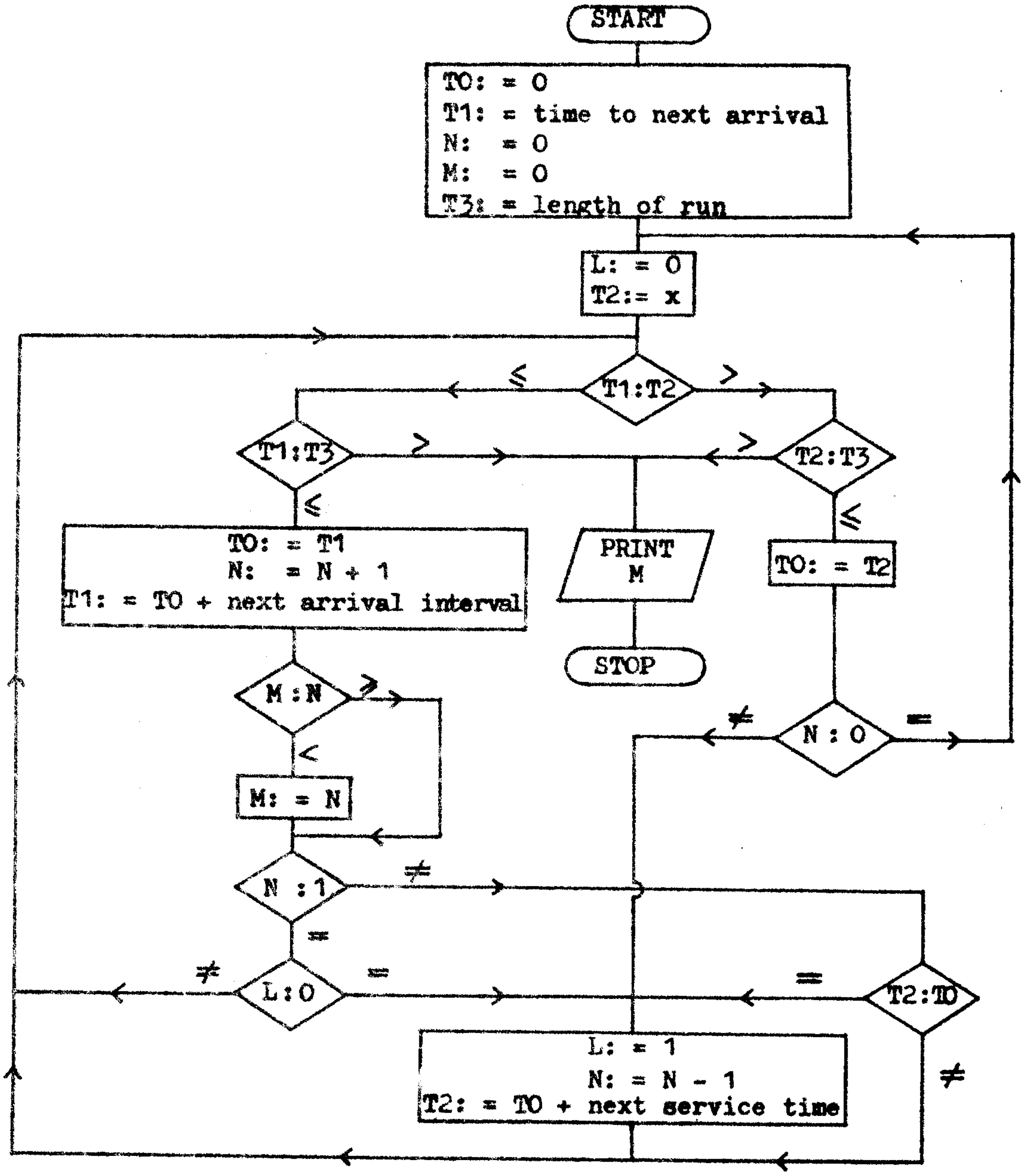


Fig. 8 Flowchart for single server with one input stream — Event-based

This problem can be solved in a more natural and straightforward way by constructing a table of the required data instead of trying to simulate the actual flow of events.

(i) The arrival time and service time for all jobs are easily computed from given data.

(ii) The time of entry into service is the difficult bit because it depends on varying conditions.

(iii) Once entry into service time is known the job-completion (exit from service) time is easily computed because the service time is already known.

(iv) When the table of data is completed it is easy to extract any desired statistics such as average waiting time.

Adopting this data-oriented approach rather than flow-charting and knowing the few essential concepts of structured programming, it is fairly easy to devise the algorithm for stage 2 of the above analysis. Job number one must be dealt with first to avoid an attempt to refer to a non-existent previous job. The rest is easy.

```
for JOB = 2 to 20
  if ARR(JOB) > COMP(JOB-1) then
    ENTRY(JOB) = ARR(JOB)
  else
    ENTRY(JOB) = COMP(JOB-1)
  endif
  COMP(JOB) = ENTRY(JOB)
    +SERVICE(JOB)
next JOB
```

3. DIFFICULTY OF REVERSING THE BRAIN DAMAGE

I have been teaching student teachers and serving teachers COMAL since 1980. Some students acquire the necessary concepts easily and write readable well-structured programs. One student with an A-level in computer science preferred to use 380Z BASIC to write a Simplex method program. After many hours and much debugging at the keyboard the program worked but it had the usual unreadable spaghetti-like control paths. In contrast three other students who chose to do the same problem presented well-structured, readable programs without any GOTO statements.

About 2 years' experience of BASIC for A-level had caused the first student to acquire GOTO/Flowchart oriented styles of programming and problem analysis. She was reluctant to learn the more modern techniques. Her brain was damaged and it proved difficult, although not impossible, to effect a cure.

Unfortunately, the vast majority of serving teachers cannot get released to attend substantial computer education courses. And the small proportion that do manage to organise something might well find that they get more of the same — more BASIC. Clive Sinclair has sold tens of thousands of ZX80 machines. The BBC is running a series of programming lessons by television. There may be a late decision to change to COMAL or other properly structured BASIC but the plans were for a cheap machine with some of the correct structures added to BASIC.

Computer education is only just beginning to get parity of esteem with such subjects as English, art, history, geography, physical education in the schools and in most teacher education colleges or departments there are more sociologists than computer scientists. At best there will be only a trickle of teachers adequately prepared for computer education, and there seems little chance of proper training for the teachers already providing the courses for CSE or GCE. In 1980 about 36,000 children took formal examinations in computer studies and the trend is sharply upwards. But there is very little teacher training.

In these circumstances reform is not easy. It is not easy to see how the brain damage is going to be reversed.

4. HOW BASIC CAN BE MADE SAFE

There is no secret about what is wrong with BASIC. From the point of view of program structure and readability it has only FOR loops, dubious versions of IF/THEN/ELSE and a very half-hearted type of subroutine. The last two things need to be tidied up and to the last should be added REPEAT/UNTIL and WHILE loops, CASE statements and long variable names. B. R. Christensen defined[6] COMAL as long ago as 1975. He wisely chose to extend rather than confront BASIC and, after 5 years of testing and development, a standard[7] was agreed in 1980 by a Danish-based committee.

Current implementations also provide system-forced indenting, automatic provision and editing of line numbers, and one implementation forces keywords of the language into lower case and other words into upper case. These are not trivial details. They make a big difference to users, especially beginners. Some examples will indicate the style of the extensions although a full exposition of all the forms of all the structures of COMAL-80 including the use of local variables and parameter passing is beyond the scope of this paper.

Program 1

Suppose that a foreman wants a worker to dig six post holes. The computer language style might use the idea of defining one sequence :

"Dig a hole"

and then instruct six repetitions of it.

```
for POST = 1 to 6
  print "Dig a hole"
next POST
```

This illustrates the FOR loop.

1. Note how long variable names (e.g. POST) help understanding.

2. Note how (system-forced) indenting helps understanding.

3. Note how the system differentiates between keywords (lower case) and other words of the program.

Program 2

Now suppose that the foreman knew only that the posts should be 15 metres apart and that they should go to the end of a field. He would say something like, "Dig holes 15 metres apart until you get to the end of the field ".

```
DISTANCE = 100
repeat
  print "Dig a hole."
  DISTANCE = DISTANCE - 15
until DISTANCE < 15
```

This is a very natural concept and necessary in a sensible approach to problem analysis and program design — whatever might be the ultimate details of coding.

Program 3

Taking things a stage further the foreman might say, "Dig holes 15 metres apart until you reach the end of the field. If the ground is soft dig a two-foot hole, otherwise dig a one-foot hole."

```
randomize
DISTANCE = 100
repeat
  SOFT = int(rnd(1)*2)
  if SOFT = 1 then
    print "Dig a two foot hole"
  else
    print "Dig a one foot hole"
  endif
  DISTANCE = DISTANCE-15
until DISTANCE<15
```

With just a small increase in complexity the indenting becomes even more helpful.

Program 4

A further complication might be the existence of a tree where a post hole should go. The instruction here is to call for a particular service: "TREECHOP".

```
randomize
DISTANCE = 100
repeat
  TREE = int(rnd(1)*2)
  if TREE = 1 then exec TREECHOP
  SOFT = int(rnd(1)*2)
  if SOFT = 1 then
    print "Dig a two foot hole"
  else
    print "Dig a one foot hole"
  endif
  DISTANCE = DISTANCE-15
until DISTANCE<15
proc TREECHOP
  print "Cut down tree"
  print "Dig out root"
endproc
```

This program illustrates the simplest form of the procedure concept. Modules can be written and tested separately and executed by a simple statement in the main program. Note that the procedure is properly named and its beginning is properly defined. It cannot be entered accidentally. The GOSUB concept of BASIC is relatively primitive and unhelpful, although COMAL has it for purposes of compatibility.

Program 5

Sometimes it is necessary to provide for exit from a loop before it is properly entered. Just as a FOR loop should be capable of no iterations.

The WHILE concept provides this.

```
DISTANCE = 12
while DISTANCE > 14 do
  print "Dig a hole."
  DISTANCE = DISTANCE - 15
endwhile
```

The REPEAT loop would give a wrong result here because the sequence must be executed at least once.

This is a very general and powerful concept:

```
while (reason for looping) do
   (statements)
endwhile
```

There is, of course, more to learn about good programming and problem analysis but five of the most important concepts have been illustrated. The reader may observe a unity between three stages of programming:

(1) Analysis and design.

(2) Graphical representation (structure diagram).[8]

(3) Program.

After a somewhat pessimistic paper it is gratifying to end on a note of optimism. There are now three implementations of COMAL in UK on the Commodore PET, the RC Piccolo and the DDE SPC 1. Other companies are known to be interested. Teachers in Denmark are the only ones in the world to have tried both BASIC and COMAL on a nationwide scale and they have unanimously rejected BASIC.

In 1980 and 1981 there was a steady flow of articles and conference papers about COMAL and Graham Beech's book[9] uses a problem analysis language which is essentially COMAL. This could be an important bridge between good computer education and the fact of having to use microcomputers without COMAL for a while. I can also report a steady flow of enquiries about COMAL.

It is the expressed view of Christensen that COMAL is easier to use than BASIC: "COMAL programs can be developed in about half the time and debugged in about one tenth of the time." I agree with the spirit of this estimate and am happy to go on record predicting that COMAL-80 signals the beginning of the end for BASIC, although, of course, it will be an unconscionable time dying.

REFERENCES

1. Thomas, G. S. P., Distributor "Interactive Programs for the PET Microcomputer". Department of Geography, University of Liverpool.

2. Doerr, Christine, "Microcomputers and the 3 r's", Hayden, 1979.

3. Dijkstra, E. W., Comm. ACM, 1968, 11, 3.

4. Bailey, B., "Data Structures", Blackie, 1980.

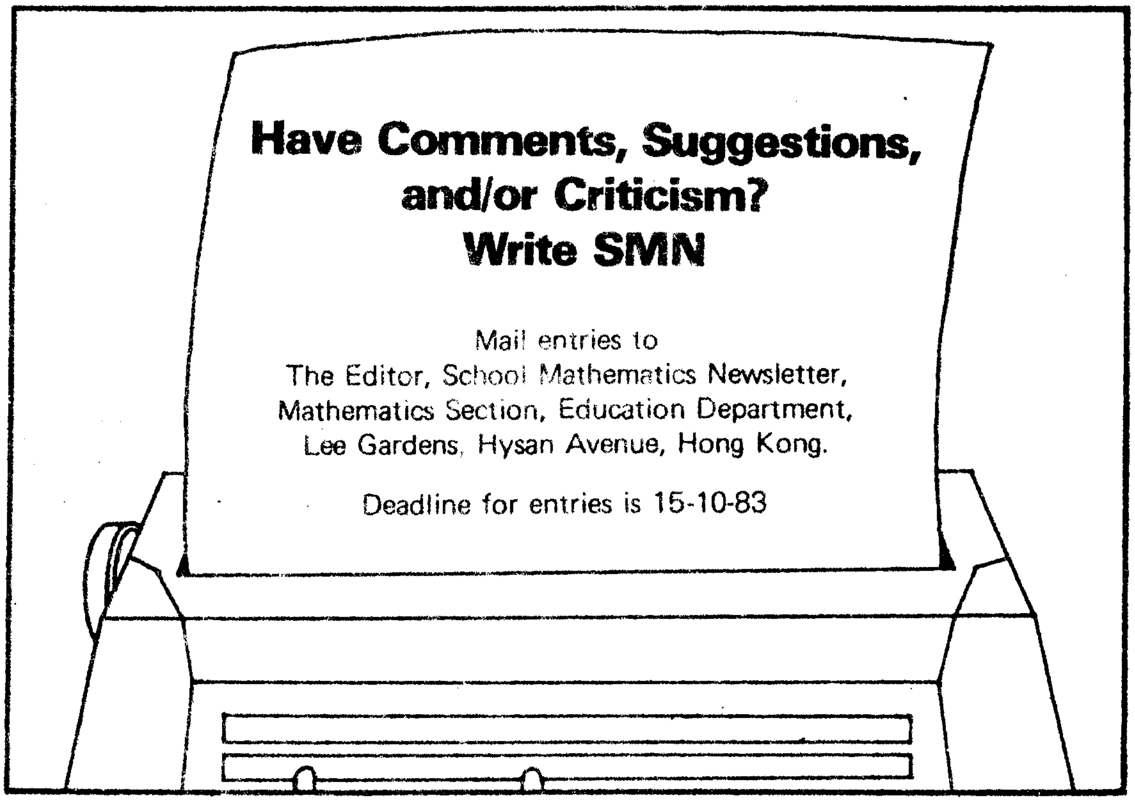
5. Oldknow, A., Computer Education, 1980, No. 36.

6. Christensen, B. R., World of Computer Education, 1975, 1, No. 8.

7. Osterby, T., Editor, "COMAL 80 (Nucleus) Definition", Technical University of Denmark, 1980.

8. Atherton, R., "Structured Programming with COMAL 80", Ellis Horwood, 1981.

9. Beech, G., "Successful Software for Small Computers — Structured Programming in BASIC", Sigma Technical Press, 1980.



PASTIMES

If you have any favourite puzzles, games, riddles, etc., please send them in for use in this column. —— Editor

PUZZLES

1. What is the missing number in the sequence?
   10, 11, 12, 13, 14, 15, 16, 17, 20, 22, 24, 31, 100, ___, 10000, and the curious final number, 1111111111111111.

2. There are four 3-digit numbers which are the sum of the cubes of their digits. They are, in ascending order of magnitude : 153, 370, ___ and 407. Can you find out the third number?

3. What is the next figure in the series?

4. Where does the "Z" go : above the line or below, and why?

   A EF HI KLMN T VWXY
   ———————————————
   BCD G J OPQRS U

5. If (a) [stand / I] represents "I understand",

   (b) [you just me] represents "just between you and me",

   what does each of the following represent?

   (c) [wheather]

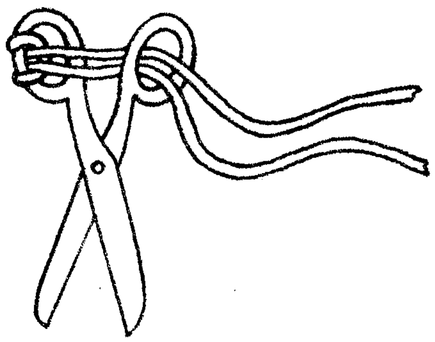
   (d) [O / Ph.D. / M.A. / B.A.]

6. Three boxes are labelled "red balls", "blue balls" and "red and blue balls". Each label is incorrect. You may stick your hand into one (and only one) box and pull out one (and only one) ball. No peeking is permitted. How can you label each box correctly?

GAMES

1. Can you cut a hole in a 8cm x 12cm note card that is big enough for your whole body to squeeze through without tearing the card?

2. A pair of scissors is threaded with a cord, as shown below. The loose ends of the cord may be tied to a doorknob or may be held by someone. Can you release the scissors without cutting the cord?



RIDDLES

Do you know the answers to the following riddles? Clues such as "My first ...., My second ...." refer to the syllables of a word. Syllables are sometimes punned, and the breaks between them may not be in the standard places.

1. My first's a trinity.
   My second's mobility.
   My third's antithetic.
   My fourth's egocentric.
   My fifth's your endeavour to put me together : The mastery of bight —— acute, obtuse, or right.

2. My first is short for tops in sports.
   My second small in mass.
   My third's an ancient dynasty.
   My whole a typed-in task.

ANSWERS WILL BE FOUND ON THE NEXT PAGE. [redacted]

ANSWERS TO PASTIMES

PUZZLES

1. The missing number is 121. (The numbers are all representations of the number 16, expressed in different bases, starting from base 16 to base 1. Thus 121 is 16 expressed in base 3.)

2. 371

3. The figures represent the lights that are left off on the digital display of a calculator or watch as the digits 0 to 7 appear. The next digit, 8, uses all the line segments, 8. So if you left your answer blank, you're right!

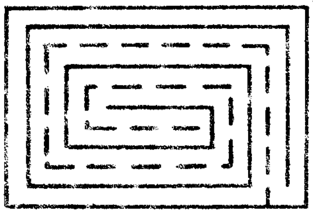
4. The "Z" goes above the line. The pattern : straight letters on top, curved ones below.

5. (c) "a bad spell of weather"
   (d) "3 degrees below zero"

6. Pick a ball from the "red and blue balls" box. Whichever colour it is, put the appropriate label on that box. Remove the incorrect "red and blue balls" label, which must belong to one of the other two boxes. Since all three labels were incorrect. The "red and blue balls" label must belong to the box still with its original label.

GAMES

1. Solution 1 : Cut the card along the dotted line and then the solid line, as shown below. Spread the paper into a ring that is large enough to squeeze through.



   Solution 2 : Make a slit along the middle of the card (Fig. 1). Fold card along the slit and cut alternately from both sides (Fig. 2). With narrow enough cuts, you can spread the paper into a ring that is large enough to squeeze through.

Fig. 1

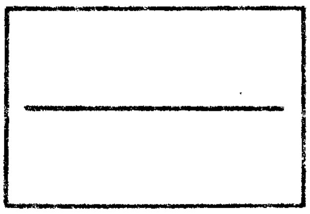
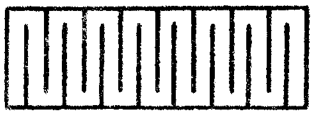


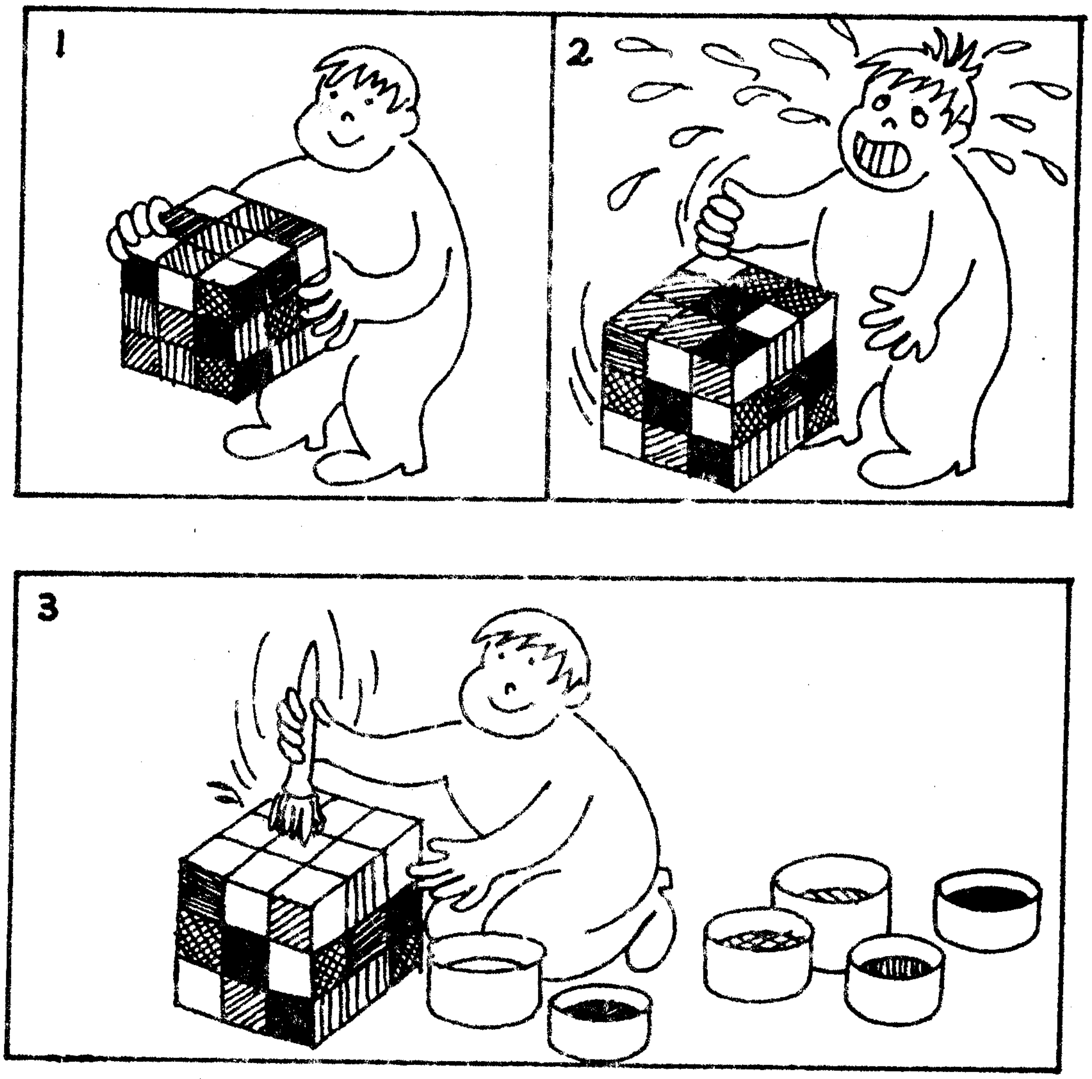
Fig. 2



2. Pull on the end of the loop that is knotted around the left handle. As this loop lengthens, slide it along the rest of the cord and thread it through the right scissors handle along the path of the rest of the cord. Continue lengthening the loop and, when it is sufficiently large, slip the scissors through it, taking care not to twist the cord. Pull the scissors and they will come free.

RIDDLES

1. tri-go-no-me-try
2. pro-gram-Ming [redacted]



NEW PUBLICATIONS

TEACHING STATISTICS IN SCHOOLS THROUGHOUT THE WORLD

Ed. V. Barnett

a new publication of the

INTERNATIONAL STATISTICAL INSTITUTE
(pp XVI, 250, 1982)

The volume presents a review of the exact nature of statistical education in approximately 20 different countries. For each country (or group of countries) considered, an individual with first-hand experience of the prevailing circumstances has presented a personal description of the present situation, of the way in which it has developed and of possible future prospects. The countries covered are from both developed and developing countries.

The general structure of school education in each country is outlined in order to clarify understanding of the details of the provisions made for teaching statistics. The outline includes information on types of schools, the pupils they cater for, principles of administration of the educational system, methods of teacher training, patterns for examinations, prospects for curriculum reform etc. The titles of the chapters are as follows:

CHAPTER 8 TEACHING STATISTICS IN SCHOOLS IN AUSTRALIA
(by J.B. Douglas)

CHAPTER 9 SCHOOL-LEVEL STATISTICAL EDUCATION IN NEW ZEALAND
(by J.B. Douglas)

CHAPTER 10 SOME PROBLEMS OF THE TEACHING OF STATISTICS IN DEVELOPING COUNTRIES - THE NIGERIAN EXAMPLE
(by J.O. Oyelese)

CHAPTER 11 STATISTICAL EDUCATION IN SCHOOLS IN UGANDA AND OTHER EAST AFRICAN STATES
(by S. Tulya-Muhika)

CHAPTER 12 THE TEACHING OF STATISTICS IN THE SECONDARY SCHOOLS OF SUDAN
(by D.Y. Downham and H. Gattar)

CHAPTER 13 STATISTICAL EDUCATION IN SOUTH AFRICAN SCHOOLS
(by J.M. Juritz)

CHAPTER 14 TEACHING STATISTICS IN SCHOOLS IN ARGENTINA
(by L.A. Santaló)

CHAPTER 15 TEACHING STATISTICS AT SCHOOL LEVEL IN MALAYSIA
(by V. Barnett)

CHAPTER 16 SOME PUBLICATIONS ON TEACHING STATISTICS IN JAPAN
(by V. Barnett)

The volume is intended to be of service not only to those involved in teaching statistics as a separate discipline within schools, but also to those teachers and educators who are involved with teaching statistics as part of other disciplines: biology, chemistry, mathematics, physics, the social sciences for example.

Orders for the Volume should be directed to:

The International Statistical Institute
428 Prinses Beatrixlaan
P.O. Box 950
2270 AZ VOORBURG
Netherlands

The price of the volume is US$ 10 or UK£ 6.

Teaching Mathematics and its Applications

(incorporating the *Journal of Mathematical Modelling for Teachers*
sponsored by Cranfield Institute of Technology and the Open University)

Editor

Professor David Burghes, FIMA,
(School of Education, University of Exeter, Tel. (0392) 52221)

Professor Burghes is a professor of education in the University of Exeter and prior to taking up this post in October 1981 he directed the Centre for Teaching Services at Cranfield Institute of Technology. There he initiated the *Journal of Mathematical Modelling for Teachers*. The latter Journal has ceased publication and has been incorporated into the new IMA Journal *Teaching Mathematics and its Applications*.

This Journal aims to provide mathematics teachers in secondary and tertiary education with interesting and stimulating material which can be used in their teaching. Articles showing applications of mathematics, which can be taught, are included as are articles that seek to enhance and improve the teaching of mathematics.

ASSISTANT EDITORS: Dr. J. Berry, FIMA (Open University); Mr. B. Bolt (School of Education, University of Exeter); Dr. I. Huntley, FIMA (Paisley College of Technology); and Michelle Drake (Institute of Mathematics and its Applications)

The following papers are among those published in the first volume.

Foreword — His Royal Highness, The Duke of Edinburgh, KG, KT
The Simplified Mathematics of Tidal Power — Professor Sir Hermann Bondi, KCB, FRS
The Mathematics of Brownian Motion — Professor Sir Sam Edwards, FRS
Convergents to $\sqrt{2}$ and "Almost Isosceles" Right Angled Triangles — P. E. Trier, CBE
Monitoring Petrol Consumption — Anne Walton
Applications of Mathematics for Pupils of Low Attainment — A. Fitzgerald
Mathematics of Voting Systems — John Berry and David Burghes
Mathematics and Rowing — Stewart Townend
Mathematics — Applied, Application, Applying — John Hersee
Find the Next Term I and II — David Hobbs
Stretching the Circle — Brian Bolt
Taking X-Rays — Susan Pirie
Chinese Mathematics 200 BC to 1400 AD — John Hiscocks
Isn't "Maths" Boring!? — Robin Urwick
Degenerate Affine Transformations of the Plane — Brian Bolt
The Alabama Paradox — Frances Robinson
Some Questions on School Examinations — Sir Harry Pitt, FRS

PRICE — 1982 — IMA Members £6 Non-members £8
1983 — IMA Members £7 Non-members £9

Orders can be placed by returning the tear-off slip together with the appropriate remittance.

To: The Secretary and Registrar, The Institute of Mathematics and its Applications,
Maitland House, Warrior Square, Southend-on-Sea, Essex, SS1 2JY.

TEACHING MATHEMATICS AND ITS APPLICATIONS

I enclose £ for (*delete as appropriate)

*a) 1982 issues of the Journal
*b) 1982 and 1983 issues of the Journal
*c) 1983 issues of the Journal

Name

Address

Grade (If IMA member) Signature

All cheques should be made payable to the Institute of Mathematics and its Applications.

FROM THE EDITOR

I wish to express my sincere thanks to those who have contributed articles and also to those who have helped in the preparation of this issue of SMN.

Grateful acknowledgement is made to the Institute of Mathematics and its Applications and to the National Education Association for their permission to reproduce articles from IMA Bulletin and NEA Today.

Readers are cordially invited to send in articles, puzzles, games, cartoons, etc. for the next issue. Contributions need not be typed. Write : The Editor, School Mathematics Newsletter, Mathematics Section, Education Department, Room 417, Lee Gardens, Hysan Avenue, Hong Kong.

For information or verbal comments and suggestions, please contact the editor at 5-8392479.