Exercise #2 Spec

Due Date: 2023/12/21 23:59:00

Red-black tree

Rule: (in textbook)

1. Every node is either red or black.
2. The root is black.
3. Every leaf (NIL) is black.
4. If a node is red, then both its children are black.
5. For each node, all simple paths from the node to descendant leaves contain the same number of black nodes.

Implement a red-black tree, including the following two operations:

- Insert a node
- Delete a node

Note: The red-black tree properties should be hold after each operation.

Input:

The first input t denotes that there are t tasks in the following inputs.
For each task, the first two inputs op, n denote that this task is either inserting elements (op = 1) or deleting elements (op = 2), and the number of the following elements is n.
The next n inputs are the elements to be operated.

Output:

Display the tree using **inorder traversal**.
Please output (1) the value, (2) the parent node, and (3) the color of each node.
P.S.
(1) Do not output NIL.
(2) The parent of the root is empty. Output empty (blank) value for the parent of root.
(3) Do not use Tab as space.The line breaks are represented as '\n'.

1. Insert node:

Example input:

2 (number of tasks)
1 8 (insert, 8 elements)
5 11 9 7 6 12 4 1
1 2 (insert, 2 elements)
2 3

Example output:

Insert: 5, 11, 9, 7, 6, 12, 4, 1
key: 1 parent: 4 color: red
key: 4 parent: 6 color: black
key: 5 parent: 4 color: red
key: 6 parent: 9 color: red
key: 7 parent: 6 color: black
key: 9 parent: color: black
key: 11 parent: 9 color: black

key: 12 parent: 11 color: red
Insert: 2, 3
key: 1 parent: 2 color: red
key: 2 parent: 4 color: black
key: 3 parent: 2 color: red
key: 4 parent: 6 color: red
key: 5 parent: 4 color: black
key: 6 parent: color: black
key: 7 parent: 9 color: black
key: 9 parent: 6 color: red
key: 11 parent: 9 color: black
key: 12 parent: 11 color: red

2. Delete node:
Example input:
2 (number of tasks)
1 8 (insert, 8 elements)
5 11 9 7 6 12 4 1
2 2 (delete, 2 elements)
11 5
Example output:
Insert: 5, 11, 9, 7, 6, 12, 4, 1
key: 1 parent: 4 color: red
key: 4 parent: 6 color: black
key: 5 parent: 4 color: red
key: 6 parent: 9 color: red
key: 7 parent: 6 color: black
key: 9 parent: color: black
key: 11 parent: 9 color: black
key: 12 parent: 11 color: red
Delete: 11, 5
key: 1 parent: 4 color: red
key: 4 parent: 6 color: black
key: 6 parent: 9 color: red
key: 7 parent: 6 color: black
key: 9 parent: color: black
key: 12 parent: 9 color: black

(Only provide the means of input and output. Please check the format of attachment testing dataset D1 display.)

Rules of programing and the datasets:

(1) All the datasets will always insert nodes first.
(2) All the datasets will not delete non-existent elements.
(3) No duplicate nodes will be in the tree.
(4) $0 < t <= 15$, $0 < n <= 20$.
(5) The data type of every element is Integer (int range).
(6) Outputs that don't break the rules of Red-Black Tree will be right.
(7) You can only use standard header files.

Exercise #2 Submission Policy

A. Language

C, C++

(Please check your program can compile successfully by gcc/g++)

B. Input / output Format

Please refer to **Example input/output** on the previous page and attachment **testing dataset D1**(input.txt and output.txt).

The line breaks are represented as '\n'.

C. Submission File

1. Main program
   - You should name your file as Exercise2_STUDENT_ID.c / Exercise2_STUDENT_ID.cpp.
   - Your program should use standard input / output. Do not read / write the text file.
2. Report
   - You can write in Chinese and English and the content must include:
     - Environment (OS, compiler version, IDE) (2%)
       - How to run your program
     - Results (4%)
       - Method or solutions
       - Anything you want to share

Please hand in your main program and report to the e3 platform.

(Do not compress files)

- Exercise2_STUDENT_ID.c / Exercise2_STUDENT_ID.cpp
- Exercise2_STUDENT_ID.pdf

D. Score

There will be 3 testing datasets: D1, D2 and D3. D1 is provided in input.txt and output.txt

- Pass D1: 60%
- Pass D2: 21%
- Pass D3: 18%
- Report: 6%

Total: 105

Penalty

- not use standard I/O -10 pts
- testing result output format error -5 pts
- filename error / submit compressed file -5 pts

- If you have submitted Exercise #2 but your grade is below 60, you will have one opportunity for a makeup submission **within seven days** of the Exercise #2 grade release. The maximum final grade achievable through makeup submission is 60 points.

E. Cheating Policies

- 0 points for any cheating on assignments.

- Allowing another student to examine your code is also considered cheating.

F. Late Submission

- Every week late from the due day will get a 10% penalty.
  For example, if you submit the homework on 12/21 23:59:01 - 12/28 23:59, your final score will be multiplied by 0.9. And if you submit it on 12/28 23:59:01 - 2024/01/05 23:59:59, your final score will be multiplied by 0.8.
- late submission after 2024/01/06 00:00 will be 0 points.

G. Asking Questions

- If you have any questions, you can choose from the following options:
  - **Email TAs**: It is recommended to email all TAs to avoid any potential issues with missing responses.You can send email through E3 mail system or gmail.
  - **Exercise #2 E3 Forum**: Post your questions on the Exercise #2 E3 forum for discussion and assistance.
  - **In-Person Assistance**: Come to EC126 after informing TAs.

- Remember that TAs may not always be able to respond instantly. It's suggested not to wait until the due day to ask questions.

- All TA responses in the HW2 discussion forum will adhere to the specifications outlined in this document.