

原 java初、中、高级面试题必备——数据结构与算法—队列&栈

2019年06月26日 18:19:32 在IT中穿梭旅行 阅读数 16 更多

编辑

队列

192.描述一下队列?

队列是一种先进先出(FIFO)的数据结构，从队尾进，从队头出



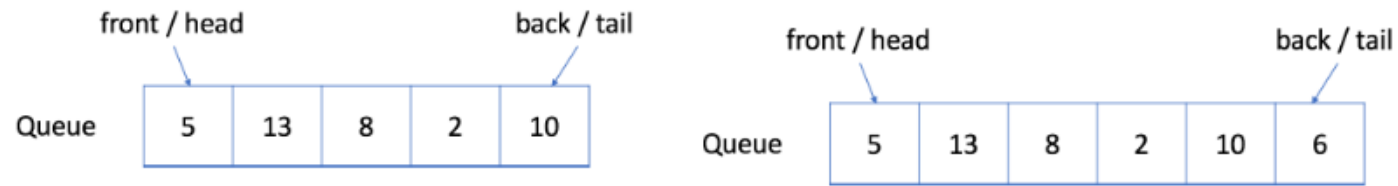
在 FIFO 数据结构中，将**首先处理添加到队列中的第一个元素**。

如上图所示，队列是典型的 FIFO 数据结构。插入（insert）操作也称作入队（enqueue），新元素始终被添加在**队列的末尾**。删除（delete）操作也被称为出队（dequeue）。你只能移除**第一个元素**。

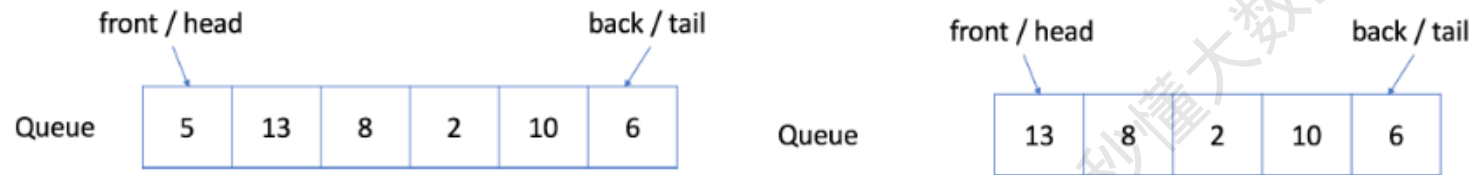
示例 - 队列

如下图所示

1.入队，将6插入



2.出队，删除5



为了实现队列，我们可以使用**动态数组**和指向队列头部的**索引**。

如上所述，队列应支持两种操作：**入队和出队**。入队会向队列追加一个新元素，而出队会删除第一个元素。所以我们需要一个索引来指出起点。

这是一个供你参考的实现：

```
1 class MyQueue {
2     // store elements
3     private List<Integer> data;
4     // a pointer to indicate the start position
5     private int p_start;
6     public MyQueue() {
7         data = new ArrayList<Integer>();
8         p_start = 0;
9     }
10    /** Insert an element into the queue. Return true if the operation is successful. */
11    public boolean enqueue(int x) {
```

```
12         data.add(x);
13         return true;
14     };
15     /** Delete an element from the queue. Return true if the operation is successful. */
16     public boolean deQueue() {
17         if (isEmpty() == true) {
18             return false;
19         }
20         p_start++;
21         return true;
22     }
23     /** Get the front item from the queue. */
24     public int Front() {
25         return data.get(p_start);
26     }
27     /** Checks whether the queue is empty or not. */
28     public boolean isEmpty() {
29         return p_start >= data.size();
30     }
31 };
32
33 public class Main {
34     public static void main(String[] args) {
35         MyQueue q = new MyQueue();
36         q.enqueue(5);
37         q.enqueue(3);
38         if (q.isEmpty() == false) {
39             System.out.println(q.Front());
40         }
41         q.deQueue();
42         if (q.isEmpty() == false) {
43             System.out.println(q.Front());
44         }
45         q.deQueue();
46         if (q.isEmpty() == false) {
47             System.out.println(q.Front());
48         }
49     }
50 }
```

缺点

上面的实现很简单，但在某些情况下效率很低。随着起始指针的移动，浪费了越来越多的空间。当我们有空间限制时，这将是难以接受的。

193.描述一下循环队列？

循环队列。具体来说，我们可以使用**固定大小的数组**和**两个指针**来指示起始位置和结束位置。目的是**重用**我们之前提到的**被浪费的存储**。



消防工程师限制专业了

消防工程师