

## 357.計算 BMI 值(1 分)

問題描述：

BMI 公式為 體重 (KG) 除以身高 (m) 之平方，分級如下：

分級	BMI 範圍
Underweight	$\text{BMI} < 18.5$
Normal	$18.5 \leq \text{BMI} < 24$
Overweight	$24 \leq \text{BMI} < 27$
Obese Class I	$27 \leq \text{BMI} < 30$
Obese Class II	$30 \leq \text{BMI} < 35$
Obese Class III	$\text{BMI} \geq 35$

輸入說明：

輸入檔名為 Problem1\_in\_BMI.txt，第一行是身高 (cm) 及體重 (kg)，中間空一個空白字元。

輸出說明：

輸出檔名為 Problem1\_out\_BMI.txt，輸出其 BMI 值至小數點第二位，次行輸出體重之等級。

範例：

Sample Input	Sample Output
173.1 82.1	27.40  Obese Class I

## 358.亂數不重複(2 分)

問題描述：

取不重複的亂數在日常生活中的應用相當常見，請製作一個程式，能夠取  $X \sim Y$  之間  $N$  個不重複的亂數值，但複雜度不得為  $O(N!)$ 。

輸入說明：

**Input** 檔案內首行輸入最大值 ( $X$ )、最小值 ( $Y$ ) 及取  $N$  個亂數，每個數字中間以空格隔開。

輸出說明：

**Output** 檔案內每行輸出一個亂數，共  $N$  個，但必須要每個亂數都不重複。

範例：

Sample Input	Sample Output
100 20 10	49
	24
	37
	65
	68
	52
	72
	25
	43
	91

# 359.和與積相等(2 分)

問題描述：

請製作一個程式讓它找出小於  $N$  位數的第一個數 (10 、 100 、 1000...) 的所有數字中，每一個數相加等於相乘的不含 0 數字。

輸入說明：

Input 檔案輸入一個數字  $N$  ，代表檢查小於  $N$  位數的第一個數的所有數字。

輸出說明：

Output 檔案輸出在  $N-1$  位數下，每一位數相加等於相乘的數字。

範例：

Sample Input	Sample Output
4	1
	2
	3
	4
	5
	6
	7
	8
	9
	22
	123

	132
	213
	231
	312
	321

## 360.分區停水(1 分)

### 問題描述：

由於氣候變遷，久旱不雨，造成水庫儲水量嚴重不足，必須進行分區停水。因此水庫管理局要開發一套自動開關輸水管線的系統，公平地分區輪流停水。假設總共有  $n$  個區域，編號  $1, 2, \dots, n$ 。一開始，所有區域依照編號順序排成一個環狀序列，隨機選取一個數字  $m$ ，從第 1 區開始數，每數  $m$  區，則該區停水，已經停過水的區域則退出環狀序列。例如，當  $n = 13, m = 8$ ，則停水順序為 8, 3, 11, 6, 1, 9, 4, 12, 7, 2, 10, 5, 13。

假設第  $n$  區為特權區域，必須是最後停水的區域。於是現在要寫一個程式，當  $n$  給定時，這個程式要能決定出最大的  $m$ ， $1 \leq m \leq n-1$ ，使得最後停水的區域為第  $n$  區。

### 輸入說明：

程式的輸入包含兩行數字，第一行包含一個正整數  $k$ ， $1 \leq k \leq 10$ ，代表第二行有  $k$  個測試資料  $n_1, n_2, \dots, n_k$ ， $2 \leq n_i \leq 1000$ ，而此  $k$  個正整數間以空格隔開。

### 輸出說明：

輸出  $k$  列答案，針對每一個測試資料  $n_i$ ，輸出一個答案  $m_i$ 。

### 範例：

Sample Input:	Sample Output:
3	1
5 13 21	8
	4

# 361.賓果遊戲(1 分)

問題描述：

小雅很喜歡玩紙上的數字賓果遊戲，但是她覺得畫賓果遊戲的圖很麻煩又很費時，請利用程式設計一個自動畫出賓果遊戲的圖來解決小雅的困 擾。

所謂的賓果遊戲就是玩家們決定一個由  $n \times n$  格子所組成正方形。每個格子內都會有一組不同號碼的數字，這些數字的組成是由  $1 \sim n \times n$  共  $n \times n$  個。玩家彼此互喊數 字， 並在正方形上的該數字格子作記號。當有記號的格子連成一條直線時 ( $n$  個號碼 )，該玩家便獲得優勝了！

輸入說明：

輸入一個正整數  $n$  ( $0 < n < 15$ ) 。

輸出說明：

印出  $n \times n$  格子所組成的正方形並且每個格子內都會有一組不同號碼的數字，這些數字的組成是由  $1 \sim n \times n$  共  $n \times n$  個，並為美觀，請將每個數字貼其格子中右 邊。( 即使輸入相同的  $n$ , 每次數字編排順序都需不同 )。

Sample1 Input:	Sample1 Output:
4	-----    8  2 10  3   -----    7 12  4  5   -----   14 13 11  1   -----    9 16  6 15   -----
Sample2 Input:	Sample2 Output:

4	<div>-----</div> <div> 12  2  5 11 </div> <div>-----</div> <div> 13  8  9  6 </div> <div>-----</div> <div>  7  1  4 15 </div> <div>-----</div> <div> 14  3 10 16 </div> <div>-----</div>
Sample3 Input:	Sample3 Output:
10	<div>-----</div> <div>  9  42  41  93  3  96  2  91  64  61 </div> <div>-----</div> <div>  24  79  85  44  59  71  4  12  36  63 </div> <div>-----</div> <div>  78  21  39  80  8  84  25  15  19  54 </div> <div>-----</div> <div>  67  34  7  56  99  86  76  5  45  58 </div> <div>-----</div> <div>  33  90  52  13  11  31  68  95  65  89 </div> <div>-----</div> <div>  60  17  75  69  46  6  53  35  14 100 </div>

	43  70  28  66  40  16  47  83  88  62
	49  81  51  97  73  29  20  27  23  26
	87  38  55  37  57  82  92  10  98  32
	94  18  22  50  72  1  77  74  30  48



## 362.英文句子字之倒轉(1 分)

題目說明：

請利用遞迴的方式撰寫一個程式，從檔案 `in.txt` 讀進一段沒有標點符號的英文文字，倒轉這段文字中字的順序，並輸出到一個文字檔 `out.txt`。例如 `in.txt` 文字檔案內容為 `This is a book and that is an apple`，則輸出檔 `out.txt` 為 `apple an is that and book a is This`。

輸入說明：

輸入一個文字檔案 `in.txt`，內容為一段沒有標點符號的英文字。

	資料	意義
第一筆	英文字母組成的英文句子	要被倒轉字的順序之英文句子

輸出說明：

將讀進的英文文字，倒轉其順序並輸出到另一個文字檔 `out.txt`。

輸入範例：

This is a book and that is an apple
-------------------------------------

輸出範例：

apple an is that and book a is This

### 363.列印包含特定元素的子集合(2 分)

題目說明：

請利用一維整數陣列儲存一個集合  $S$ ， $S$  含有  $N$  個元素，元素分別為  $1 \sim N$  的整數。請撰寫一個程式，產生集合  $S$  的所有子集合  $P$ ，假設  $P$  不含空集合  $f$  與  $S$  本身，請將  $P$  存入一個二維整數陣列中，並將  $P$  中含有元素  $1$  與  $N$  的子集合列印出來。例如：假設集合  $S=\{1, 2, 3\}$ ， $S$  的子集合  $P=\{\{1\}, \{2\}, \{3\}, \{1,2\}, \{1,3\}, \{2,3\}\}$ ，則包含元素  $1$  與  $3$  的子集合為  $\{1\}, \{3\}, \{1,2\}, \{1,3\}, \{2,3\}$ 。

輸入說明：

輸入一個整數  $N$ ，則此集合的元素為  $\{1, 2, \dots, N\}$ 。

	資料	意義
第一筆	大於 1 的整數	集合元素的個數

輸出說明：

產生集合的元素  $\{1, 2, \dots, N\}$  的子集合，存進二維陣列，並列印出。

輸入範例：

3
---

輸出範例：

1,

3,

1, 2,

1, 3,

2, 3,

### 364.最佳排程(1 分)

問題描述：

尾牙到了，因經濟不景氣，許多表演團體紛紛表示有意願前來表演餘興節目，員工希望在節目秀場中看到最多的表演，但受限於尾牙派對時間長度，公司必須決定邀請哪些團體來表演。請為公司寫一個程式讀入公司尾牙派對的時間和所有有意願前來表演的團體所需的表演時間，計算出公司最多可邀請多少團體來表演。例如，若公司派對長度為 200 分鐘，各表演團體所需的時間為

27 6 50 21 3 14 16 8 42 33 21 9

則最多可排入 12 個團體表演（ $27+6+21+3+14+16+8+42+33+21+9=200$ ）。

輸入說明：

第一行為一個正整數  $N$ ，代表共有幾家公司。之後接下來有  $N$  行，每行第一個為正整數  $M$ （），代表該公司的派對時間限制，接下來有數個（不超過 100 個）整數，以 0 結束，代表前來申請的各個表演所需的表演時間。

輸出說明：

將每個公司最多可邀請的表演團體數目輸出於一行。

範例：

Sample Input	Sample Output
3	11
200 27 6 50 21 3 14 16 8 42 33 21 9 0	7
70 20 5 3 6 12 8 4 0	4
30 13 12 10 5 3 6 15 0	

## 365.天際線查詢 (Skyline query) (1 分)

找尋天際線資料是近年來資料處理、決策分析的一個熱門的應用。請仔細閱讀底下的說明，並依據說明，將資料集中的天際線資料找出來。

假設一位使用者要購買手機，且這位使用者以 (1) 重量、(2) 價格以及 (3) 內附相機的像素數來決定購買手機的準則。在下表中列出了市場中一部分手機的資料：

手機編號	重量	價格	像素 ( 百萬像素 )
1	50	10000	300
2	55	9000	300
3	58	12000	250
4	50	10000	280
5	50	10000	300

正常而言，使用者會希望手機重量越輕越好、價格越低越好且像素越高越好。但市場上通常不可能出現這種「完美」的手機，因此使用者勢必得在一群「不完美」的手機中，挑選自己「偏愛」的手機。而「天際線查詢」的目的，便可以在一大群手機資料中，「自動地」幫使用者過濾掉他 / 她絕不可能選擇的手機（換句話說，沒被過濾掉的手機，都「可能」是使用者所偏好的），方便使用者做後續的決策。

以上表為例，何謂使用者絕不可能選擇的手機呢？很明顯的，手機 3 是使用者絕不可能選擇的手機。為什麼？因為它比手機 1 重，比手機 1 貴，比手機 1 的像素數少。簡而言之，它樣樣不如手機 1，所以這種手機就可以過濾掉。用術語來說的話，我們會說，手機 1「擊敗」(dominate) 了手機 3。

**定義 1：**若我們說一筆資料 A 它「擊敗」(dominate) 了另一筆資料 B，則 A 必需符合底下兩個條件：

1. A 的所有屬性值都不能比 B 的所有屬性值「差」，且

2. A 擁有一個以上的屬性「優於」資料 B。

套用上面的定義 1，我們知道手機 1 擊敗了手機 3。此外，我們也可以知道，手機 1 也擊敗了手機 4。為什麼？因為手機 1 無論在「重量」、「價格」以及「像素數」這 3 個屬性，都不比手機 4 來得差（符合條件 1）；但手機 1 的「像素數」這個屬性「優於」手機 4（符合條件 2），所以手機 4 被手機 1 所擊敗。

那麼手機 1 有沒有擊敗手機 5？答案是沒有。因為手機 1 對手機 5 而言，只符合了條件 1，也就是說，它的每個屬性都不比手機 5 的每個屬性「差」。但手機 1 沒有任何一個屬性「優於」手機 5，所以手機 1「無法」擊敗手機 5（同理，手機 5 亦無法擊敗手機 1）。

那麼有其他的手機可以擊敗手機 1 嗎？答案是沒有，因為手機 2 的重量比手機 1 重，所以致少在重量這個屬性上，手機 2 比手機 1「差」。而手機 1 可以擊敗手機 2 嗎？也沒法子，因為手機 1 在「價格」上敗給了手機 2。

於是，在上表中，手機 1，手機 2 以及手機 5 是無法被別人所擊敗的。我們稱這些手機為「天際線」資料。用通俗的話來說，就是這些手機都或多或少擁有了一些讓人無法忽略的優點。於是我們就要把這 3 個手機的資料呈現給使用者，讓使用者做最後的決策。

**定義 2：**資料集中的一筆資料 A 若被稱為天際線資料 (skyline object)，則若且唯若 A 無法被資料集中的其它任何資料給擊敗。

天際線查詢的目的，就是在一群資料集中，將這些天際線資料給過濾出來，方便使用者做最後的決策。

**輸入說明：**

我們會給你的程式底下的輸入：

3

0.23,4.00,0.69

2.75,4.58,3.69

3.58,0.26,4.33

第一行代表所要輸入資料的筆數。在本例中，共有 3 筆資料。

之後的每一行，都代表一筆資料。每個屬性都是浮點數。每筆資料的每個屬性，用逗點分隔（也就是說，上述的每一筆資料都有 3 個屬性）。

你的程式要將其中的天際線資料給過濾出來，並滿足底下幾個條件：

1. 假設數值較「大」的屬性值為較「優」。所以上述的第 2 筆資料的屬性 1「優於」第 1 筆資料的屬性 1（因為  $2.75 \geq 0.23$ ）。

2. 要將天際線資料過濾出來，並輸出。以上述的資料為例，程式要輸出 2.75,4.58,3.69

3.58,0.26,4.33

這兩筆資料。

3. 每輸出一筆資料就斷行一次。

範例：

Sample Input:	Sample Output:
3	
0.23,4.00,0.69	2.75,4.58,3.69
2.75,4.58,3.69	3.58,0.26,4.33
3.58,0.26,4.33	

## 366.立柱包牌法(1 分)

### 題目描述:

包牌一直是許多彩卷迷用來提高中獎機率的手段，而立柱包牌法則是其中一種包牌的手法。立柱包牌法的順序如下：

I 選擇可能會出現的號碼。彩迷要由 1~49 個號碼（以大樂透為例）中選出若干個在下期很有機會會出現的號碼。舉例而言，令這位彩迷選了 2，18，19，22，26，27，28，31，33，37，42，48，49。

II 將號碼分成六類。接著彩迷會設立 6 個柱子，並把選出的號碼分別歸類到不同的柱子上。每個柱子至少會有一個數字。舉例而言，下面是這位使用者的歸類。

柱 1	柱 2	柱 3	柱 4	柱 5	柱 6
2	18	22	26	33	42
48	19	27	28	37	49
			31		

III 最後，使用組合的方式，把所有可能的號碼組合都列出來，而這些組合就是彩迷所要投注的號碼。挑選的方式很簡單，上述共有 6 個柱子，每個柱子上一次可以取出一個號碼。因為共有 6 個柱子，所以每一次都可以挑出 6 個號碼來。以上述的例子而言，所有的排列組合數共有  $2*2*2*3*2*2=96$  組號碼。

立柱包牌有一個特色，那就是屬於同一柱子的號碼一定不會出現在同一組的號碼中。這個特色可以大量地減少組合的個數。舉例而言，這位彩迷共選出 13 個號碼，在 13 個號碼中取 6 個號碼，總共的組合數為  $C(13,6) = 1716$ ，若要將這些號碼都買下來，金額為  $1716*50=85800$ ，是一個很龐大的開銷。而若使用立柱包牌，總共的組合數只有 96 組，這是因為立柱包牌法把所有“不會一起出現號碼”的組合都給刪掉了（舉例而言，這位彩迷認為 18 號與 19 號不可能一起出現，於是將它們歸於同一柱中，所以同時出現 18 及 19 的組合就會被忽略掉）。就算把所有的組合都買下來，也只要花  $96*50=4800$  元。

請寫一個程式，接受使用者的設定，並輸出所有的立柱組合。

### 輸入說明：

我們的輸入包含 6 行。每一行代表一個柱子的內容。一個柱子中會包含 1 個以上的數字，數字間彼此用逗點隔開。

請將所有的組合算出，並輸出。每一行為一個組合。

範例：

Sample Input:	Sample Output:
2,48 18,19 22,27 26,28,31 33,37 42,49	2,18,22,26,33,42
	2,18,22,26,33,49
	2,18,22,26,37,42
	2,18,22,26,37,49
	2,18,22,28,33,42
	2,18,22,28,33,49
	2,18,22,28,37,42
	2,18,22,28,37,49
	2,18,22,31,33,42
	2,18,22,31,33,49
	2,18,22,31,37,42
	2,18,22,31,37,49
	2,18,27,26,33,42
	2,18,27,26,33,49
	2,18,27,26,37,42
	2,18,27,26,37,49
	2,18,27,28,33,42



	2,18,27,28,33,49
	2,18,27,28,37,42
	2,18,27,28,37,49
	2,18,27,31,33,42
	2,18,27,31,33,49
	2,18,27,31,37,42
	2,18,27,31,37,49
	2,19,22,26,33,42
	2,19,22,26,33,49
	2,19,22,26,37,42
	2,19,22,26,37,49
	2,19,22,28,33,42
	2,19,22,28,33,49
	2,19,22,28,37,42
	2,19,22,28,37,49
	2,19,22,31,33,42
	2,19,22,31,33,49
	2,19,22,31,37,42
	2,19,22,31,37,49
	2,19,27,26,33,42
	2,19,27,26,33,49

	2,19,27,26,37,42
	2,19,27,26,37,49
	2,19,27,28,33,42
	2,19,27,28,33,49
	2,19,27,28,37,42
	2,19,27,28,37,49
	2,19,27,31,33,42
	2,19,27,31,33,49
	2,19,27,31,37,42
	2,19,27,31,37,49
	48,18,22,26,33,42
	48,18,22,26,33,49
	48,18,22,26,37,42
	48,18,22,26,37,49
	48,18,22,28,33,42
	48,18,22,28,33,49
	48,18,22,28,37,42
	48,18,22,28,37,49
	48,18,22,31,33,42
	48,18,22,31,33,49
	48,18,22,31,37,42

	48,18,22,31,37,49
	48,18,27,26,33,42
	48,18,27,26,33,49
	48,18,27,26,37,42
	48,18,27,26,37,49
	48,18,27,28,33,42
	48,18,27,28,33,49
	48,18,27,28,37,42
	48,18,27,28,37,49
	48,18,27,31,33,42
	48,18,27,31,33,49
	48,18,27,31,37,42
	48,18,27,31,37,49
	48,19,22,26,33,42
	48,19,22,26,33,49
	48,19,22,26,37,42
	48,19,22,26,37,49
	48,19,22,28,33,42
	48,19,22,28,33,49
	48,19,22,28,37,42
	48,19,22,28,37,49

	48,19,22,31,33,42
	48,19,22,31,33,49
	48,19,22,31,37,42
	48,19,22,31,37,49
	48,19,27,26,33,42
	48,19,27,26,33,49
	48,19,27,26,37,42
	48,19,27,26,37,49
	48,19,27,28,33,42
	48,19,27,28,33,49
	48,19,27,28,37,42
	48,19,27,28,37,49
	48,19,27,31,33,42
	48,19,27,31,33,49
	48,19,27,31,37,42
	48,19,27,31,37,49

## 367.Bernoulli 裝錯信封問題(1 分)

### 問題敘述:

有個人寫了  $a$  封信要給  $a$  個人 ( 令  $1 \leq a \leq 9$  ) 。每一封信對應到一個信封。但這位老兄居然把每一封信都裝錯了信封。請問全部裝錯的種類共有幾種？

假設  $a=5$  , 令數字  $i(1 \leq i \leq a)$  為要給第  $i$  個人的信。所以 1 代表要給第 1 個人的信, 2 代表要給第 2 個人的信...依此類推。則信和信封的對應有三種情況：

#### 1. 每封信都裝在 “ 正確 ” 的信封

我們用序列 12345 代表第 1 封信裝在第 1 個信封, 第 2 封信裝在第 2 個信封, ...依此類推。

#### 2. 部分信裝錯, 部分信裝對

底下的序列就是部分信裝進了對的信封, 但部分信裝錯了信封。

21345

代表第 1 及第 2 封信裝在錯誤的信封中, 而第 3 、 4 、 5 封信則裝在正確的信封中。

#### 3. 所有的信都裝在 “ 錯誤 ” 的信封中

54321

代表了第 5 封信裝在第 1 個信封, 第 4 封信裝在第 2 個信封...依此類推。也就是所有的信, 都裝在錯誤的信封中。再舉一例 54213 也是一個所有信都裝在錯誤信封中的序列。

在本題中, 你要將所有屬於第 3 種狀況 ( 即所有信都裝在錯誤信封中 ) 的序列都找出來, 並計算出序列的個數。

### 輸入說明:

我們的輸入只有一行, 即變數  $a$  。在本題中, 我們假設  $1 \leq a \leq 9$  。

你 / 妳要將所有的可能序列都輸出。在第一行中, 先將可能序列的個數給輸出。舉例而言, 若  $a=5$  , 那總共有 44 種可能性, 可以將所有的信都裝在錯誤的信封中。接著, 你 / 妳要把所有可能的序列都輸出。在本例中, 你 / 妳需要輸出 44 個可能的序列 ( 見下表 ) 。

範例：

**Sample Input: Sample Output:**

44

21453

21534

23154

23451

23514

24153

24513

24531

25134

5

25413

25431

31254

31452

31524

34152

34251

34512

34521

35124

35214

35412

35421

41253

41523

41532

43152

43251

43512

43521

45123

45132

45213

45231

51234

51423

51432

53124

53214

53412

53421

54123

54132

54213

54231



## 368.連寫數整除問題(2 分)

二維空間中有 16 個點（如圖 1(a) 所示），請問你可以用一條線串起所有的點，且讓線段不交叉嗎？這樣的線，我們稱為 **space filling curve**，圖 1(b) 及圖 1(c) 分別展示了兩種不一樣的 **space filling curve**。

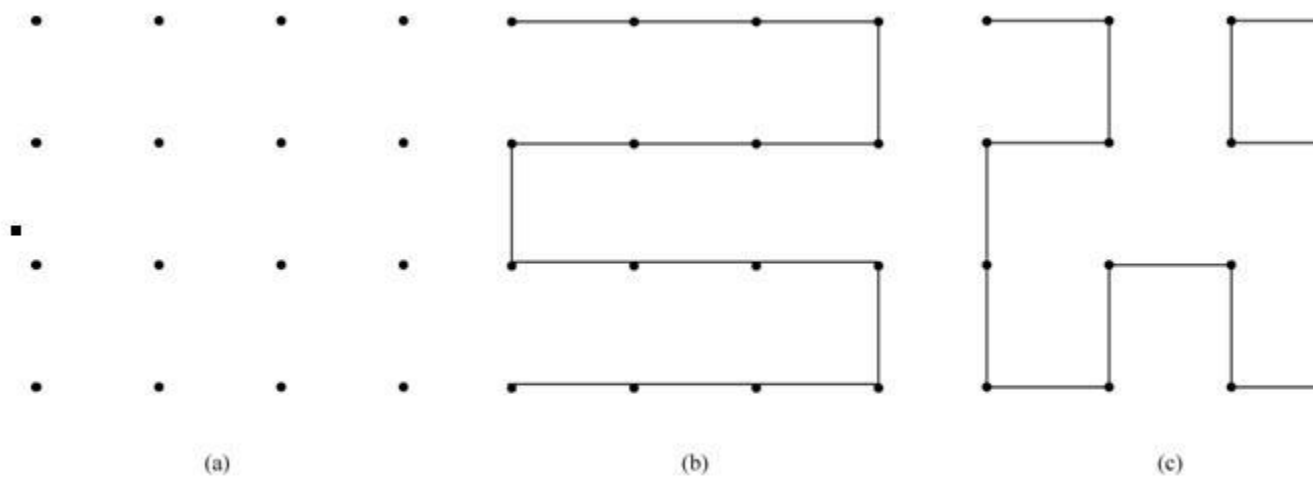


圖 1

圖 1(c) 的 **space filling curve** 特別有名，稱為 **Hilbert Curve**，它是數學家 David Hilbert 在 1981 年時提出來的。這種曲線常被用在資料庫領域以及影像處理領域中，它有一個很有趣的特性，那就是 **locality-preserving** 特性。這個特性是說，高維空間中若兩點相鄰越近，在 **Hilbert Curve** 上它們有很高的機率也會十分接近對方。下圖 2 中進一步展示了 2 維空間中 4 點以及 64 點的 **Hilbert Curve**。

圖 2(a) 的 **Hilbert Curve** 稱為 **first order Hilbert Curve**，圖 1(c) 稱為 **second order Hilbert Curve**，而圖 2(c) 則稱為 **third order Hilbert Curve**。讀者應該可以猜出來， $i$ -th order **Hilbert Curve** 可以將空間中  $2^{2i}$  個點連起來（舉例而言， $2^{\text{nd}}$  order **Hilbert Curve**（見圖 1(c)）可以將空間中  $2^{2 \times 2} = 16$  個點串連起來）。

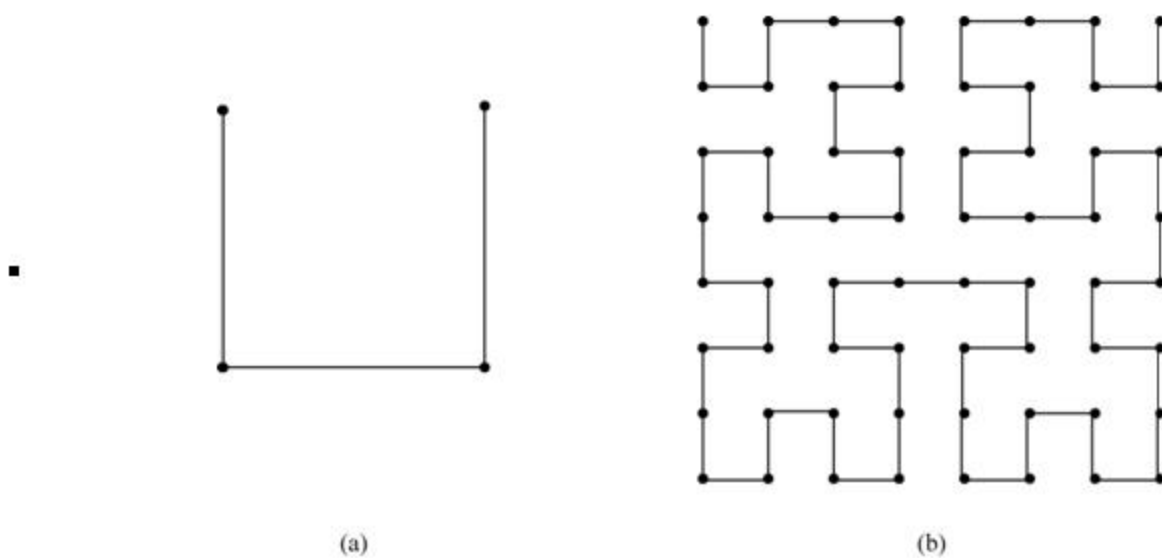


圖 2

Hilbert 由 cup 以及 joint 所組成。Cup 如圖 3(a) 所示。依照 cup 的開口方向，可有四種不同的變化（如圖 3(a)~3(d) 所示）。Joint 則是連接 cup 和 cup 之間的線。

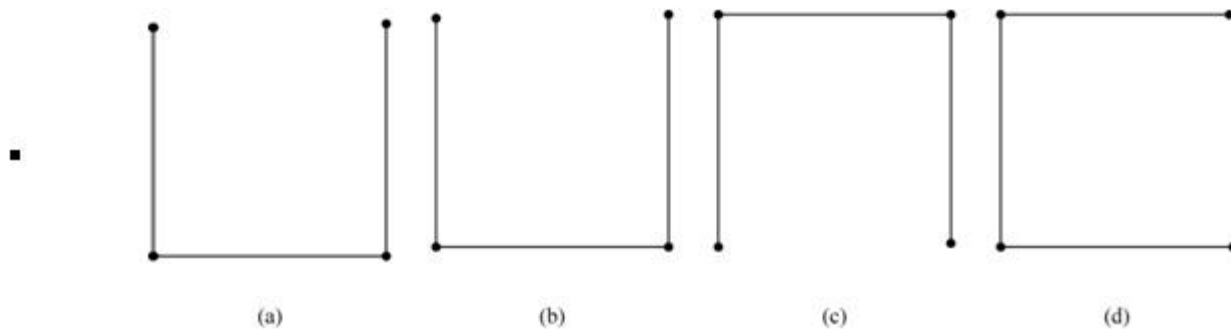


圖 3

Cup 的變化有一定的規則，如圖 4 所示。

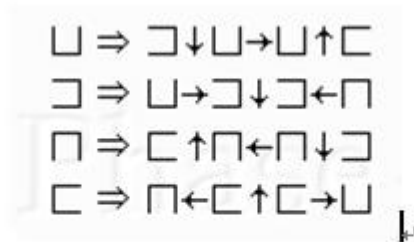


圖 4 摘自 <http://www.compuphase.com/hilbert.htm>

圖 4 再配合底下的圖 5，你會發現其中存在一個遞迴的關係。而利用這個遞迴的關係，我們就可以畫出 Hilbert Curve 來。

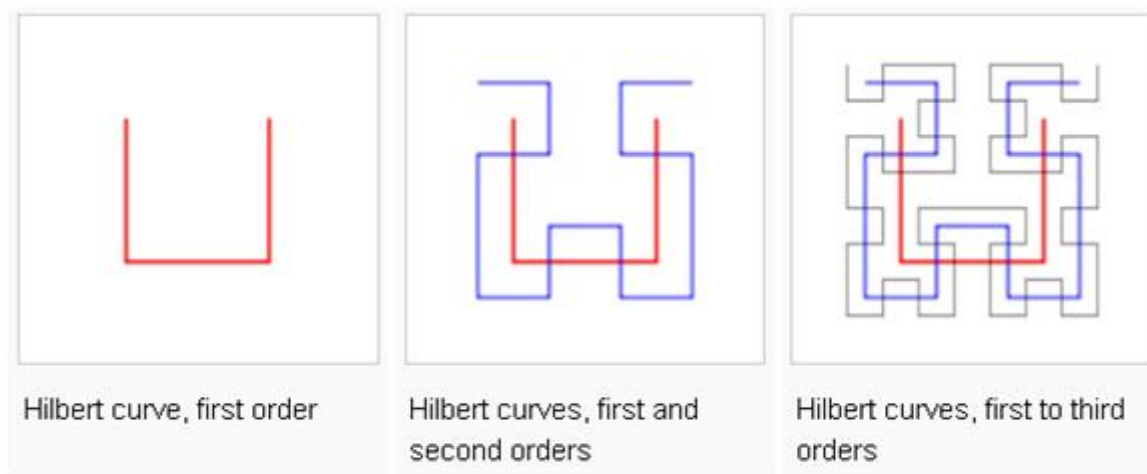


圖 5 摘自 [http://en.wikipedia.org/wiki/Hilbert\\_curve](http://en.wikipedia.org/wiki/Hilbert_curve)

我們用一個例子來解釋要如何畫出 Hilbert Curve。假設我們要畫出 3<sup>rd</sup> order 的 Hilbert Curve。這意味著空間中有 64 個點（如圖 6(a) 所示）。我們以點 (0,0) 做為起點。

接著由 first order Hilbert Curve 開始畫起，假設 first order Hilbert Curve 開口朝上（即圖 3(a)）。那麼 first order Hilbert Curve 就如圖 6(b) 所示。請注意，空間中我們畫了 4 個方框，這些是輔助線，讓讀者們可以更明白我們的解說。當有了 first order Hilbert Curve 後，我們便可以使用圖 4 的規則了。圖 4 的規則跟我們說，開口向上的 cup 可以演變成另外 4 個 cup，其順序為開口向左、向上、向上、以及向右。而這 4 個 cups 剛好組成了 second order Hilbert Curve，見圖 6(c)。注意，圖 4 中的箭號代表了 joint 的走向。舉例而言，圖 4 第一行中，左開口的 cup 後接一個向下的箭號。這代表這個 joint 要向下走，其後接著一個向上開口的 cup。接著 joint 要向右走，然後跟著一個向上開口的 cup，依此類推。

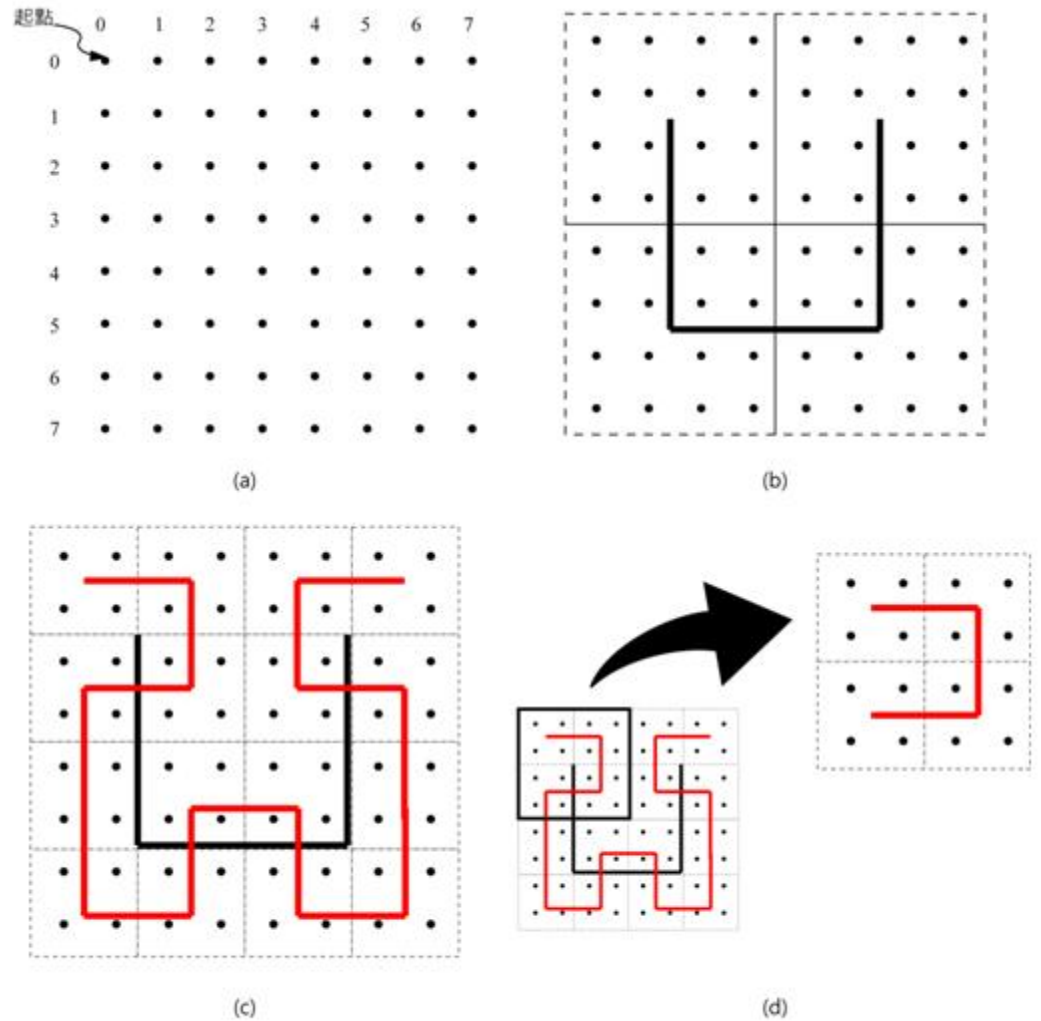


圖 6

接著我們到達了目的地，也就是 **third order Hilbert Curve**。由於 **second order Hilbert Curve** 是由 4 個 **cups** 所組成，而每個 **cup** 又可以依照圖 4 的規則繼續拆成另外 4 個子 **cups**，將這些 **cups** 都組合起來，就成為一個 **third order Hilbert Curve**。我們以圖 6(d) 為例。圖 6(d) 中，我們將圖 6(c) 的左上角放大。可以發現它是一個朝左開口的 **cup**。依照圖 4 的規則，它可以拆成朝上、朝左、朝左以及朝下四個子 **cups**。分別將這四個子 **cups** 畫上，你就完成了一部分的 **Hilbert Curve**。另外的三個部分也可以依照前述的想法畫出。

#### 輸入說明：

我們的輸入只有一行，它代表畫出的 **Hilbert Curve order** 數。若這個數字是 2，代表程式要畫出 2<sup>nd</sup> order 的 **Hilbert Curve**。

你 / 妳要將每個座標的順序輸出（參考下表）。請注意，我們令 (0,0) 為起點，所以它的順序為 1。在下表中，要輸出 2nd order **Hilbert Curve**，(0,0) 為起點 1，下一個點為 (0,1)，所以它的順序

為 1，再下一個點為 (1,1)，所以它的順序為 2，依此類推。輸出時，請依照“座標，順序”的格式輸出 (e.g., (0,0),1)，逗點間請不要填入空白。

範例：

**Sample Input: Sample Output:**

	(0,0),1
	(0,1),2
	(1,1),3
	(1,0),4
	(2,0),5
	(3,0),6
	(3,1),7
2	(2,1),8
	(2,2),9
	(3,2),10
	(3,3),11
	(2,3),12
	(1,3),13
	(1,2),14
	(0,2),15
	(0,3),16

### 369.斜率(1 分)

問題描述：

以知兩點坐標，用  $M = \frac{\Delta Y}{\Delta X}$  公式算出兩點斜率為何

輸入說明：

每列輸入一組座標 (x,y) ，共需輸入兩組 (x,y) 與 (x1,y1) 。  $0 < x,y,x1,y1$  ，且為整數。

輸出說明：

列印出斜率數值 m 。

範例：

輸入範例	輸出範例
1 1 2 2	-3
2 2 3 3	-3
1 2 2 1	-1

### 370.九九乘法表(1 分)

問題描述：

輸入兩數M、N整數，去設定九九乘法表的底限，列印出M × N為最大值之乘法表。

輸入說明：

輸入M、N兩整數，分別以一空格為間隔。

輸出說明：

列印出M、N兩整數為最大值之乘法表。

範例：

輸入範例	輸出範例
1 1	1 x 1 = 1
1 2	1 x 1 = 1 1 x 2 = 2
3 1	1 x 1 = 1 2 x 1 = 2 3 x 1 = 3
2 2	1 x 1 = 1 1 x 2 = 2 2 x 1 = 2 2 x 2 = 4

## 371.簡易記帳簿 (2 分)

### 問題描述：

寫一個簡易記帳簿程式，由鍵盤輸入目前剩下的現金，從檔案讀入收支項目，計算每筆收支記錄後的餘額顯示於螢幕上，並將最終餘額寫入檔案中。

### 輸入說明：

1. 輸入檔案 (q4.in) 中的項目名稱為 ASCII 字串，且不包含空白，以 `income` 表示收入，`outgoing` 表示支出，金額為正整數。

2. 由鍵盤輸入現金數目，從檔案 (q4.in) 輸入收支紀錄，資料格式如下：

[item] [income/outgoing] [money] ，中間以空白隔開。

q4.in：

Book outgoing 1000

Water outgoing 20

Salary income 2000

### 輸出說明：

於螢幕輸出每筆交易後的剩餘金額，並將最後剩餘金額寫入檔案 (q4.out) 中。

### 範例：

Sample Input	Sample Output
2000	Item In/Out Money Remain
file input(q4.in):	Book outgoing 1000 1000
Book outgoing 1000	Water outgoing 20 980
Water outgoing 20	Salary income 2000 2980



Salary income 2000	file output(q4.out):  2980
1000  file input(q4.in):  Book outgoing 1000  Water outgoing 20  Salary income 2000	Item In/Out Money Remain  Book outgoing 1000 0  Water outgoing 20 -20  Salary income 2000 1980  file output(q4.out):  1980
500  file input(q4.in):  Book outgoing 1000  Water outgoing 20  Salary income 2000	Item In/Out Money Remain  Book outgoing 1000 -500  Water outgoing 20 -520  Salary income 2000 1480  file output(q4.out):  1480

## 372.哪台筆電最適合我—skyline problem(2 分)

問題描述：

Alice 想購買筆電，於是對各家品牌的筆電進行資料蒐集，在考慮 CPU、硬碟空間、顯示卡等級情況下，Alice 優先考慮的筆電為不差於其他機型的筆電，即為在這些筆電中的 skyline points，請幫 Alice 找出這些筆電。

Skyline 問題定義：

給定一個 D- 維空間和點集合 S，從 S 中選出一個子集，該子集中的點不能被 S 中任意一個點給支配 (dominate)，則該子集內的點稱為 Skyline point。

支配 (Dominate) 的定義

在 D- 維空間，存在兩個點 P 和 Q 屬於 S，如果 P 在所有 D 維度上的值不比 Q 差，且至少在一維度上優於 Q，則稱 P dominate Q。（考慮數值越大越好的情況）

範例：

考慮 3 維空間，點集合  $S = \{A, B, C\}$

A 5 4 3

B 4 3 3

C 4 5 4

-> 在此例中 A 和 C 可以支配 B  $(5 > 4, 4 > 3, 3 = 3)$   $(4 = 4, 5 > 3, 4 > 3)$ ，而 A 和 C 卻無法支配彼此。因此在此例 A，C 為 skyline points.

輸入說明：

資料的格式為：

A 5 4 3

A ，代表筆電型號， 5 代表 CPU 速度， 4 代表硬碟空間， 3 代表顯示卡等級

( 數字越大代表性能越好 )

輸出說明：

輸出所有的 Skyline points 。

範例：

Sample Input	Sample Output
A 7 8 9 B 5 8 5 C 1 2 8 D 6 8 9 E 9 5 4 F 4 9 5 G 3 6 6	A 7 8 9 E 9 5 4 F 4 9 5
A 6 7 8 B 5 8 4 C 1 7 5 D 6 8 9 E 9 5 9 F 4 9 4 G 3 6 7 H 4 8 9	D 6 8 9 E 9 5 9 F 4 9 4
A 7 8 9 B 5 8 5	A 7 8 9 E 9 5 4

C 1 2 8	F 4 9 5
D 6 8 9	H 8 8 8
E 9 5 4	I 7 8 9
F 4 9 5	
G 3 6 6	
H 8 8 8	
I 7 8 9	
J 4 5 6	

### 373.哪隻手機適合我—k-Skyline problem(3 分)

問題描述：

Elric 想購買手機，於是對各家品牌的手機進行資料蒐集，在考慮待機時間、電池容量、額外應用程式個數、通話時間後，想將其中任兩個屬性( $k=2$ )不比它牌手機差的手機列入候選清單( $k$ -skyline problem)，請幫 Elric 找出這些候選手機。

k-Skyline 問題定義：

給定一個  $D$ -維空間，點集合  $S$  和變數  $k$ ，從  $S$  中選出一個子集，該子集中的點不能被  $S$  中任何的點  $k$ -支配 ( $k$ -dominate)，該子集內的點稱為  $k$ -dominate skyline point。

$k$ -支配 ( $k$ -dominate) 定義

在  $D$ -維空間，存在兩個點  $P$  和  $Q$  屬於  $S$ ，如果  $P$  存在  $k$  個維度上的值不比  $Q$  差，且至少在一維度上優於  $Q$ ，則稱  $P$   $k$ -dominate  $Q$ 。（考慮數值越大越好的情況）

Note 🤖 反之若  $Q$  存在  $k$  個維度上的值不比  $P$  差，且至少在一維度上優於  $P$ ，則  $Q$   $k$ -dominate  $P$ ，非遞移性，且有 Cyclic 可能發生）

範例：

考慮 4 維空間，點集合  $S=\{A,B,C\},k=2$

A 5 4 5 5

B 4 3 3 4

C 4 5 4 3

-> A 2-dominate B 在維度 和 C 在維度，而 B,C 無法 2-dominate A，因此在此例，A 為 2-skyline points。

輸入說明：

輸入資料的格式為：

A 5 4 3 5

A，代表手機型號，5 代表待機時間，4 代表電池容量，3 代表額外應用程式個數、5 代表通話時間。

(其中數字越大代表功能越好)

輸出說明：

輸出符合條件的手機。

範例：

Sample Input	Sample Output
A 7 8 9 9 B 5 8 5 5 C 1 2 8 4 D 6 8 9 3 E 9 5 4 4 F 4 9 5 2 G 3 6 6 4	A 7 8 9 9
A 6 7 8 8 B 9 8 4 9 C 1 7 5 5 D 6 8 7 9 E 9 8 9 9 F 4 9 4 5 G 3 6 7 8	E 9 8 9 9

H 4 8 9 3	
A 7 8 2 5	J 7 8 8 6
B 5 8 5 4	
C 1 2 8 3	
D 6 8 5 1	
E 9 5 4 5	
F 4 9 5 4	
G 3 6 6 3	
H 6 8 4 2	
I 7 8 8 6	
J 4 5 6 1	

## 374.學生成績輸入(1 分)

### 問題描述：

某間國小第 2 次段考結束了，老師要幫小學生輸入新的段考成績，分別有國文、英文、數學，成績可以用整數或倍精數輸入，輸入完畢後會一一印出小學生的成績。

宣告類別：Std 內含三個 private 整數型態成員變數 chn 、 eng 、 math ，分別代表國文、英文與數學成績。建構方法有三種版本：

(1) 傳入三個整數 (int chn, int eng, int math) ，以傳入數值預設其成績初始值。

(2) 傳入兩個倍精數 ( double chn, double eng) ，其輸入值被轉型成整數後預設初始值，而 math 預設為 -1 。

(3) 無傳入參數之建構方法，呼叫第 (1) 版本方法，並預設初始值為 chn=0 、 eng=-1 、 math=-1 。另寫一個 public void disp() 方法，可顯示其成績。

### 輸入說明：

Main 方法宣告 Std 類別之物件名稱 s1,s2,s3, 在建立物件時，

s1 傳入之 3 個整數值 (75,55,45) ；

s2 傳入之 2 個倍精數 (82.5,72.3) ； s3 無傳入參數；

在 main() 方法中， s1 、 s2 、 s3 各自呼叫 disp 方法

### 輸出說明：

輸出 s1 75 55 45

### 範例：

Sample Input	Sample Output
75 55 45	s1 75 55 45
82.5 72.3	s2 82 72 -1
no input	s3 0 -1 -1



## 375.計算矩形立體方塊(1 分)

### 問題描述：

某天阿土伯在田裡工作挖到 3 個寶箱和一張紙，紙上說只要分別算出，這 3 個矩形立方體的寶箱的立方體積，紙上寫了一些小提示，第一個寶箱沒有任何參數傳入 ( 預設  $W=15$  、  $D=5$  ，  $H=10$  ) ，第二個寶箱給了固定的  $H=5$  其它  $W$  、  $D$  要自己填上，第三個寶箱給了 3 個數要自己填上，完成以上的計算後寶箱就會自動打開。

宣告類別：**block** 描述一個矩形立體方塊，具有三個 **double** 成員變數  $W$  、  $D$  、  $H$  分別代表寬度、深度與高度。三種建構方法，版本一：可依傳入三個 **double** 數值作  $W$  、  $D$  、  $H$  初始值設定，版本二：傳入兩個 **double** 數值，可設定  $W$  、  $D$  之數值，而高度  $H$  內定為 5 。版本三：無傳入任何數值，但會呼叫版本一建構方法，並預設  $W$  、  $D$  、  $H$  為 (15, 5, 10) 。

一個方法 **double volume()** 可計算立體方塊之體積並回傳。

### 輸入說明：

**main**    方法宣告 **block** 類別之物件名稱 **a,b,c**, 建構 **a** 方塊時沒有傳參數，

建構 **b** 方塊時，以版本二建構方法，預設  $W=6$  、  $D=8$ ;

建構 **c** 方塊時，預設  $W=4$  、  $D=6$  、  $H=15$ ;

呼叫 **volume**    方法，顯示 **a,b,c** 之體積

### 輸出說明：

750.0

240.0

360.0

### 範例：

Sample Input	Sample Output
6 8	750.0
4 5 15	240.0

	360.0
--	-------

## 376.計算員工薪水(1 分)

### 問題描述：

某間公司月底要發薪水，所以需要程式來計算員工的薪水，薪資計算是以（底薪扣除請假之薪資，請假每小時扣薪資 120 元）計算的，A 員工 ID 是 1001 請假時數 10.5 小時，B 員工 ID 是 1002 請假時數 5 小時，請問計算後 A、B 員工薪資是多少。

宣告一員工類別：**EmpRec**，內含整數型態成員變數 **id** 代表員工編號；整數型態成員變數 **salary** 與 **basepay** 分別代表薪資與底薪；**float** 型態變數 **hours** 代表請假時數。

一個 **disp()** 方法可顯示員工編號、請假時數與薪資；

一個 **void setid(int id, int basepay)** 方法可依傳入之數值，用來設定 **id** 與 **basepay**；

一個方法 **void sethours(float hours)** 傳入請假時數設定成員變數 **hours**，並計算薪資後，設定變數 **salary** 之值：薪資計算以底薪扣除請假之薪資，請假每小時扣薪資 120 元。

### 輸入說明：

**main** 方法中，宣告兩個 **EmpRec** 類別之物件 **a,b**,

在 **main()** 方法中，(1) 以 setid 方法

設定物件 **a** 之 **id=1001**, **basepay=n**;

設定物件 **b** 之 **id=1003**, **basepay=n**;

(2) 以 sethours 方法 設定物件 **a** 與 **b** 請假時數分別為 10.5 與 5 小時，

(3) 呼叫 **disp** 方法列印薪資報表

※ **n** 為接收一個整數。

### 輸出說明：

24740

### 範例：

Sample Input	Sample Output
--------------	---------------

26000	24740
18500	17900

## 377.速度競賽(1 分)

問題描述：

小明最近新買的一台重機，想與隔壁的小王比賽。因為小明很臭屁自己騎車技術很好又說自己機車很好，所以叫小王開車與他比賽，他們又邀請了小文來紀錄比賽秒數，比賽開始時間是 10 點 10 分 20 秒小文開始記錄，但小王的汽車因為過熱停在原地不動，小明到終點時間是 15 點 15 分 20 秒，他們請小文算出是誰速度最快，然後又快了多少。

宣告類別：Clock; 內含

三個 `private` 整數型態成員變數 `hh, mm, ss`，分別代表 時，分，秒；

兩種建構方法，版本一：可依傳入之時分秒作時間設定，版本二：無傳入參數，呼叫版本 1，並預設時間為 15:15:20。

一個 `public` 方法 `void seccal(Clock t)` 傳入一個 `Clock` 物件，並計算 `t` 物件與呼叫物件之時間差。

一個 `public` 方法 `void dispcnt`，可顯示時間

輸入說明：

`main` 方法中，宣告 `Clock` 類別之物件名稱 機車與汽車，

機車預設時間初始值為 10 點 10 分 20 秒；機車建構時未傳參數；

呼叫 `dispcnt` 方法，顯示機車與汽車時間，

呼叫 `seccal` 方法，顯示機車與汽車之時間差；

輸出說明： 汽車比機車快 300 秒

範例：

Sample Input	Sample Output
15 10 20	300
15 15 20	

## 378.記錄汽車出廠表單(1 分)

### 問題描述：

某間汽車製造公司，製造出來的汽車要做記錄，記錄表上有出廠年份、定價、業務員編號，第一批出廠的汽車年份 2009、定價 800000( 初始值 )，但必須給這台汽車配上一個業務員 102，如果該業務員業績不佳即可換掉。

宣告汽車類別：Car; 內含 3 個 private 整數成員變數 year, price 與 sales 如下：

static int 成員變數 year，代表出廠年份；static final int 成員變數 price 代表定價，

在 static 初始區塊 預設出廠年份為 2009，price 為 800000；

一個 private int 型態成員變數 sales 代表業務員編號。建構方法可預設其業務員編號。

一個 public 方法 void setsales (int sales)，可以傳入整數值，設定該車之負責業務員。

一個 toString() 方法可回傳列印字串，包含出廠年份、售價及業務員編號。

### 輸入說明：

main 方法宣告 Car 類別物件 c1, 與 c2，以建構方法預設其 sales 值分別為 102 與 105，利用 toString() 方法，列印 c1 與 c2 物件訊息，c1 呼叫 setsales 方法，修改 c1 業務員編號為 101，再次利用 toString() 方法，列印 c1 與 c2 物件訊息。

### 輸出說明：

c1：年份 2009, 定價 800000,sales 102

### 範例：

Sample Input	Sample Output
102	c1：年份 2009, 定價 800000,sales 102  c2：年份 2009, 定價 800000,sales 105

	<div>修改後 ...</div> <div>c1：年份 2009, 定價 800000,sales 101</div> <div>c2：年份 2009, 定價 800000,sales 105</div>
--	--

## 379.計算到校時間(1 分)

問題描述：

小明今天早上 9 點起床，如果他現在立刻出到學校，小明會在 10:10:0 到學校。

如果小明吃完早餐再去學校，則小明到校時間會晚 30 分到學校。

如果小明先去修機車和吃早餐再去學校，則小明到校時間會晚一個小時又 30 分。

宣告類別：Clock;

內含三個整數型態成員變數 hh=10, mm=10, ss=0，代表初始時間 10 時,10 分,00 秒；

一個方法 void disp，在方法中會顯示當時之時間。

另一個方法 void adj (int v)，在方法中 若 v=1, 則 ss 會加 1；若 v=2, 則 mm 會加 30；若 v=3, 則 hh 會加 1。

輸入說明：

main 方法中，宣告 Clock 類別之物件名稱 w,

在 main 方法中：

- (1) 呼叫 disp 方法以顯示時間；
- (2) 再執行“分”的調整，然後再次顯示時間；
- (3) 再執行“時”的調整，然後再次顯示時間，

輸出說明：到校時間 10:10:0

範例：

Sample Input	Sample Output
2	10 10 0
3	10 40 0
	11 40 0



## 380.多重型態計算成績(1 分)

問題描述：

數學老師要做一個數字的計算分別，有三種計算方式，第一種計算兩種整數的平均值、第二種計算兩種浮點數的平均值、第三種計算三種整數的平均值，老師希望能一次計算三種方法所以希望有程式幫他。

宣告類別：`Math`；內含一個多重定義之方法 `mean`，可將傳入參數之平均值回傳。方法 `mean` 為多重定義，有三種版本，(1) 傳入兩個整數，回傳 `double` 型態平均值，(2) 傳入兩個 `float` 型態數值，回傳 `float` 型態平均值，(3) 傳入三個整數，回傳 `double` 型態平均值。

輸入說明：

`main` 方法宣告 `Math` 類別之物件名稱 `test`，呼叫 3 次 `mean` 方法，並列印結果。

輸出說明：

`mean(65, 72)=68.5`

範例：

Sample Input	Sample Output
23	54.0
56	39.5
56	55.666666666666664
23	
52	
63	
52	

## 381.數學遞迴計算(1 分)

### 問題描述：

今天林老師心情很好，剛好今天又要幫學生考數學，所以他決定今天小考放一點水，他考了一題很簡單的數學題，題目是  $1+2+3...+n$ ，依照輸入的  $n$  值計算出對應的結果。

宣告類別：`Math`；內含一個方法 `int ACC(int n)`，此一方法傳入整數值  $n$ ，計算  $1$  加到  $n$  的結果，即  $1+2+3...+n$ ，並回傳“ $1$  加到  $n$  的結果”。

### 輸入說明：

`main` 方法宣告 `Math` 類別之物件名稱 `test`,

由鍵盤輸入一個整數  $m$ ; 呼叫 `ACC` 方法並印出“ $1$  加到  $m$  的結果”。

### 輸出說明：

輸入 50

答案 1275

### 範例：

Sample Input	Sample Output
5	15

## 382.多重定義傳值(1 分)

問題描述：

操場上有 5 位阿兵哥，今天班長想找這五位阿兵哥排成一列，請問 若從這五位阿兵哥中選出 1 位阿兵哥排成一列，試問共有多少種可能的線性安排？

若從這五位阿兵哥中選 2 位阿兵哥排成一列，試問共有多少種可能的線性安排？

若從這五位阿兵哥中選 3 位阿兵哥排成一列，試問共有多少種可能的線性安排？

(b) 宣告類別：Math; 內含多重定義之方法 multiply，可將傳入參數之“乘積”傳

回，傳入之三數有三種版本，(1) 傳入一個 int 數，(2) 傳入兩個 int 數，

(3) 傳入三個 int 整數

輸入說明：

Main 方法宣告 Math 類別之物件名稱 test,

呼叫 3 次 test.multiply 方法，

並列印結果：

輸出說明：

共有 5 種可能的線性安排

範例：

Sample Input	Sample Output
3	3
5	15
2	30

### 383.計算 BMI 值(1 分)

問題描述：

健康中心這次要幫大家測量 BMI 值 ( $\text{BMI} = \text{體重} / (\text{身高})^2$ ) 測量步驟要先記錄同學身高、體重之後顯示剛設定的同學身高、體重，最後再做側量 BMI 值得顯示，健康中心希望有個能幫助他們測量的程式。

宣告類別名稱 `Body` 內含二個成員變數：身高 `height`，體重 `weight`，資料型態均為 `float`。有一方法 `void shape(float h, float w)`，在方法中設定物件之身高、體重之值。另一方法 `void disp()`，印出該物件之身高、體重之值。第三方法 `float BMI()`，計算物件之 BMI 值並回傳。若 BMI 值  $< 18.5$  則體重過輕若 BMI 值  $> 24$  則體重過重若介於兩者之間，則為體重適中在 `main()` 方法中，宣告類別之物件為 `ROBERT`。

輸入說明：

`main` 方法中，

在 `void shape` 設定 `ROBERT` 的身高為 172.0，體重為 62.5 公斤。

叫出方法 `void disp()` 印出剛設定之值。

呼叫方法 `BMI` 後，算出 BMI 後印出，印出結果如下所示。

輸出說明： `ROBERT` 身高為 172.0 公分，體重為 62.5 公斤

BMI 值為 21.126284, 體重適中

範例：

Sample Input	Sample Output
120	ROBERT
20	Height:180.0Cm, Weight:50.0Kg
	BMI:
	15.432099,The weight is excessively light

### 384.以遞迴函數計算階乘(1 分)

#### 1. 問題描述：

寫一個程式以遞迴函數計算階乘。

#### 輸入說明：

一整數  $n$ ， $n \geq 0$ 。

#### 輸出說明：

以遞迴函數計算  $n!$ 。

#### 範例：

輸入範例	輸出範例
0	1
1	1
2	2
8	40320
5	120

## 385.判斷一個含 N 個整數的陣列 H 是否為最大堆積 (max-heap) (1 分)

1. 問題描述：判斷一個含 N 個整數的陣列 H 是否為最大堆積 (max-heap)

寫一個程式判斷一個含 N 個整數的陣列 H 是否為最大堆積 (max-heap)。

輸入說明：

第一行為 N

第二行為一整數序列，以空白鍵隔開 N 個整數。

輸出說明：

輸出 YES 或 NO 回應是否為最大堆積。

範例：

輸入範例	輸出範例
5	
4 2 3 5	
8	NO
43 37 24 28 30 20 23 7	YES
9	YES
37 30 32 28 25 30 0 26 24	
5	NO
77 70 68 80 44	YES
9	
39 36 25 30 25 0 0 25 25	

## 386.數字翻轉(1 分)

### 問題描述

輸入任意數字，並將其數字全部倒轉

### 輸入說明

一個整數，不超過  $2^{31}$  (i.e.  $2^{31}=2147483648$ )

### 輸出說明

翻轉過後的數字，若翻轉後前面有 0 的話應消除

### 範例

Sample Input	Sample Output
12345678	87654321
135790	97531
24681357	75318642

## 387.刪除數字(1 分)

### 問題描述

給定一個正整數  $a$ ，令其小於 200 位數。再給定另一個整數  $b$ ，由  $a$  中刪除  $b$  個數字，得到另一個新的整數  $c$ ，使得  $c$  為最小。舉例而言： $a=9876$ ， $b=2$ ，那麼  $c=76$ 。

### 輸入說明：

我們的輸入只有一行。一行中包含兩個數字，這兩個數字用逗點隔開。第一個數字是  $a$ ，第二個數字則是  $b$

### 輸出說明：

請將  $c$  輸出。

### 範例：

**Sample Input: Sample Output:**

9876,2          76



## 388.猜數字(1 分)

### 問題描述

魔術師請觀眾想一個三位數  $abc$ ，並且請觀眾把  $acb$ ， $bac$ ， $bca$ ， $cab$ ， $cba$  這 5 個數相加，得到一個和  $x$ 。只要觀眾說出  $x$ ，那麼魔術師便可以說出  $abc$  為何。請寫一個程式來扮演魔術師，當輸入  $x$  時，將  $abc$  給輸出。

### 輸入說明：

我們的輸入只有一行，也就是  $x$ 。

### 輸出說明：

將 3 位數  $abc$  輸出

### 範例：

**Sample Input: Sample Output:**

1589                  631

## 389.壓縮序列(1 分)

### 問題描述

一個序列由  $n$  個整數組成。令此序列為  $a_1, a_2, a_3, \dots, a_n$ 。所謂的壓縮的操作指的是在此序列中取出兩個數， $a_x$ ，以及  $a_y$ ，將這兩個數由序列中刪除，並且加入一個新的數  $2a_x \times 2a_y$ 。當做完  $n-1$  次的壓縮操作後，整個序列會只剩下一個數  $x$ 。請寫一個程式，其可以在輸入的序列中執行  $n-1$  次壓縮操作，使得最後剩下的數  $x$  為最大。

### 輸入說明：

輸入為一行字串，即為輸入的序列，序列中每個整數用逗號隔開，且中間不會有空格。例如底下的序列中有 6 個數字（即  $n=6$ ）

6,7,4,3,5,8

### 輸出說明：

將  $x$  輸出。

### 範例：

**Sample Input: Sample Output:**

6,7,4,3,5,8      20643840

### 390.我們結婚吧! (2 分)

**問題描述：** 中國傳統結婚禮俗中，訂婚與結婚是不可同日的。今日林小羊歡欣鼓舞的挑了兩個日子，他相當期待結婚日的到來，準備從訂婚完後，開始倒數節日那日的來臨。哪麼可不可以請你幫他算一算，訂婚日與結婚日式間隔幾日呢？

※ 每四年一閏，每百年不閏，每四百年一閏

**輸入說明：** 第一列為訂婚日，第二列為結婚日，日期格式依照西元年月日。

**輸出說明：** 輸出兩時間點間隔多少日。

**範例：**

Sample Input	Sample Output
2010 11 11	3
2010 11 15	
Sample Input	Sample Output
2010 11 20	2
2010 11 23	

## 391.生命靈數(1 分)

### 問題描述：

生命靈數簡單地說，就是每個人人生來都帶著的密碼，由西洋出生年月日組合而成。而生命靈數的流年運勢，則是根據每年宇宙年（例如今年為 2010 年， $2+0+1+0=3$ ），加上每個人生日的月份與「個性數字」（出生日期）總和來論。

將你的西元出生年月日加起來，例如 1985 年 2 月 20 日出生的人就是： $1+9+8+5+0+2+2+0=27$ ，這時出來的數位如果是一位元數以上的話再相加（注：如果相加之後還是一位數以上的話，就再相加，直到變成一位數）。

例如：先前出來的是 27 那就是  $2+7=9$

**輸入說明：** 以西元年月日輸入，年月日之間以空格隔開。

**輸出說明：** 算出生命靈數數字。

**範例：**

Sample Input	Sample Output
1987 2 20	2

### 392.數字之最(1 分)

**問題描述：** 從一連串數字之中，計算阿拉伯數字 0~9 ，那些出現次數最多。如有相同的最多次數，一併顯示。

**輸入說明：** 讀取一串數字。

**輸出說明：** 計算哪個阿拉伯數字出現最多次。

**範例：**

Sample Input	Sample Output
6789766678677967866387976	Number is 6

### 393.ISBN 驗證(1 分)

問題描述：

ISBN(International Standard Book Number) 是一種世界共通的書籍編碼方法，世界上任何一本書籍之出版，皆有著唯一的一組 ISBN 碼。此碼由十個位數組成，每一位數可以為 0~9 的任何一個數字，或者為 X，代表此位數為 10。其判斷方法如下，首先，將此 ISBN 碼的十個位數分開，自左而右依次為第一位數，第二位數至第十位數，接著進行第一次的累加，使得第二位數成為第一位數到第二位數的和，第三位數為第一位數到第三位數的累加和，第十位數為第一位數到第十位數的累加和；進行完第一次的累加和後，接著再依照相同之方法進行第二次的累加動作，我們稱此時最後所求得之累加和為此 ISBN 碼之識別碼，倘若此識別碼為 11 的倍數，則此 ISBN 碼為合法的。例如，若輸入之 ISBN 碼為 0 1 3 1 6 2 9 5 9 X，則其運算之過程如下表所示：

ISBN 碼	0	1	3	1	6	2	9	5	9	10(X)
第一次累加和	0	1	4	5	11	13	22	27	36	46
第二次累加和	0	1	5	10	21	34	56	83	119	165

經由計算可得其識別碼為 165，乃是 11 之倍數，故此為一合法之 ISBN 碼。

輸入說明：輸入一串 ISBN 碼，以空格隔開。

輸出說明：合法 ISBN 碼，印出 "YES"，不合法 ISBN 碼，印出 "NO"。

範例：

Sample Input	Sample Output
0 1 3 1 6 2 9 5 9 X	YES

### 394.猴子偷吃桃(1 分)

問題描述：

猴子吃桃問題：猴子第一天摘下若干個桃子，當即吃了一半，還不過癮，又多吃了一個第二天早上又將剩下的桃子吃掉一半，又多吃了一個。以後每天早上都吃了前一天剩下的一半又一個。到第  $n$  天早上想再吃時，只剩下一個桃子了。求第一天共摘了多少。

提示：

(1). 請先輸入一個數代表第  $n$  天。

輸入說明：

10

輸出說明：

$10 = 1$

$9 = 4$

$8 = 10$

$7 = 22$

$6 = 46$

$5 = 94$

$4 = 190$

$3 = 382$

$2 = 766$

$1 = 1534$

範例：

Sample Input	Sample Output
10	10 = 1  9 = 4  8 = 10  7 = 22  6 = 46  5 = 94  4 = 190  3 = 382  2 = 766  1 = 1534



### 395.判斷回文數(1 分)

問題描述：

一個 5 位數，判斷它是不是回文數。即 12321 是回文數，個位與萬位相同，十位與千位相同。

提示：

先輸入一組回文數。

輸入說明：

12321

輸出說明：

yes

範例：

Sample Input	Sample Output
12321	yes

## 396.判斷三角形(1 分)

問題描述：

試撰寫一程式，判斷這三個整數是否能構成三角形的三個邊長（註：三角形兩邊長之和必須大於第三邊）。

提示：

(1) 輸入三個正整數。

(2) 兩邊長之和大於第三邊長就印出 `Bigger than the third side`，否則印出 `Less than the third side`。

輸入說明：

1

2

3

輸出說明：

Less than the third side

範例：

Sample Input	Sample Output
1	Less than the third side
2	
3	

## 397.規畫作息時間程式(1 分)

問題描述：

請撰寫一支規畫作息時間程式，計算每個人 24 小時，所花費的時間在那些事情上較多，並以百分比做呈現。

提示：

(1). 輸入 4 數分別代表 1. 睡覺及休息、 2. 吃飯、 3. 上課及上班、 4. 娛樂。



計算公式如下：

$$(\text{娛樂時間} / 24) * 100\% = \text{娛樂} \%$$

$$(\text{總花費時間} - 24) / 24 * 100\% = \text{閒置} \%$$

(2). 當如果睡覺及休息時間小於 33% 時，將會提醒使用者多多休息。

輸入說明：

5

6

3

4

輸出說明：

Sleep and rest:20.833334

Eat:25.0

School and work:12.5

Entertainment:16.666668

Idle:25.0

More rest

範例：

Sample Input	Sample Output
5	Sleep and rest:20.833334
6	Eat:25.0
3	School and work:12.5
4	Entertainment:16.666668
	Idle:25.0
	More rest