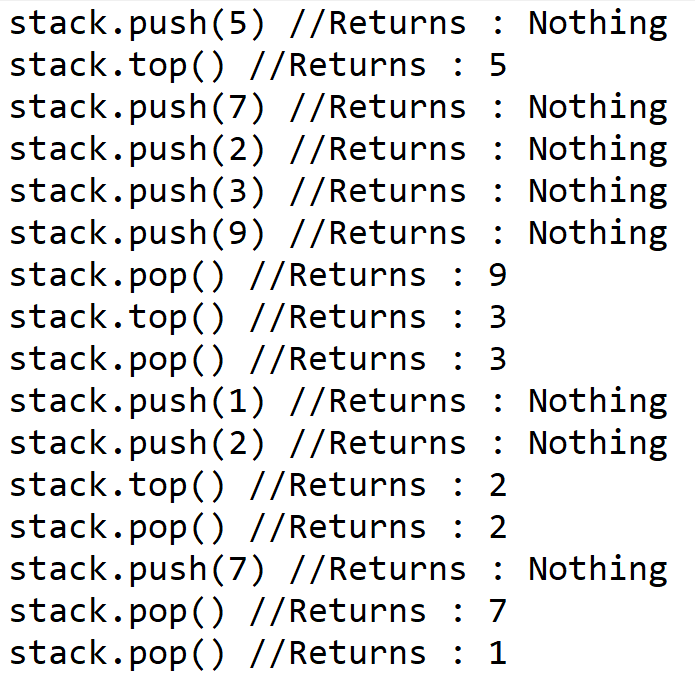
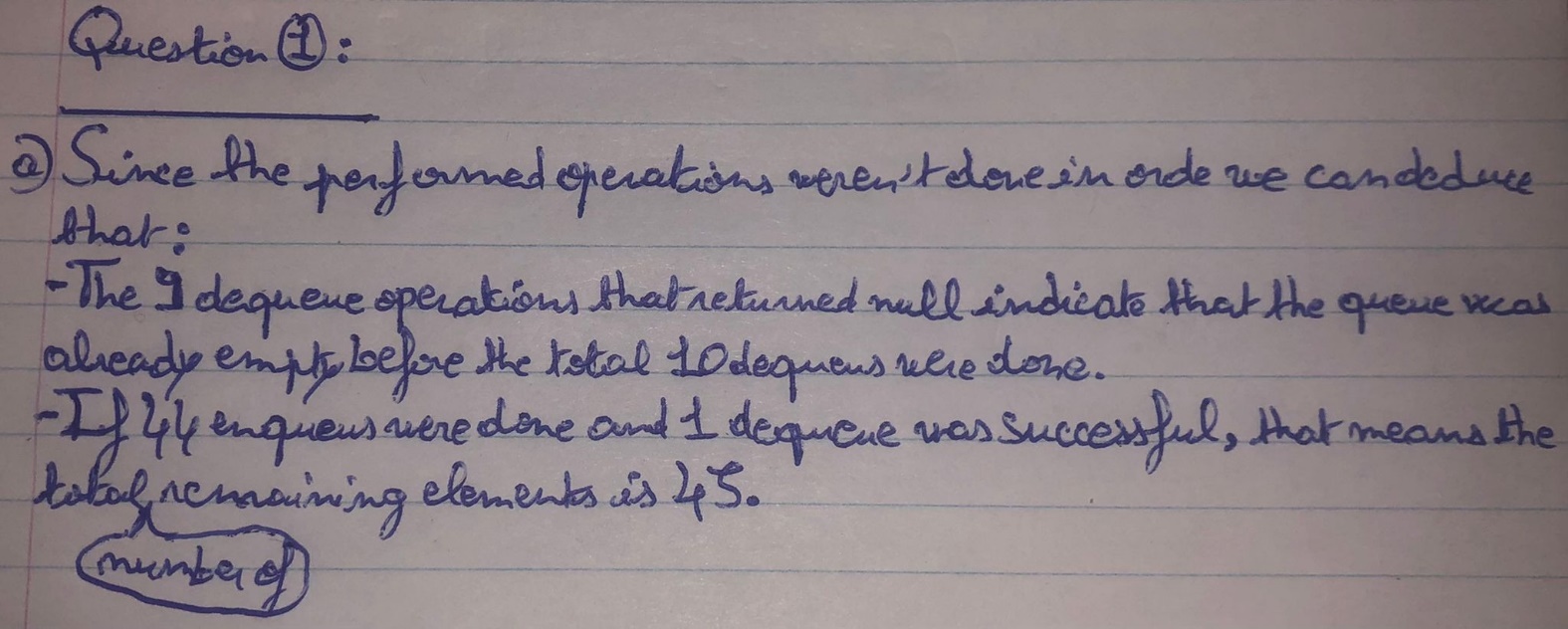
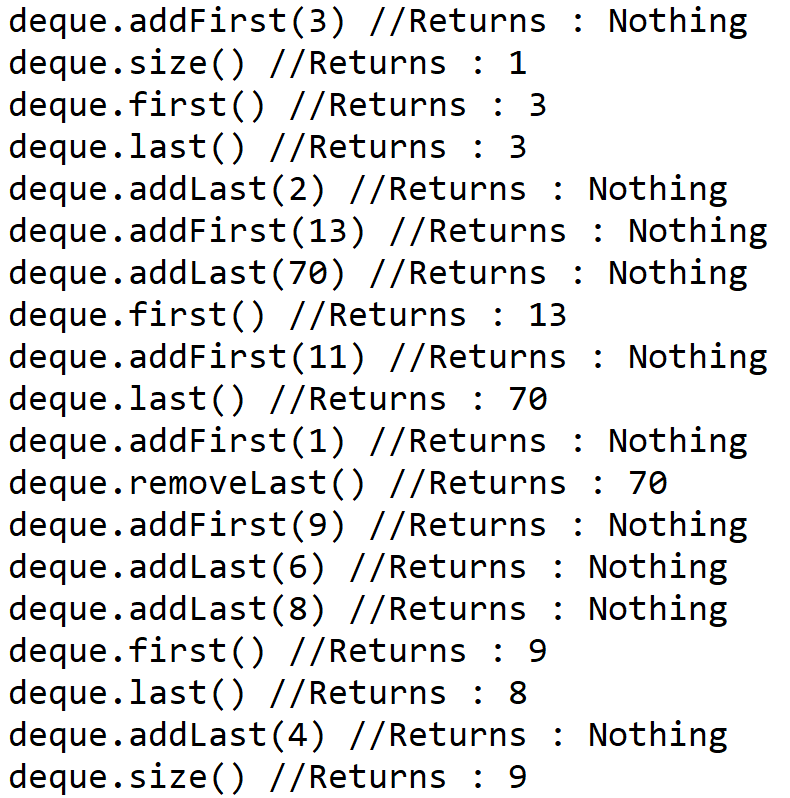
******Question 1 :**b) In order : 5,9,3,3,2,2,7,1

****c) In order : 1,3,3,13,70,70,9,8,9

**Question 2 :**a) *(Java implementation available on the Question2 folder)*b)   
**My algorithm is correct because it outputs the expected results, takes into consideration all possible edge cases (empty array/list of intervals, a value of T that’s strictly inferior to zero), and the three possible base cases in the type of implementation I decided to adopt.**  
**My implementation is O(n) because the least efficient operations performed in it are two loops that travel through a data structure linearly :  
-The first one is a while loop used to navigate the array of busy intervals in a helper method. [O(n)]  
-The second one is used to navigate the array of merged busy intervals in the freeTimesToMeet() method. [O(n)]**