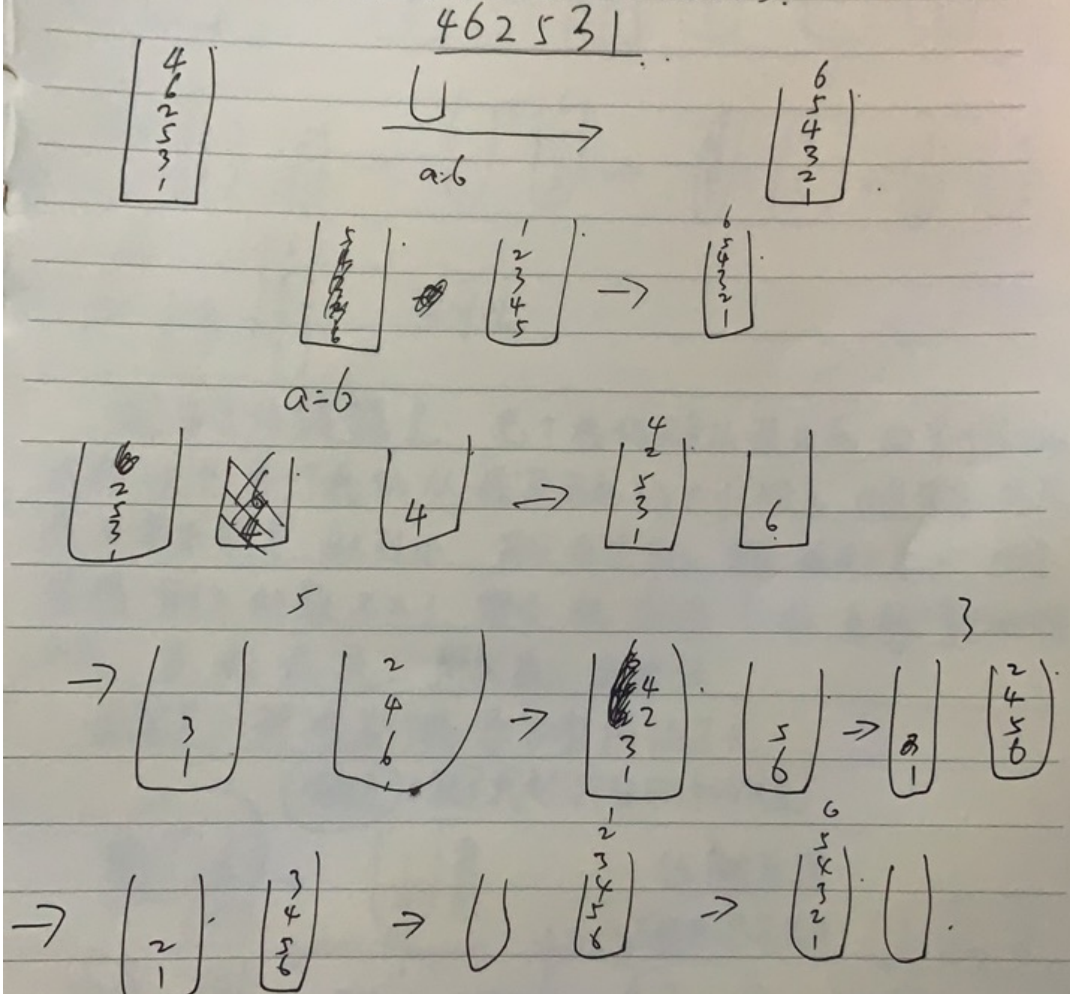
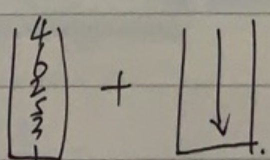
Date

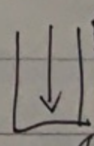
用一个栈实现另一个栈的排序。

将该栈从顶到底按从大到小的顺序排列。



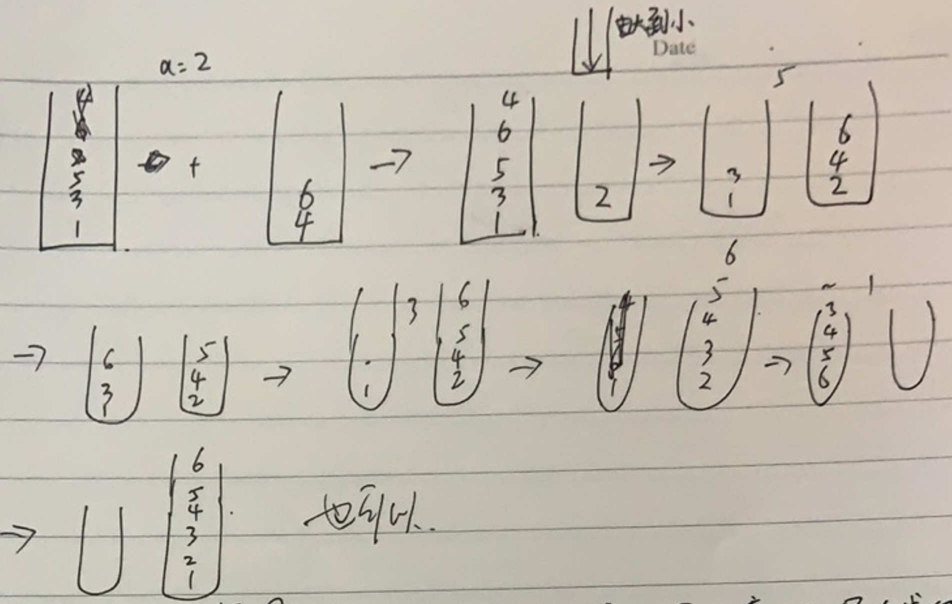


由小到大。，或者 另一个栈直接



由大到小 也应该可以。

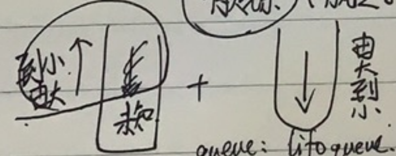
twy



这个方法的精髓是：把一个栈的元素从顶由底向另一个栈倒入，选择选定另一个栈的从栈顶到底的大小顺序。这里直接指定题目要求顺序，由大到小。所以往里倒入要求越来越大。如果发现倒入的数不大了，那么就倒回B栈直到拿出的元素大于B栈栈顶，那就放入并继续。

这就是一种往另一个栈添加数据的方式。

修改元素（就是找到这个待定元素的位置）



这就原理

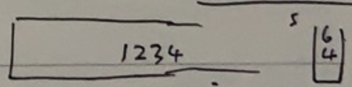
[illegible]

queue: lifoqueue. [illegible]

stack. ~~list~~ pop(). append(). [-1]. peek 完美契合

def. f(stack1, stack2)：将stack1倒入stack2，并stack2顶→底 由大到小。

```
def f(stack1, stack2):
    while stack1:
        a = stack1.pop()
        while a < stack2[-1] and stack2:
            stack1.append(stack2.pop())
        stack2.append(a)
```



stack2 不空，并且 a < stack2[-1]