

全國高級中等學校 100 學年度工業類科學生技藝競賽電腦軟體設計

壹、試卷說明：

1. 請將寫好之程式原始檔依題號命名資料夾存檔，第一題取姓名_Q1，第二題取姓名_Q2，依序命名存檔，並存於 C 碟之資料夾姓名_Contest 中。
2. 競賽時間 4 小時。
3. 請將程式編譯成執行檔儲存在 C 碟之資料夾姓名_Contest 及隨身碟片中。

貳、評分說明：本試卷共六題，每題配分不一。

1. 每題評分只有對與錯兩種，對則給滿分，錯則不給分(即以零分計算)。
2. 每解答完一題上傳，評審人員將針對該題進行測試，若解題正確則回應正確，若解題錯誤則扣該題一分至該題零分為止，答錯之題目可繼續作答。

試題一：設計一個可讓使用者輸入正多邊形邊數及顏色並繪製出正多邊形的程式。(17分)

說明：N 邊形，有 N 個頂點、N 個邊、一個面。邊由兩個頂點連線組成，面由多個邊圍成。由此可知，頂點是多邊形組成的基本元素，因此只要找到多邊形的各頂點座標，就可以輕易地連線畫出多邊形了！

要找到多邊形的頂點，我們可以利用圓來內接多邊形，利用 360 度角(2π)，切割成 n 等分，再利用餘弦函數和正弦函數來求得頂點的 x, y 座標值。N 邊形頂點公式：

$$X : [\text{圓心到頂點之長(圓半徑)}] * \cos(2\pi/n) + [\text{圓心 X 值}]$$

$$Y : [\text{圓心到頂點之長(圓半徑)}] * \sin(2\pi/n) + [\text{圓心 Y 值}]$$

請設計一程式：可輸入邊數 N，範圍需大於等於 3，小於等於 10， $3 \leq N \leq 10$ ；圖形顏色可讓使用者選擇 (B:black, R:red, G:green, L:blue)；半徑及圓心圖形界面請自行設計。

輸入範例：

N=9

Color=B

輸出範例：如圖 1-1 所示

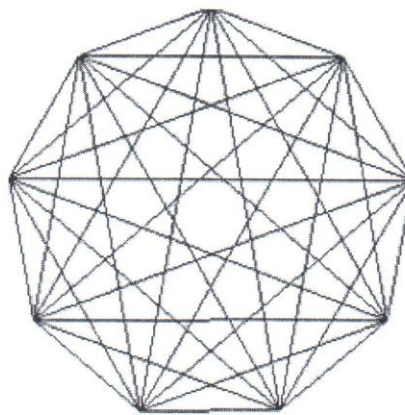


圖 1-1

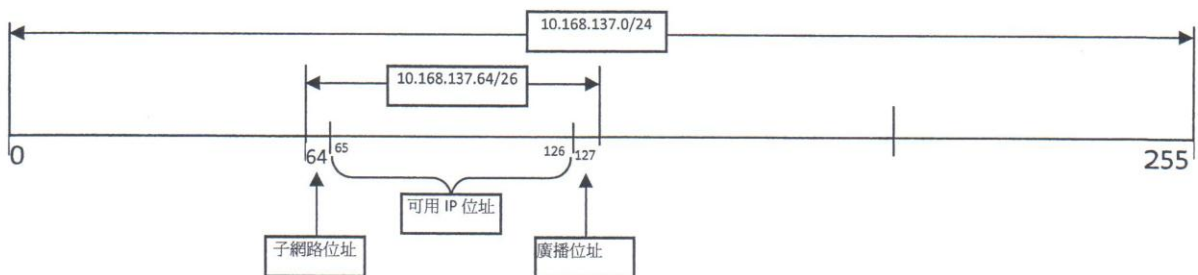
試題二：計算 IP 相關位址 (17 分)

說明：32 位元的 IP 是由網路位元(network bits)及主機位元(host bits)組成，網路位址最少 8bits、最多 30bits。

若網路位元有 24 位元、則主機位址有 8 位元，則其子網路遮罩為 255.255.255.0，是由連續的 24 個 1 及 8 個 0 所組成，可表示成 /24。若網路位址有 30 位元，則其子網路遮罩為 255.255.252，可表示成 /30。

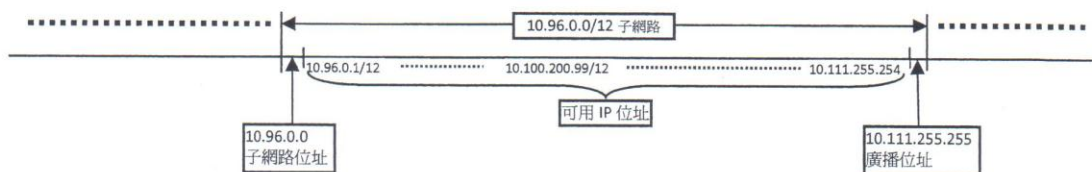
請寫一個程式能依所給的 IP 位址及子網路遮罩(/n)來計算出該 IP 所在的網路位址(network address)、廣播位址(broadcast address)、該網路的可用 IP 數。例如：已知某 IP 位址 10.168.137.88/26，如下圖所示：

- 網路位元數有 26bits，計算方法：保留左邊 26bits 的數值，故其網路位址為 10.168.137.64。
- 廣播位址為 10.168.137.127，計算方法是：將 6(=32-26)位元的主機位元均設為 1，再與廣播位址 OR 起來，如 0.0.0.63 OR 10.168.137.64 得到 10.168.137.127。
- 該網路的可用 IP 數為 62 個，計算方法是： $2^{(32-26)}-2$ 。

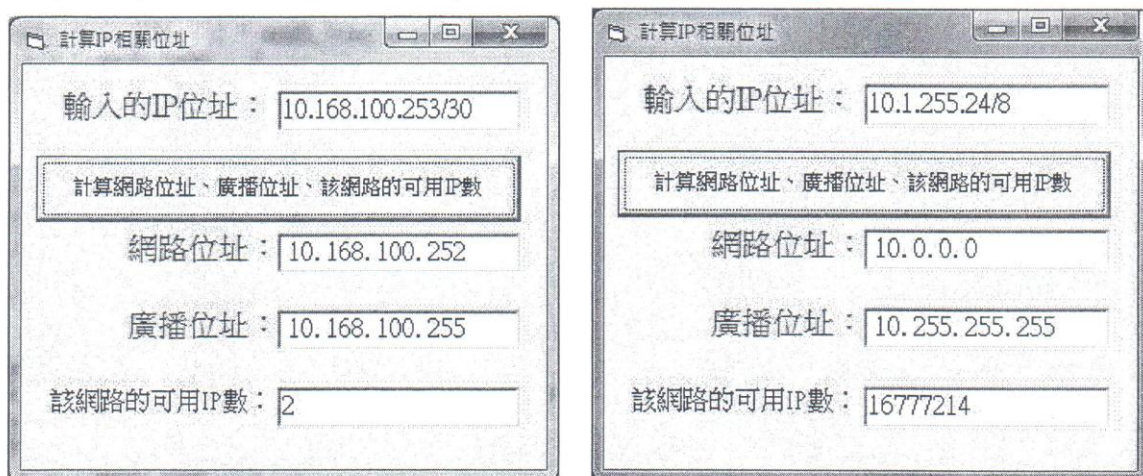


另舉一例，10.100.200.99/12 的 IP 位址，如下圖所示：

- 網路位元數有 12bits，計算方法：保留左邊 12bits 的數值，故其網路位址為 10.96.0.0。
- 廣播位址為 10.111.255.255，計算方法是：將 20(=32-12)位元的主機位元均設為 1，再與廣播位址 OR 起來，形成 0.15.255.255 OR 10.96.0.0 得 10.111.255.255。
- 該網路的可用 IP 數為 1048574 個，計算方法是： $2^{(32-12)}-2$ 。



程式測試：IP 位址的測試範圍 10.0.0.1 ~ 10.255.255.254，子網路遮罩 /8 ~ /30，均可得到正確的網路位址、廣播位址、該網路的可用 IP 數。參考畫面如下：



試題三：電腦螢幕倒數計時器(16分)

說明：(1)在一些比賽的場合，例如：歌唱比賽、話劇比賽、演講比賽、籃球比賽、程式設計比賽等，有競爭就有公平公正公開的要求，而清楚地、公開地顯示時間，就是其中的一項要求。從而裁判、選手、領隊及觀眾等，均可清楚地了解當下「時間」的資訊。

(2)設計一電腦螢幕計時器，可在介面輸入時間，或直接在程式中設定，以秒為單位，計時過程由所設定的數值遞減至 0 為止。

(3)為清晰地、清楚地顯示時間，字體必須放大至 400 及設定為紅色，以便可在不同場合，透過投影機投射到螢光布幕，眾多人等均可清楚看到。

(4)在此命題僅設計「分」與「秒」即可，可使用相關軟體設計。

3. 執行結果參考畫面：

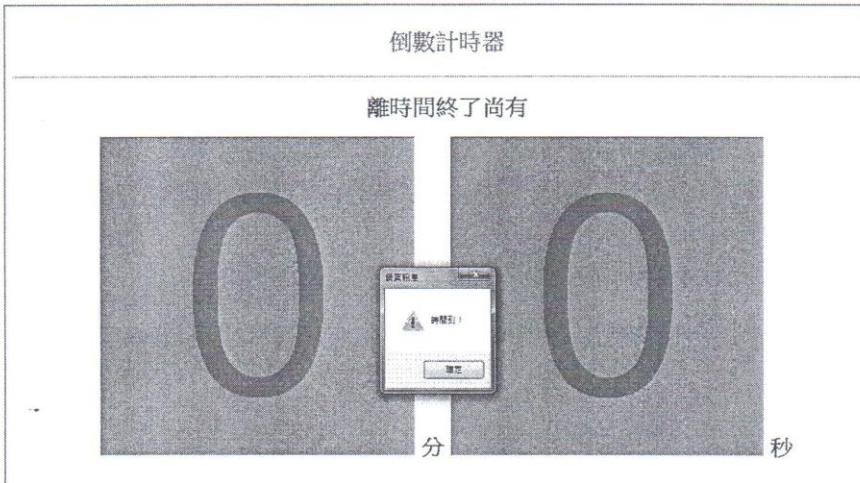
(1)例如：將時間設定為 2 分 0 秒，如下畫面：



(2)執行過程的畫面之一：



(3)時間計時終了了的畫面：



試題四：灰階影像直方圖 (Histogram of Gray Level Image) (17分)

說明：在有些時候，想要估測所拍攝照片的亮度分佈為何，最常使用的方式是，統計所拍攝照片的灰階影像之亮度分佈情況，這種統計即灰階影像直方圖，要達到此目的，其演算法如下：

1. 輸入一張彩色影像(RGB Image)，彩色影像的大小是高度(Height)×寬度(Width)×3，其中，每一點色彩像素(color pixel)，是由紅色 (R)，綠色 (G)，藍色 (B) 三種顏色來組成，每一 R, G, B 分別用 8 位元來表示，所以，一張 RGB 彩色影像總共有 24 位元，可以表示 $2^{24}=16,777,216$ 個顏色。
2. 將彩色影像轉為灰階(Gray)影像(Image)，其轉換公式如下：
$$\text{Gray}(y, x) = 0.3 \times R(y, x) + 0.59 \times G(y, x) + 0.11 \times B(y, x) \quad (1)$$
3. 對灰階影像求灰階值出現的頻率，此頻率一般稱為直方圖 (Histogram)，其中，灰階影像的灰階值是從 0 (黑色) 到 255 (白色)，總共有 256 個變化，即亮度有 256 種變化。
4. 將直方圖做正規化，這就是所謂的機率密度函數(Probability Density Function)，其公

式如下：

$$pdf(g) = \frac{n_g}{N}, \quad g = 0, 1, 2, \dots, 255. \quad (2)$$

其中， N 是影像中的像素總和， n_g 是影像灰階值 g 的出現頻率（像素個數）

程式功能：

請利用上述演算法，寫一個程式，能完成以下功能要求：

- (1) 能讓使用者選取要觀測之照片圖檔。
- (2) 能將輸入的彩色影像轉為灰階影像。
- (3) 求出和顯示灰階影像直方圖。
- (4) 求出和顯示最小灰階和最大灰階值(即最暗亮度和最亮亮度)。
- (5) 求出和顯示出現機率最大的灰階值(即出現最多之亮度)和此機率值。

程式執行範例和要求：

- (1) 當程式剛執行時會出現一視窗，如下圖（一）所示，使用者可以利用功能表，來選取功能。



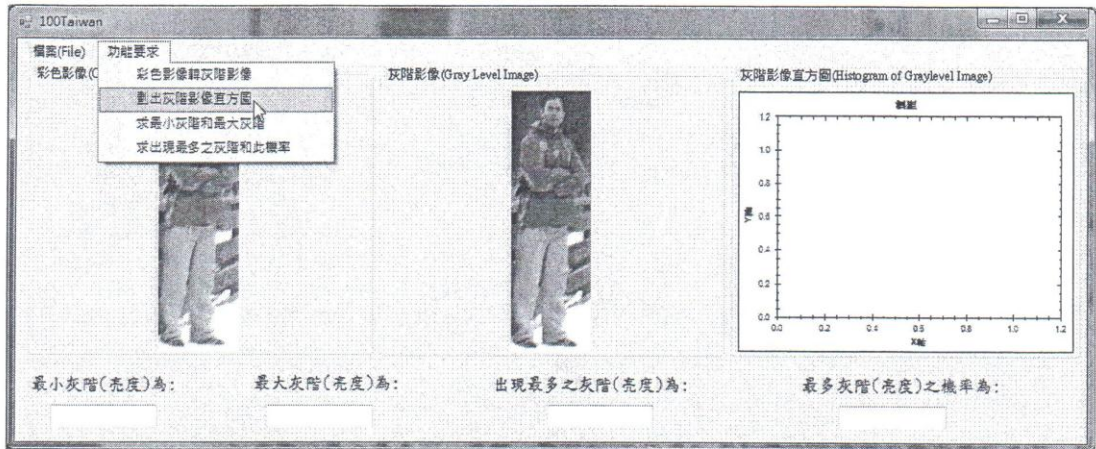
圖（一）

- (2) 當使用者選取功能表之開啟彩色影像檔之後，程式會自動顯示此影像檔，如圖（二）所示。



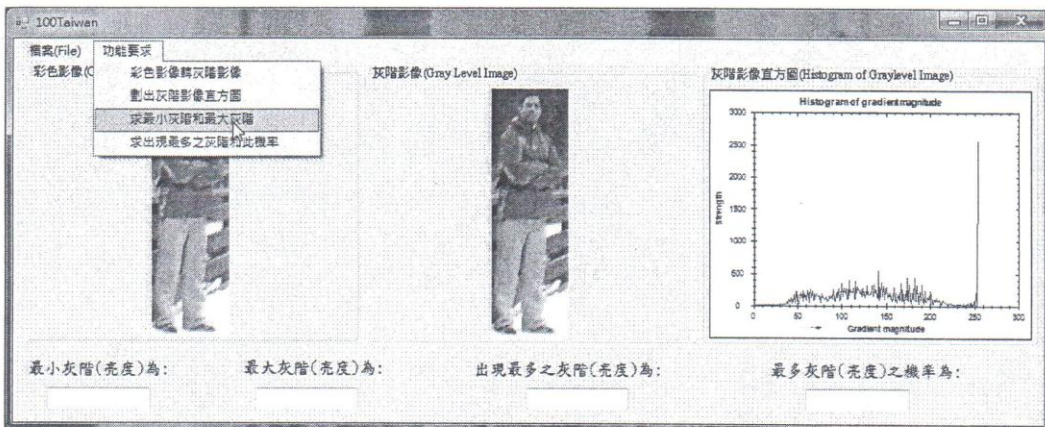
圖（二）

(3) 當使用者選取功能表之功能要求: 彩色影像轉為灰階影像, 程式會顯示灰階影像, 如圖(三)所示。



圖(三)

(4) 當使用者選取功能表之功能要求: 劃出灰階影像直方圖, 程式會顯示灰階影像直方圖, 如圖(四)所示



圖(四)

(5) 當使用者選取功能表之功能要求: 求出最小灰階和最大灰階, 程式會顯示最小灰階和最大灰階值, 如圖(五)所示。

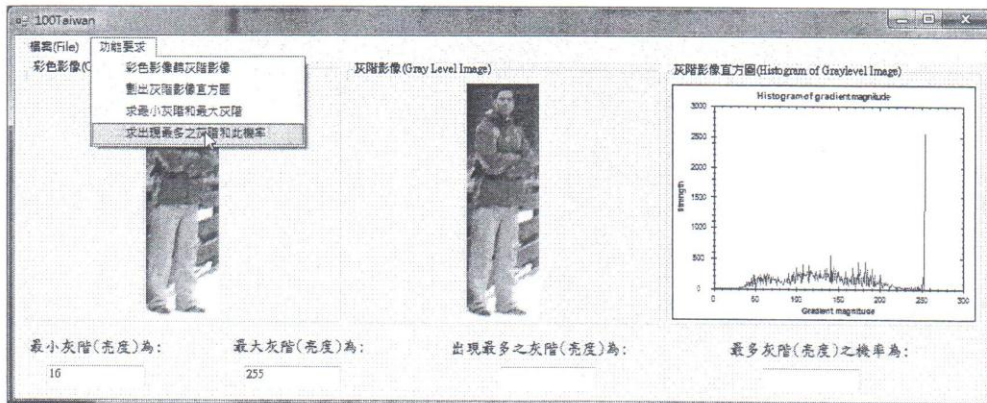
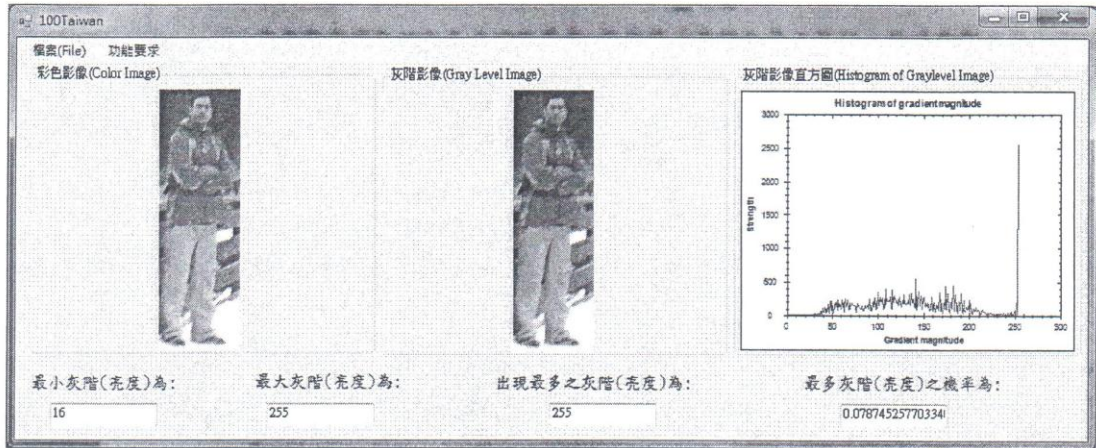


圖 (五)

(6) 當使用者選取功能表之功能要求: 求出現最多之灰階和此機率, 程式會顯示出現最多之灰階和其機率值, 如圖 (六) 所示。



(7) 若妳(你)的程式都完成上述功能和要求, 才可以要求檢查功能。

試題五：實數與二進位數之編碼解碼 (16 分)

請撰寫一程式, 根據實數變數 x 的範圍(range)以及精確度(precision)的要求, 程式必須能夠將使用者所輸入之實數轉換成相對應之二進位數, 或是將二進位數轉換成相對應之實數。

說明：

我們希望使用二進位向量(binary vector)來代表一實數變數 x , 該二進位向量的長度很顯然與所需要的精確度相關。

假設變數 x 的範圍是 $[-1.0 \ 2.0]$, 編碼的精確度為小數點以下 6 位, 則 $[-1.0 \ 2.0]$ 需要切割成：

$$(2.0 - (-1.0)) \times 10^6 = 3000000 \text{ 個等分, 這表示我們需要使用 22 位元來進行編碼:}$$

$$2097152 = 2^{21} < 3000000 < 2^{22} < 4194304$$

以得到二進位字串：

$$(b_{21}b_{20}b_{19} \dots b_0)$$

亦即：

二進位字串(000 ... 0)將代表實數值-1.0, (111 ... 1) 將代表實數值 2.0。

如果要進行解碼, 由一個二進位字串轉換成一個實數值 x , 作法如下：

1. 將二進位字串：

$$(b_{21}b_{20}b_{19} \dots b_0)$$

由基底 2 轉換為基底 10：

$$(b_{21}b_{20}b_{19} \dots b_0)_2 = \left(\sum_{i=0}^{i=21} b_i \cdot 2^i \right)_{10} = x'$$

2. 求得對應之實數值 x ：

$$x = -1.0 + x' \cdot \frac{(2.0 - (-1.0))}{2^{22} - 1}$$

範例一：假設實數變數 x 的範圍是 $[-1.0 \ 2.0]$ ，編碼的精確度為小數點以下 6 位。
則二進位字串

(1000101110110101000111)

即是代表實數值 0.637197，因為

$$x' = (1000101110110101000111)_2 = 2288967_{10}$$

$$x = -1.0 + 2288967 \cdot \frac{(2.0 - (-1.0))}{2^{22} - 1} = -1.0 + 2288967 \cdot \frac{3}{4194303} = 0.637197$$

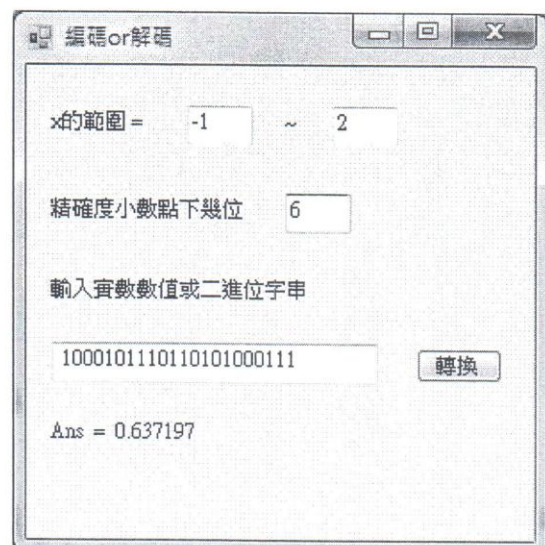
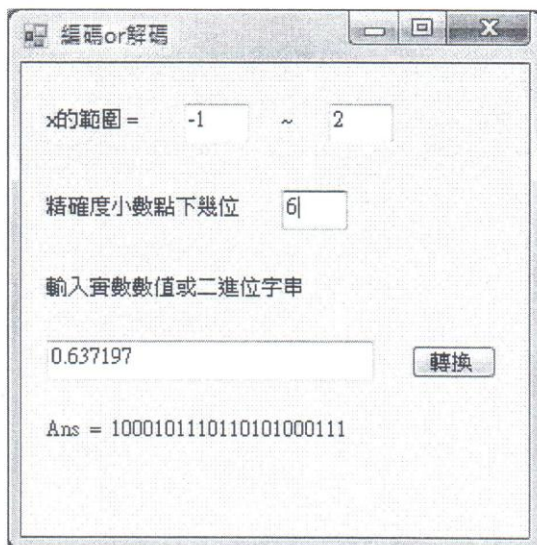
範例二：假設變數 x 的範圍是 $[-3.0 \ 12.1]$ ，編碼的精確度為小數點以下 4 位，則我們需要以 18 位元來進行編碼。

因此，二進位字串：(010001001011010000)

即是代表實數值 1.0524，因為

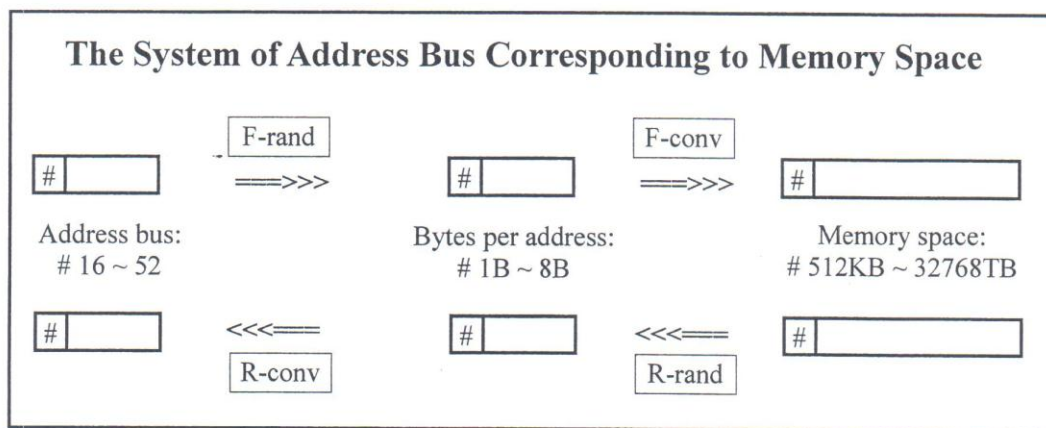
$$x = -3.0 + (010001001011010000)_2 \cdot \frac{(12.1 - (-3.0))}{2^{18} - 1} = -3.0 + 70352 \cdot \frac{15.1}{262143} = 1.0524$$

程式執行範例和要求：當程式剛執行時會出現一視窗，並提示使用者先輸入實數變數 x 的範圍，以及編碼的精確度(小數點以下位數)，程式首先需要求得編碼所需之位元數。當使用者輸入一實數值之後，程式需能自動轉換該實數值成相對應之二進位字串；當使用者輸入一個二進位字串之後，程式需能自動轉換該二進位字串成相對應之實數，如下圖所示。



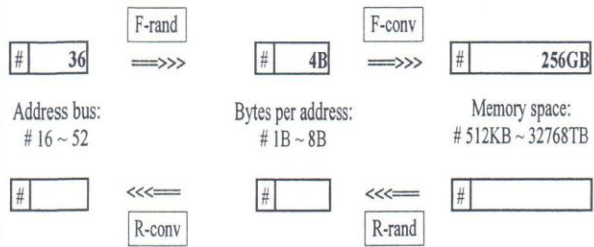
試題六：位址線與記憶體空間之對等轉換系統說明：(17分)

- 記憶體空間的基本單位為位元組(Byte, 簡稱B), 且常用的四個單位為: $2^{10} \times 1B$ 對等於 1KB、 $2^{20} \times 1B$ 對等於 1MB、 $2^{30} \times 1B$ 對等於 1GB 及 $2^{40} \times 1B$ 對等於 1TB。而計算機組織中之位址匯流排與主記憶體空間維持一定的對等轉換關係, 以順向對等轉換為例: 已知位址匯流排有 24 條線, 且每個位址一次可存取 2B 資料, 則對等主記憶體空間為 $2^4 \times 2B = 2^4 \times 2^0 \times 2B = 32MB$; 以反向對等轉換為例: 已知主記憶體空間為 24GB, 且每個位址一次可存取 4B 資料, 則對等位址匯流排需 33 條線(即 $24GB = 24 \times 2^{30}B / 4B = 3 \times 2^{33} / 2^2 < 2^2 \times 2^{31} = 2^{33}$)。
- 請依說明 1, 設計如下圖所示初始狀態之位址匯流排與記憶體空間之對等轉換系統(The System of Address Bus Corresponding to Memory Space), 使其具有下列功能:
 - 每當滑鼠左鍵點一下左上角之 F-rand時, 則為如圖中上半部由左至右(==>>>)之順向對等轉換功能, 此時上半部之位址匯流排(Address bus)與每位址可存取位元組數(Bytes per address) 兩顯示區分別隨意產生 16~52 條線之間與 1B~8B 之間, 且可分別以手動更改為各自區間之任一值; 同時其餘顯示區立即被清除為空白。接著以滑鼠左鍵點一下 F-conv 時, 則記憶體空間(Memory space) 顯示區顯示對等轉換的記憶體容量。
 - 每當滑鼠左鍵點一下右下角之 R-rand時, 則為如圖中下半部由右至左(<<<==)之反向對等轉換功能, 此時下半部之記憶體空間(Memory space)與每位址可存取位元組數(Bytes per address) 兩顯示區分別隨意產生 512KB~32768TB 之間與 1B~8B 之間, 且可分別以手動更改為各自區間之任一值; 同時其餘顯示區立即被清除為空白。接著以滑鼠左鍵點一下 R-conv 時, 則位址匯流排(Address bus)顯示區顯示對等轉換的位址線數。



範例：系統執行之初始狀態如上圖螢幕所示。當滑鼠左鍵點一下 F-rand 則為順向對等轉換(==>>>), 上半部之 Address bus 與 Bytes per address 兩顯示區分別隨意產生 36 與 4B, 其餘顯示區被清除為空白; 接著滑鼠左鍵點一下 F-conv 則 Memory space 顯示對等轉換的記憶體容量為 256GB, 如左下圖所示。當滑鼠左鍵點一下 R-rand 則為反向對等轉換(<<<==), 下半部之 Memory space 與 Bytes per address 兩顯示區分別隨意產生 12TB 與 8B, 其餘顯示區被清除為空白; 接著滑鼠左鍵點一下 R-conv 則 Address bus 顯示對等轉換的位址線數為 41, 如右下圖所示。

The System of Address Bus Corresponding to Memory Space



The System of Address Bus Corresponding to Memory Space

