8. Find a binary tree whose preorder and inorder traversals create the same result.

16. Find the root of each of the following binary trees:

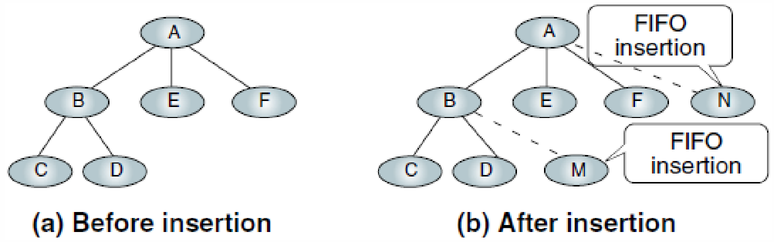
   a. tree with postorder traversal: FCBDG
   b. tree with preorder traversal: IBCDFEN
   c. tree with inorder traversal: CBIDFGE

18. A binary tree has eight nodes. The postorder and inorder traversals of the tree are given below. Draw the tree.

   Postorder: FECHGDBA
   Inorder: FCEABHDG

*只要寫出一種 case 即可

22. Draw the corresponding binary tree of Figure 6-21(b).



(a) Before insertion

(b) After insertion

FIGURE 6-21 FIFO Insertion into General Trees

26. Find the infix, prefix, and postfix expressions in the expression tree of Figure 6-27.

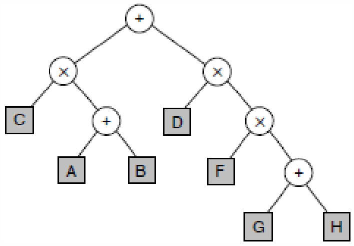
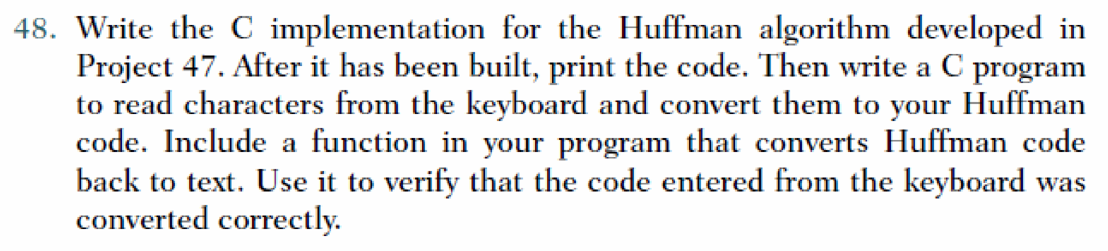


FIGURE 6-27 Expression Tree for Exercise 26

36. Write an algorithm that determines whether a binary tree is complete.

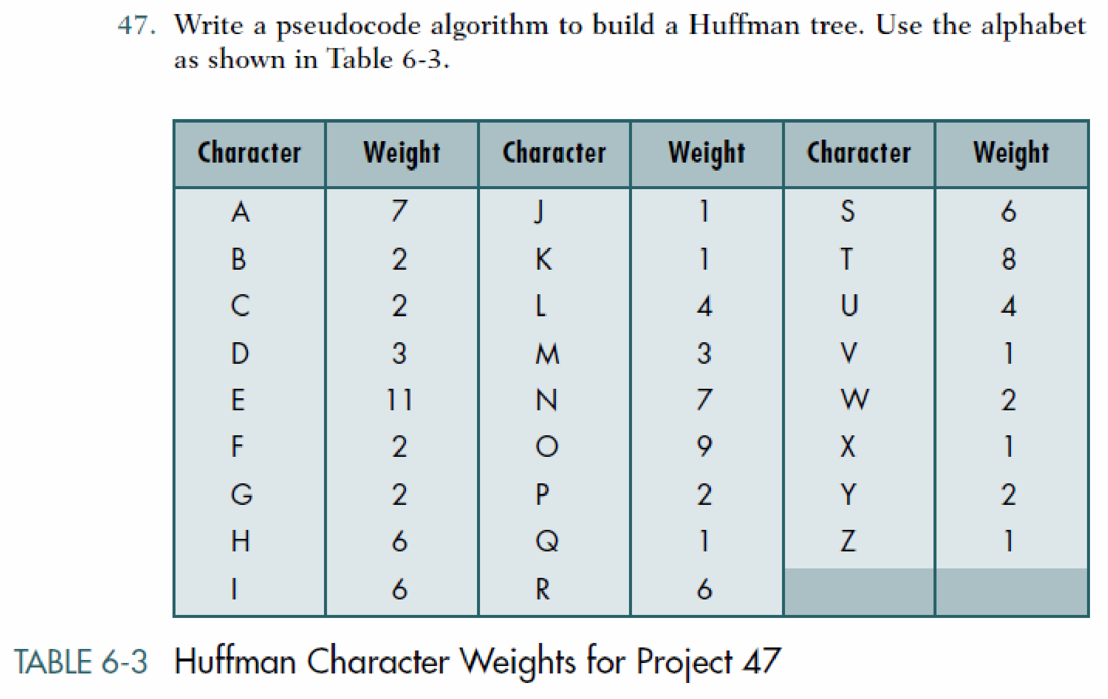
40. Rewrite the binary tree postorder traversal algorithm using a stack instead of recursion.



48. Write the C implementation for the Huffman algorithm developed in Project 47. After it has been built, print the code. Then write a C program to read characters from the keyboard and convert them to your Huffman code. Include a function in your program that converts Huffman code back to text. Use it to verify that the code entered from the keyboard was converted correctly.

***48 題參考**

1. Table 6-3 在 HW6_48.txt
2. Huffman tree 建立後，print 各個英文字對應的 code
3. 程式接下來要求使用者輸入一串文字，print 一串文字轉換成 code 的結果
4. 程式最後要求輸入 code，並且轉換回英文字



47. Write a pseudocode algorithm to build a Huffman tree. Use the alphabet as shown in Table 6-3.

Character	Weight	Character	Weight	Character	Weight
A	7	J	1	S	6
B	2	K	1	T	8
C	2	L	4	U	4
D	3	M	3	V	1
E	11	N	7	W	2
F	2	O	9	X	1
G	2	P	2	Y	2
H	6	Q	1	Z	1
I	6	R	6		

TABLE 6-3 Huffman Character Weights for Project 47