

**General Note:**

- The design mark distribution for each function is given next to it. **Write the design only for those functions.**
  - For implementation, you are permitted to use the code that you have submitted as an assignment in Eduserver.
  - Verify the correctness of your program with **the two sample test cases that are uploaded in Eduserver.**
2. NITC wants to recall the students to the campus after the lockdown period. The Institute would like to record the *vaccination\_status* of each student (whether the student is *vaccinated* (Y) or *not vaccinated* (N)), along with their admission number (*Ad\_No*). Use AVL tree data structure to implement the following operations:
- *Insert*( $T, k$ ) : Insert a new student with *Ad\_No* =  $k$  to the AVL tree  $T$ . At first, mark the *vaccination\_status* of the student as *not vaccinated*.  
[1.5 Marks]
  - *Update*( $T, k$ ) : Update the *vaccination\_status* of the student with *Ad\_No* =  $k$  to *vaccinated*.  
[1.5 Marks]
  - *FindDiff*( $T, k$ ) : Let  $n$  be the number of students who are not vaccinated among all the students with *Ad\_No* <  $k$ , and  $y$  be the number of students who are vaccinated among all the students with *Ad\_No* <  $k$ . Find and print the difference between  $n$  and  $y$  ( $|n - y|$ ).  
[1.5 Marks]
  - *PrintTree*( $T, k$ ) : Print the tree rooted at *Ad\_No* =  $k$  in its parenthesis format.  
[1 Mark]

**Input/Output Format**

The input consists of multiple lines. Each line starts with a character from  $\{i, u, f, p, t\}$  followed by zero or more integers or characters. The integers, if given, are in the range  $[1, 10^5]$ .

- Character '*i*' : Character '*i*' will be followed by a positive integer  $k$  representing the admission number of a student. Read and store the details of the student with admission number  $k$  to the AVL tree  $T$ , using *Insert*() function.
- Character '*u*' : Character '*u*' will be followed by a positive integer  $k$  representing the admission number of a student. Update the vaccination status of the student with admission number  $k$  to *vaccinated*, using *Update*() function.
- Character '*f*' : Character '*f*' will be followed by a positive integer  $k$  representing the admission number of a student. Find and print the difference between the number of students that are *not vaccinated* and the number of students that are *vaccinated* among all the students with *Ad\_No* <  $k$ , using *FindDiff*() function.
- Character '*p*' : Character '*p*' will be followed by a positive integer  $k$  representing the admission number of a student. Print the tree rooted at  $k$  in the parenthesis format which is recursively defined as:
  - The string  $()$  represents an empty tree.
  - The string  $( k \ n \ y \ L \ R )$  represents a tree whose root node is a student with *Ad\_No* =  $k$ , where

- \*  $n$  is the number of students who are not vaccinated among all the students with  $Ad\_No < k$ .
- \*  $y$  is the number of students who are vaccinated among all the students with  $Ad\_No < k$ .
- \* L is the left subtree of the root node in Parenthesis Representation.
- \* R is the right subtree of the root node in Parenthesis Representation.
- Character ' $t$ ' : Terminate the program.

### Sample Input and Output

#### Input 1

```
i 401
i 600
i 323
i 200
i 156
p 401
f 401
u 200
u 156
u 401
p 401
f 401
i 601
p 401
f 200
u 601
p 600
t
```

#### Output 1

```
( 401 3 0 ( 200 1 0 ( 156 0 0 ( ) ( ) ) ( 323 0 0 ( ) ( ) ) ) ( 600
0 0 ( ) ( ) ) )
3
( 401 1 2 ( 200 0 1 ( 156 0 0 ( ) ( ) ) ( 323 0 0 ( ) ( ) ) ) ( 600
0 0 ( ) ( ) ) )
1
( 401 1 2 ( 200 0 1 ( 156 0 0 ( ) ( ) ) ( 323 0 0 ( ) ( ) ) ) ( 600
0 0 ( ) ( 601 0 0 ( ) ( ) ) ) )
1
( 600 0 0 ( ) ( 601 0 0 ( ) ( ) ) )
```