You are given the head of a singly linked list where each node contains an integer. Write a function filter\_within\_range(head, low, high) to filter out all numbers **inside of a given inclusive range** (i.e., numbers equal or greater than low or equal or lower than high). After filtering, create a new linked list that contains only the numbers that are inside the inclusive range [low, high], preserving the original order. The function should return another linked list, containing those elements in **reversed order**. If there is no such element present, return None.

No other data structures can be used other than linked lists.   
Consider that node class is already provided.

| **Sample Input** | **Sample Output & Explanation** |
| --- | --- |
| 4 -> 12 -> 7 -> 10 -> 18 -> 1  low = 10  high = 20 | **Sample Output**: 18-> 10 -> 12  12, 10, 18 are inside the inclusive range [10, 20] |
| 14 -> 12 -> 17 -> 20  low = 30  high = 40 | **Sample Output**: *None*  None of the Input Linked List are outside the range [30,40] |