895. Maximum Frequency Stack

Implement `FreqStack`, a class which simulates the operation of a stack-like data structure.

`FreqStack` has two functions:

- push(int x), which pushes an integer `x` onto the stack.
- pop(), which removes and returns the most frequent element in the stack.
    - If there is a tie for most frequent element, the element closest to the top of the stack is removed and returned.

```
class FreqStack {

    HashMap<Integer, Integer> freq;
    HashMap<Integer, Stack<Integer>> m;
    int maxFreq;

    public FreqStack() {
        freq = new HashMap<>();
        m = new HashMap<>();
        maxFreq = 0;
    }

    public void push(int x) {
        int f = freq.getOrDefault(x, 0) + 1;
        freq.put(x, f);
        maxFreq = Math.max(maxFreq, f);

        // create a stack for each frequency
        if (!m.containsKey(f))
            m.put(f, new Stack<Integer>());

        m.get(f).add(x);
    }

    public int pop() {
        int x = m.get(maxFreq).pop();
        freq.put(x, maxFreq -1);

        if (m.get(maxFreq).size() == 0)
            maxFreq--;

        return x;
    }
}

/**
 * Your FreqStack object will be instantiated and called as such:
 * FreqStack obj = new FreqStack();
 * obj.push(x);
 * int param_2 = obj.pop();
 */
```