

子題 2：霍夫曼編碼（Huffman Coding）。(程式執行限制時間: 2 秒) 16 分

霍夫曼編碼（Huffman Coding），又譯為哈夫曼編碼、赫夫曼編碼，是一種用於無損資料壓縮演算法。由大衛·霍夫曼在 1952 年發明。

在電腦資料處理中，霍夫曼編碼使用變長編碼表對源符號（如檔案中的一個字母）進行編碼。

例如，在英文中，e 的出現機率最高，而 z 的出現機率則最低。當利用霍夫曼編碼對一篇英文進行壓縮時，e 極有可能用一個位元來表示，而 z 則可能花去 25 個位元（不是 26）。用普通的表示方法時，每個英文字母均占用一個字元組，即 8 個位元。二者相比，e 使用了一般編碼的 1/8 的長度，z 則使用了 3 倍多。倘若我們能對於英文中各個字母出現機率較準確的估算，就可以大幅度提高無失真壓縮的比例。

霍夫曼編碼（Huffman Coding）要先建立霍夫曼樹(Huffman Tree):

建立霍夫曼樹(Huffman Tree)步驟：

(一)針對相異字元，統計其出現的次數：

字元	A	B	C	D	E	F	G
出現的次數	2	3	6	8	13	15	19

(二)為每個字元建立一顆只有一節點的樹，每棵樹的根節點之關鍵值(紅色字)為其字元出現的次數.

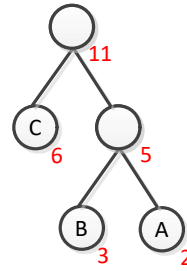
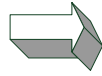
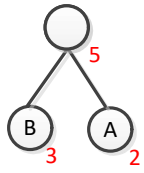


(三)找出根節點關鍵值(出現次數)最小的兩顆樹。

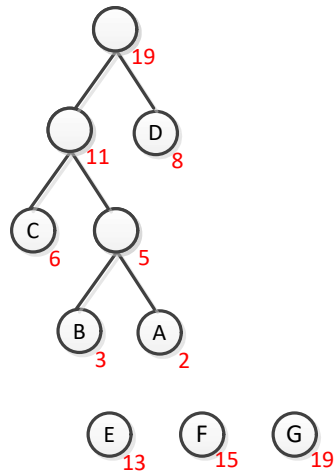
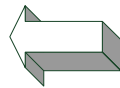
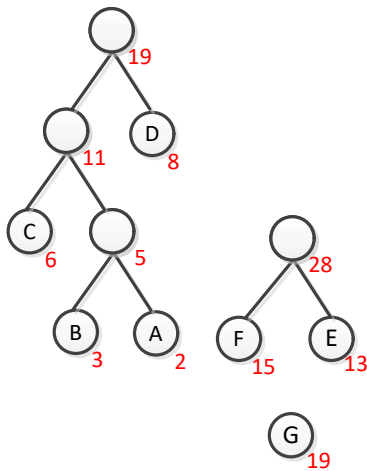
(四)產生一個新的根節點，並將找到的兩棵樹分別當作此新的根節點之左右子樹(節點關鍵值大的放左樹，關鍵值小的放右樹或是節點關鍵值小的放左樹，關鍵值大的放右樹)，而根節點的關鍵值為左右子樹節點之關鍵值(紅色字)的和。

(五)重複步驟 (三) 與 (四)，直至全部節點合併為一棵樹。

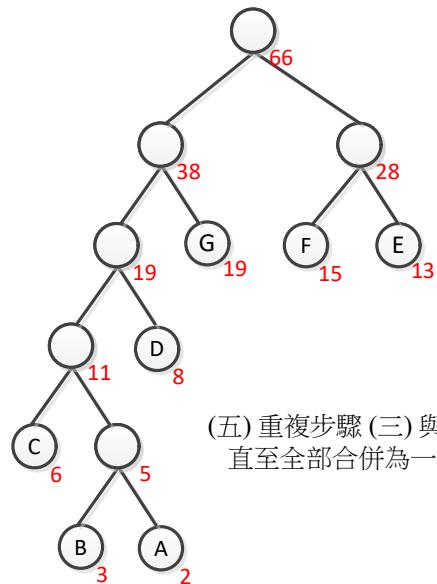
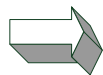
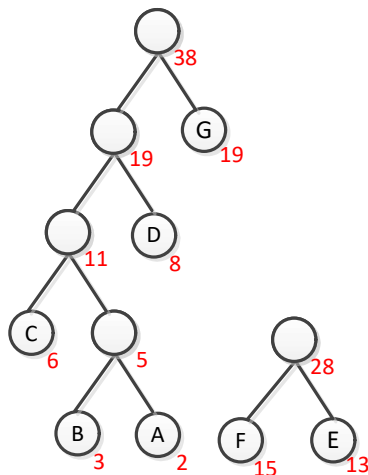
(四) 產生一個根節點，節點關鍵值大的放左樹，關鍵值小的放右樹，而根節點的關鍵值為左右子樹節點之關鍵值(紅色字)的和 $2+3=5$ 。



(三) 找出根節點關鍵值(出現次數)最小的兩顆樹2,3。



(三) 找出根節點關鍵值(出現次數)最小的兩顆樹13,15。(三) 找出根節點關鍵值(出現次數)最小的兩顆樹8,11。



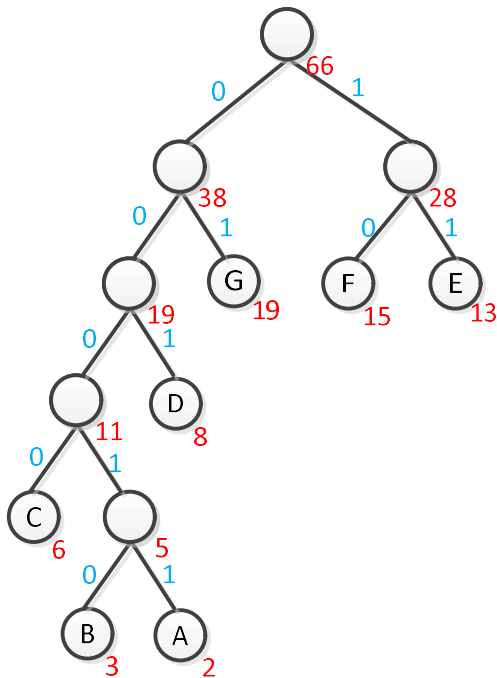
(五) 重複步驟(三)與(四)，直至全部合併為一棵樹。

(三) 找出根節點關鍵值(出現次數)最小的兩顆樹19,19。二個關鍵值一樣，可以依照排序先後秩序。(三) 找出根節點關鍵值(出現次數)最小的兩顆樹28,38。

產生霍夫曼編碼(Huffman Code):

(一)在霍夫曼樹(Huffman Tree) 中，針對每個節點，將連至左子樹的邊標為 0，將連至右子樹的邊標示為 1。

* 注意一：霍夫曼樹(Huffman Tree) 的每個葉節點代表一個相異字元，且葉節點的個數恰等於相異字元的個數。



(二)針對每個由根節點至葉節點的路徑，將其所經過邊的標示連結起來，並指派給對應葉節點所代表的字元，此即霍夫曼編碼(Huffman Code):

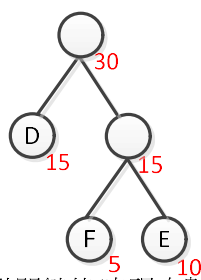
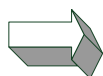
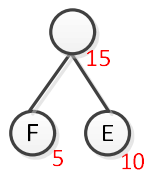
字元	A	B	C	D	E	F	G
出現的次數	2	3	6	8	13	15	19
霍夫曼編碼	00011	00010	0000	001	11	10	01
霍夫曼編碼 位元數(長度)	5	5	4	3	2	2	2

字元	A	B	C	D	E	F
出現的次數	26	25	20	15	10	5
霍夫曼編碼	10	01	00	110	1111	1110
霍夫曼編碼 位元數(長度)	2	2	2	3	4	4

壓縮檔案：

當產生所有字元的霍夫曼編碼(Huffman Code)後，我們可以利用霍夫曼編碼(Huffman Code)來取代檔案中的所有字元。

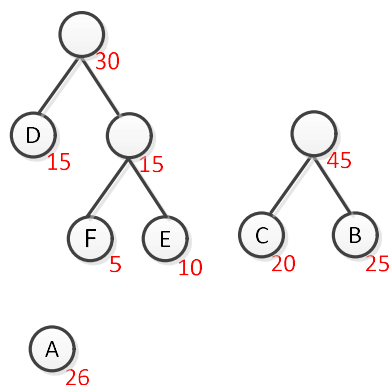
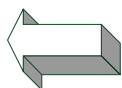
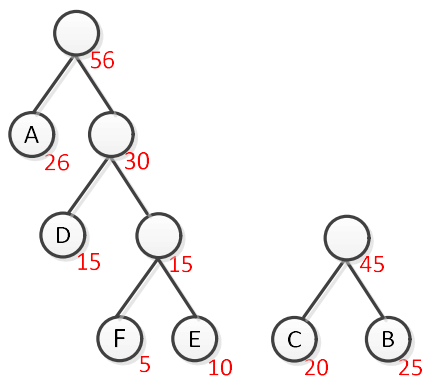
(四) 產生一個根節點，節點關鍵值小的放左樹，關鍵值大的放右樹，而根節點的關鍵值為左右子樹節點之關鍵值(紅色字)的和 $5+10=15$ 。



(三) 找出根節點關鍵值(出現次數)最小的兩顆樹5,10。

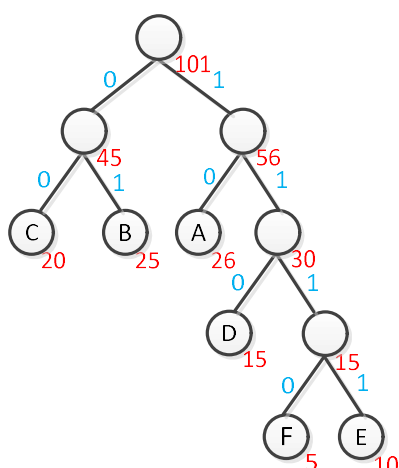
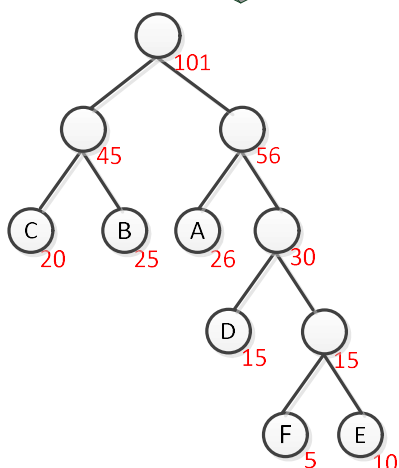


(三) 找出根節點關鍵值(出現次數)最小的兩顆樹15,15。二個關鍵值一樣，可以安照排序先後秩序。左右順序任意放不會影響答案霍夫曼編碼位元數(長度)。



(三) 找出根節點關鍵值(出現次數)最小的兩顆樹26,30。

(三) 找出根節點關鍵值(出現次數)最小的兩顆樹20,25。



(三) 找出根節點關鍵值(出現次數)最小的兩顆樹45,56。

產生霍夫曼編碼(Huffman Code)

輸入說明：

第一列的數字 n 代表有幾筆資料要測試， $2 \leq n \leq 8$ ，第二列起為測試資料，之後每列為每組測試資料，每組測試資料至少有 2 個正整數最多有 26 個正整數，正整數數字 N ， $1 \leq N \leq 65535$ 。各個數字間以“,”隔開，分別代表各字元的出現的次數。每組資料都有 M 個相異數字 ($2 \leq M \leq 26$)，讓你去建立霍夫曼樹(Huffman Tree)。會避免 3,3,8,8 這樣輸出結果不唯一的測試資料。

輸出說明：

在測試資料中所建建立霍夫曼樹(Huffman Tree)，完成霍夫曼編碼(Huffman Code)及計算霍夫曼編碼(Huffman Code)位元數(長度)，這組測試資料輸出一列，各個數字間以“,”隔開，分別代表各字元的霍夫曼編碼(Huffman Code)位元數(長度)。

輸入檔案 1：【檔名：in1.txt】

```
2
2,3,6,8,13,15,19
4,8,5
```

輸入檔案 2：【檔名：in2.txt】

```
3
26,25,20,15,10,5
4,3,5,6
2,1
```

輸出範例：【檔名：out1.txt】

```
5,5,4,3,2,2,2
2,1,2
```

輸出範例：【檔名：out2.txt】

```
2,2,2,3,4,4
2,2,2,2
1,1
```