画解剑指 Offer - LeetBook - 力扣 (LeetCode) 全球极客 挚爱的技术成长平台

leetcode.cn/leetbook/read/illustrate-lcof/58gyst

口剑指 Offer 59 - II. 队列的最大值 - 解决方案

题目描述

请定义一个队列并实现函数 max value 得到队列里的最大值,要求函数 max value 、 push_back 和 pop_front 的均摊时间复杂度都是 O(1)O(1)。

若队列为空, pop_front 和 max_value 需要返回-1

示例 1:

输入:

["MaxQueue", "push_back", "push_back", "max_value", "pop_front", "max_value"] [[],[1],[2],[],[],[]]

输出: [null, null, null, 2, 1, 2]

示例 2:

输入:

["MaxQueue", "pop_front", "max_value"] [[],[],[]] 输出: [null, -1, -1]

限制:

- 1 <= push_back,pop_front,max_value的总操作数 <= 10000
- 1 <= value <= 10^5

解题方案

思路

- 标签:单调队列、辅助队列
- 整体思路:
 - 。 这道题和 59 I 的思路很像,最难的地方在于取出最大值函数的时间复杂度如 何降为 O(1)O(1), 如果每次入队的时候都维护一个最大值, 那么时间复杂度很 明显不满足条件
 - 。 使用一个额外的辅助单调队列,该队列单调递减即可,保证最大值在队列头 部,这样的话就可以在取出最大值的时候时间复杂度减低

• 复杂度:

- 。 时间复杂度: O(1)O(1)。取 max_value 时只需要取单调队列头部即可
- 。 空间复杂度:O(n)O(n)。一共需要维护两个队列,普通队列和辅助单调队列, 最差的情况是两个队列里面 nn 个数字

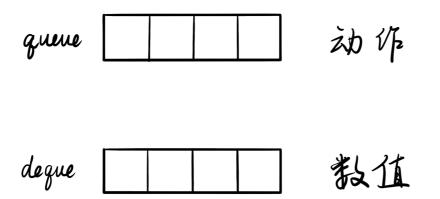
算法流程

- 1. 构造函数中初始化一个普通队列 queue 和一个辅助单调队列 deque
- 2. max_value 函数中只需要每次从辅助单调队列 deque 取头部值即可,如果没有则 返回 -1
- 3. push_back 函数中首先将值放入 queue 中,然后将 deque 尾部所有小于该值的 元素都剔除,最后将该值放入尾部,保证单调队列递减
- 4. pop_front 函数中首先获取 queue 头部值 head ,如果不存在则返回 -1 ,然后判断 deque 的头部值是否和 head 相等 ,如果相等则将其头部值也移除

代码

```
class MaxQueue {
    Queue<Integer> queue;
    Deque<Integer> deque;
    public MaxQueue() {
        queue = new LinkedList<>();
        deque = new LinkedList<>();
    }
    public int max_value() {
        return deque.size() > 0 ? deque.peek() : -1;
    }
    public void push_back(int value) {
        queue.offer(value);
        while(deque.size() > 0 && deque.peekLast() < value){</pre>
            deque.pollLast();
        }
        deque.offerLast(value);
    }
    public int pop_front() {
        int head = queue.size() > 0 ? queue.poll() : -1;
        if(deque.size() > 0 && deque.peek().equals(head)){
            deque.poll();
        }
        return head;
    }
}
 * Your MaxQueue object will be instantiated and called as such:
 * MaxQueue obj = new MaxQueue();
 * int param_1 = obj.max_value();
 * obj.push_back(value);
 * int param_3 = obj.pop_front();
 */
```

画解



⋈1/13 **⋈**

花絮

最热♪



匿名用户

来自上海2021-11-19

push_back不是要o(1)么

₾

1





-

来自浙江2021-08-27



不懂就问引入head值的目的到底是什么 pop_front 函数中首先获取 queue 头部值 head ,如果不存在则返回 -1,然后判断 deque 的头部值是否和 head 相等,如果相等则将其头部值也移除 这句话的后半句 deque 的头部值是否和 head 相等,如果相等则将其头部值也移除 的意思是指pop

可以这么说吧,辅助队列里保存的值,就是队列里可能成为最大值得数值。而且这个辅助队列是降序,所以当pop操作时,取出的head需要和辅助队列的头进行比较,判断这个数值是否需要也从辅助队列里移除

Δ

1





naruuu



来自河南2021-08-25

不懂就问 引入head值的目的到底是什么

pop_front 函数中首先获取 queue 头部值 head,如果不存在则返回 -1,然后判断 deque 的 头部值是否和 head 相等,如果相等则将其头部值也移除

这句话的后半句 deque 的头部值是否和 head 相等,如果相等则将其头部值也移除 的意思是指pop_front方法有一定的可能性是移除queue和deque 最上面的元素么?

₾

1





来自上海2022-03-11



push_back不是要o(1)么

均摊时间复杂度

