An application requires a data structure to store a list of char strings. It should be possible to obtain the sub-list of all strings with the same starting character in O(1) time.

A file input.txt contains a sequence of space-separated integers in increasing order. This file needs to be read only once and processed using a data structure which writes the sequence to another file output.txt but in reverse order of the sequence in O(N) time where N is the number of elements in the file. (E.g. input.txt: 90 10 23 output.txt: 23 10 90).

An application requires a data structure where a new item can be inserted either as a predecessor (previous element) or successor (next element) of a given item in O(1) time. Assume that a pointer is available for the node containing the given item.

Question

What does the code do here ?

```
public void func(Node first) {
    Node p, q;
    p = first;
    q = p.next;
    while (q.next != null){
        p = q;
        q = q.next;
    }
    p.next = null;
}
```

Which of the following permutation can be obtained in the same order using a stack assuming that input is the sequence 5, 6, 7, 8, 9 in that order?

- 7,8,9,5,6
- 5,9,6,7,8
- 7,8,9,6,5
- 9,8,7,5,6

Stacks:
https://leetcode.com/problems/remove-duplicate-letters/description/