

2004 / 2005 學年教學設計獎勵計劃

Pascal 原來可以這樣學

參選編號：C032

學科名稱：資訊

適合程度：高一

簡介

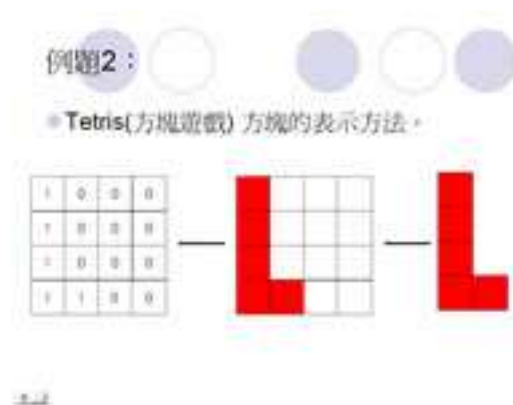
設計目的



使學生能用電腦編寫程序解決日常生活中的問題，進一步學習**實際問題轉換成數字形式**的表示方法及**數模轉換等技巧**，從設計程序中，增強編寫程序的**動手實作能力**，加深了解和認識電腦的運作過程，理解電腦的長處和不足，從而對將來使用或設計電腦程序等方面更得心應手。

內容摘要

本單元的設計是承接上一單元《寫程序解決現實生活中的問題 I》，以實現及講解上一單元所提示的改進解法為基礎。引入新的題目及 IQ 題，如方塊遊戲(Tetris)、蝸牛問題、迷宮(Maze)的實現技巧及算法分析(二維數組的使用及遞歸算法的思路)。誘發學生使用電腦編程解決問題，增加學生學習電腦編寫程序的興趣。



創意與特色



本教學設計的設計特色在於應用**多元智慧**的理論於教學中，以**小組實習形式**，培養學生們的**人際智慧**，使學生更接近現實生活工作中的**集體編程情況**。有別於一般基礎編程教學課程。而在數模轉換及解 IQ 題的學習中，對學生的**數學-邏輯智慧及空間智慧訓練**有著重要的作用，這都是傳統以教授語法為主的編程教學中所缺乏的。

本課程有助激發學生的**創造能力**、通過調試和比較各組所編寫的程序，更正學生一些錯誤的編程概念，把當中較好的方法和思路向各組推廣，最後再把前人所設計的方法作介紹，集思廣益。脫離以往死板的板書講授程序撰寫，大大增強了學生學習電腦程序設計的興趣，提高教學效益。

使用該設計作品的建議、需要的條件或限制

建議跟據學校自身的課程安排，設計一些與其他學科相關的題目給學生思考及設計，使學生更加投入。(如 解決化學、物理、會計等科目的問題)

使用設備：投影機、資訊科技設備、已經安裝 Pascal 編譯程序的 PC

教材教具使用指引



光碟目錄說明

目錄	說明
doc	存放本教學設計之所有文檔(包括教案、封面及其他補充資料)
powerpoint	存放本教學設計所用到的電腦播放性文件(如 powerpoint 文件 *.ppt)
photo	存放本教學設計的試教相片
image	存放本教學設計所用之圖片
video	存放本教學設計所用到的視頻播放文件(如 *.AVI , *.MPG , *.WMV)

根目錄文件說明

文件名	說明
autorun.inf	如各下沒有把光碟機的自動播放功能關閉，將順利進入 autorun.exe 的菜單(menu)
autorun.exe	菜單(menu)文件
autorun.ico	本光碟之圖標文件
default.htm	如各下把光碟機的自動播放功能關閉，請打開 default.htm 作為本光碟之入口點

教學程序 (教學目標)	教學活動		時間 分配	效果評量
	教師活動	學生活動		
(第一課時) 溫故知新	講解 《寫程序解決現實生活中的問題 I》三個問題的改進解法。(收銀機程序、百錢百鷄問題、一元二次方程求解。)	思考問題	10'	學生能把自己的編寫程序技巧提高
理解問題	新問題引入及討論 例題 1 蝸牛回家(問題引入) 例題 1 解(參考答案) 例題 1 討論及總結	思考練習	25'	例子生動有趣,學生能積極參與
提升動手編程能力	請同學在電腦上完成例題的測試	上機實習	5'	大部分達到要求
(第二課時) 問題的前提知識	例題 2 : Tetris(方塊遊戲)的設計基礎 矩陣旋轉的算法 方塊的表示方法	思考分析	15'	學生熟悉的遊戲,容易理解
理解問題	例題 2 解(參考答案) 例題 2 解(參考答案 2) 例題 2 討論及總結 總結	思考分析、分組討論	15'	學生能積極參與嘗試
提升分析及動手能力	例題 2 附加思考題:	上機實習	10'	學生被難倒,前後知識要求跨度過大
(第三課時) 擴闊思維方式	例題 3 圍圈遊戲(問題引入) 例題 3 討論及總結 總結	思考分析、分組討論	20'	基本理解
綜合練習	思考練習題	上機實習	20'	基本問題,基本過關

目錄

1.教學計劃內容說明.....	7
2.教學活動.....	8
前言.....	8
《寫程序解決現實生活中的問題 I 》	
三個問題的改進解法.....	8
收銀機程序.....	8
百錢百鷄問題.....	9
一元二次方程求解.....	10
新問題引入及討論.....	11
例題 1 蝸牛回家.....	11
例題 1 解(參考答案).....	11
例題 1 討論及總結.....	12
例題 2：Tetris(方塊遊戲)的設計基礎.....	12
矩陣旋轉的算法.....	12
方塊的表示方法。.....	12
例題 2 解(參考答案).....	14
例題 2 解(參考答案 2).....	14
例題 2 討論及總結.....	15
總結.....	5
例題 2 附加思考題：.....	15
思考題.....	15
例題 3 圍圈遊戲.....	16
例題 3 討論及總結.....	17

總結.....	17
思考練習題.....	17
小組活動.....	17
3.試教評估.....	17
4.反思與建議.....	18
5.參考資料.....	19

1.【教學計劃內容說明】

A. 參選編號..... C032

B. 學科名稱..... 資訊科

C. 單元名稱..... Pascal 原來可以這樣學

D. 教學對象..... 高中一年級/高中二年級

E. 學生人數..... 55

F. 教學目標

使學生能用電腦編寫程序解決日常生活中的問題，進一步學習**實際問題轉換成數字形式**的表示方法及**數模轉換等技巧**，從設計程序中，增強編寫程序的**動手實作能力**，加深了解和認識電腦的運作過程，理解電腦的長處和不足，從而對將來使用或設計電腦程序等方面更得心應手。

G. 教學重點/難點

本教學設計的重點在於訓練同學們編寫程序的**動手實作能力**，這是很大部分同學在學習完程序設計課後都缺乏的，值得我們重視。

而本設計難點亦在於此，很多學生都會因短期的多次失敗而放棄。要學生們**長時期的專注學生程序設計**是很難的，所以要多加一些活動，先培養他們程序設計的興趣。

H. 教學時數 3 教節（每教節 40 分鐘）

I. 創意與特色

本教學設計的設計特色在於應用**多元智慧**的理論於教學中，以**小組實習形式**，培養學生們的**人際智慧**，使學生更接近現實生活工作中的**集體編程情況**。有別於一般基礎編程教學課程。而在數模轉換及解 IQ 題的學習中，對學生的**數學-邏輯智慧及空間智慧訓練**有著重要的作用，這都是傳統以教授語法為主的編程教學中所缺乏的。

本課程有助激發學生的**創造能力**、通過調試和比較各組所編寫的程序，更正學生一些錯誤的編程概念，把當中較好的方法和思路向各組推廣，最後再把前人所設計的方法作介紹，集思廣益。脫離以往死板的板書講授程序撰寫，大大增強了學生學習電腦程序設計的興趣，提高教學效益。

J. 教材架構

問題提出→分析問題→參考解答→小組實習→實習總結→成果討論。

K. 教學準備

投影機、資訊科技設備、已經安裝 pascal 編譯程序的 PC

L. 教學評量

1.在設計的評量方式分為兩個部分，第一個部分是**創意**部分。原因是創意對現今軟件業以及其他行業的發展都十分重要，單靠使用前人的理論或成果，現在已經開始不足應付社會發展的需要了。在創意方面佔評量的 50% 比重。

第二個部分是程序的**完整性**。雖然創意的確是目前發展的主流，但最終的成品都是要靠完整性來衡量的，程序設計更是如此，一個不完整或經常出錯的軟件是不能被使用者所接受的。在完整性方面佔評量的 50% 比重。

2.【教學活動】

前言

本單元的設計是承接上一單元《寫程序解決現實生活中的問題 I》，以實現及講解上一單元所提示的改進解法為基礎。引入新的題目及 IQ 題，如方塊遊戲(Tetris)、蝸牛問題、迷宮(Maze)的實現技巧及算法分析(二維數組的使用及遞歸算法的思路)。誘發學生使用電腦編程解決問題，增加學生學習電腦編寫程序的興趣。

本教學設計的目標在於要同學們在完成本教程後，能用電腦編寫程序解決日常生活中的問題及進一步學習實際問題轉換成數字形式的表示方法及數模轉換等技巧，從設計程序中，加深了解和認識電腦的運作過程，理解電腦的長處和不足，從而對將來使用或設計電腦程序等方面更得心應手。

《寫程序解決現實生活中的問題 I》三個問題的改進解法

收銀機程序

百錢百雞問題

一元二次方程求解

說明：在程式部分中，黑色字為原有的解法，**紅色**字為改進解法及註釋有關上一單元的內容可以參閱《寫程序解決現實生活中的問題 I》



(上單元) 例一·改進結果

收銀機程序

1. Var n:integer;
2. cost,total:real;
3. Begin
4. Total:=0;
5. N:=1;

6. repeat
7. n:=n+1;
8. Write('你現正輸入第',n,'件貨品之價格:');
9. Readln(cost);
10. Total:=total+cost;
11. until cost<0 {價錢小於零是代表輸入結束}
12. Writeln('你總共購入了',n,'件貨品');
13. Writeln('總共',total:7:2,'元');
14. End.

(上單元) 例二 改進結果

百錢百雞問題

1. Var chicken,hen,cock:integer;
2. Begin
3. For chicken:=0 to 100 do {由於小雞不可能多於100隻}
4. For hen:=0 to 100 div 5 do {母雞數到100 div 5已把錢花光}
5. For cock:=0 to 100 div 3 do {與母雞數同一道理}
6. If chicken*0.5+hen*5+cock*3=100 then
7. begin
8. Writeln('chicken:',chicken);
9. Writeln('hen:',hen);
10. Writeln('cock:'cock);
11. Writeln('-----');
12. End;
13. End.



(上單元) 例三. 改進結果

一元二次方程求解

1. *Var a,b,c:real;* {改變了系數的定義域後就可以解法更多高數上的問題了}
2. *Delta:real;*
3. *Begin*
4. *Writeln('請輸入一元二次方程的三個系數:');*
5. *Write('a=');*
6. *Readln(a);*
7. *Write('b=');*
8. *Readln(b);*
9. *Write('c=');*
10. *Readln(c);*
11. *Delta:=sqr(b)-4*a*c;*
12. *If delta<0 then*
13. *Writeln('此方程無解:')*
14. *Else*

(上單元) 例二. 改進結果



為',-b/(2*a))

1 為',(-b+sqrt(delta))/(2*a));

rt(delta))/(2*a));

新問題引入及討論

經過上一單元的學習，同學們已經體驗了如何寫程序解決實際生活的問題，但這個只是我們初步的目標。

接下來本教學設計將會帶領同學們去到編寫程序的另一個新台階。我們來嘗試通過寫程序來處理一些 IQ 題目、及數理邏輯分析吧。

例題 1 蝸牛回家

一隻蝸牛以 1cm/s 的速度在一條 10cm 長的橡皮繩上由下往上爬，由於重力的關係，橡皮繩以 10cm/s 的速度均勻地延長，到底在這種情況，蝸牛能不能爬到橡皮繩的頂部然後回家？

例題 1 問題分析：

本題著重分析，程序其實可以非常簡單，乍看之下，蝸牛永遠不能爬到終點。但試想想，第一秒末繩子長為 10cm，這秒內蝸牛爬了繩子的 1/10，第二秒末繩長 20cm，蝸牛仍然爬了 1cm，即爬了繩子的 1/20，如此類推到了第 N 秒末，蝸牛爬了繩子的 $1/(N*10)$ 。

同學可能會問，繩子不斷延長，怎樣爬得完呢？但同學們可能忽略了一個重點就是繩子是均勻地延長，即第一秒爬過的 1/10，到了第 N 秒這 1/10 仍然是 1/10。

例題 1 問題分析：

例題1問題分析：

- 那麼蝸牛爬過的繩子長度就可以列成下式：
- TOTAL=1/10 第一秒
- TOTAL=1/10+1/20 第二秒
- TOTAL=1/10+1/20+1/30 第三秒
- TOTAL=1/10+1/20+.....+1/(N*10) 第N秒
- 只要TOTAL的總和大於或等於1，蝸牛就爬完繩子了。

那麼蝸牛爬過的繩子長度就可以列成下式：

- TOTAL=1/10 第一秒
- TOTAL=1/10+1/20 第二秒
- TOTAL=1/10+1/20+1/30 第三秒

- TOTAL=1/10+1/20+.....+1/(N*10) 第 N 秒
- 只要 TOTAL 的總和大於或等於 1，蝸牛就爬完繩子了。

例題 1 解(參考答案)：

1. Var s:longint;
2. total:real;
3. Begin

```
4.     total:=0;
5.     s:=0;
6.     repeat
7.         s:=s+1;
8.         total:=total+1/(s*10);
9.     until total>=1;
10.    writeln('蝸牛總共需要爬',total,'秒才能到達繩頂');
11. End.
```

例題 1 討論及總結：

總結：

本題的解法其實類似一些代數求極限的問題，只要懂得其思想，很多原來很複雜的問題都會變成很單的程序

討論問題：

你在日常生活中遇到

類似的問題嗎？試編

程序解法之。

例題 2：Tetris(方塊遊戲)的設計基礎

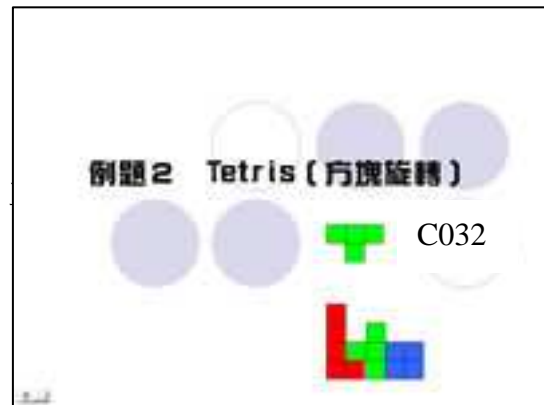
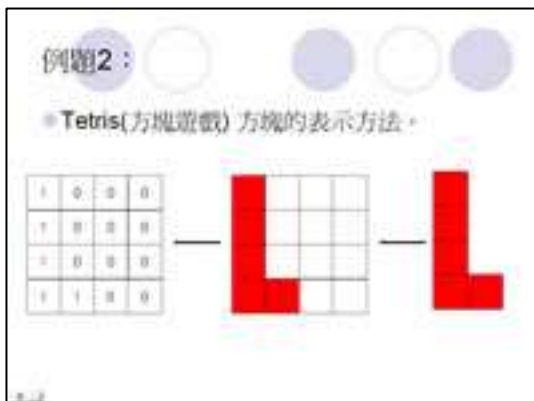
Tetris 又稱俄羅斯方塊，這類遊戲的設計，旋轉及運算)

下面就討論矩陣旋轉的算法

Tetris(方塊遊戲)的矩陣旋轉的算法

- 本旋轉方法主要是實現二維矩陣的旋轉
- 在作旋轉前首先，我們討論方塊的表示方法。

Tetris(方塊遊戲) 方塊的表示方法。



- 二維矩陣旋的實現。

●

- 以一個 3*3 的矩陣為例：
- 上圖為 2 次順時針方向轉 90 度的結果
- 但我們感興趣的只是數字的排列而非方向

例題2：

- 二維矩陣旋的實現。
- 以一個 3*3 的矩陣為例：

1	2	3
4	5	6
7	8	9

↻

7	4	1
8	5	2
9	6	3

↻

9	8	7
6	5	4
3	2	1

- 上圖為 2 次順時針方向轉 90 度的結果
- 但我們感興趣的只是數字的排列而非方向

- 二維矩陣旋的實現。
- 所以剛才的旋轉可以表示成：
- 接下來我們分析轉的規律
- 二維矩陣旋轉的規律的分析。
- 右表為旋轉前後 A、B、C 三點的座標對照表

例題2：

- 二維矩陣旋轉的規律的分析。

A	B	C
4	5	6
7	8	9

↻

7	4	1
8	5	2
9	6	3

↻

9	8	7
6	5	4
3	2	1

- 右表為旋轉前後 A、B、C 三點的座標對照表

座標	旋轉前		旋轉後	
	X	Y	X	Y
A	1	1	1	3
B	1	2	2	3
C	1	3	3	3

例題2：

- 二維矩陣旋的實現。
- 所以剛才的旋轉可以表示成：

1	2	3
4	5	6
7	8	9

↻

7	4	1
8	5	2
9	6	3

↻

9	8	7
6	5	4
3	2	1

- 接下來我們分析轉的規律

- 由右圖分析可知：
- 旋轉前後
 - X 座標等於 Y 座標
 - 即 X 旋轉後 = Y 旋轉前
 - Y 座標等於矩陣邊長一去 X 座標
 - Y 旋轉後 = N - X 旋轉前

例題2：

- 由右圖分析可知：
- 旋轉前後
 - X 座標等於 Y 座標
即 $X_{\text{旋轉後}} = Y_{\text{旋轉前}}$
 - Y 座標等於矩陣邊長一去 X 座標
即 $Y_{\text{旋轉後}} = N - X_{\text{旋轉前}}$

座標	旋轉前		旋轉後	
	X	Y	X	Y
A	1	1	1	3
B	1	2	2	3
C	1	3	3	3

例題 2 解(參考答案) :

```
1. var a,b:array[1..3,1..3] of byte;
2.     i,j:integer;
3. begin
4.     for i:=1 to 3 do
5.         for j:=1 to 3 do read(a[i,j]); {輸入部分}
6.     writeln;
7.     for i:=1 to 3 do
8.         for j:=1 to 3 do b[i,j]:=a[n-j+1,i];
9.     for i:=1 to 3 do {輸出部分}
10.        for j:=1 to 3 do
11.            begin
12.                write(b[i,j]:2);
13.                if j=3 then writeln;
14.            end;
15. end.
```

例題 2 解(參考答案 2) :

```
1. var a,b:array[1..10,1..10] of byte; {假設上限為 10}
2.     n,i,j,k:integer;
3. begin
4.     write('n=');
5.     readln(n); {輸入矩陣邊長}
6.     for i:=1 to n do {輸入矩陣內容}
7.         for j:=1 to n do read(a[i,j]);
8.     writeln;
9.     for k:=1 to 4 do {旋轉 4 次}
10.        begin
11.            for i:=1 to n do
12.                for j:=1 to n do b[i,j]:=a[n-j+1,i]; {內容轉換}
13.            for i:=1 to n do {輸出矩陣內容}
14.                for j:=1 to n do
15.                    begin
16.                        write(b[i,j]:2);
17.                        if j=n then writeln;
18.                    end;
19.            writeln;
20.            a:=b;
21.        end;
22. end.
```

例題 2 討論及總結：

總結：

本題的解法是解決二維空間旋轉問題的基本解法。很多的問題(如圖像處理)、電腦遊戲等都需要用到此算法，只是在使用上不會這樣簡單而是加上其他的一些內容作優化，學懂此內容為基礎，對於電玩設計、圖像處理等都十分有幫助。

討論問題：

你在日常生活中遇到
類似二維空間旋轉的
問題嗎？試編程序解法之。

例題 2 附加思考題：

思考題：

本題的解法是解決二維空間旋轉問題的基本解法。但實際上，現今流行的都是三維空間的遊戲，試編寫程序解決三維空間的旋轉問題。此類問題可以解決類似魔方(扭計色)、3D Tetris 等。

注意

要先做好座標分析，才編程
否則容易引起混亂。



例題 3 圍圈遊戲

13 個同學圍成一圈，順時針方向 1、2、3、4 報數，每逢數到 4 的同學被 out，餘下的同學繼報數，最後會誰會留下來呢？

例題 3 分析



- 根據情況，本題有兩個迴路同時在執行著，但每一個迴路本身的中間過程都會影響到另一個迴路的結果，所以不適合用 for 迴路這種固定迴路次數的指令。
- 建議使用 While do 或 repeat until 迴路



例題 3 解(參考答案)：

```

1.  var a:array[1..13] of boolean;
2.      out,i,n:integer;
3.  begin
4.      for i:=1 to 13 do          {初
      始化程序}
5.          a[i]:=true;
6.          out:=0;
7.      i:=0;
8.      n:=0;
9.      while out<12 do          {判斷是否最後一個人}
10.         begin
11.             n:=(n mod 13)+1;    {由1到13循環地數}
12.             if a[n]=true then
13.                 i:=i+1;
14.             if (i=4) then      {由1到4循環地數}
15.                 begin
16.                     a[n]:=false;
17.                     out:=out+1;    {統計被out人數}
18.                     i:=0;
19.                 end;
20.             end;
21.         for i:=1 to 13 do      {印出最後結果}
22.             write(a[i]:6);
23.         readln;
24.     end.

```

例題 3 討論及總結：

總結：

本題的要點是要把問題的數理邏輯要點化為程序編寫出來，很多 IQ 遊戲都是有類似的問題，就是你一定要明白過中的機制才可以正確的推出必勝的方案。在這個過程中，機制的推導當然是十分重要，但後期大量的計算就更加是不可以出錯的，而這些繁複的計算工作，最好就是編寫程序來解決了。

討論問題：

你在日常生活中遇到

類似的問題嗎？試編程序解法之。

思考練習題

1. N 個小朋友排一直線玩猜大王，數 1,2,3 報數，數到 3 落選，數到最尾則由尾數上頭，如此往返的報數，最後留下的是大王，最後誰是大王呢？
2. 某一月餅生產線上有兩台不同型號，一新一舊的制餅機，已知新機的產量是舊機的兩倍。在生產了 6 箱餅後發現原來新機器漏放了蛋黃這種原材料，現在只知道每箱所含的月餅數為 78,94,86,87,82,80，根據以上條件，你能找到那箱是有蛋黃，那箱是沒有的嗎？
3. 森林探險隊一行 4 人，要通過一條獨木橋。現在只有一支手電筒，已知獨木橋每次只可兩人同時通過，而且由於天黑必需拿著電筒才能保證安全。設每人過橋時所花的時間分別為 1、2、5 和 10 分鐘，求最少要多少分鐘才能使全部隊員過到對面。



小組活動

- 分組討論及設計程序
- 借助此形式，激發學生的創造能力、通過調試和比較各組所編寫的程序，更正學生一些錯誤的編程概念，把當中較好的方法和思路向各組推廣。
- 最後再把前人所設計的方法作介紹，集思廣益。大大增強了學生學習電腦程序設計的興趣。

3. 【試教評估】

由於程式設計的概念對學生來說比較抽象，要求每個學生都能獨立完成程序是不實際的，以分組形來設計程式，目的是希望一些好的同學能帶領新手的同學一起完成一個設計，這樣即使是編程能力比較差的同學也能參與全個設計過程，而他們又可以用自己的視點給其他組員一些點子，互補不足。本設計舉的例子及做實習的機會都很多，讓學生不會認為這科是一門理論課，而是切切實實的一門技能。

4. 【反思與建議】

完成了本教學設計後，學生一般能解決或發現更多有趣的問題，除了本設計所選用的例題和習題之外，建議及學生多一些與他們生活或其他學科有關的題目，使學生比較感興趣去探索。在最後的幾個思考題裏，同學們都提出了很多不錯的解法，這正好說明這些內容擴闊了學生的眼界，啟發了他們的編寫程序潛能，也推動了學生去思考。

5. 【參考資料】

本校電腦科組，Turbo Pascal 程序語言設計
王志強，中學資訊奧林匹克訓練習題
黃信斌，寫程序解決現實生活中的問題 I
譚浩強、田淑清，Pascal，程序設計 電子工業出版社
嚴蔚敏、吳偉民，數據結構，清華大學出版社