

CSE 12 – Basic Data Structures and Object-Oriented Design

Lecture 21

Greg Miranda, Spring 2021

Announcements

- Quiz 21 due Wednesday @ 12pm
- PA7 due Wednesday, May 26 @ 11:59pm (closed)
- Survey 8 due Friday @ 11:59pm
- Exam 2 – this week!
 - Released Friday @ 2pm
 - Due Saturday @ 6pm
 - Topics:
 - Cumulative
 - Big topics – lectures 9 - 17
 - Big O, Big Theta run-time analysis
 - Sorting algorithms
 - Hash tables/maps
- No class on Friday (but there is discussion)

Topics

- Heaps Applications
- Questions on Lecture 21?

MedianTracker (draw the picture and arrays)

- Draw the picture and the arrays for the following:
 - Add the following elements to the MedianTracker (in this order):
 - 5, 10, 15, 20, 25, 30, 35
 - What is the result of the call to `get()` after adding all the elements?

```
class Tracker {
    PriorityQueue<Integer> pq1 = new PriorityQueue<>{Collections.reverseOrder(Integer::compare)};
    PriorityQueue<Integer> pq2 = new PriorityQueue<>{Integer::compare};
    void add(int n) {
        if(pq2.size() == 0 && pq1.size() == 0) {
            pq2.add(n);
            return;
        }
        int current = get();
        if(n >= current) {
            pq2.add(n);
        }
        else {
            pq1.add(n);
        }
        int sizeDifference = pq2.size() - pq1.size();
        if(sizeDifference > 1) { pq1.add(pq2.poll()); }
        else if(sizeDifference < -1) { pq2.add(pq1.poll()); }
    }

    int get() {
        if(pq2.size() == pq1.size()) { return (pq2.peek() + pq1.peek()) / 2; }
        if(pq2.size() > pq1.size()) { return pq2.peek(); }
        else { return pq1.peek(); }
    }

    public String toString() {
        return "" + pq1 + " " + this.get() + " " + pq2;
    }
}
```