Data Structure and Programming, Spring 2024
programming assignment #4
Monotonic Routing

Chiao-Yun, Chin
b10901154@ntu.edu.tw

Shao-Ming, Chen
r11942095@ntu.edu.tw

Pei-Yuan, Wu
peiyuanwu@ntu.edu.tw

DUE DATE: 05/23/2024

1 Problem Description

The median is the middle value in an ordered integer list. If the size of the list is even, there is no middle value, and the median is the mean of the two middle values.

- For example, for array = [2,3,4], the median is 3.
- For example, for array = [2,3], the median is (2 + 3) / 2 = 2.5

In this homework, you are expected to implement both MinHeap and MaxHeap classes, and utilize them to further implement a FindMedian class.

2 TODO

We provide the templates for **MinHeap**, **MaxHeap**, and **FindMedian** in **Median.py**. Your task is to complete the sections marked as **TODO**.

1. MinHeap
   - insert(self, item): input: a value. Insert a new item with the given value into the MinHeap. You don't need to return or print anything in this function.
   - removeMin(self): Remove the minimum item from the MinHeap.
2. MaxHeap

- insert(self, item): input: a value. Insert a new item with the given value into the MaxHeap. Don't return or print anything with this function.
- removeMax(self): Remove the maximum item from the MaxHeap.

3. FindMedian

    - __init__(self): initialization your own data structure. Implementing with heap structure is highly recommended.
    - AddNewValues(self, NewValues): Input a list of values. Add these new values (a list of items) into your data structure.
    - ShowMedian(self): Return the median of your data structure. If the size of the list is even, there is no middle value, and the median is the mean of the two middle values. The return value should always be a float number.
    - RemoveMedian(self): Remove the median of your data structure. If there is an even number of elements, remove the larger one. For example, if array=[1, 2, 3, 5], remove 3.

Note:

1. Do not modify the interface of the functions, but you can add your own functions.
2. Any statistics library is not allowed. You should not use heapq library.
3. The time complexity of **AddNewValues** method for a single item should be $O(\log n)$, **RemoveMedian** method should be $O(\log n)$, and **ShowMedian** method should be $O(1)$.

3 Evaluation

Use the commands "python3 0001.py" and "python3 0002.py", then compare your result with golden_1.txt, golden_2.txt respectively to check your code.

4 Submission

Please put Median.py into a directory named studentID and compress the directory into studentID.zip to NTUCOOL. The homework is due on 05/23, at 23:59

5 Grading policy

This programming assignment will be graded based on correctness and time complexity. We have 5 test cases in total, and we provide you two of these test cases: 0001.py, 0002.py, and their corresponding golden output: golden 1.txt, golden 2.txt. If your program can pass one of the test cases, you will get 10% each. If your program can pass one of the test cases with the desired time complexity, you will get addition 10% each. The total point is 100.