

# 剑指Offer 09-用两个栈实现队列

## 题目描述

用两个栈实现一个队列。队列的声明如下，请实现它的两个函数 `appendTail` 和 `deleteHead`，分别完成在队列尾部插入整数和在队列头部删除整数的功能。（若队列中没有元素，`deleteHead` 操作返回 -1）

示例 1:

```
输入:
["CQueue", "appendTail", "deleteHead", "deleteHead"]
[[], [3], [], []]
输出: [null, null, 3, -1]
```

示例 2:

```
输入:
["CQueue", "deleteHead", "appendTail", "appendTail", "deleteHead", "deleteHead"]
[[], [], [5], [2], [], []]
输出: [null, -1, null, null, 5, 2]
```

提示:

- `1 <= values <= 10000`
- 最多会对 `appendTail`、`deleteHead` 进行 10000 次调用

## 思路

1. 栈的特性：先进后出
2. 队列的特性：先进先出
3. 两个栈，一个作输入栈，只管输入操作，另一个作输出栈，负责输出返回
4. 两次先进后出，负负得正即为先进先出
5. 在构造函数中完成两个栈的初始化工作，在 `appendTail` 函数中向其中一个栈 `stack1` 结尾插入整数，在 `deleteHead` 函数中如果 `stack2` 为空，则将 `stack1` 的值全部弹出放到 `stack2` 中，再从 `stack2` 中取值，这样达到了负负为正的队列效果

## 代码

```
1 class CQueue {
2     private:
3         stack<int> in,out;    //输入栈和输出栈
4     public:
5         CQueue()
6         {
```

```

7
8     }
9
10    void appendTail(int value)
11    {
12        in.push(value); //压入输入栈
13    }
14
15    int deleteHead()
16    {
17        if(!out.empty())    //输出栈不为空 直接弹出即可 注意
C++ STL中的stack的pop方法返回值为void
        //应该 先top()取出再pop()弹出 以免误处理
18        {
19            int v_1 = out.top();    //取值
20            out.pop();    //弹出
21            return v_1;
22        }
23        else
24        {
25
26            //把所有的输入栈中的元素压入输出栈中 两次先进后出即
为先进先出
27            while(!in.empty())
28            {
29                int v_1 = in.top();
30                in.pop();
31                out.push(v_1);
32            }
33            if(out.empty())
34            {
35                return -1;
36            }
37            else
38            {
39                int v_2 =out.top();
40                out.pop();
41                return v_2;
42            }
43
44        }

```

```

45     }
46 };
47
48 /**
49  * Your CQueue object will be instantiated and called
  as such:
50  * CQueue* obj = new CQueue();
51  * obj->appendTail(value);
52  * int param_2 = obj->deleteHead();
53  */

```

```

1  //简化版本
2  class CQueue {
3  public:
4      CQueue() {}
5
6      void appendTail(int value)
7      {
8          in.push(value);
9      }
10
11     int deleteHead()
12     {
13         if(out.empty())
14         {
15             while(!in.empty())
16             {
17                 out.push(in.top());
18                 in.pop();
19             }
20         }
21         if(!out.empty())
22         {
23             int res = out.top();
24             out.pop();
25             return res;
26         }
27         else{
28             return -1;
29         }

```

```
30 |     }  
31 private:  
32     stack<int> in,out;  
33 };
```



易行行行行  
2023.01.25 12:18



详情

写题解

C++



点击图片查看分布详情

备注

在此输入备注

相关标签

选择标签

0/5