

Problem P

霍夫曼編碼

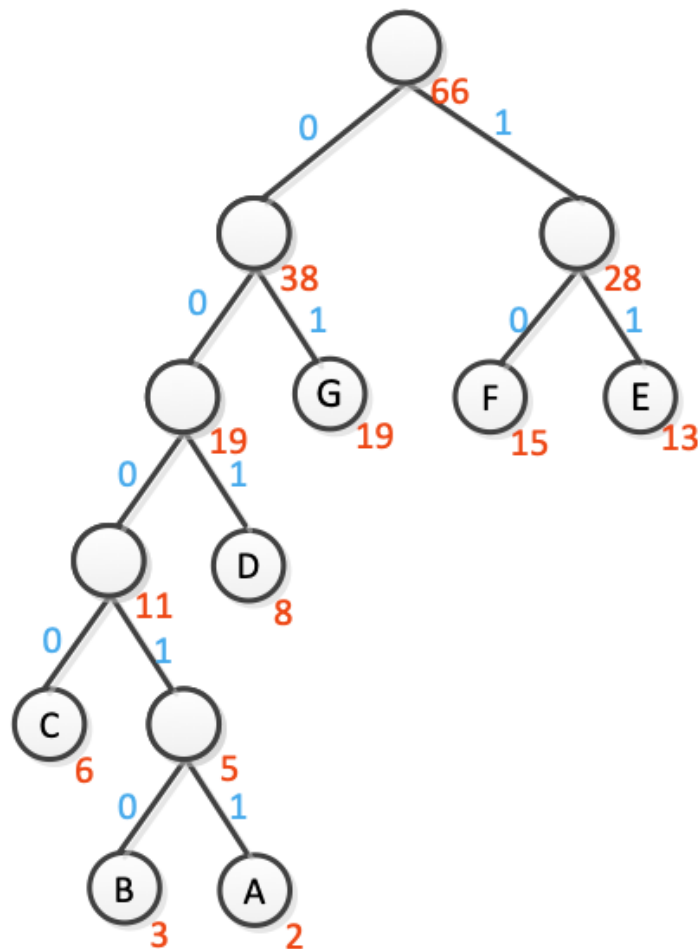
Problem Description

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。針對相異字元，統計其出現的次數：A:2。B:3。C:6。D:8。E:13。F:15。G:19。產生霍夫曼編碼(Huffman Code)：

Huffman 編碼法利用建立二元樹的方式，首先將每個符號以一個自由節點(freenode)表示這些自由節點的權重(weight)即是符號的頻率。接著，從自由節點中找出權重最小的兩個節點，為這兩個節點做一個父節點，此父節點的權重則為這兩個子節點的權重和。之後，將父節點加入自由節點的行列，並將兩個子節點從自由節點的行列中去除。

接著，重覆選取兩個權重最小的節點，造出父節點並更新自由節點的過程，直到最後只剩一個自由節點為止。

(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。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)：當得到這棵編碼樹後，我們就可以從

二元樹的根結點(root node)開始往下走每往左子樹(left subtree)走一層就給定一個位元的編碼0每往右子樹(right subtree)走一層就給定一個位元的編碼1依此方式逐層往下走並同時編碼直到走到葉節點(leaf node)為止。

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

編碼所得到的總位元數: $5+5+4+3+2+2+2=23$ 。

請撰寫一個程式，根據已知的頻率表(尚未排序過)，計算以Huffman 編碼後的總位元數。

Input Format

第一行為一個整數 N ，代表共有幾個符號。 $(1 < N \leq 2^{16})$ 。

接下來的 N 個數字代表這 N 個符號的頻率，每個數字為介於1 與 2^{16} 之間的正整數，每兩個數字以一個空格隔開。

Output Format

對每一個輸入的頻率表，計算以Huffman 編碼後的總位元數。

Sample Input 1

```
7
2 3 6 8 13 15 19
```

Sample Output 1

```
23
```

Sample Input 2

```
6
26 25 20 15 10 5
```

Sample Output 2

```
17
```

Hint