White House Shuffle

Purpose: Trees

Due: June 21st

Like most successful presidential campaigns, President Donald Trump staffed his initial administration with individuals who helped him get elected. However since then, there has been a record amount of turn over in the administration and they feel they need help keeping things straight. The white house's hierarchical structure can be represented by a tree. (Consider the example shown below.) Trump is the head of the organization. He has two direct subordinates, Sessions and Pompeo. members of the the organization who are direct subordinates of the same member are ranked by their respective seniority. In the diagram, the seniority of such members decreases from left to right. For example, Sessions has higher seniority than Pompeo.

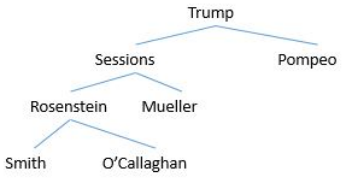

Figure 1: Early White House

When a member hires a direct subordinate, the newly hired subordinate has lower seniority than any other direct subordinates of the same member. For example, if Trump hires Kelly then Trumps direct subordinates are Sessions, Pompeo and Kelly in order of decreasing seniority.

There is a new policy when firing an employee to simplify the process of replacement. There are two possible scenarios. If the victim (the person who gets fired) had no subordinates, then he or she will simply be dropped from the white house's hierarchy. If the victim had any subordinates, then his or her highest-ranking (by seniority) direct subordinate will be promoted to fill the resulting vacancy. The promoted person will also inherit the victim's seniority. Now, if the promoted person also had some subordinates, then his or her highest-ranking direct subordinate will similarly be promoted, and the promotions will cascade down the hierarchy until a person having no subordinates has been promoted.

In Figure 1, if Sessions gets fired, then Rosenstein will be promoted to Sessions's position and seniority, and Smith will be promoted to Rosenstein's previous position and seniority. Figure 2 shows the hierarchy resulting from Figure 1 after (1) Trump hires Devos and (2) Sessions gets fired.

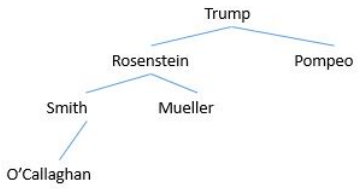

Figure 2: White House after hiring Devos and firing Sessions

Input

The first line of the input contains the name of the current president followed by hires, followed by another name. All names in the input file consist of 2–20 characters, which may be upper- or lowercase letters, apostrophes, and hyphens (but no blank spaces). Each name contains at least one uppercase and at least one lowercase letter. The first line will be followed by one or more additional lines. The format of each of these lines will be determined by one of the following three rules of syntax:

- `[Existing member] hires [new member]`
- `Fire [existing member])`
- `Print`
- `Quit`

Here [Existing member] is the name of any individual who is already a member of the white house, and [new member] is the name of an individual who is not a member of the white house yet. The three types of lines (hires, fire, and print) can appear in any order, any number of times. You may assume that at any time there is at least one member (who is the president) and no more than 1000 (assume no restriction for analysis) members in the white house.

Output

For each print command, print the current hierarchy of the white house, assuming all hires and fires since the beginning of the input have been processed as explained above. Tree diagrams (such as those in Figures 1 and 2) are translated into textual format according to the following rules:

- Each line in the textual representation of the tree will contain exactly one name.
- The first line will contain the president's name, starting in column 1.
- The entire tree, or any subtree, having the form shown in Figure 3 will be represented in textual form as shown in Figure 4.

There will not be any blank lines in the output.

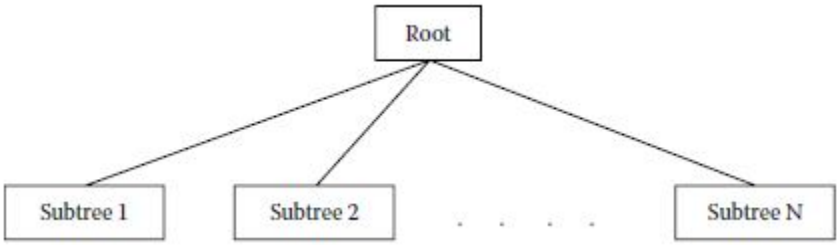

Figure 3: Arbitrary Tree

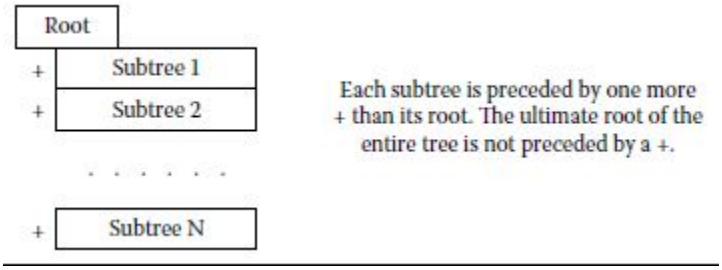

Figure 4: Tree Diagram

Sample Input and Corresponding Output

Sample Input	Sample Output
Trump hires Sessions Trump hires Pompeo Sessions hires Rosenstein Rosenstein hires Smith Rosenstein hires O'Callaghan Sessions hires Mueller Print tt+Pompeo Quit	Trump +Sessions ++Rosenstein +++Smith +++O'Callaghan ++Mueller
Trump hires Sessions Trump hires Pompeo Sessions hires Rosenstein Rosenstein hires Smith Rosenstein hires O'Callaghan Sessions hires Mueller Trump hires Devos Fire Sessions Print Quit	Trump +Rosenstein ++Smith +++O'Callaghan ++Mueller +Pompeo +Devos

How the program will be graded

Memo

What	pts
Name	1
Time Analysis O() of every function[1,2] (in terms of the words input	5
Space Analysis O() of every function[1,2]	5
A class diagram	10

Source Code Document

What	pts
Name	1
Description[5]	4
Style	10
pre/post conditions	10
Functionality	54
`Org.find(string)`	11
`Org.print()`	11
`Org.hire(string boss, string sub)`	16
`Org.fire(string sub)`	16

[1]The main() is a function.

[2]All analysis should be worst case based on the number of input words.

[5]The description should be written to some one who knows NOTHING about the program. It should discuss what the program does (in your own words). After reading the description the user should be able to create legal input and predict the output.

Data Structure

You will need to create a tree. Here each node can an unlimited number of children. I suggest you make each node have a vector (list) of pointers to nodes (nodes) as well as the name and a pointer to its parent.