Leetcode Bootcamp Homework

1. Implement queue using stacks

```
class MyQueue(object):
    def __init__(self):
        self.in_stack = []
        self.out_stack = []

    def push(self, x):
        """
        :type x: int
        :rtype: None
        """
        self.in_stack.append(x)

    def move_in_to_out(self):
    # Transfer elements from in_stack to out_stack if out_stack is empty
        if not self.out_stack:
            while self.in_stack:
                self.out_stack.append(self.in_stack.pop())
    def pop(self):
        """
        :rtype: int
        """
        self.move_in_to_out()
        return self.out_stack.pop()


    def peek(self):
        """
        :rtype: int
        """
        self.move_in_to_out()
        return self.out_stack[-1]


    def empty(self):
        """
        :rtype: bool
        """
        return not self.in_stack and not self.out_stack
```

2. Daily Temperatures

```
class Solution(object):
    def dailyTemperatures(self, temperatures):
        """
        :type temperatures: List[int]
        :rtype: List[int]
        """
        n = len(temperatures)
        ans = [0] * n
        stack = []

        for i, temp in enumerate(temperatures):
            # Pop from stack while current temp is warmer than stack top
temp
            while stack and temp > stack[-1][0]:
                prev_temp, prev_index = stack.pop()
                ans[prev_index] = i - prev_index
            stack.append((temp, i))

        return ans
```

3. Number of people aware of a secret

```
class Solution(object):
    def peopleAwareOfSecret(self, n, delay, forget):
        """
        :type n: int
        :type delay: int
        :type forget: int
        :rtype: int
        """
        mod = 10**9 + 7
        dp = [0] * (n + 1)
        dp[1] = 1  # On day 1, one person knows the secret
        for day in range(2, n + 1):
            for share_day in range(max(1, day - forget + 1), day - delay
+ 1):
                dp[day] = (dp[day] + dp[share_day]) % mod
        return sum(dp[max(1, n - forget + 1): n + 1]) % mod
```