Experiment Title 3

Student Name: Praduman Kumar **UID: 20BCS9446**
Branch: BE-CSE **Section/Group: 20BCS-DM-714/A**
Semester: 6 **Subject .Code: 20CSP-351**
Subject Name: Competitive Coding-II **Date of Performance: 16-03-2023**

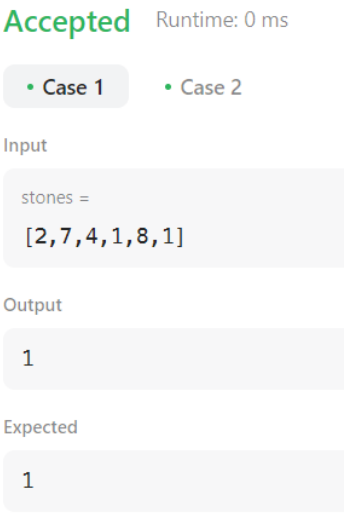
1. **Aim:** To demonstrate the concept of Heap model problem 1

2. **Objective:** The objective is to solve problem of heap and its methods and priority queue.

3. **Code and output:**
   Problem 1: Last Stone Weight
   Code:

```
class Solution {
    public int lastStoneWeight(int[] stones) {
        Queue<Integer> queue = new
PriorityQueue<>(Collections.reverseOrder());
        for (int i : stones) queue.offer(i);
        while (queue.size() > 1) {
            int first = queue.poll();
            int next = queue.poll();
            if (first != next) {
                queue.offer(first - next);
            }
        }
        return queue.isEmpty() ? 0 : queue.poll();
    }
}
```

   Output:



Problem 2: Distant Barcodes

Code:

```
class Solution {
    class Bar {
        int barcode;
        int freq;
        Bar(int barcode, int freq) {
            this.barcode = barcode;
            this.freq = freq;
        }
    }
    public int[] rearrangeBarcodes(int[] barcodes) {
        Map<Integer, Integer> map = new HashMap<>();
        for (int n : barcodes) {
            map.put(n, map.getOrDefault(n, 0) + 1);
        }
        PriorityQueue<Bar> pq = new PriorityQueue<>(
                (a, b) -> b.freq - a.freq
        );
        for (int key : map.keySet()) {
            pq.offer(new Bar(key, map.get(key)));
        }
        Bar prev = null;
        for (int i = 0; i < barcodes.length; i++) {
```

```
            Bar curr = pq.poll();
            barcodes[i] = curr.barcode;
            curr.freq--;
            if (prev != null) {
                pq.offer(prev);
            }
            if (curr.freq > 0) {
                prev = curr;
            }
            else prev = null;
        }
        return barcodes;
    }
}
```

Output: