Leetcode(python) 队列难题

363矩形区域不超过K的最大数值和

题目：给定一个非空二维矩阵 *matrix*和一个整数 *k*，找到这个矩阵内部不大于 *k* 的最大矩形和。

**示例:**

**输入:** matrix = [[1,0,1],[0,-2,3]], k = 2

**输出:** 2

**解释:** 矩形区域 [[0, 1], [-2, 3]] 的数值和是 2，且 2 是不超过 k 的最大数字（k = 2）。

**说明：**

1. 矩阵内的矩形区域面积必须大于 0。
2. 如果行数远大于列数，你将如何解答呢？

思路：

1. 代码：**class** Solution(object):
2. **def** maxSumSubmatrix(self, matrix, k):
3. """
4. :type matrix: List[List[int]]
5. :type k: int
6. :rtype: int
7. """
8. m = len(matrix)
9. n = len(matrix[0]) **if** m **else** 0
11. M = max(m, n)
12. N = min(m, n)
13. ans = None
14. **for** x **in** range(N):
15. sums = [0] \* M
16. **for** y **in** range(x, N):
17. slist, num = [], 0
18. **for** z **in** range(M):
19. sums[z] += matrix[z][y] **if** m > n **else** matrix[y][z]
20. num += sums[z]
21. **if** num <= k:
22. ans = max(ans, num)
23. i = bisect.bisect\_left(slist, num - k)
24. **if** i != len(slist):
25. ans = max(ans, num - slist[i])
26. bisect.insort(slist, num)
27. **return** ans **or** 0

862和至少为K的最短子数组

题目：返回 A 的最短的非空连续子数组的**长度**，该子数组的和至少为 K 。

如果没有和至少为 K 的非空子数组，返回 -1 。

**示例 1：**

**输入：**A = [1], K = 1

**输出：**1

**示例 2：**

**输入：**A = [1,2], K = 4

**输出：**-1

**示例 3：**

**输入：**A = [2,-1,2], K = 3

**输出：**3

**提示：**

1. 1 <= A.length <= 50000
2. -10 ^ 5 <= A[i] <= 10 ^ 5
3. 1 <= K <= 10 ^ 9

思路：使用动态规划，考虑dp[i]代表包含第i个的连续和

代码：

1. **class** Solution(object):
2. **def** shortestSubarray(self, A, K):
3. """
4. :type A: List[int]
5. :type K: int
6. :rtype: int
7. """
8. n,dp=len(A),[0]
9. **for** v **in** A: dp.append(dp[-1]+v)
10. queue,res=collections.deque(),n+1
11. **for** i **in** range(n+1):
12. **while** queue **and** dp[i]<=dp[queue[-1]]:
13. queue.pop()
14. **while** queue **and** dp[i]-dp[queue[0]]>=K:
15. res=min (res,i-queue.popleft())
16. queue.append(i)
17. **return** res **if** res<n+1 **else** -1