

Java

```
/**
 * Definition for singly-linked list.
 * public class ListNode {
 *     int val;
 *     ListNode next;
 *     ListNode() {}
 *     ListNode(int val) { this.val = val; }
 *     ListNode(int val, ListNode next) { this.val = val; this.next = next; }
 * }
 */
class Solution {
    public ListNode frequenciesOfElements(ListNode head) {

    }
}
```

JavaScript

```
/**
 * Definition for singly-linked list.
 * function ListNode(val, next) {
 *     this.val = (val===undefined ? 0 : val)
 *     this.next = (next===undefined ? null : next)
 * }
 */
/**
 * @param {ListNode} head
 * @return {ListNode}
 */
```

```
var frequenciesOfElements = function(head) {  
  
};
```

TypeScript

```
/**  
 * Definition for singly-linked list.  
 * class ListNode {  
 *     val: number  
 *     next: ListNode | null  
 *     constructor(val?: number, next?: ListNode | null) {  
 *         this.val = (val===undefined ? 0 : val)  
 *         this.next = (next===undefined ? null : next)  
 *     }  
 * }  
 */  
  
function frequenciesOfElements(head: ListNode | null): ListNode | null {  
  
};
```

C++

```
/**  
 * Definition for singly-linked list.  
 * struct ListNode {  
 *     int val;  
 *     ListNode *next;  
 * }
```

```

*     ListNode() : val(0), next(nullptr) {}
*     ListNode(int x) : val(x), next(nullptr) {}
*     ListNode(int x, ListNode *next) : val(x), next(next) {}
* };
*/
class Solution {
public:
    ListNode* frequenciesOfElements(ListNode* head) {

    }
};
-----

```

C#

```

/**
 * Definition for singly-linked list.
 * public class ListNode {
 *     public int val;
 *     public ListNode next;
 *     public ListNode(int val=0, ListNode next=null) {
 *         this.val = val;
 *         this.next = next;
 *     }
 * }
 */
public class Solution {
    public ListNode FrequenciesOfElements(ListNode head) {

    }
}
-----

```

