P1
↘ 5 → 6 → 3 → 1 → 8 → 7

4. 2. 1. 3. 3.

4
2 4
1 2 4
1 2 3 4

P2 → (d) → 5
-Infinity

- getNode

- insert

need to handle cases with duplicates.

→ d → 5 → 6

```
const fn = (head) => {
    const dummy = new ListNode(-Infinity);
    let p1 = head;


    while(p1) {
        const next = p1;
        p1 = p1.next;
        insert(dummy, next);
    }
    return dummy.next;
}
```

1 → 3 → 5
4.

n1
d → null
n:=2

d → 1 → 3 → 3 → 3 → 5
4

```
function insert(h1, h2) {
    let ptr = h1;
    while(ptr) {
        if(!ptr.next ||
            ptr.val < h2.val) {
            h2.next = ptr.next;
            ptr.next = h2;
            return;
        }
    }
}
```

```
ptr.val <= h2.val &&
h2.val <= ptr.next.val
```