**General Note:**

- Read and understand the question carefully.
- For implementation, you are permitted to use the code that you have submitted as an assignment.
- You are not permitted to use global variables and/or static variables.
- Assume that all the inputs in the test cases are valid.

2. The details of the students ($reg\_id$ and CGPA ($cgpa$)) are stored in the institute server using a Binary Search Tree (BST) with $cgpa$ as the key.

   In the server, CGPA level ($cl$) of a BST is defined as the number of edges between its root and the farthest leaf in that tree. A BST with a single node has $cl = 0$ and an empty BST has $cl = -1$.

   The training and placement unit of the institute wants to upgrade the server by storing the details in a **structured_BST** $T1$ along with a BST $T2$. A **structured_BST** satisfies the following property in addition to the BST property.

   - For every node in a **structured_BST**, the difference of $cl$ of right subtree from $cl$ of left subtree is either $0$ or $1$. That is,

   $$cl \text{ of left subtree} - cl \text{ of right subtree } \in \{0, 1\}$$

   Given the details of students, write a program to implement the following functions as per the given function prototypes.

   - $main()$: Repeatedly read a character '$i$', '$p$', or '$c$' from the console and call the corresponding functions, as described below, until character '$t$' is encountered.
   - $insert\_student(T1, T2, student)$: Insert $student$ into the **structured_BST** $T1$. If $T1$ is not a **structured_BST** after the insertion, delete the parent node of $student$ in $T1$ and insert it into the BST $T2$. [2 Marks]
   - $print\_tree(T)$: Print the $reg\_id$ of the students stored in the BST $T$ in the order of its post-order traversal, separated by a space. Print NULL if $T$ is empty. [1 Mark]
   - $find\_cl(T)$: Find and return the CGPA level of the BST $T$. Return $-1$ if $T$ is empty. [2 Marks]

   **Notes:**

   - Efficient designs will have more weightage.
   - Implement BST operations using the pseudocodes in CLRS.

### Input/Output Format

The input consists of multiple lines. Each line starts with a character from $\{i, p, c, t\}$ followed by zero or one string and/or integers.

- Character '$i$' : Character '$i$' will be followed by the $reg\_id$ (string of maximum length 10) and $cgpa$ (float) $\in [0, 10]$ of a student. Insert this into the **structured_BST** using $insert\_student()$ function.
- Character '$p$' : Character '$p$' will be followed by an integer $x \in \{1, 2\}$. If $x = 1$, print the $reg\_id$ of the employees stored in the **structured_BST** $T1$ and if $x = 2$, print the $reg\_id$ of the students stored in the BST $T2$, using $print\_tree()$ function.

- Character '$c$' : Character '$c$' will be followed by an integer $x \in \{1, 2\}$. If $x = 1$, print the $cl$ of the **structured_BST** $T1$ and if $x = 2$, print the $cl$ of the BST $T2$ using $find\_cl()$ function.
- Character '$t$' : Terminate the program.

**Sample Input and Output**

**Input 1**

```
i B14715 7.4
i B1189 5.6
i B2142 8.9
i B2101 8.0
p 1
p 2
i B220 2.4
i B564 8.5
c 1
p 1
c 2
p 2
t
```

**Output 1**

```
B1189 B2101 B14715
B2142
2
B220 B1189 B564 B14715
1
B2101 B2142
```

**Input 2**

```
i B14715 7.2
i B2189 6.1
i B2142 7.4
i B1101 2.0
i B21011 7.9
i B12101 4.6
i B1105 3.3
c 1
c 2
i B11045 7.4
i B11021 5.5
p 1
p 2
i B22015 7.3
p 1
c 2
```

```
p 2
i B12022 8.5
i B13224 8.7
i B12224 8.5
p 2
p 1
c 1
c 2
t
```

**Output 2**

```
2
2
B11021 B2189 B11045 B21011 B14715
B1105 B12101 B1101 B2142
B11021 B2189 B22015 B21011 B14715
3
B1105 B12101 B1101 B11045 B2142
B1105 B12101 B1101 B13224 B12022 B11045 B2142
B11021 B2189 B22015 B12224 B21011 B14715
2
3
```